CHAPTER 1

INTRODUCTION AND PROBLEM STATEMENT

1.1 Asynchronous Transfer Mode (ATM) Networks

With the convergence of telecommunication, entertainment and computer industries, computer networking is adopting a new paradigm called *Asynchronous Transfer Mode (ATM)* [14, 24]. ATM was selected by the telecommunication (carrier) industry as the technology to deliver the Broadband Integrated Services Digital Network (B-ISDN) carrier service. ATM is designed to handle different kinds of communication traffic (voice, audio, video and data) in an integrated way. It is first technology to promise seamless interworking between the LAN and WAN network environments. The international standards for ATM networks are being formulated by the ATM Forum [34] and ITU-T [114].

ATM uses short, fixed size (53-byte) packets, called "cells" which is an attractive option because: a) the transmission time per cell is fixed (which reduces the variability in queuing delays), and b) the transmission time is small (which allows building pipelined hardware architectures to process cells in switches). The resulting low mean delay, and low delay variance characteristics are the features that facilitate cell-based voice and video transmissions. However, each cell has five bytes (or 9.43%)

header information which limits the maximum possible efficiency of data transmission, especially on LANs. Further, the loss of one cell results in the loss of an entire packet (which may consist of several cells). But the cell switching (as opposed to expensive packet routing) and sophisticated traffic management technology in ATM networks allows the real efficiency to be close to the maximum possible (unlike the Ethernet technology where the efficiency drops off rapidly as load increases). This feature makes ATM attractive for data communications as well.

The development of the ATM technology has also resulted in several elegant total or compromise solutions to facilitate high-speed integrated services networking. These include: the use of shared switches (as opposed to using shared media), connection-oriented technology (to deliver guarantees, and simplified management and control), the use of short switch-assigned labels in cell headers instead of addresses (for scalability), the development of a true QoS-based routing (PNNI) protocol, and introduction of features such as LAN Emulation (LANE) and Multiprotocol over ATM (MPOA) which has triggered off work in the field of internetworking (running technology "X" over technology "Y") [24, 4].

In this dissertation, we focus on the problem of supporting data applications efficiently within the integrated services framework. Note that, in addition to providing a viable solution for any one of voice, video, or data transmission in isolation, ATM allows all these applications to be supported efficiently in a single network. This is a key feature differentiator when compared with current data network technologies like Ethernet. This feature, when complemented with traffic management capabilities allows the integrated network to be fully utilized while delivering the quality of service requested by applications.

1.2 The Available Bit Rate (ABR) Service

ATM networks provide multiple classes of service to support the quality of service (QoS) requirements of diverse applications, [35]. The current set of classes specified are: the constant bit rate (CBR), real-time variable bit rate (rt-VBR), non-real time variable bit rate (nrt-VBR), available bit rate (ABR), and unspecified bit rate (UBR). The CBR service is aimed at supporting voice and other synchronous applications, the VBR (rt- and nrt-) service are designed to support video and audio applications (which do not need isochronous transfer), while the ABR and UBR services are designed to primarily support data applications.

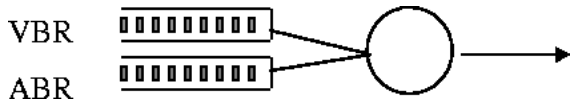

Figure 1.1: ATM ABR and VBR traffic sharing a link

Typically, the CBR and VBR classes are assigned higher "priority" by the network switches and get a share of the link bandwidth first. The "left-over" capacity is used by the ABR and UBR services, with ABR typically having priority over UBR. In figure 1.1, we show a link being shared by a "higher priority" VBR class and a "lower priority" ABR class. Note that VBR and ABR cells are queued separately.

The ABR service class includes an elaborate traffic management framework which allows the efficient handling of data traffic. On the other hand, there exist no standard method of managing traffic on the UBR service. Switches can provide proprietary traffic management mechanisms for UBR, but they cannot coordinate with other

switches since a standard does not exist. In the next section, we define the traffic management problem and discuss its role in the success of ATM as an integrated services networking technology.

1.3 Traffic Management vs Congestion Control

A key issue in ATM and in *any* network architecture design is resource management, i.e., how to make the best use of available resources. Maintaining high utilization of resources while satisfying the users' traffic contracts is the only way the high investment on the networking infrastructure can be recouped. However, striving for high utilization of a resource without proper allocation may lead to long queuing delays, and losses resulting in a low throughput (degradation of user-perceived quality of service).

Traffic management is a resource management problem which deals exclusively with the mechanisms required to control traffic on the network. A related problem is *"congestion"* which occurs when the aggregate demand for a resource (typically link bandwidth) exceeds the available capacity of the resource. In other words, congestion happens whenever the demand is more than the available capacity:

$$\sum_i \text{Demand}_i > \text{Available Capacity}$$

. There are two sets of mechanisms to handle congestion. *"Congestion control"* mechanisms typically come into play *after* the network is overloaded, i.e., congestion is detected. *"Congestion avoidance"* mechanisms come into play *before* the network becomes overloaded, i.e., congestion is predicted. "Congestion management" is a term used to to denote the combination of congestion avoidance and control mechanisms [67].

Congestion management involves the design of mechanisms and schemes to statically limit the demand-capacity mismatch, or dynamically control traffic sources when such a mismatch occurs. Congestion is a problem associated with the dynamics of the network load and capacity, it has been shown that static solutions such as allocating more buffers, or providing faster links, or faster processors does not solve the problem [67, 65]. In fact, the partial deployment of these static alternatives has led to more heterogeneity in the network and increased the possibility of congestion.

Observe that congestion management deals with the problem of matching the demand and capacity for a *single* network traffic class. Traffic management, even for a single traffic class, deals with the problem of ensuring that the network bandwidth, buffer and computational resources are efficiently utilized while meeting the various Quality of Service (QoS) guarantees given to sources as part of a traffic contract. The general problem of network traffic management involves all the available traffic classes. In ATM networks, the general traffic management problem involves the mechanisms needed to control the multiple classes of traffic (like CBR, VBR, ABR and UBR) while ensuring that *all* the traffic contracts are met. The components of traffic management other than congestion management schemes include scheduling mechanisms, traffic contract negotiation, admission control, and traffic policing. In this dissertation, we address the problem of designing traffic management mechanisms for one class - the ABR service class in ATM networks.

