Integration of Rate and Credit Schemes for ATM Flow Control[†]

K. K. Ramakrishnan
AT&T Bell Laboratories
600 Mountain Ave
Murray Hill, NJ 07974
<kkrama@research.att.com>

and

Peter Newman
Ipsilon Networks Inc.
2465 Latham St., Suite 100
Mountain View, CA 94040
<pn@ipsilon.com>

ATM is the first switching technology that can support both fixed bandwidth services similar to circuit switching, and highly variable bandwidth services similar to packet switching, in a single integrated environment [1]. Definitions for traffic management for services with a fixed traffic profile were completed by the ATM Forum in their version 3.0 specification [2]. For the last year and a half the Traffic Management Group of the ATM Forum has been working on flow control for the highly variable bandwidth services typically supported by packet switching networks. In such services there is no explicit contract between the network and the user specifying the traffic profile and quality of service expected. Rather, the network is expected to provide each user with a fair share of the amount of bandwidth dynamically available. It is expected that if the user adjusts the transmission rate according to the feedback from the network then cell loss will remain low. The ATM Forum has termed such services available bit rate (ABR) services.

A congestion control loop is required between the network and the user to support an ABR service. Two separate schools of thought (religions) developed during the ABR debate as to how to implement the control loop: rate, and credit. In the rate-based view the network sends information to the user specifying the bit rate at which the user should be transmitting and the control loop may extend end-to-end across the network. The credit-based approach sends information about the available buffer space independently on each link of the network and is thus a link-by-link mechanism. A third alternative was also proposed which observed that both rate and credit solutions have their pros and cons and that to a large extent they can be viewed as complementary.

This third alternative was the integrated proposal which attempted to allow these different control mechanisms to coexist. The integrated proposal suggested that rate control was the most appropriate for the wide area but that static credit control had distinct advantages in the local area (i.e., it had been built and proven to work). It was an attempt to combine the advantages of both approaches into a single proposal for ATM flow control. In this paper we present the argument in support of an integrated approach (while remaining cognizant of the fact that rate-based control was selected by the ATM Forum in their September meeting).

Why Rate in the Wide Area

Wide area networks may be classified as such for one very simple reason: they operate across a wide geographical area. Distance means propagation delay and propagation delay may in fact be greater than queueing delay in a wide area, high-speed network [3]. The same dynamic response available in a local area network is simply not available from a wide area network due to the limitations imposed by the speed of light. This is physics — it will not change with improved implementation.

Several observations result from the fact that the propagation delay is not negligible. The buffer sizes that would be required to support a static hop-by-hop credit scheme are impractical because of the propagation delay of the long distance links. So the only credit scheme that could be applied to the wide area would require dynamic buffer allocation. Also, the speed at which a user can adjust to changing network conditions is a function of the propagation delay and will be much slower in the wide area than the local area. In addition, the customer understands this and has a different expectation of the performance available from the wide area than that of the local area.

The customer does not expect the same performance from a server physically located on the other side of the country as is expected from a local server on campus. It will always be true that bandwidth is plentiful in the local area and will always be a more expensive and more highly shared

[†] IEEE Network Mag., Mar. 1995, pp. 49–56.

in the wide area. Thus we have a different expectation from a wide area network than we do for a local area network. So we might expect a different performance from a control loop designed for the wide area than one optimized for the local area.

Conversely, we do not wish to impose the performance limitations of the wide area upon the local area by mandating a single control mechanism that cannot be optimized for both. During the development of the rate-based approach several proposals were made in which a user began transmission of a burst at a minimum rate and slowly ramped up to a fair share of the bandwidth over a period of about 5 ms. This may be a reasonable method in the wide area but the local area expects immediate access to the full bandwidth available. Remote procedure call and client server applications with moderate size but bursty traffic would perform poorly in such an environment.

The rate-based approach seems the more natural choice for public wide area networks. With the distances involved in the public network a static credit scheme requires an inordinate amount of buffering to permit a very large number of connections to operate at high speed. While credit schemes with dynamic buffer allocation [4, 5] have been proposed, they are not yet of sufficient maturity. It would be unduly optimistic to expect the public carriers to endorse a dynamic credit scheme at its current stage of development.

In addition, some of the public carriers have a very different view of the timescales involved in an ABR service. They have a very different concept of the rate of variation of bandwidth on a connection than that envisaged in the local area. For example, one of the rate-based solutions proposed by a carrier suggests an ABR service in which the bandwidth of a connection changes in the order of once every 30 seconds. Clearly one would employ a rate based mechanism to implement such a service.

