INTERNATIONAL JOURNAL OF DIGITAL AND ANALOG COMMUNICATION SYSTEMS, VOL. 3, 199–209 (1990)

# AN APPROACH TO CONTROLLING CONGESTION IN ATM NETWORKS

A. E. ECKBERG, JR., D. T. LUAN AND D. M. LUCANTONI
*AT&T Bell Laboratories, Crawfords Corner Road, Holmdel, New Jersey 07733-1988, U.S.A.*

## SUMMARY

Determining approaches to congestion and flow control, especially real-time components in an overall strategy, is recognized as one of the fundamental challenges facing broadband 'packet-based' information transport, as, for instance, in the case of BISDN/ATM. In this paper we summarize basic issues underlying this subject, and describe a particular approach to achieving a multilayer broadband congestion, flow and error-control architecture, based on a 'core' congestion control strategy that we term *bandwidth management*. The modular and layered nature of this control architecture is described, and shown to lend itself to a structured approach to characterizing the control architecture performance.

KEY WORDS Traffic monitoring Violation tag Control layering Control separation Leaky bucket

## 1. INTRODUCTION

Broadband ISDN (BISDN) architectures, built on a technology base of information transport via asynchronous transfer mode (ATM),[1,2] offer a means for the efficient integrated transport for a broad spectrum of services, from voice and low-speed data to high-speed data and video telephony. Although well-thought-out engineering and administration rules will probably result in excellent end-to-end performance in such networks most of the time, a fundamental challenge in the design of robust BISDN architectures is the structuring of a comprehensive strategy for congestion and flow controls, which will be essential during those (hopefully rare) instances of resource overloads due to unforeseen traffic focusing and peaking, as well as during network component failures. Unfortunately, the multi-megabit-per-second speeds that will be responsible for the high performance/capacity of BISDN/ATM networks will also severely limit the number of implementable and stable congestion control strategies.

In this paper we address a number of basic issues in the area of congestion and flow control strategies suitable for BISDN/ATM, and suggest a specific BISDN/ATM congestion/flow/error-control architecture. We begin, in Section 2, by summarizing goals, issues, and several guiding principles that underlie the selection of such a control architecture. Section 3 describes a *bandwidth management*-based congestion/flow/error control architecture that is motivated by some of the considerations laid out in Section 2. Then, Section 4 describes a modular approach to achieving a traffic/performance characterization of this control architecture. Finally, in Section 5 we address some commonly asked questions regarding the proposed congestion control strategy.

## 2. UNDERLYING ASSUMPTIONS, MAJOR ISSUES AND GUIDING PRINCIPLES

There are two, fundamentally different, approaches to determining control architectures for complex systems such as BISDN/ATM networks: an *inside-out* approach in which each component in an overall architecture is looked at individually, without excessive attention given to component interworking, and an *outside-in* approach, in which a global, 'system' view is taken of the entire system. It is the inside-out approach which typically must be taken during experimentation in technology advancement and component prototyping, in particular to determine component functionalities that are feasible to implement. However, at an appropriate point in technology development it becomes necesary to switch to an outside-in approach, so that the entire system can be characterized as a single entity, and so that system-wide strategies that may make global sense are not unnecessarily prevented by individual component limitations.

It is an outside-in approach which we attempt to take in this paper focusing on issues and characteristics of entire BISDN/ATM information transport systems. Therefore, in this section we list some of the assumptions we are making concerning the characteristics of BISDN/ATM networks and the information movement and management environments they will be supporting. These assumptions then lead to a number of high-level objectives and

1047–9627/90/020199–11$05.50


*Received 5 January 1990*
*Revised 19 April 1990*

guiding principles for a BISDN/ATM congestion/flow/error control architecture.

### 2.1. *Assumptions*

We first clarify the framework underlying the control architecture by listing several key assumptions:

(i) We restrict attention to BISDN/ATM networks, and in particular, the *connection-oriented* information transport service they will provide.* In generic terms, we think of such networks as consisting of network 'nodes' and a network 'edge', where the latter provides the BISDN/ATM interface between the 'network' and ATM end-devices. We specifically include these ATM end-devices within the definition of the 'BISDN/ATM information transport system', because there is a fundamental role such end-devices will play in an overall BISDN/ATM control architecture, particularly in the area of flow and error controls. These ATM end-devices may be end-terminals that handle the ATM protocol themselves, or terminal adaptors (TAs) allowing end-terminals associated with a variety of non-ATM services and native protocols to adapt to the common ATM transport. See Figure 1 for a depiction of this ATM networking environment.

(ii) For the scope of this paper, we will not concern ourselves with control issues associated with interconnected networks.

(iii) We specifically assume that the BISDN/ATM network will provide the integrated transport for a variety of different services, only some of which are data services. Moreover, there is a great uncertainty associated with many of these services and applications, particularly with regard to their traffic characteristics and their performance needs (e.g. low-delay or low-loss); see Figure 1. This uncertainty affects the desirable characteristics of a control architecture, since it points to the need for considerable robustness and flexibility in accommodating diverse services, some of which may not even be conceived of until they have been stimulated by the existence of BISDNs.

(iv) We also point out that ATM end-devices cannot be assumed to be under the total control of the network, particularly when the network is a public, shared one. One implication of this observation is that the network cannot assume a 'friendly community' of end-devices, and in particular, cannot assume that individual end-device actions will be in the best interest of the network and other end-devices. Even if end-device algorithms were to be completely standardized, there is still the possibility that one or more end-devices would not conform to the standards in certain critical and difficult-to-detect areas.

*This does not preclude support of a network service which appears to be connectionless at higher protocol layers.