Historically, traditional data networks supported only one class of service (data). In such networks, the term "traffic management" was synonymous with "congestion control." In passing, we also note the difference between "flow control" and "congestion control." Flow control deals with the control of a particular flow, whereas

congestion control deals with the control of a group of flows sharing a group of network resource. It is possible to design congestion control schemes which essentially control flows individually at every hop. This makes the problem similar to flow control. An example of such a design is the hop-by-hop flow-controlled virtual circuit [86] or credit-based framework proposal for ATM discussed later in the dissertation.

1.4 Traffic Management for the ABR Service

In this dissertation, we shall address the problem of designing traffic management mechanisms for the ABR service class of ATM networks.

1.4.1 Problem Statement

Traffic management for ABR involves using end-to-end feedback control to match the variable ABR bandwidth at the network ABR queuing points with the variable demand of ABR sources. The statement of the abstract control problem(s) is(are) as follows:

Consider a bottleneck queuing point fed by a set of ABR sources. Define the following per-source variables:

$d_i(t)$ **:** is the desired demand (rate) of the i^{th} source at time t

$r_i(t)$ **:** is the network-assigned rate of the i^{th} source at time t

$T^i{}_d$ **:** is the propagation delay from the i^{th} source to the bottleneck ($T_d^- \leq T_d^i \leq T_d^+$).

Define the following bottleneck variables:

B : The buffer size at the bottleneck (constant)

C : The total capacity of the bottleneck (constant)

N : The total number of ABR virtual circuits through the bottleneck (constant).

$q_a(t)$ **:** the number of ABR cells at the bottleneck buffer at time t ($q_a(t) < B$)

$q_v(t)$ **:** the number of VBR cells at the bottleneck buffer at time t ($q_v(t) = 0$ since VBR is immediately serviced and never queued).

$C_a(t)$ **:** the available capacity for the ABR service ($K \times C < C_a(t) < C$, where $0 \leq K \leq 1$).

$C_v(t)$ **:** the capacity used by the VBR service ($0 \leq C_v(t) \leq (1 - K) \times C$).

$d_v(t)$ **:** the aggregate VBR demand. Note that $C_v(t) = d_v(t)$.

$\rho_a(t)$ **:** the ABR utilization factor at time t ($0 \leq \rho_a(t) \leq 1$, and, $\rho_a(t) = 1$, when $q_a(t) > 0$)

$n_a(t)$ **:** the number of active ABR sources at time t : $(0 \leq n_a(t) \leq N)$

The "open-loop" system is defined as follows. The bottleneck is loaded by both VBR and ABR. VBR is not controllable, whereas the ABR load and capacity display the following relation:

$$\sum_{i=1}^{n_a(t)} min(r_i(t - T^i{}_d), d_i(t - T^i{}_d)) = \frac{dq_a(t)}{dt} + \rho_a(t) \times C_a(t)$$

The equation gives a relation between ABR demand, capacity, queues. and utilization. The *left hand side* of the equation is the *aggregate ABR demand* at time t. It is simply the sum of the demands of the active ABR sources ($\sum_{i=1}^{n_a(t)}$) staggered by their respective time delays ($\tau = t - T^i{}_d$). Each ABR source demand is the minimum of the network-assigned rate ($r_i(\tau)$) and the desired source demand ($d_i(\tau)$). The *right*

hand side of the equation is the sum of the rate of growth of the ABR queue ($\frac{dq_a(t)}{dt}$) and the capacity ($C_a(t)$) scaled by the utilization factor ($\rho_a(t)$). In other words, the ABR demand directly affects the ABR utilization and the rate of growth of the ABR queue.

We desire that the system calculate and feedback rate assignments $r_i(t)$ which satisfy a desired set of goals. Since the goals are many, we elaborate the goals later in chapter 3. More generally, the problem we consider is the design of traffic management mechanisms for the ABR service. We consider five aspects of this problem in this dissertation. Firstly, the service requires a mechanism to carry rate feedback from the switches to the sources. We also design switch algorithms which calculate the rate allocations $r_i(t)$ to satisfy a given set of goals. Secondly, we design a set of source mechanisms which respond to feedback, and perform control when feedback is disrupted or is stale. Thirdly, we validate the performance of the service for various ABR and VBR demand patterns ($d_i(t)$ and $d_v(t)$). Specifically, we study the case of Internet traffic over ABR. Fourthly, we consider the switch design issues for a specific ABR framework option called the "Virtual Source/Virtual Destination" option. The detailed problem specifications and goals are considered in the respective portions of the dissertation.

Our general methodology for tackling this problem is the use of experimentation and simulation techniques, rather than rigorous mathematical analysis. This technique helps us build models which are closer to the real-world systems than mathematical models. However, we rely on simple analytical tools and techniques (such as metric design, and correlation of feedback with control) to ensure stability of the designed system.

1.4.2 Thesis Organization

The ATM Forum has defined a traffic management standard includes a rate-based framework to facilitate end-to-end feedback control. In this framework, ABR sources are allowed to send data at a network-directed rate (r_i, also called the "Allowed Cell Rate"). Periodically, the sources send control cells which are used by the switches to give feedback to the sources. We present the ABR traffic management framework in Chapter 2.

The first part of this dissertation covers the design and performance analysis of distributed algorithms (or "schemes") whose components run independently at different switches in the ATM network, and calculate the feedback for sources. In Chapter 3, we enumerate the goals and limitations of switch schemes.

Typical performance goals are: **a)** "efficiency" - to provide maximum link bandwidth utilization while minimizing queue length and computational overhead; **b)** "fairness" - to divide the available bandwidth fairly among all active sources; **c)** "transient response" - to respond quickly to changes in the load; and **d)** "steady state" - to be stable with minimal load oscillations. Finally, the system should be tuned to work for a wide variety of realistic workloads, and should provide a cost-effective implementation option.