A public carrier will need to deploy high-speed switches with a large number of access ports. It is felt that implementing per-VC queueing, as required for a credit scheme, on such a switch will incur unacceptable expense. Further, the larger carrier switches will operate with port speeds of 2.4 Gb/s and above. It is felt that at these speeds it will be difficult to implement anything more complex than EFCI marking. Also, EFCI marking is the only congestion control mechanism specified in the ATM Forum Version 3.0 specification so compatibility with EFCI marking switches was seen as important to the carriers and wide area network vendors. Finally, public carriers like to charge for their services. Many felt that it was much easier to perform billing when the rate was adjusted explicitly by the network than for a credit-based scheme.

Why Credit Based Control in Local Area Networks

The initial adoption of ATM in the local area will mostly be driven by the desire to efficiently support existing applications in a high-speed LAN. The ABR class of service will essentially be used as a “best effort” service that emulates the current behavior of existing LAN technologies. This means that bandwidth must be available on a timely basis, allowing applications to “almost instantaneously” utilize all or most of the available bandwidth, while maintaining packet loss low enough for existing applications to work well. It is important to maintain low loss, as a very small cell loss rate results in a significantly higher packet loss rate. The natural question arises as to what cell loss rate is acceptable: packet loss results in inefficient usage of both the network’s and the end systems’ resources.

To allow for a wide dynamic range of parameters for higher layer protocols, it would be preferable for congestion management schemes to maintain the overall congestion related packet loss rate as low as possible. In essence, the smaller the packet loss rate the better. Among the proposals put forth at the ATM Forum [4–6], the hop-by-hop per-VC credit flow control scheme (FCVC, based on static allocation of buffers) achieves the “ideal” from this perspective: the packet loss rate due to congestion is zero. This makes the overall behavior of LAN applications, which use existing protocol stacks (e.g., TCP/IP, UDP/IP, IPX etc.) much more predictable. The fact that the loss probability is zero makes the behavior of applications relatively insensitive to parameter settings in many of the algorithms used throughout the protocol stack (e.g., TCP retransmission timers, the extent of the change on the window size when a retransmission occurs, sensitivity to retransmission algorithms, network layer timers etc.). In a Local Area Network, the link distances are modest. The number of cell buffers needed for a VC to fully utilize the link is of the order of 10 or 12 cells. The number of VCs that a typical LAN switch needs to support are also claimed to be in the 1K range, with smaller switches having substantially smaller numbers of active VCs to support and larger switches being capable of supporting at most an order of magnitude larger. As a result, even for a switch supporting 1K VCs, the amount of cell buffering needed is of the order of 10K cell buffers, which is about 0.5 Mbytes of memory, which may be considered a reasonable amount of memory per port. Thus, the use of the static allocation of buffers for the hop-by-hop per-VC credit flow control scheme appeared to be quite suitable for the LAN.

The competition from both shared LANs (e.g., different Fast Ethernet proposals, FDDI) and switched versions of these LANs promise cost and aggregate bandwidth efficiencies that compete well with those possible on ATM. Therefore, from a purely ABR perspective, the peak available bandwidth of these competing technologies are comparable to the “typical” ATM bandwidths being currently considered. The perception that the QoS guarantees provided by ATM are advantageous will come gradually. We see other service classes, such as CBR and VBR coexisting with ABR service, and being in more widespread use in the future. However, in the immediate future, it is to the advantage of ATM LANs to achieve high efficiency in its use for data communication using ABR. The opportunistic behavior of ABR in utilizing unused bandwidth will initially (and likely for the long term) be very important. We see efficient use of unused bandwidth, and the responsiveness to the changes in the available bandwidth as significant design considerations for a congestion control scheme in the local area.

It is possible that CBR and VBR flows use bandwidth in a fashion that the “troughs” in their bandwidth usage may last from a few microseconds to several tens of milliseconds. A scheme for controlling ABR flows that is responsive in using a link’s available bandwidth within a few microseconds (few cell times) of its being available is highly desirable. We see the hop-by-hop schemes have a potential for this desired responsiveness. With a static buffer allocation for the hop-by-hop credit based scheme, it has been demonstrated that the efficiency of using this available bandwidth is quite high.

Perhaps one of the most significant performance advantages of hop-by-hop credit over end-end rate is that its performance is independent of the incident traffic pattern. Data traffic contains a large proportion of short, transient, bursts of traffic. An end-end scheme can only control traffic bursts that are longer than the round trip time of the connection. Transient bursts can only be accommodated by providing sufficient buffering. The static hop-by-hop credit scheme, however, allows control of all traffic, short transient bursts as well as large file transfers. It offers “zero congestive loss” performance regardless of the arriving traffic pattern.