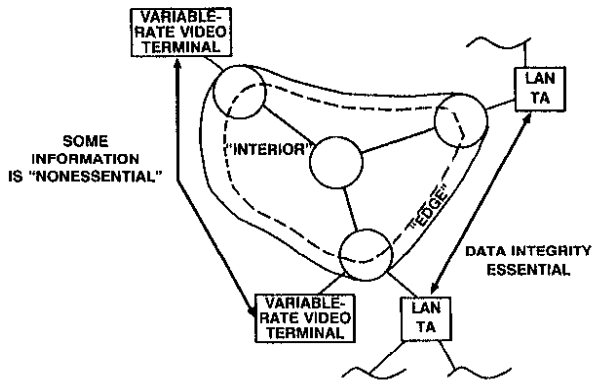



Figure 1. A BISDN/ATM environment

### 2.2. *Major issues in the design of broadband control architectures*

A basic characteristic of packet-based information transport, such as with BISDN/ATM, is the sharing of critical network resources (memory, processor real-time, transmission bandwidth) between large numbers of information flows. Controls are needed to protect such shared resources from depletion and to allocate them fairly, and the difficulty in achieving effective controls generally increases with increasing transport speeds. At broadband speeds three factors become particularly notable:

(i) The level of control algorithm complexity that can be implemented in individual devices is limited due to the costs associated with specialized high-speed hardware.

(ii) The effectiveness of sharing explicit control and status information between the various ATM cell-handling devices along a transport path, to trigger continuous, real-time control actions, is limited by large propagation-delay-to-cell-queue-time-constant ratios.†

(iii) Complete transport integrity (without reliance on suitable recovery mechanisms, e.g. at the ATM Adaptation Layer) can become a difficult goal in many settings, since large and expensive buffering memories may be needed to eliminate ATM cell losses in the network; but such large buffering

†For example the transmission time for a 53-octet ATM cell is approximately 2·8 μs at 150 MBPS, and the resulting time-constant in a cell queue will be of the order of tens of microseconds for moderate utilizations, whereas the end-to-end signal propagation delay through fibre for a 1000 km connection is approximately 5 ms.

memories would add significantly to end-to-end delays.

This last factor implies that ATM cell loss during broadband transport is a phenomenon that may be difficult to 'design away',‡ and thus it becomes important that error control strategies be derived jointly with congestion and flow control strategies.

### 2.3. *Guiding principles in the design of a broadband control architecture*

The above-mentioned assumptions and issues motivate the following principles to guide the design of broadband congestion/flow/error control architectures:

(i) The control architecture should be *flexible*, capable of accommodating the characteristics and requirements of yet-to-be-determined mixes of services and applications.

(ii) The control architecture should be *robust*, in that its operation and resulting performance should not depend strongly on assumptions concerning the traffic characteristics of services, the co-operativeness of end-devices or even basic transport characteristics of the network (actual buffering capacities, propagation delays, etc.). In particular, we feel that systems of controls that are tightly coupled between the network and end-devices tend to be intrinsically non-robust; and that therefore if loosely-coupled, distributed controls can be designed, such controls may exhibit considerably more robustness.

(iii) The control architecture should be based on *as simple as possible* control actions, in order to increase the chances of implementability at broadband speeds, and to result in an intuitively understandable overall system.

(iv) The control architecture should be separable into three distinct domains, each with its own set of objectives:

  (a) a set of network congestion controls, allowing the network to protect its shared resources and to allocate them 'fairly' in accordance with resource-demand agreements between the network and end-devices (bandwidth should be allocated in proportion to agreed to bandwidth parameters); moreover, this protection and fair allocation should not depend on end-device actions

  (b) flow control actions (e.g protocol window reduction and/or 'traffic-shaping') taken by, and under the control of, the end-devices; such actions should be motivated by end-device self-interest, but should result in actions for the common good

  (c) error control actions, such as forward error correction,[3] retransmission, or source coding to allow cell-dropping,[4–6] that are synergistic to the congestion and flow controls.

Figure 2 depicts this desirable separation of network congestion controls from end-device flow and error controls.

(v) The control architecture should be 'layered', with simple real-time, distributed 'core' actions, overlaid by call-level actions acting over longer time scales. This layering principle is also depicted in Figure 2.

## 3. CONGESTION, FLOW AND ERROR CONTROL STRATEGIES BASED ON BANDWIDTH MANAGEMENT

This section provides a motivation and description of congestion flow and error control strategies based on *bandwidth management* (BWM). We begin with a summary of the BWM 'core' network congestion controls themselves, then describe call-level controls needed as an overlay to the BWM 'core', and finally observe how flow and error controls implemented in end-devices or adaptors could be structured to complement the network congestion controls. It can be noted that the overall control framework described in this section is similar to ideas in other recent papers.[7–9] Moreover, the BWM core controls have been implemented in wideband packet technology devices.[10]

### 3.1. *The BWM core network congestion control*

BWM is motivated by three principal control objectives: (i) *protection of shared network resources*, (ii) *fairness* and (iii) *robustness*; and by two principal implementation objectives: (i) *simplicity* and (ii) *flexibility*. These five objectives reflect the end goal of achieving a manageable control strategy with

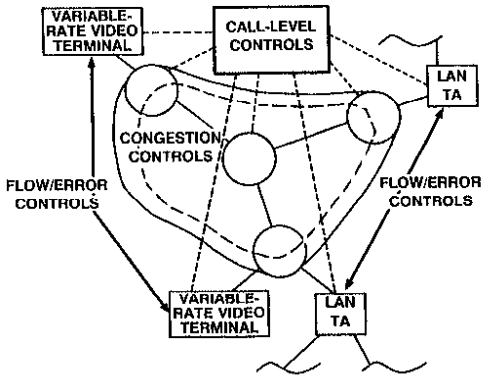



Figure 2. A BISDN/ATM environment with a layered/separated control structure

‡In fact, *selective packet discard* can form the basis of a robust congestion control.

good performance. The BWM approach to achieving resource protection is to rely on congestion relief through cell discard; and fairness is achieved with BWM by making the cell discard process *selective*, based on traffic agreements and real-time traffic monitoring. The simplicity objective is approached through the primary role played by selective cell discard (an intrinsically simple operation), and through a *distributed control structure* that involves local control actions, and no real-time co-ordination between network components, or between the network and end devices. Flexibility is achieved through the explicit role of traffic agreements and monitoring. Robustness is approached through non-real-time-intensive couplings between the BWM 'core controls' and 'call-level' controls.