We survey related work in the area of ABR switch scheme design in Chapter 4. We then describe the three switch schemes designed as a part of this dissertation work: the OSU scheme (Chapter 5, the ERICA and the ERICA+ schemes (Chapter 6). The work done as part of the development of these schemes helped design the ATM Forum Traffic Management Specification 4.0 [35], and introduced several concepts which are part of later switch schemes. These chapters also include extensive performance

analyses which are used to validate the schemes, and to illustrate the methodology of switch scheme testing.

The second part of this dissertation deals with design of source end-system control mechanisms (Chapter 7). Such mechanisms are inherently "open-loop" in the sense that sources may unilaterally reduce rates without feedback from switches. Cases where such an approach will be useful includes : a) the case when a network link becomes broken and feedback does not reach the sources, b) the case when a source which is granted a high rate becomes idle temporarily and later uses its retained rate. If the network is heavily loaded, both these cases may result in unpredictably large queuing delays. The mechanisms also determine how RM and data cells are scheduled, especially for low bit-rate sources.

The third part of the dissertation deals with issues in supporting Internet applications like file transfer and world wide web (which run over the TCP/IP protocol) over ATM ABR, with different models of higher priority VBR background traffic (Chapter 8). We study the dynamics and quantify buffer requirements to support zero-loss transmission under such conditions.

The fourth part of this dissertation deals with the switch design issues for a specific ABR framework option called the "Virtual Source/Virtual Destination" option (Chapter 9). In this option, the switch splits the network into two segments and shortens the feedback loop for both segments.

We briefly look at implementation issues in Chapter 10 and proceed to summarize and conclude this dissertation in Chapter 11.

Appendix A quotes the source, destination and switch rules from the ATM Traffic Management 4.0 specification. Appendices B, C and C.3 detail the complete

pseudo-code for the OSU scheme, ERICA schemes and VS/VD alternatives, including the optional features of each. Finally, appendix D provides a glossary of common acronyms used in this dissertation.

CHAPTER 2

THE ABR TRAFFIC MANAGEMENT FRAMEWORK

ABR mechanisms allow the network to divide the available bandwidth fairly and efficiently among the active traffic sources. In the ABR traffic management framework, the *source end systems* limit their data transmission to rates allowed by the network. The network consists of *switches* which use their current load information to calculate the allowable rates for the sources. These rates are sent to the sources as feedback via *resource management (RM)* cells. RM cells are generated by the sources and travel along the data path to the *destination end systems*. The destinations simply return the RM cells to the sources. The components of the ABR traffic management framework are shown in Figure 2.1. In this tutorial, we explain the source and destination end-system behaviors and their implications on ABR traffic management.

The ABR traffic management model is called a "rate-based end-to-end closed-loop" model. The model is called "rate-based" because the sources send data at a specified "rate." This is different from current packet networks (for example, TCP), where the control is "window based" and the sources limit their transmission to a particular number of packets. The ABR model is called "closed-loop" because there is a continuous feedback of control information between the network and the source.

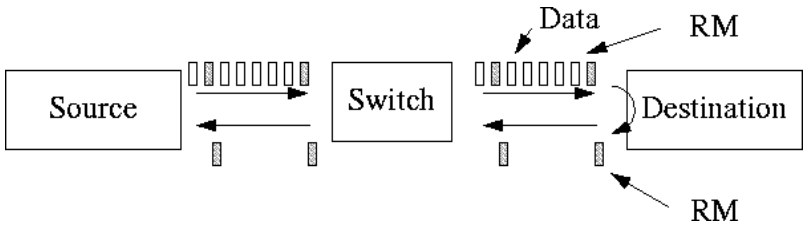

Figure 2.1: ABR Traffic Management Model: Source, Switch, Destination and Resource Management Cells

If more sources become active, the rate allocated to each source is reduced. The model used for CBR and VBR traffic, on the other hand, is "open-loop" in the sense that rates are negotiated at the beginning of the connection and do not change dynamically. Finally, the model is called "end-to-end" because the control cells travel from the source to the destination and back to the source. The alternative of "hop-by-hop" control in which each switch would give feedback to the previous switch [86] was considered and not accepted due to its complexity. However, one can achieve the hop-by-hop control in TM4.0 using the virtual source/virtual destination (VS/VD) feature discussed later in this section.

When there is a steady flow of RM cells in the forward and reverse directions, there is a steady flow of feedback from the network. In this state, the ABR control loop has been established and the source rates are primarily controlled by the network feedback (closed-loop control). However, until the first RM cell returns, the source rate is controlled by the negotiated parameters, which may or may not relate to the current load on the network. The virtual circuit (VC) is said to be following an "open-loop" control during this phase. This phase normally lasts for one round-trip

time (RTT). As we explain later, ABR sources are required to return to the open-loop control after long idle intervals. Traffic sources that have active periods (bursts) when data is transmitted at the allocated rate and idle periods when no data is transmitted are called "bursty sources" Open-loop control has a significant influence on the performance of bursty traffic particularly if it consists of bursts separated by long idle intervals.

There are three ways for switches to give feedback to the sources:

1. First, each cell header contains a bit called Explicit Forward Congestion Indication (EFCI), which can be set by a congested switch. This mechanism is a modification of the DECbit scheme [63]. Such switches are called "binary" or "EFCI" switches. The destination then aggregates these EFCI bit information and returns feedback to the source in an RM cell. An initial version of the binary feedback scheme is illustrated in figure 2.2. In the current specification, the RM cell is sent by the source periodically and is turned around by the destination with the bit-feedback.