An obvious, but inadequate, implementation of hop-by-hop credit-based flow control “without per-VC buffering” would not be sufficient. This would not assure that the queueing delays through the network are maintained at sufficiently small levels (operating at the “knee” of the throughput and delay curves [7]). However, by separating the flows on a per-VC basis, we ensure that the queue for one VC does not interfere with that for another VC [8]. Given enough credits for each of the VCs so that it can potentially fully utilize a link, each VC can opportunistically take advantage of the available bandwidth on a hop without interfering with other VCs, while operating at the “knee.”

An important area, particularly relevant in a LAN, is the overall fairness and efficiency of operation even when there exist one or more sources (users of the network) that do not cooperate with the other flows in the congestion management scheme. This has been a long-standing question impeding the widespread implementation of schemes for congestion management that require/assume cooperation by all the users of the network. While a policing mechanism at the entry into a WAN is frequently considered, such a policing mechanism may be expensive in a LAN, for an end-to-end rate based scheme. If it is necessary, it would in fact have to be implemented in every node (particularly switches) in the network, based on our current understanding. Thus, it is desirable to have the non-cooperating flows separated in such a way that they do not interfere with those that are cooperating or “well behaved” from the view of the congestion management scheme. Per-VC credit based flow control implicitly allows the network to protect itself from a non-cooperating user, and isolates these from the well behaved users. When a non-cooperating user misbehaves, the consequence is additional loss for that VC only, without any concomitant loss experienced by other users. We see this as being desirable for use of a technology, particularly in LANs.

Another complementary issue is that of a switch not participating in the congestion management scheme. Consider a credit scheme in the LAN operating with one or more intervening switches not participating in the credit protocol. This has been addressed using the concept of “tunneling”. The two switches on either side of the portion of the subnet

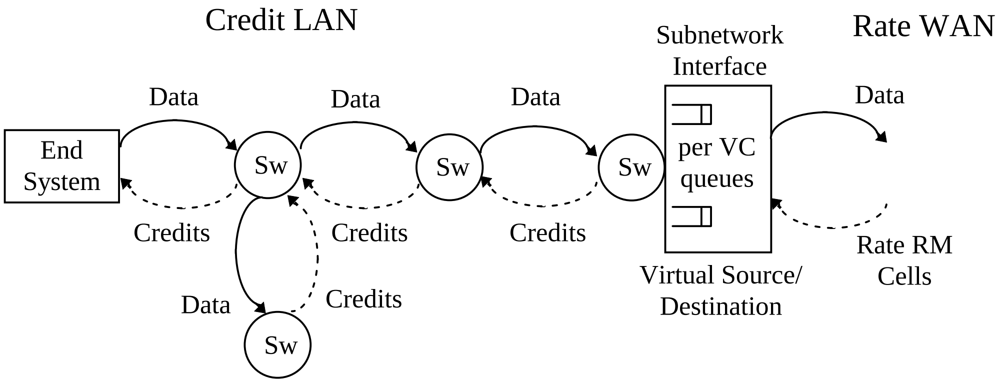

Figure 1: Subnetwork interface connecting a credit LAN to a rate WAN.

that has non-credit switches (let us call it a "non-credit cloud") that participate in the credit update protocol (CUP) communicate the credit cells on a VC that is set up across the "non-credit cloud". This allows for reasonable management of the buffers at the switches that participate in CUP. If the other switches that exist in the path that do not participate in CUP have adequate buffering, then the burst load into these switches are limited by the credits issued by the remote "cooperating switch." This minimizes the amount of loss experienced by the VCs that span the "non-credit cloud," while not providing the strict guarantee of loss-free behavior. The idea of tunneling is explained in more detail in [9].

Interworking Multiple Congestion Control Schemes

Rate-based control is the only solution acceptable to the public carriers for congestion control across the wide area. Static credit control requires excessively large buffers and dynamic credit algorithms are still under development. Rate-based control is very flexible and permits a variety of implementations within the switch. This allows product differentiation while maintaining compatibility with the standard. The more complex rate-based implementations will offer a higher performance service, yet the expectations of the customer are lower for a wide area service than for one within the local area.

Static credit-based control permits the lowest cost adapters and the highest performance in the LAN. While there is increased complexity in the switch to support the required per-VC queueing, this may not translate to greatly increased cost for the modest sizes of switch likely to be found in a LAN (5 Gb/s capacity with 2000 VCs per OC-3 port). Indeed it is not certain that an explicit rate based approach will offer a much lower cost to attain a similar performance.

Therefore, in order to combine the flexibility of rate with the performance of credit several integrated approaches were considered:

- Rate in the WAN/credit in the LAN.
- Rate is default, credit is optional.
- One Size Fits All.

The first two approaches are combinations of rate and credit while the third is a true integration.

Rate in the WAN, Credit in the LAN

This is perhaps the simplest attempt at an integrated solution yet it suffers the major drawback of creating two types of ATM interfaces. The proposal is simply that a rate-based scheme be used in the wide area and a static credit-based scheme in the local area.