The BWM core controls are applied to the cell flow of a virtual circuit (VC), and are based on a set of traffic parameters, established at VC set-up for each of the VC end-devices, defining the characteristics of traffic from that end-device that the network is willing to support. These traffic parameters are directly related to a method for monitoring traffic in real-time and establish a traffic throughput level and degree of burstiness that the network must transport with a high degree of integrity. The end-device may exceed this traffic agreement, but then should not expect complete transport integrity. The core controls are as follows:

(i) For the duration of the VC, traffic entering the network from each end device is monitored in real-time at the network edge; and through such real-time monitoring, traffic that falls within the agree-to traffic parameters is discriminated from the remaining traffic (the *excessive* traffic), if any. Cells corresponding to excessive traffic are then marked with a *violation tag* in their header, indicating that they should be tranported, but at the risk of possible discard should these excessive-traffic cells encounter congestion along the VC route.

(ii) Moreover, 'pre-marking' of cells with a violation tag at an end device can also be permitted. The monitoring of traffic from the end device is not influenced by such 'pre-marked' cells; rather, these will be ignored in the traffic monitoring and made to retain their original violation tag. This allows an end-device flexibility in designating those cells it may wish to have transported 'at risk', but monitoring is still always performed at the network edge for protection. For example, voice and video end-devices could pre-mark non-essential cells.

(iii) Violation-tagged cells are discarded whenever they encounter moderate congestion at any ATM cell queue along the virtual circuit, so that the presence of excessive-traffic cells causes only negligible impact to the delays, and virtually no impact to the loss, of cells not carrying a violation tag.

It can be seen that the BWM core controls involve simple and distributed operations. In particular, no explicit passing of congestion status information between ATM cell-handling devices along the VC is involved; rather, simple local actions are taken in response to the sizes of local ATM cell queues. Once initialized by the establishment of traffic monitoring parameters, they are 'free-running' controls. Yet, cells are discarded only when congestion relief is needed, thus conceptually 'closing the loop' between the detection of congestion and a control action to help relieve congestion in real-time.

The distribution of control for this set of BWM core network congestion controls, with traffic monitoring and violation-tagging at the edge and selective cell discard in the interior, is depicted in Figure 3.

3.1.1. *Violation-tagging vs. policing.* Note that whenever traffic is monitored in real-time at a network edge, there are two general classes of control actions possible when 'excessive' traffic has been detected:

(i) policing, where the network unconditionally discards the excessive traffic after its detection

(ii) traffic violation-tagging, where the excessive traffic is identified and marked, to be discarded later in the network only if congestion is encountered

and, in fact, both of these control actions are under consideration as elements in an overall control architecture. The BWM core congestion controls are based on violation-tagging, rather than policing, for the following reason.

Policing tends to be a 'hard' control action, whereas violation-tagging is 'softer', in that, depending on the network congestion conditions, very little violation-tagged traffic may actually be discarded. Because the eventual fate of 'excessive' traffic depends on which of these control actions is to be taken, one can expect that different objectives would be placed on the accuracy of the traffic monitoring,

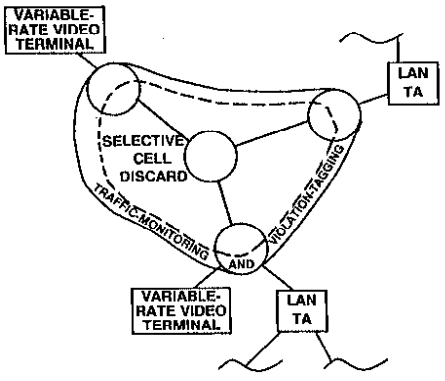



Figure 3. Bandwidth management actions in the network

depending on whether a policing or a violation-tagging approach is taken. In particular, because violation-tagging is 'softer', one can afford to incorrectly identify 'non-excessive' traffic as 'excessive' more frequently.

In fact, because a set of traffic parameters does not unambiguously define the demarcation between allowable and excessive traffic, any traffic monitoring scheme will be characterized by the following attributes:

(i) the scheme's responsiveness, meaning how rapidly changes in traffic characteristics into the 'excessive traffic domain' can be detected
(ii) the scheme's probability of false alarm, meaning with what probability segments of stationary, and non-excessive, traffic will be incorrectly identified as 'excessive'
(iii) the amount of 'margin' that the scheme requires (because we would like a monitoring scheme to be responsive, but not exhibit a high probability of false alarm, there is a 'margin' that a scheme must allow in traffic above and beyond the agreed-to traffic characteristics; this 'margin' can potentially be exploited by a sophisticated and malicious end-device so as to send traffic consistently at an excessive rate, but such that none of the traffic is identified as excessive).

One could want a monitoring scheme to be responsive, to have a low probability of false alarm, and to need only a modest 'margin'. Clearly, these three performance parameters are involved in basic performance trade-offs, and it would be impossible to simultaneously achieve very small response time, very small probability of false alarm, and a very small margin. For example, to achieve a very small probability of false alarm with a very small margin would require considerable traffic averaging over long intervals, and this would result in poor responsiveness.

Because of these considerations, it is preferable to take the 'softer' action of violation-tagging, rather than the 'harder' action of policing, against traffic that is taken to be 'excessive'.