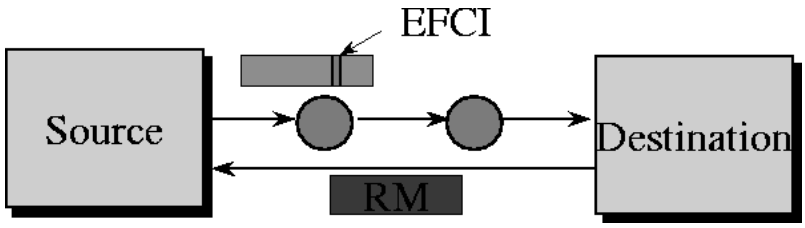

Figure 2.2: Initial Binary Feedback Scheme

2. Second, RM cells have two bits in their payload, called the Congestion Indication (CI) bit and the No Increase (NI) bit, that can be set by congested switches. Switches that use only this mechanism are called relative rate marking switches.

3. Third, the RM cells also have another field in their payload called explicit rate (ER) that can be reduced by congested switches to any desired value. Such switches are called explicit rate switches. The explicit rate mechanism is shown in figure 2.3.

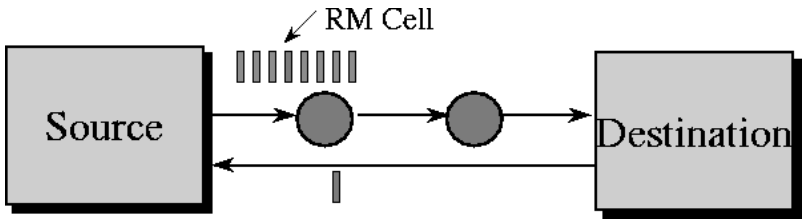

Figure 2.3: Initial Explicit Rate Scheme

Explicit rate switches normally wait for the arrival of an RM cell to give feedback to a source. However, under extreme congestion, they are allowed to generate an RM cell and send it immediately to the source. This optional mechanism is called backward explicit congestion notification (BECN).

Switches can use the VS/VD feature to segment the ABR control loop into smaller loops. In a VS/VD network, the switches additionally behave both as a (virtual) destination end system and as a (virtual) source end system. As a destination end system, it turns around the RM cells to the sources from one segment. As a source end system, it generates RM cells for the next segment. This feature can allow

feedback from nearby switches to reach sources faster, and allow hop-by-hop control as discussed earlier.

2.1 ABR Parameters

At the time of connection setup, ABR sources negotiate several operating parameters with the network. The first among these is the peak cell rate (PCR). This is the maximum rate at which the source will be allowed to transmit on this virtual circuit (VC). The source can also request a minimum cell rate (MCR) which is the guaranteed minimum rate. The network has to reserve this bandwidth for the VC. During the data transmission stage, the rate at which a source is allowed to send at any particular instant is called the allowed cell rate (ACR). The ACR is dynamically changed between MCR and PCR. At the beginning of the connection, and after long idle intervals, ACR is set to initial cell rate (ICR).

During the development of the RM specification, all numerical values in the specification were replaced by mnemonics. For example, instead of saying "every 32nd cell should be an RM cell" the specification states "every Nrmth cell should be an RM cell." Here, Nrm is a parameter whose default value is 32. Some of the parameters are fixed while others are negotiated. A complete list of parameters used in the ABR mechanism is presented in Table 2.1. The parameters are explained as they occur in our discussion.

2.2 In-Rate and Out-of-Rate RM Cells

Most resource management cells generated by the sources are counted as part of their network load in the sense that the total rate of data and RM cells should not

Label	Expansion	Default Value
PCR	Peak Cell Rate	-
MCR	Minimum Cell Rate	0
ACR	Allowed Cell Rate	-
ICR	Initial Cell Rate	PCR
TCR	Tagged Cell Rate	10 cells/s
Nrm	Number of cells between FRM cells	32
Mrm	Controls bandwidth allocation between FRM, BRM and data cells	2
Trm	Upper Bound on Inter-FRM Time	100 ms
RIF	Rate Increase Factor	1/16
RDF	Rate Decrease Factor	1/16
ADTF	ACR Decrease Time Factor	0.5 ms
TBE	Transient Buffer Exposure	16,777,215
CRM	Missing RM-cell Count	$\lceil$ TBE/Nrm $\rceil$
CDF	Cutoff Decrease Factor	1/16
FRTT	Fixed Round-Trip Time	-

Table 2.1: List of ABR Parameters

exceed the ACR of the source. Such RM cells are called "in-rate" RM cells. Under exceptional circumstances, switches, destinations, or even sources can generate extra RM cells. These "out-of-rate" RM cells are not counted in the ACR of the source and are distinguished by having their cell loss priority (CLP) bit set, which means that the network will carry them only if there is plenty of bandwidth and can discard them if congested. The out-of-rate RM cells generated by the source and switch are limited to 10 RM cells per second per VC. One use of out-of-rate RM cells is for BECN from the switches. Another use is for a source, whose ACR has been set to zero by the network, to periodically sense the state of the network. Out-of-rate RM cells are also used by destinations of VCs whose reverse direction ACR is either zero or not sufficient to return all RM cells received in the forward direction.

Note that in-rate and out-of-rate distinction applies only to RM cells. All data cells in ABR should have CLP set to 0 and must always be within the rate allowed by the network.

2.3 Forward and Backward RM cells

Resource Management cells traveling from the source to the destination are called "forward RM" (FRM) cells. The destination turns around these RM cells and sends them back to the source on the same VC. Such RM cells traveling from the destination to the source are called Backward RM (BRM) cells. Forward and backward RM cells are illustrated in Figure 2.4. Note that when there is bi-directional traffic, there are FRMs and BRMs in both directions on the Virtual Channel (VC). A bit in the RM cell payload indicates whether it is an FRM or BRM. This direction bit (DIR) is changed from 0 to 1 by the destination.