Figure 1 illustrates the concept. There is an interface located on the trunk port card of the campus backbone switch that connects to the wide area. This interface acts as a virtual source and virtual destination terminating the credit and rate control loops and interconnecting them. So the LAN looks like a rate-based source to the WAN, and the WAN looks like a credit-based subnet to the LAN. The credit scheme on the LAN side is terminated by returning credit cells for VCs whose data cells are forwarded on to the WAN. The buffer at the interface from the LAN to the WAN direction needs to be sized only to the extent of having an adequate amount of buffering for each VC to go at the full rate of the LAN link. However, the interface from the WAN to the LAN link may in fact need substantially more buffer to accommodate the potentially large end-to-end round-trip feedback delay.

Fundamentally, this virtual source/destination function is a traffic shaper for the ABR service. It is very likely that a traffic shaper will be required in this location to shape the other services, VBR and CBR, on entrance to the wide area. Thus it is not necessarily an excessive burden to require such a function at this interface. In addition, even were the LAN also rate-based, a virtual source/destination function is likely to be required at the LAN/WAN boundary in order to police the ABR service.

With this solution, we have achieved the performance of credit in the local area and a simple, low-cost end-station adapter card. However, we have divided the ATM world into two types of interface, credit-based LAN and rate-based WAN. Our experience with the Ethernet/Token Ring duality suggests that it would be far better to attempt a single interface even at the expense of somewhat increased cost. Also this solution perpetuates the boundary between the LAN and the WAN. This boundary is artificial. Indeed it is more of a political boundary than a physical boundary and there will be many instances when it is unclear which side of the fence a particular product falls thus requiring the support of both types of interface. Interoperability problems also surface in this approach. It is unclear how one would propose to support an adapter card only capable of rate-based operation within a credit based LAN, and also how to support a version 3.0 switch within a credit based LAN.

Rate is Default, Credit is Optional

In this solution, rate-based control is required in the WAN and it is the default scheme in the LAN. Static credit-based control is permitted as an option within the LAN, selected on a per connection basis, when a connection is established. (Selection on a per link basis is impractical because it will force every switch port to be capable of implementing the virtual source/destination function.)

In this proposal the simple, low-cost, credit-only end-station adapter must be forfeit. All adapters must be capable of the default rate mechanism and multiple control loops may coexist within the same set of nodes. The scheduling hardware for the rate mechanism is the most complex component of the adapter. The additional hardware required to support the credit approach is modest in comparison. If we accept that the adapter must be capable of rate control and should optionally be capable of credit control then the control scheme may be selected on a per-VC basis. If all entities on the path of the VC are capable of credit then credit control may be selected. If not the rate based default is used. This will permit the customer the choice of low-cost rate control switches in the LAN or high performance credit switches.

The virtual source/destination interface between the LAN and the WAN is still required if the credit option is selected on the local area portion of the connection. If credit-based control is only selected for connections that remain within the local area then wide area connections may use rate-based control throughout the entire connection. Thus the virtual source/destination need no longer be mandatory and no artificial boundary need be defined to separate LAN from WAN. However, a virtual source/destination function is still likely to be necessary at the boundary between a LAN and a public ATM service for traffic shaping and to assist in policing the ABR service.

For the adapter manufacturer the cost of adopting this compromise is that market pressure will probably force them to implement both schemes. For the switch manufacturer, the credit switch must implement a default rate scheme. Also congestion control protocol selection is required as part of the signalling process. While this does add additional complexity it allows multiple traffic management protocols to coexist in the same network. This is not necessarily a bad idea considering the speed at which the ABR congestion control scheme is being developed by the ATM Forum.

One Size Fits All

The third proposal is to use an encoding in the resource management (RM) cell to provide not only rate information in the cell specifying the rate that a VC can flow but also have a validity count field in the RM cell that is associated with the rate. The validity count field may be interpreted as the number of cells transmitted by a VC before the rate that it is currently using becomes invalid. At that point the source has to cease transmission on the VC until a new RM cell is received updating the rate to use and its validity. Another interpretation of the validity count field is that it is a time value, particularly suited for the end-end rate scheme. This would allow a much longer period for which the rate conveyed in an RM cell would be valid.

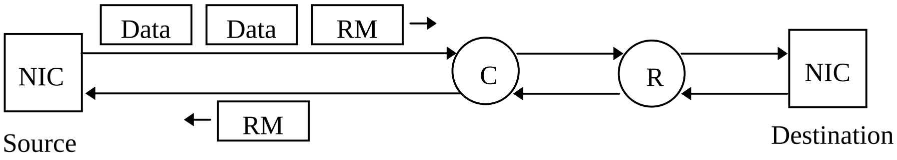

Figure 2: Interworking credit and rate switches.