3.1.2. *A specific traffic monitoring scheme.* Many traffic monitoring schemes have been suggested, but one that exhibits both simplicity of implementation and monitoring performance (in terms of the responsiveness, false alarm probability, and margin performance parameters described above) is the 'leaky-bucket' monitoring algorithm.[11, 12] There are numerous variants of the leaky-bucket monitoring algorithm, but a typical one involves maintaining a counter with simple operations at successive cell arrivals on a VC as follows:

(i) The counter value, $X$, is initialized to 0.
(ii) At each new cell arrival, the counter value is first decremented as $X \leftarrow \max(X-cT,0)$, where $c$ is a specified parameter, and $T$ is the elapsed time since the last cell arrival (but, as indicated, $X$ is not decremented below 0).
(iii) The counter value $X$ is then compared with a prescribed value $M$ and
  (a) if $X \geq M$, the cell is violation-tagged and passed on
  (b) if $X < M$, the cell is passed untagged, and the counter is incremented as $X \leftarrow X+1$.

This algorithm allows a sustained, untagged, throughput rate of $c$ cells per time unit, and a burstiness that is controllable (for a given value of $c$) through the selection of the value of $M$. One can envision the leaky-bucket algorithm as a *throughput-burstiness filter* applied to the cell flow entering the network from an end device on a VC. Note that the operation of the leaky-bucket is identical to that of a finite-capacity single-server queue with a deterministic service time. Figure 4 depicts the operation of this monitoring algorithm.

3.1.3. *Specification of traffic parameters.* The final aspect of the BWM core controls that merits more discussion is the specification of traffic parameters with respect to which, and with a monitoring algorithm, discrimination between 'excessive' and 'allowable' traffic is to be effected. As mentioned above, no monitoring scheme can simultaneously achieve optimum performance for all three of the performance parameters: responsiveness, false alarm probability and margin. Moreover, the difficulty of achieving good monitoring performance is increased by the ambiguities inherent in the use of a finite set of traffic characteristics (e.g. average rate, peak rate, 'burstiness') to accurately distinguish between excessive and non-excessive traffic. We are then led to the conclusion that to minimize such ambiguity the best approach would be for the network and individual end-devices (whose traffic is to be monitored) to use the traffic monitoring algorithm

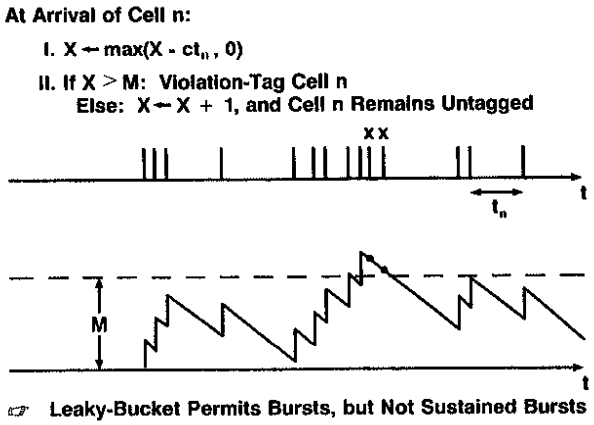



Figure 4. Example of traffic-monitoring via a leaky-bucket

(and its specific set of parameters) itself as the basis for defining allowable (i.e. non-excessive) traffic characteristics. Thus, for the leaky-bucket algorithm, allowable traffic would be umambiguously specified in terms of the parameters $M$ and $c$.

### 3.2. *BWM call-level controls to overlay the BWM core*

As noted above, the BWM core controls are simple, distributed controls. They can be expected to provide the necessary resource protection and fairness when allowed to 'free-run' over significant time intervals (i.e. 'significant' with respect to characteristic time-constants of the network). However, they require initialization, and also may require minor 'tuning', although not very frequently. This is the purpose of the 'call-level' controls, which provide an additional 'closed-loop' aspect to the overall network congestion control strategy.

In particular, at each VC set-up there needs to be a negotiation to establish the leaky-bucket parameters for the traffic monitoring/tagging process. This entails determination by the network whether or not a requested set of traffic parameters can be supported, and thus whether the VC set-up should be accepted, denied or modified. This determination could depend in part on a quantitative prediction of the throughput and burstiness of the traffic that will be allowed to pass without a violation tag, if a particular set of traffic parameters is allowed. Thus, a traffic characterization of the throughput-burstiness filter is a building-block for the VC acceptance/denial/modification process.

Other operations that need to be performed at the call level are as follows. Unless VC set-up requests are handled very conservatively, there is always the chance that too much non-violation-tagged traffic will have been permitted, and that consequently network resources are being stressed. If such an event occurs, it will be necessary to invoke call-level controls to reduce the amount of non-violation-tagged traffic. Possible control actions include tear-down of selected VCs, renegotiation of traffic parameters on existing VCs, or simply denying all new VC set-up requests for a fixed period of time.

### 3.3. *End-device flow and error controls to complement the BWM core*

As was noted earlier, whereas with BWM the network-implemented congestion controls require no explicit coupling with end-device-implemented flow and error controls, the *strategies* underlying these controls need to be co-ordinated and consistent. This co-ordination could be very simple, or could be quite sophisticated, depending on the degree to which an end device wishes to exploit the transport characteristics that BWM provides. In particular, it may be noted that a VC through a BWM-controlled network has the appearance of a 'statistical VC', with certain fixed characteristics such as 'guaranteed' throughputs and burstiness that will be supported, and other statistical/dynamical characteristics (that change with network congestion) relating to how much excessive traffic can be sent effectively, albeit always at risk.

An end-device may choose to implement flow controls which tend to keep traffic at levels where only a negligible fraction of its cells are discarded. This could be accomplished at the end-device by 'shaping' its traffic, e.g. via a leaky-bucket algorithm and buffering or other flow control schemes, so that none of the cells will be violation-tagged by the network.