2.4 RM Cell Format

The complete format of the RM cells is shown in figure 2.5. Every RM cell has the regular ATM header of five bytes. The payload type indicator (PTI) field is set to 110 (binary) to indicate that the cell is an RM cell. The protocol id field, which is one byte long, is set to one for ABR connections. The direction (DIR) bit distinguishes forward and backward RM cells. The backward notification (BN) bit is set only in switch generated BECN cells. The congestion indication (CI) bit is used by relative rate marking switches. It may also be used by explicit rate switches under extreme congestion as discussed later. The no increase (NI) bit is another bit available to explicit rate switches to indicate moderate congestion. The request/acknowledge,

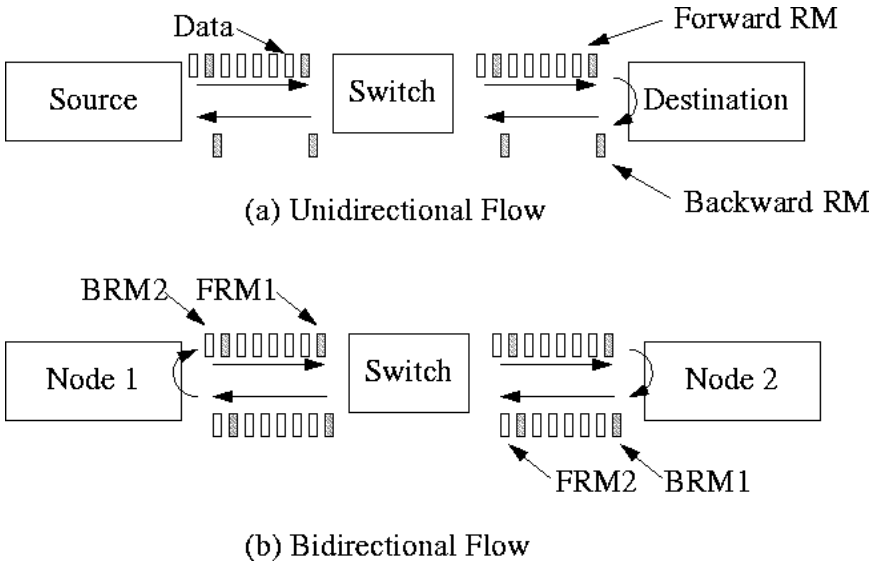

Figure 2.4: Forward and Backward Resource Management Cells (FRMs and BRMs)

queue length, and sequence number fields of the RM cells are for compatibility with the ITU-T recommendation I.371 and are not used by the ATM Forum.

The Current Cell Rate (CCR) field is used by the source to indicate to the network its current rate. Some switches may use the CCR field to determine a VC's next allocation while others may measure the VC's rate and not trust CCR. The minimum cell rate (MCR) field is redundant in the sense that like PCR, ICR, and other parameters it does not change during the life of a connection. However, its presence in the RM cells reduces number of lookups required in the switch.

The ER, CI and NI fields are used by the network to give feedback to the sources. The ER field indicates the maximum rate allowed to the source. When there are multiple switches along the path, the feedback given by the most congested link is the one that reaches the source.

Data cells also have an Explicit Forward Congestion Indication (EFCI) bit in their headers, which may be set by the network when it experiences congestion. The destination saves the EFCI state of every data cell. If the EFCI state is set when it turns around an RM cell, it uses the CI bit to give (a single bit) feedback to the source. When the source receives the RM cell from the network, it adjusts its ACR using the ER, CI, NI values, and source parameters.

ATM Header	5 Bytes	
Protocol ID	1 Byte	1 = ABR
Direction	1 bit	0 = Forward
Backward Notification	1 bit	1 = Switch/dest generated
Congestion Indication	1 bit	1 = High Congestion
No Increase	1 bit	1 = Mild congestion
Request/Acknowledge*	1 bit	
Reserved	3 bits	
Explicit Rate	2 Bytes	
Current Cell Rate	2 Bytes	
Minimum Cell Rate	2 Bytes	
Queue Length*	4 Bytes	
Sequence Number*	4 Bytes	
Reserved	30.75 Bytes	
CRC-10	10 bits	

Figure 2.5: Resource Management (RM) Cell Fields

All rates (e.g., ER, CCR, and MCR) in the RM cell are represented using a special 16-bit floating point format, which allows a maximum value of 4,290,772,992 cells per second (1.8 terabits per second). During connection setup, however, rate parameters are negotiated using an 24-bit integer format, which limits their maximum value to 16,777,215 cells per second or 7.1 Gb/s.

2.5 Source End System Rules

TM4.0 specifies 13 rules that the sources have to follow. This section discusses each rule and traces the development and implications of certain important rules. In some cases the precise statement of the rule is important. Hence, the source and destination rules are quoted from the TM specification [35] in appendix A.

- **Source Rule 1**: Sources should always transmit at a rate equal to or below their computed ACR. The ACR cannot exceed PCR and need not go below MCR. Mathematically,

$$\text{MCR} \leq \text{ACR} \leq \text{PCR}$$

$$\text{Source Rate} \leq \text{ACR}$$

- **Source Rule 2**: At the beginning of a connection, sources start at ICR. The first cell is always an in-rate forward RM cell. This ensures that the network feedback will be received as soon as possible.

- **Source Rule 3**: At any instant, sources have three kinds of cells to send: data cells, forward RM cells, and backward RM cells (corresponding to the reverse flow). The relative priority of these three kinds of cells is different at different transmission opportunities.

  First, the sources are required to send an FRM after every 31 cells. However, if the source rate is low, the time between RM cells will be large and network feedback will be delayed. To overcome this problem, a source is supposed to send an FRM cell if more than 100 ms has elapsed since the last FRM. This introduces another problem for low rate sources. In some cases, at every transmission

opportunity the source may find that it has exceeded 100 ms and needs to send an FRM cell. In this case, no data cells will be transmitted. To overcome this problem, an additional condition was added that there must be at least two other cells between FRMs.

An example of the operation of the above condition is shown in the figure 2.6. The figure assumes a unidirectional VC (i.e., there are no BRMs to be turned around). The figure has three parts. The first part of the figure shows that, when the source rate is 500 cells/s, every 32nd cell is an FRM cell. The time to send 32 cells is always smaller than 100 ms. In the second part of the figure, the source rate is 50 cells/s. Hence 32 cells takes 640 ms to be transmitted. Therefore, after 100 ms, an FRM is scheduled in the next transmission opportunity (or slot). The third part of the figure shows the scenario when the source rate is 5 cells/s. The inter-cell time itself is 200 ms. In this case, an FRM is scheduled every three slots, i.e., the inter-FRM time is 600 ms. Since *Mrm* is 2, two slots between FRMs are used for data or BRM cells.