With this proposal, a switch that is part of an ATM network may choose to participate in one or the other of the congestion control algorithms. Figure 2 illustrates this concept. “C” is a credit switch, and “R” is a rate switch. If the switch wants to participate in the end-end rate-based congestion management algorithm, it would mark the RM cell with an explicit rate R1. R1 may be based on its calculation of the desired rate for that VC (possibly an allocation based on the max-min fairness criterion [10]). Along with the rate, the switch would mark the RM cell with (potentially) a large value for the validity count field. The count may be set based on the period over which the switch re-computes the rate, or the maximum time it perceives is safe within which it has to see a new RM cell for communicating a new rate to the VC. This adds a degree of robustness to the algorithm to allow recovery from the loss of RM cells. The validity field may be a constant value, that is inserted in each of the RM cells communicated to the VCs. If necessary, depending on the conservativeness of the switch designer, this value may in fact be variable, set to smaller values as the switch gets more and more congested.

If the switch wants to participate in the hop-by-hop credit scheme, then the explicit rate R2 that is conveyed in the RM cell would be the peak rate assigned to the VC, and the validity count field would be a much smaller variable count that indicates to the upstream node the number of cells that it may send at the explicit rate R2. When the upstream node exhausts the count by transmitting R2 cells, it will then have to await a replenishment of credits from a downstream node, just as in the hop-by-hop credit scheme defined in [4].

As far as the end-station adapter Network Interface Card (NIC) is concerned, there is only one interface. The NIC transmits cells at the rate conveyed in an RM cell until the count runs out.

The end-end rate scheme operates as follows: The source of the VC indicates the desired rate in the RM cell. This RM cell is on the same VC, and is therefore allowed to flow all the way to the destination end-system. The destination NIC reflects the RM cell, with an indicator to show that the RM cell is now making progress in the reverse direction. The intermediate switches then mark down the rate (the explicit rate allocated to the VC) in the reverse RM cell. The smallest allocation is therefore the value in the RM cell when it reaches the source. The source may then use this rate for subsequent transmissions until a new RM cell is received.

The hop-by-hop credit scheme operates as follows: The source indicates the rate (peak rate, if it wants to set it to a value lower than the link rate) it desires to transmit at in the forward RM cell. The destination end-system NIC reflects the RM cell, and the switch then updates the count field in the reverse RM cell. The switch also needs the ability to generate additional RM cells on the local link so as to enable a more timely hop-by-hop indication of credits to the upstream node when necessary (e.g., to initiate data flow on a VC or to support dynamic buffer allocation).

This scheme permits rate and credit switches to be mixed within the local area without any special interface equipment. A rate switch can operate downstream from a credit switch because the RM cell contains the sum of the rate and credit control information. The source will not transmit at a rate greater than that permitted by the rate switch and will send no more cells than it is permitted by the credit switch. To support a credit switch downstream from a rate switch the credit switch must fill in the count field in the backward RM cell with the number of credits that is allowed for the flow, and then rate switch fills in the explicit rate field in the same backward RM cell. This way, the source is limited by both the rate at which the “rate-switch” can service the VC as well as the number of cells that the downstream credit switch is willing to buffer for that VC and ensure no cell loss.

In addition, to support a credit switch downstream from a rate switch the credit switch must fill in the explicit rate field in the backward RM cell. It would do this by computing the actual throughput of the VC within the credit switch and use this value to mark the RM cells. This is particularly important when the outbound link of the credit switch is the bottleneck. This rate is fed back to the source in the RM cell, to minimize cell loss. It is also likely that the credit switch may require larger per-VC buffers than a pure static credit switch to cope with the additional delay through the rate switch. Thus, the network looks like a rate subnet up to the input to the credit switch, and as a credit subnet downstream from that switch. This requires additional hardware to that of a pure credit switch but seems a small price to pay for an integrated scheme.

Implementation Issues

The choice among different congestion management mechanisms influences the implementations in a significant way. In the past, with lower speed networks, the issue of implementation of the congestion management algorithms was somewhat less important, since much of it was implemented in software. However, because ATM is meant to be scalable to much higher speeds, the implementation of a significant part of the congestion management algorithms needs to be performed in hardware, both in switches and in the end systems.

The implementation of the two ATM congestion control schemes has been viewed to have widely varying complexity. To a large extent this is due to the differing perspectives of the designers of the switches versus those implementing the end system. It is important to consider the implementation of the scheme from a complete system perspective. Addressing the implementation complexity from an entire system perspective is not often feasible in a standards environment, since representatives tend to focus on one or the other depending on their particular “corporate” emphasis.