Alternatively, the end device may maintain a handshaking process, at the ATM adaptation layer or a higher layer, with its peer device, to determine to what extent its excessive traffic is getting through. Note that if sending excessive traffic is to be attempted, this traffic must either be considered 'expendable', or it must be protected via end-to-end error controls. When excessive traffic is being discarded with a high enough likelihood, the end-device can then apply self-throttling or traffic-shaping methods to reduce the amount of excessive traffic, and thus reduce or eliminate the amount of cell discard. Such adaptive techniques for flow and error control could, for instance, be obtained via adaptive windowing when suitable protocol windows apply. Other possible approaches for recovering from lost cells are simple forward error correction schemes, e.g. as described in Reference 3, or selective 'frame-based' retransmission schemes possibly with adaptive frame sizes as described in Reference 13.

Some of the possible end-to-end flow/error controls that are synergistic with the BWM core network congestion controls are suggested in Figure 5.

### 3.4. *An example of BWM-based congestion/flow/error controls: LAN-To-LAN interconnect*

We now sketch through a hypothetical networking application that illustrates the possible operation of

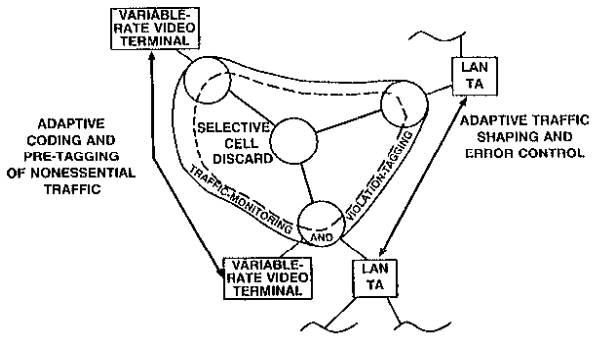



Figure 5. End-device complementary flow error controls

the modules in the BWM-based control architecture. Consider a BISDN network with ATM transport, and suppose that several of the end devices, connected via a set of VCs, are Ethernet LAN (ELAN) terminal adaptors (TAs). We assume that each of these TAs supports a number of 10 Mb/s ELANs, and that each TA has a 150 Mb/s access link to the BISDN network. Traffic from one ELAN to another ELAN not served by the same TA will be carried by a unique VC associated with that ELAN pair.

We assume that traffic presented by an ELAN for transport on a VC originates as ordinary Ethernet frames, which are variable in size, but no larger than 1518 octets, and also that the average ELAN-to-ELAN traffic on a given VC is 1 MBPS (a fraction of the 10 Mb/s total ELAN bandwidth). We also assume an ATM adaptation layer (AAL) structure which segments an Ethernet frame into ATM cells in such a way that the 48 octet cell payload is partitioned into 44 octets to carry user information plus 4 octets to carry AAL control information. The average rate at which information-carrying cells must be transported on a VC is then 2·84 cells/ms, although they originate in bursts of size up to 35 (for the maximum 1518 octets in an Ethernet frame). Finally, we also assume a flow of AAL-control-containing cells (the purpose of which will not be dealt with here) equal to 10 per cent of the flow of information-carrying cells, or 0·28 cells per millisecond.

Moreover, we assume that the network applies BWM controls to these VCs, using the following leaky-bucket parameter values: $c = 4$ cells/ms, and a value of $M$ in the range $10 \leq M \leq 50$. Thus, the network agrees to transport an average throughput roughly 28 per cent greater than the VC long-term average throughput of 3·12 cells/ms, but the allowed burstiness may be severely restricted. Note that the 35 cells from a maximum-sized Ethernet frame will take slightly less than 100 μs to enter the network, if they are sent at the full 150 Mb/s access rate.

There are several options from which a TA could choose in sending ATM cells; two options that illustrate flow controlling possibilities are

(i) For every Ethernet frame, send all the resulting ATM cells at the full 150 Mb/s rate into the network. This will typically result in many of the ATM cells being violation-tagged, but could be a good TA strategy if the network is only lightly loaded, depending on how easily recovery from cell discard can be effected.

(ii) For every Ethernet frame, separate successive ATM cells by an idle time of $\tau$ ms. This can be thought of as a 'traffic-shaping' form of flow control at the TA. In particular, if the choice $\tau = 0{\cdot}250$ ms is made, then none of the ATM cells will be violation-tagged, but the transfer of an entire Ethernet frame may be unnecessarily slowed down. The operation of such a 'traffic-shaper' can be thought of as a single-server queue with a deterministic service time, $\tau$.

## 4. TRAFFIC/PERFORMANCE MODELS UNDERLYING BWM

A global model or simulation would capture significant interactions. However, it would be too complicated to capture all interactions, so that submodels are needed for additional insight. It can be seen that the overall control architecture can be viewed as four separate control components, each of which can be studied through a separate model or set of models. In particular,

(i) An SCD (selective cell discard) network node model would focus on the performance characteristics of a simple discard mechanism for violation-tagged cells, e.g. based on an instantaneous local queue threshold, and would allow the cell traffic entering the node to be broken into its two components (violation-tagged and non-violation-tagged), with each component separately characterized by its intensity and a measure of burstiness, e.g. *peakedness*.[14] This model would allow quantification of delays and losses for non-violation-tagged and violation-tagged cells individually. The inputs for an SCD analysis module would be obtained as the outputs from other SCD modules as well as the TBF1,2 modules described below.

(ii) A pair of models, TBF1 and TBF2 (TBF = throughput-burstiness filter), would characterize the traffic impact of throughput-burstiness filtering through the leaky-bucket traffic monitoring. These models would allow the VC cell stream being filtered to be characterized by its original throughput and burstiness, and would produce the resulting throughputs and burstiness parameters of the two resulting cell streams: violation-tagged and non-violation-tagged. TBF1 focuses on the resulting non-violation-tagged traffic, and TBF2 focuses on the violation-tagged traffic.