Second, a waiting BRM has priority over waiting data, given that no BRM has been sent since the last FRM. Of course, if there are no data cells to send, waiting BRMs may be sent.

Third, data cells have priority in the remaining slots.

The second and third part of the this rule ensure that BRMs are not unnecessarily delayed and that all available bandwidth is not used up by the RM cells.

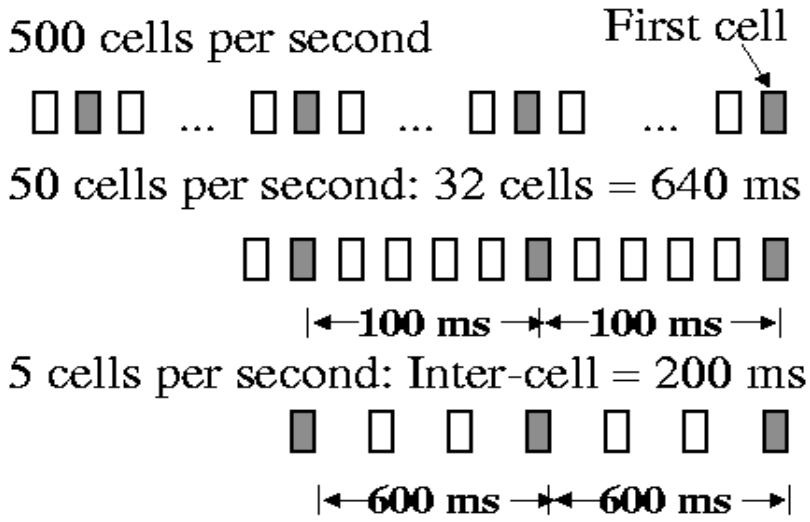

Figure 2.6: Frequency of forward RM cells.

Figure 2.7 illustrates the scheduling of FRMs, BRMs and data cells. In the first slot, an FRM is scheduled. In the next slot, assuming that a turned around BRM is awaiting transmission, a BRM is scheduled. In the remaining slots data is scheduled. If the rate is low, more FRMs and BRMs may be scheduled.

- **Source Rule 4**: All RM cells sent in accordance with rules 1-3 are in-rate RM cells and have their cell loss priority (CLP) bit set to 0. Additional RM cells may be sent out-of-rate and should have their CLP bit set to 1. For example, consider the third unidirectional flow of Figure 2.6. It has an ACR of 5 cells/s.

Figure 2.7: Scheduling of forward RM, backward RM, and data cells.

It is allowed to send only one in-rate RM cell every 400 ms. If necessary, it can send a limited number of out-of-rate RM cells with CLP set to 1.

The frequency of FRM is determined by parameters Nrm, Trm, and Mrm, whose default values are 32, 100 ms, and 2, respectively. During the debate on credit vs rate based alternatives for traffic management [69], the rate based group selected a default value of 32 for Nrm. This ensured that the control overhead was equivalent to that of credit based alternative which claimed an overhead of approximately 6%. During normal operation 1/32th or 3% of all cells are FRM cells. Similarly, another 3% of cells are BRM cells resulting in a total overhead of 6%.

In practice, the choice of Nrm affects the responsiveness of the control and the computational overhead at the end systems and switches. For a connection running at 155 Mb/s, the inter-RM cell time is 86.4 μs while it is 8.60 ms for the same connection running at 1.55 Mb/s. The inter-RM interval determines the responsiveness of the system. While most end-systems and switches will do ABR computations in hardware, it has been shown that it is possible to do them in software on a Pentium^{TM} system provided Nrm is set to 192 or higher on a 155 Mb/s link.

- **Source Rule 5**: The rate allowed to a source is valid only for approximately 500 ms. If a source does not transmit any RM cells for this duration, it cannot use its previously allocated ACR particularly if the ACR is high. The source should re-sense the network state by sending an RM cell and decreasing its rate to the initial cell rate (ICR) negotiated at connection setup. If a source's ACR is already below ICR, it should stay at that lower value (and not increase it to ICR).

  The timeout interval is set by the ACR Decrease Time Factor (ADTF). This parameter can be negotiated with the network at connection setup. Its default value is 500 ms.

  This simple rule was the cause of a big debate at the Forum. It is intended to solve the problem of *ACR retention*. If a source sends an RM cell when the network is not heavily loaded, the source may be granted a very high rate. The source can then retain that rate and use it when the network is highly loaded. In fact, a source may set up several VCs and use them to get an unfair advantage. To solve this problem, several so called *use it or lose it* (UILI) solutions were proposed. Some of them relied on actions at the source while others relied on actions at the switch. The source based solutions required sources to monitor their own rates and reduce ACR slowly if was too high compared to the rate used.

  UILI alternatives were analyzed and debated for months because they have a significant impact on the performance of bursty traffic that forms the bulk of data traffic. The ATM Forum chose to standardize a very simple UILI policy at the source. This policy provided a simple timeout method (using ADTF

as the timeout value) which reduces ACR to ICR when the timeout expires. Vendors are free to implement additional proprietary restraints at the source or at the switch. A few examples of such possibilities are listed in the Informative Appendix I.8 of the specification [35]. We survey the proposed UILI alternatives and present our design later in this dissertation.

- **Source Rule 6**: If a network link becomes broken or becomes highly congested, the RM cells may get blocked in a queue and the source may not receive the feedback. To protect the network from continuous in-flow of traffic under such circumstances, the sources are required to reduce their rate if the network feedback is not received in a timely manner.