Switch Implementation Issues

Switches are relatively expensive to design and build. There is a wide range of switch capacities, with small LAN switches having a small number of ports and a relatively small maximum number of VCs supported per port, to large WAN carrier switches that may have a large number of ports and also potentially a large maximum number of VCs supported per port. There is also the fact that a large number of switch designs are in existence and there is an existing base of deployed switches. This influences what we can accomplish with the introduction of a new congestion management algorithm into an environment that has already deployed older switch designs.

One of the major issues for switch designs is that of buffering, and where it is located relative to the switching function. There are three major ways buffers are placed in switches [11, 12]:

- Input buffered switches.
- Output buffered switches.
- Shared memory switches.

With a hop-by-hop credit scheme, the perception has been that an input buffered switch is most suitable. This is because the occupancy of the buffer is sensed, and when a cell is forwarded from the buffer, is quickly available and can be communicated to the upstream node in a timely fashion. However, even in such a switch, there is the need for feedback across the switch fabric to communicate the credits received from the upstream node. In the output buffered switch, there is the need to communicate credit information across the switch fabric back to the upstream node. In either case, the need to pass credit information across the fabric to the point of control is unavoidable.

The implementation of the hop-by-hop credit scheme in the switch involves a reasonable amount of complexity. This is to maintain state on a per-VC basis, at each port (input or output) and the need to recognize that it is time to communicate credits by transmitting a resource management (RM) cell when a threshold has been reached [4, 5]. In addition, the switching function has to recognize that a credit is available for a VC before the cell is forwarded.

The implementation of the end-end rate scheme has a range of complexity, depending on the particular rate scheme adopted. With almost all of the schemes proposed, there has to be some form of monitoring of the buffer occupancy to determine if the switch or switch links are congested. With the simple EFCI scheme [7, 13], the ability to indicate congestion in the cell header is needed, which requires a small amount of hardware support. However, this function is already part of the existing standards and is implemented in current switches. With the schemes which require indication of congestion in an RM cell, there is the added complexity of receiving and processing the RM cells.

The significant complexity arrives when an explicit rate computation has to be performed for each VC, and the rate has to be communicated in an RM cell by the switch [14]. In this, a fair allocation of the capacity of the bottleneck capacity is provided to each VC. The fair allocation is based on a criterion termed as "max-min" fairness. The need for allocation of capacity based on a max-min fair allocation has been extensively studied in the recent past [10, 15, 16]. The notion is to provide all VCs that have a “low” demand of the capacity of a resource their entire requirement. The VCs that have a “higher” demand are then provided at least an equal share of the left over bandwidth. This allows VCs that have been bottlenecked elsewhere to not be further limited by this switch, and the VCs for which this switch is the bottleneck are the ones that will be bottlenecked. In informal terms, a switch is supposed to not limit a VC to a rate that is lower than it’s demand if there are other VCs that are flowing through this switch which have a higher allocation of bandwidth. A straightforward implementation of the explicit rate scheme requires state to be maintained for each VC, as to the rate that has been allocated for the VC. A further modified proposal [17] for explicit rate based congestion attempts to estimate the fair allocation of the capacity for a VC by performing an exponentially weighted averaging of the rates seen from each of the VCs. In this way, there is no need to maintain the explicit rates for each of the VCs, and instead only an estimate of the fair share. A comparison has to be performed between the average rate (which is an approximation of the max-min fair share) and the requested rate by the source of the VC, that is indicated in the RM cell. Furthermore, the implementation needs to be able to capture an RM cell flowing in the reverse direction and write the allocated rate. There is a need to perform a small amount of arithmetic in the switch as well.

The most significant issue related to switch implementation complexity lies in the fact that the hop-by-hop per-VC based credit scheme requires buffering/queueing on a per-VC basis. This allows each of the VCs to flow independently. If there was a common queue for all the VCs, this potentially results in deadlocks. With a single FIFO queue for all VCs, when a VC is bottlenecked downstream, this results in credits not being issued to the upstream node. This can cause other VCs that may potentially take a different path to also be blocked at the upstream node. Not only does this cause unfairness, but it is relatively easy to arrive at topologies and workloads in which deadlocks may arise. One of the ways this is avoided is to use routing that is “deadlock-free”. However, a better approach that not only solves this problem but also provides considerable benefits in other ways is to provide queueing on a per-VC basis. This implies that the buffer (e.g., a buffer for an input port) may now be allocated on a per-VC basis (i.e., the accounting of the occupancy of the buffer is performed on an individual VC basis). This results in additional complexity in the switch, which needs to be considered in the early stages of the switch design. With an end-end rate based congestion control scheme, there is no strict dependency on the availability of per-VC queueing in the switches. Thus, the complexity of the switch due to queueing and buffering is considerably reduced.