It can be shown (see, for instance, some of the numerical results in Reference 15) that via this throughput-burstiness filtering the burstiness of the non-violation-tagged traffic is considerably reduced, and thus becomes much easier for the network to handle. Moreover, when network utilizations become high, so that violation-tagged traffic is very likely to be discarded, the detection of lost cells would trigger additional traffic-shaping in end-devices, and this also has a burstiness-reducing effect.

(iii) In an overall network congestion control analysis procedure, the outputs of the TBF1 and TBF2 modules would feed a network of

SCD modules, to provide a quantification of the congestion-dependent edge-to-edge transport characteristics of the network.

(iv) Finally, given this edge-to-edge transport characteristic, i.e. for the 'statistical VC' mentioned in Section 2.3, the end-to-end flow and error controls can be studied. For a specific set of end-device controls, the corresponding FEC (flow/error control) module would couple with this edge-to-edge transport characteristic to quantify overall end-to-end performance.

Related papers[15,16] address methodologies for characterizing the module TBF1, and ongoing activities are being similarly focused at the remaining modules, as well as analytic methods for coupling these modules together.

## 5. COMMONLY ASKED QUESTIONS AND ANSWERS

In this final section, we address some commonly asked questions regarding the proposed congestion control strategy. Several of the 'answers' may be regarded as 'responses', since further work, which is currently in progress, is needed to fully resolve some of the issues.

Q1. *How does the end-user pick appropriate traffic parameters?*

A1. In an ideal situation, an end-user may have accurate information about its traffic statistics, as well as the monitoring scheme implemented in the network. The user can then just pick the traffic parameters to be the appropriate monitoring parameters (e.g. leaky-bucket parameters). In reality, a user may not have this detailed traffic information, at least initially. In this case a conversion must be carried out that maps user-specifiable parameters to monitoring parameters. This could be accomplished either at the user terminal or in the network. Initially, the mapping could be based on *a priori* knowledge of the user traffic and may tend to be conservative. As we gain more experience, we may be able to modify the mapping to take advantage of additional information obtained through normal operation of the application, e.g. an adaptive learning algorithm. In the beginning, the user might only need to specify an application or class from a finite (small) number of applications or classes. As end-devices become more sophisticated and if cost incentives exist to encourage accurate information from the end-user, the above learning algorithms could be programmed in the end-devices and be completely transparent to the user.

Q2. *Why not use more parameters to characterize the traffic?*

A2. Although we emphasized a two-parameter characterization of the traffic (which could be directly related to the monitoring parameters) we feel that the peak rate is also a natural and necessary parameter to be specified, which leads to a three-parameter characterization. More parameters than three could lead to a more precise calibration of the user traffic and monitoring algorithm; however the added complexity is probably not justified.

Q3. *How do you guarantee that violation-tagged traffic has minimal effect on untagged traffic?*

A3. It depends on how the violation-tagged traffic gets dropped during congestion. One simple scheme is to set a moderate threshold on the total traffic in the output buffer of a node so that violation-tagged traffic is allowed to enter the buffer only when this threshold is not exceeded. This is very simple to implement and guarantees that the violation-tagged traffic has virtually no effect on untagged traffic loss and negligible effect on its delay. More advanced schemes such as dropping violation-tagged cells in the buffer only when an untagged cell encounters a full buffer are also being considered. These types of schemes offer a clear throughput advantage but are harder to implement in real-time, although, progress is being made in this direction.

Q4. *Why not have several priorities of violation-tagged cells to create an even higher degree of fairness?*

A4. This could be done; however, once again, we are striving for simplicity.

Q5. *Consider two VC's, AB and CD, over two separate paths which traverse a common congested link. Suppose that other than this congested link, path AB is not congested but path CD has several other congested links. Then, ideally, violation-tagged traffic on path CD should not be allowed to affect the violation-tagged traffic on path AB since the AB violation-tagged traffic could be delivered. Does this scheme allow for the protection of violation-tagged traffic over such paths?*

A5. No. This scheme cannot cure everything. More sophisticated tagging schemes as proposed above could potentially remedy this; however, our simple strategy takes the philosophy that any excess traffic is transported 'at risk' and no attempt is made to distinguish different types of excess traffic.

Q6. *The congestion control strategy proposed here is based on the concept of 'loss-priorities'. Couldn't the same effect be achieved through the more traditional 'delay-priorities'?*

A6. A large number of applications, e.g. real-time voice, video, etc., can tolerate moderate loss but not delay; loss priority is certainly the way to go for these applications. Delay priorities may achieve some of the same load-shedding effects; however, they may introduce additional problems (e.g. out-of-sequence cells) if not properly administered.

Q7. *When higher layer data units are segmented into ATM cells and several cells are violation-tagged and subsequently dropped, the entire higher layer data unit needs to be recovered. If a small number of cells from many higher layer data units are dropped won't this severely degrade the performance in terms of throughput?*

A7. Yes, but this can be resolved by monitoring and tagging the higher layer data units and propagating the tag to all cells into which an excess data unit is segmented. In this way violation-tagged cells that might be dropped would be clustered in a smaller number of data units.

Q8. *Can the end-user really benefit from the violation-tagging and transport of excess traffic?*

A8. Yes. Performance modelling of this congestion control strategy applied to a LAPD frame-relaying environment at wideband speeds[17] showed that end-users employing very simple adaptive window schemes can significantly increase their throughput to exploit transient surpluses of network capacity. Modelling of the higher speed environments is currently in progress and initial results indicate similar advantages. In high-speed applications that do not employ a window, simple adaptive shaping algorithms can be used.