  Normally under steady state, sources should receive one BRM for every FRM sent. Under congestion, BRM cells may be delayed. If a source has sent CRM FRM cells and has not received any BRM, it should suspect network congestion and reduce its rate by a factor of CDF. Here, CRM (missing RM cell count) and CDF (cutoff decrease factor) are parameters negotiated at the time of connection setup. BECN cells generated by switches (and identified by BN=1) are not counted as BRM.

  When rule 6 triggers once, the condition is satisfied for all successive FRM cells until a BRM is received. Thus, this rule results in a fast exponential decrease of ACR. An important side effect of this rule is that unless CRM is set high, the rule could trigger unnecessarily on a long delay path. CRM is computed from another parameter called transient buffer exposure (TBE) which is negotiated at connection setup. TBE determines the maximum number of cells that may

suddenly appear at the switch during the first round trip before the closed-loop phase of the control takes effect. During this time, the source will have sent TBE/Nrm RM cells. Hence,

$$\text{CRM} = \lceil \frac{\text{TBE}}{\text{Nrm}} \rceil$$

The fixed part of the round-trip time (FRTT) is computed during connection setup. This is the minimum delay along the path and does not include any queuing delay. During this time, a source may send as many as ICR $\times$ FRTT cells into the network. Since this number is negotiated separately as TBE, the following relationship exists between ICR and TBE:

$$ICR \times FRTT \leq TBE$$

or

$$ICR \leq TBE/FRTT$$

The sources are required to use the ICR value computed above if it is less than the ICR negotiated with the network. In other words:

ICR used by the source =

Min{ICR negotiated with the network,

TBE/FRTT}

In negotiating TBE, the switches have to consider their buffer availability. As the name indicates, the switch may be suddenly exposed to TBE cells during the first round trip (and also after long idle periods). For small buffers, TBE should be small and vice versa. On the other hand, TBE should also be large enough to prevent unnecessary triggering of rule 6 on long delay paths.

It has been incorrectly believed that cell loss could be *avoided* by simply negotiating a TBE value below the number of available buffers in the switches. We have shown [70] that it is possible to construct workloads where queue sizes could be unreasonably high even when TBE is very small. For example, if the FRM input rate is x times the BRM output rate (see Figure 2.8), where x is less than CRM, rule 6 will not trigger but the queues in the network will keep building up at the rate of $(x-1) \times$ ACR leading to large queues. The only reliable way to protect a switch from large queues is to build it in the switch allocation algorithm. The ERICA+ algorithm presented in this dissertation is an example of one such algorithm.

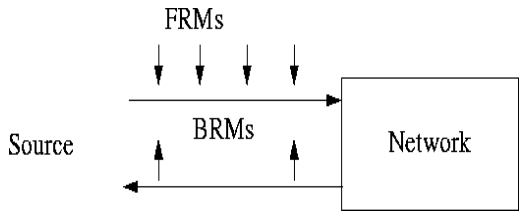

Figure 2.8: Source Rule 6 does not trigger if BRM flow is maintained

Observe that the FRTT parameter which is the sum of fixed delays on the path is used in the formula for ICR. During the development of this rule, an estimate of round trip time (RTT), including the fixed and variable delays was being used instead of FRTT in the ICR calculation. We argued that RTT estimated at connection setup is a random quantity bearing little relation to the round trip delays during actual operation [74]. Such parameter setting could trigger source

Rule 6 unnecessarily and degrade performance. Hence, the Forum decided to use FRTT parameter instead of RTT.

Note that it is possible to disable source Rule 6, by setting CDF to zero.

- **Source Rule 7**: When sending an FRM, the sources should indicate their current ACR in the CCR field of the RM cells.

- **Source Rules 8 and 9**: Source Rule 8 and 9 describe how the source should react to network feedback. The feedback consists of explicit rate (ER), congestion indication bit (CI), and no-increase bit (NI). Normally, a source could simply change its ACR to the new ER value; but this could cause a few problems as discussed next.

  First, if the new ER is very high compared to current ACR, switching to the new ER will cause sudden queues in the network. Therefore, the amount of increase is limited. The rate increase factor (RIF) parameter determines the maximum allowed increase in any one step. The source cannot increase its ACR by more than RIF $\times$ PCR.

  Second, if there are any EFCI switches in the path, they do not change the ER field. Instead, they set EFCI bits in the cell headers. The destination monitors these bits and returns the last seen EFCI bit in the CI field of a BRM. A CI of 1 means that the network is congested and that the source should reduce its rate. The decrease is determined by rate decrease factor (RDF) parameter. Unlike the increase, which is additive, the decrease is multiplicative in the sense that

$$\text{ACR} \leftarrow \text{ACR}(1 - \text{RDF})$$

NI	CI	Action
0	0	ACR $\leftarrow$ Min(ER, ACR + RIF $\times$ PCR, PCR)
0	1	ACR $\leftarrow$ Min(ER, ACR $-$ ACR $\times$ RDF)
1	0	ACR $\leftarrow$ Min(ER, ACR)
1	1	ACR $\leftarrow$ Min(ER, ACR $-$ ACR $\times$ RDF)

Table 2.2: Source End System actions upon CI and NI bits

It has been shown that additive increase and multiplicative decrease is sufficient to achieve fairness [21]. Other combinations such as additive increase with additive decrease, multiplicative increase with multiplicative decrease, and multiplicative increase with additive increase are unfair.

The no-increase (NI) bit was introduced to handle mild congestion cases. In such cases, a switch could specify an ER, but instruct that, if ACR is already below the specified ER, the source should not increase the rate. The actions corresponding to the various values of CI and NI bits are listed in Table 2.2.

$$\text{ACR} \leftarrow \text{Max(ACR,MCR)}$$

If there are no EFCI switches in a network, setting RIF to 1 allows ACRs to increase as fast as the network directs it. This allows the available bandwidth to be used quickly. For EFCI networks, or a combination of ER and EFCI networks, RIF should be set conservatively to avoid unnecessary oscillations.

Once the ACR is updated, the subsequent cells sent from the source conform to the new ACR value. However, if the earlier ACR was very low, it is possible that the very next cell is scheduled a long time in the future. In such a situation,

it is advantageous to "reschedule" the next cell, so that the source can take advantage of the high ACR allocation immediately [73] (also see chapter 7).