The queueing on a per-VC basis has additional desirable characteristics that have been propounded in the past in the context of Fair Queueing for congestion management [8]. Having a separate queue for each of the VCs allows the scheduling of switching cells from each VC in a fair manner. This also allows the delay and loss behavior of individual VCs to be isolated from each other. When there is a single common queue for all the VCs, this can lead to potentially undesirable interactions between different flows. For example, a VC that has a small amount of data to transfer may have its cells queued behind a large burst of cells from another VC that in fact may be going through a more severe bottleneck downstream. This form of transient “head of the line” blocking results in unpredictable interaction between the VCs. At a high level, the fact that cells rather than larger packets are switched mitigates the effect of this interaction. But, the notion of an ABR service where a VC may flow at the peak rate still allows for a large burst of cells from a VC being queued ahead of cells for a latency sensitive VC. A per-VC queue with a suitably fair (e.g., weighted round-robin) service policy allows for controlling latency better.

In the future, we may envisage being able to specify at least weak bounds on the end-end latency experienced by a cell/packet. Specification of the latency bounds is more easily accomplished when we have a per-VC queue with a fair per-VC service policy [18]. While this notion of specifying delay bounds is strictly not envisaged for ABR service, it may still be desirable in the future for the incorporation of weak real-time traffic (e.g., video teleconferencing). In any case, we clearly see the need for separating the flows from different VCs on a class by class basis in a switch. This requires implementation support for recognizing the class a VC belongs to and queueing and servicing in accordance with the class characteristics. The incremental complexity of per-VC queueing and service may not be particularly significant beyond the existing cost of per service-class queueing.

Many of the first generation switches were implemented with discrete FIFO queues which made per-VC queueing impossible and in general two or at most four per service-class queues were offered. In more recent designs custom silicon is being employed to reduce cost and increase capacity and functionality. Frequently the queueing function is now implemented with custom silicon and static RAM permitting queues to be implemented as linked lists of cells. With this implementation the additional cost of per-VC queueing over per service-class queueing is very low and the flexibility of the scheduler becomes the more significant differentiator. Some advanced designs are already considering incorporating traffic shaping within the main switch queueing function by implementing complex scheduling capabilities within custom silicon.

While most shared memory designs implement queueing with linked lists of cells, all of the queues are implemented in a single, centralized shared memory. This makes it difficult for such designs to offer per-VC queueing or advanced scheduling due to the large number of VCs passing across the center of the switch.

Input or output buffered designs are more likely to offer per-VC queueing than shared memory designs.

Adapter Design Issues

The most cost-sensitive network component for an ATM network appears be the end station adapter card. The switch port is cost sensitive but there is more opportunity for sharing functions across a number of ports. The adapter stands alone. The cost of an adapter is very often determined by the amount of memory used. For ATM, the use of reassembly buffering is an important reason for adding memory on the adapter, in contrast to other technologies. The smaller this is, the more competitive ATM technology would be relative to Fast Ethernet, FDDI, etc.

On the receive side of the ATM adapter, the amount of buffering needed for reassembly in the most simplistic case, without taking advantage of statistical multiplexing is given by: { number of active VCs × max. packet size}. This can be excessive. However, statistical multiplexing arguments typically suggest that the amount of buffering needed is much smaller than this. The downside of this reduced buffering is that when the buffers fill up, because there are too many packets undergoing simultaneous reassembly, we can encounter significant packet loss. In fact, with cheaper adapter designs, we may see that the primary point where loss occurs is in the receiving end system, rather than at the more expensive switches which can be designed to have more buffering.

With a credit based adapter, we can choose to provide credits to a selected number of VCs to enable reassembly, while allowing the other VCs to more wisely use the buffering at the source end system or in the intermediate switches. This is especially true when static buffer allocation schemes are used. It allows us to bring down the buffering requirement at the receive side of an adapter to “almost arbitrarily” small levels, so that the cost of building an ATM adapter is not substantially more than that of building a Fast Ethernet or FDDI adapter. One must note that the added cost of an ATM switch port for every connection into an ATM network (in contrast to Fast Ethernet, for example) encourages us to make the cost of an ATM adapter as small as possible. Therefore, we see using a credit based scheme allows us to drive down the cost of building an ATM adapter.

On the transmit side, SAR chips often will queue packets awaiting segmentation on a per-connection basis. The simplest possible scheduler is a list of connections to serve that have packets queued for segmentation. The simplest enhancement of this scheduler that is capable of implementing an ABR service is to queue only those connections that have permission to send. This how the scheduler for the credit scheme is implemented.

For a rate-based scheme each connection needs to be transmitted at its own particular rate. This is the same problem as traffic shaping. A scheduler capable of supporting a reasonable number of connections for the rate-based scheme is certainly more complex to implement than one for the credit scheme. The rate-based scheme in effect requires both the SAR function and a traffic shaper function.