Q9. *What is the best approach for applications that send short, high-speed transfers (e.g. files)?*

A9. We don't know the best approach yet, but we see three alternatives:

(i) Set up a 'long' holding time, 'low' activity VC for the high-speed transfer and carefully specify the traffic parameters to take into account the burstiness due to the VC alternating between file transfers and silence. The adaptive traffic-shaping algorithms mentioned above could also be used to gain further efficiency.

(ii) Same as above, except that an activate/confirm procedure is implemented before sending a file to 'reserve' the necessary bandwidth.

(iii) Set up short-duration VCs for each file transfer.

The trade-off between these alternatives is the efficient use of network resources and the VC set-up/tear-down processing at each node.

Q10. *Are there applications that can take advantage of pre-marking (see Section 3.1)?*

A10. Yes, packetized voice and variable-rate packetized video are applications of two services that can benefit from pre-marking non-essential cells.

Q11. *It is likely in a BISDN environment that an end-to-end virtual path will be set up, e.g. between a pair of PBXs. The network would not discriminate between cells belonging to separate VCIs. What would be monitored in this case and how can it be ensured that service is being managed appropriately between VCIs?*

A11. In this case the network would monitor the entire virtual path. To ensure that cells from well-behaved VCIs were not inappropriately violation-tagged because of cells belonging to ill-behaved VCIs, the PBX would monitor and tag the individual VCIs. This is another example of the usefulness of pre-marking. Note that in this case, all relevant monitoring and violation-tagging is being done by the end-system, so the monitoring performed by the network is simply a protection mechanism.

Q12. *Are there other methods of recovery of lost cells other than retransmission?*

A12. Yes. In fact several very simple forward error correcting schemes which recover from lost cells are known.[3] These could also provide additional motivation for the consideration of loss priorities.

Q13. *Is the leaky-bucket monitoring algorithm optimal?*

A13. We do not know, but we feel that it is reasonable. It has been shown to be better than fixed or sliding window monitoring schemes.[18]

Q14. *Traditional data transport has striven to minimize data loss, whereas this scheme encourages and attempts to take advantage of data loss. Can this be justified?*

A14. See A6 above. Also, in order to minimize the loss of data, additional buffering must be supplied by the network. This adds delays which might not be tolerated well by many services. Also, this may require complex service disciplines to ensure that the delays are fairly allocated, e.g. shorter delays to well-behaved VCs.

Q15. *With this congestion control strategy are we guaranteed that untagged traffic never causes congestion and never needs to be dropped?*

A15. This could be achieved but possibly not in a

practical sense. In particular, by limiting the allowable traffic parameters, by limiting the number of simultaneously set-up VCs, and by providing sufficient buffering capacity at all nodes, discard of untagged cells can be totally eliminated. However, this may place severe restrictions on traffic that is allowed and achievable network utilizations. More practically, these types of restrictions would be relaxed somewhat so that higher network utilizations could be achieved. This will result in situations where 'untagged overloads' could occur, but these events can be kept rare by appropriate engineering together with call-level controls.

Q16. *What specific methods and actions would the network use to deal with congestion caused by untagged traffic?*

A16. As mentioned in Section 3.2, higher layer call-level controls will be incorporated to ensure that untagged overloads will be rare. When it does occur, several options exist, such as adaptively changing the monitoring parameters or even dropping VCs. These are currently under consideration.

Q17. *Can you clarify the advantage of violation-tagging over policing?*

A17. Owing to the bursty nature of the traffic that will be transported on a broadband packet network even if the traffic was completely and accurately characterized by a small set of parameters, any monitoring scheme which is used for policing (e.g. dropping violating traffic) would need to set its parameters loosely to guarantee that only a negligible amount of traffic is lost. The situation is exacerbated due to the additional uncertainty in the traffic characteristics. Such loose monitoring parameters would offer very little protection to the network, since they would allow ill-behaved users to send much more than what was originally negotiated for. Violation-tagging on the other hand is a much 'softer' control since much of the time violation-tagged traffic would be delivered. Therefore, the monitoring parameters can be tighter and the network achieves much better protection.

Q18. *To avoid the necessity of providing a violation tag indication in an ATM cell header, would an equivalent method be to simply re-monitor each VC at each successive node along a path through the network, identify locally within each node those cells that are violation-tagged and those that are not, discard internally-violation-tagged cells as needed at the outputs of the node, but not pass the violation-tag indication along to the next node?*

A18. Although this method has strong similarities to the violation-tag-based BWM method, there are several important differences:

(i) Without an explicit violation tag indication in the cell header, it is not possible for the end-device to specify which cells are considered 'more expendable' than others.

(ii) There may be many situations where a full VC indication need not be examined at every node along a path. For example for a virtual path service only the VPI in the cell header needs to be examined by network nodes. However, traffic monitoring should be done on the basis of individual VCs to be more robust.

(iii) Finally, traffic statistics are inevitably altered as cells traverse network nodes and queue for transmission. Therefore, if traffic monitoring is to be done at successive nodes along a path, there must be an added 'margin' to achieve the same 'false alarm probability' (see Section 3.1.1), and consequently performance is degraded.