Summary

We have suggested a means of integrating rate- and credit-based control for ABR traffic management. Why? Because at the time the ATM Forum was due to vote to select a single control mechanism, both the rate- and dynamic credit-based proposals were still under development. At that time it was not clear that one could select between rate and credit as a fundamental approach in the absence of a stable and implementable mechanism.

Static credit control was simple in concept and could be proven to work. It would satisfy the most stringent demands of the local area both now and into the future. It also offered the simplest and lowest cost adapter cards for ABR service. The local area market is a fast moving market, it was ready for ATM product, and it required a stable traffic management scheme rather than one that was still under development.

However, a static credit scheme is not suitable for the wide area. The wide area requires a rate scheme for both technical and political reasons, not the least of these being that that is what the carriers want. Carriers need to provision virtual circuits, and tariff and police on the rate at which the VC is sending information. The wide area market moves more slowly and the same stringent performance may not be required in the wide area as for the local area. There seemed to us less of an immediate need for ABR service in the wide area so it appeared there was more time to develop a rate scheme for the wide area. By decoupling the schemes for the local and wide areas there was an opportunity to enhance the wide area scheme over time.

Due to these differing requirements we felt that one cannot mandate a single congestion control mechanism for all application areas for all time. There is also a desirable goal of being able to support future development. So, it seemed that if we could show how these schemes might interwork, it would be reasonable to permit a choice of control schemes in the local area. This was the basis for the integrated proposal, described here. This proposal appeared particularly attractive as the different congestion control schemes being combined offered different optimizations of cost and performance.

The ATM Forum has chosen to use a rate scheme for traffic management of the ABR service. Since that decision, considerable development has taken place on the end-end rate scheme, most significantly the inclusion of an explicit rate control capability as an option and the focus on congestion avoidance techniques within the switch. Work continues on the scheme, and the explicit rate enhancements appear to be addressing the requirements of the local area.

References

[1] P. Newman, “Traffic management for ATM local area networks,” IEEE Commun. Mag. Aug. 1994, pp. 44–50.

[2] ATM Forum, “ATM user-network interface specification,” Version 3.0, Prentice-Hall, Sep 1993.

[3] L. Kleinrock, “The latency/bandwidth tradeoff in gigabit networks,” IEEE Commun. Mag. Apr. 1992, pp. 36–40.

[4] H. T. Kung, “The FCVC (flow-controlled virtual channels) proposal for ATM networks,” Proc. Int. Conf. Network Protocols, San Francisco, Oct. 1993.

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

[6] H. T. Kung and R. Morris, “Credit-based flow control for ATM networks,” IEEE Network Mag., Mar. 1995, pp. 40–48.

[7] K. K. Ramakrishnan, R. Jain, “A Binary Feedback Scheme for Congestion Avoidance in Computer Networks,” ACM Transactions on Computer Systems, May 1990.

[8] A. Demers, S. Keshav, S. Shenker, Xerox PARC, “Analysis and Simulation of a Fair Queuing Algorithm,” Proc. ACM Sigcomm Symposium, 1989.

[9] D. Hunt et. al., “Flow Controlled Virtual Connections for ATM Traffic Management,” ATM Forum contribution 94-0632R2, Ottawa, Canada, Sep. 1994.

[10] D. Bertsekas and R. Gallagher, “Data Networks,” Prentice Hall, Englewood Cliffs, N.J., 1987.

[11] P. Newman, “ATM technology for corporate networks,” IEEE Commun. Mag. Apr. 1992, pp. 90–101.

[12] R. Rooholamini, V Cherkassy, M Garver, “Finding the right ATM switch for the market,” IEEE Computer Mag. Apr. 1994, pp. 16–28.

[13] M. Hluchyj et. al., “Closed-Loop Rate-Based Traffic Management,” ATM Forum contribution 94-0438R2, Ottawa, Canada, Sep. 1994.

[14] A. Charny, “An Algorithm for Rate Allocation in a Packet Switching Network with Feedback,” MIT Thesis, MIT/LCS TR-601, Apr. 1994.

[15] K. K. Ramakrishnan, D.-M. Chiu, R. Jain, “Congestion Avoidance in Computer Networks with a Connectionless Network Layer. Part IV: A Selective Binary Feedback Scheme for General Topologies,” (DEC TR-510, November 1987).

[16] S. Shenker, “A Theoretical Analysis of Feedback Flow Control,” Proc. ACM Sigcomm Symposium, 1990.

[17] L. Roberts, “Enhanced Proportional Rate Control Algorithm,” ATM Forum contribution 94-0735R1, Ottawa, Canada, Sep. 1994.

[18] A. Parekh, “A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks,” MIT Thesis, MIT/LIDS-TH 2089, Feb. 1992.