## REFERENCES

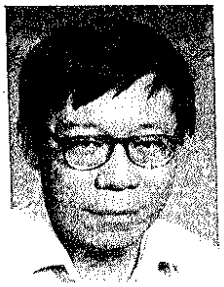

1. 'Broadband Aspects of ISDN', *Recommendation I.121, Vol. III, Fascicle III.7*, CCIT's November 1988 Recommendations (Blue Book).
2. 'Meeting Report of Sub-working Party 8/1—ATM', *CCITT COM XVIII-TD.14-E*, Geneva, June 1989.
3. N. Shacham, 'Packet recovery and error correction in high-speed wide-area networks', *Proc. MILCOM'89*.
4. D. W. Petr, L. A. DaSilva, Jr. and V. S. Frost, 'Priority discarding of speech in integrated packet networks', *IEEE J. Selected Areas in Comm.*, 7, (5), 644–656 (1989).
5. C. Chamzas and D. L. Duttweiller, 'Encoding facsimile images for packet-switched networks', *IEEE J. Selected Areas Commun.*, 7, (5), 857–864 (1989).
6. F. Kishino, K. Manabe, Y. Hayashi and H. Yasuda, 'Variable bi-rate coding of video signals for ATM networks', *IEEE J. Selected Areas Commun.*, 7, (5), 801–806 (1989).
7. R. G. H. Rogers, P. S. Richards and G. M. Woodruff, 'An assessment of network control configurations for a packetized multimedia communications network', *IEEE COMSOC Int. Workshop on Future Prospects of Burst/Packetized Multimedia Communications*, Osaka, 1987.
8. G. M. Woodruff, R. G. H. Rogers and P. S. Richards, 'A congestion control framework for high-speed integrated packetized transport', *Proc. IEEE GLOBECOM'88*, November, 1988.
9. G. Gallassi, G.Rigolio and L. Fratta, 'ATM: bandwidth assignment and bandwidth enforcement policies', *Proc. IEEE Globecom'89*, Dallas, 27–30 November 1989, pp. 49.6.1–49.6.6.
10. D. Sparrell, 'Wideband packet technology', *Proc. IEEE GLOBECOM'88*, November, 1988.
11. J. S. Turner, 'New directions in communications (or which way to the information age?)', *IEEE Communications Magazine*, October 1986.
12. R. L. Cruz, 'Maximum delay in buffered multistage interconnection networks', *Proceedings of the IEEE INFOCOM'88*, April 1988.
13. W. S. Lai, 'Coping with congestion losses in high-speed networks', to be submitted for publication. December 1990.

14. A. E. Eckberg, 'Generalized peakedness of teletraffic processes', *Proceedings of the 10th International Teletraffic Congress*, Montreal, 1983.
15. A. E. Eckberg, D. T. Luan and D. M. Lucantoni, 'Bandwidth management: a congestion control strategy for broadband packet networks—characterizing the throughput-burstiness filter', *Int. Teletraffic Congress Specialists Seminar*, Adelaide, 1989.
16. A. E. Eckberg and D. M. Lucantoni, 'A traffic/performance analysis of the bandwidth management throughput-burstiness filter', to appear in *Proc. 29th Conf. on Decision and Control,* December 1990.
17. D. T. Luan, and D. M. Lucantoni, 'Throughput analysis of an adaptive window-based flow control subject to bandwidth management', in M. Bonati (ed.), *Teletraffic Science for New Cost-Effective Systems, Networks and Services*, ITC-12, 1989.
18. T.-C. Hou, D. M. Lucantoni and A. E. Eckberg, 'Traffic monitoring/policing mechanisms for high-speed integrated services packet networks', *Fourth Annual Workshop on Computer Communications*, Dana Point, CA, 30 October–1 November 1989.

*Authors' biographies:*

**A. E. Eckberg** received the B.S., M.S. and Ph.D. degrees in Electrical Engineering from M.I.T., specializing in estimation and control. He consulted at Raytheon Co. and Lincoln Laboratories in the areas of signal processing, communications and air traffic control systems analysis. Following his Ph.D. in 1973, he joined AT&T Bell Laboratories, where he has been engaged in performance modelling and analysis activities directed at switching system and network service designs. Early performance issues that were addressed include switching system overload control strategies and algorithms and performance analysis of 'first generation' packet switching architectures.

He currently supervises a group in the Teletraffic Theory and System Performance Department of AT&T Bell Laboratories, Holmdel, NJ, which is responsible for performance, traffic and capacity studies of high-speed packet-switching systems and networks for the integrated transport of voice, data, image and video, as well as other forward-looking studies. Particular attention is being given to broadband transport mechanisms, including ATM and adaptation to ATM, with a focus on congestion and flow control strategies that may be appropriate for such technologies.



**Daniel T. Luan** received the B.S. degree in Electrical Engineering from National Taiwan University, Taipei, Taiwan, in 1972, the M.S. degree in Management Science from National Chiao Tung University, Hsinchu, Taiwan, in 1976 and the Ph.D. degree in Operations Research from Yale University, New Haven, CT, in 1982. His graduate research was in the areas of dynamic programming and stochastic optimization.

He joined AT&T Bell Laboratories in 1981 as a Member of Technical Staff. He worked initially on developing data network engineering methodologies and tools for the Bell System Operating Companies. Since 1984 he has been with the Teletraffic Theory and System Performance Department, where he has worked on various performance issues related to integrated networks, with a special emphasis on congestion and flow control.

He was on leave from AT&T Bell Laboratories from September 1988, to August 1989, and went back to Taiwan, ROC, where he held teaching positions in Chung Yuan Christian University and National Chiao Tung University. He has since returned to AT&T Bell Laboratories and worked on performance issues arising from distributed architectures.

**David M. Lucantoni** received the B.S. degree in Mathematics from Towson State University, Baltimore, MD, in 1976 and received the M.S. degree in Statistics in 1978, and the Ph.D. degree in Operations Research in 1981, both from the University of Delaware, Newark, DE. He was awarded the Allan P. Colburn Prize for the best dissertation in the Engineering and Mathematical Sciences at the University of Delaware in 1982.

Since 1981 he has been at AT&T Bell Laboratories, where he is a Distinguished Member of Technical Staff in the Teletraffic Theory and Systems Performance Department. His work has included the overload control design and performance analysis of mobile telephone systems, switching systems and integrated voice and data networks. In 1986 he was the co-recipient of the IEEE Communications Society Stephen O. Rice Prize Paper Award in the Field of Communication Theory. His current research interests are in the areas of the algorithmic analysis of stochastic models and queueing theory, and in the design and performance analysis of flow and congestion control architectures for broadband data communication networks.