- **Source Rule 10**: Sources should initialize various fields of FRM cells as follows. For virtual path connections (VPCs), the virtual circuit id (VCI) is set to 6. For virtual channel connections (VCCs), the VCI of the connection is used. In either case, the protocol type id (PTI) in the ATM cell header is set to 6 (110). The protocol id field in the payload of the RM cell is set to 1. The direction bit should be set to 0 (forward). The backward notification (BN) bits should be set to 0 (source generated). Explicit rate field is initialized to the maximum rate below PCR that the source can support. Current cell rate is set to current ACR. Minimum cell rate is set to the value negotiated at connection setup. Queue length, sequence number, and request/acknowledge fields are set in accordance with ITU-T recommendation I.371 or to zero. All reserved octets are set to 6A (hex) or 01101010 (binary). This value is specified in ITU-T recommendation I.610 (whose number coincidently is also 6-A in hex). Other reserved bits are set to 0. Note that the sources are allowed to set ER and NI fields to indicate their own congestion.

- **Source Rule 11**: The out-of-rate FRM cells generated by sources are limited to to a rate below the "tagged cell rate (TCR)" parameter, which has a default value of 10 cells per second.

- **Source Rule 12**: The EFCI bit must be reset on every data cell sent. The alternative of congested sources being allowed to set EFCI bit was considered but rejected due to insufficient analysis.

- **Source Rule 13**: Sources can optionally implement additional Use-It-or-Lose-It (UILI) policies (see discussion of source Rule 5 and also later in this dissertation).

2.6 Destination End System Rules

- **Destination Rule 1**: Destinations should monitor the EFCI bits on the incoming cells and store the value last seen on a data cell.

- **Destination Rule 2**: Destinations are required to turn around the forward RM cells with minimal modifications as follows: the DIR bit is set to "backward" to indicate that the cell is a backward RM-cell; the BN bit is set to zero to indicate that the cell was not generated by a switch; the CCR and MCR fields should not be changed. If the last cell has EFCI bit set, the CI bit in the next BRM is set and the stored EFCI state is cleared.

  If the destination has internal congestion, it may reduce the ER or set the CI or NI bits just like a switch. Observe that this rule is used in the VS/VD configuration where the virtual destination is bottlenecked by the allowed rate in the next segment. In any case, the ER is never increased.

- **Destination Rules 3-4**: The destination should turn around the RM cells as fast as possible. However, an RM cell may be delayed if the reverse ACR is low. In such cases destination rules 3 and 4 specify that old out-of-date information can be discarded. The destinations are allowed a number of options to do this. The implications of various options of destination Rule 3 are discussed in the

Informative Appendix I.7 of the TM specification [35]. Briefly, the recommendations attempt to ensure the flow of feedback to the sources for a wide range of values of ACR of the reverse direction VC. If the reverse direction ACR is non-zero, then a backward RM cell will be scheduled for in-rate transmission. Transmitting backward RM cells out-of-rate ensures that the feedback is sent regularly even if the reverse ACR is low or zero (for example, in unidirectional VCs).

Note that there is no specified limit on the rate of such "turned around" out-of-rate RM cells. However, the CLP bit is set to 1 in the out-of-rate cells, which allows them to be selectively dropped by the switch if congestion is experienced.

- **Destination Rule 5**: Sometimes a destination may be too congested and may want the source to reduce its rate immediately without having to wait for the next RM cell. Therefore, like a switch, the destinations are allowed to generate BECN RM cells. Also, as in the case of switch generated BECNs, these cells may not ask a source to increase its rate (CI bit is set). These BECN cells are limited to 10 cells/s and their CLP bits are set (i.e., they are sent out-of-rate).

- **Destination Rule 6**: An out-of-rate FRM cell may be turned around either in-rate (with CLP=0) or out-of-rate (with CLP=1).

2.7 Switch Behavior

The switch behavior specifies that the switch must implement some form of congestion control, and rules regarding processing, queuing and generation of RM cells.

- **Switch Rule 1**: This rule specifies that one or more methods of feedback marking methods must be implemented at the switch. The possible methods include :

  EFCI Marking: This defines the binary (bit-based) feedback framework, where switches may set the EFCI bit in data cell headers. We have noted earlier that the destinations maintain an EFCI state per-VC and set the CI bit in backward RM cells if the VC's EFCI state is set. Note that the VC's EFCI state at the destination is set and reset whenever an incoming data cell has its EFCI set or reset respectively.

  Relative Rate Marking: This option allows the switch to set two bits in the RM cell which have a specific meaning to when they reach the source end systems. The CI bit when set asks the source to decrease, while the NI bit tells the source not to increase beyond its current rate, ACR. Observe that the source rate may be further reduced using the explicit rate indication field. These bits allow the switches some more flexibility than the EFCI bit marking. Specifically, the switches can avoid the "beat-down" fairness problem seen in EFCI marking scenarios. The problem occurs because connections going through several switches have a higher probability of their EFCI bits being set, than connections going through a smaller number of switches.

  Explicit Rate Marking: Allows the switch to specify exactly what rate it wants a source to send at. To ensure coordination among multiple switches in a connection's path, the switch may reduce (but not increase) the ER

field in the RM-cells (in the forward and/or backward directions). This dissertation deals mainly with explicit rate feedback from switches.

VS/VD Control: In this mode, the switch may segment the ABR control loop by appearing as a "virtual source" to one side of the loop and as a "virtual destination" to the other side. We study the implications of this mechanism on the ABR service later in this dissertation.

- **Switch Rule 2**: This rule specifies how a switch may generate an RM cell in case it is heavily congested and doesn't see RM cells from the source. Basically, the rule allows such RM cells to only decrease the source rate, and these RM cells are sent out-of-rate. This rule contains aspects of the Backward Explicit Congestion Notification proposal [92] and the OSU scheme proposal described later in this dissertation.

- **Switch Rule 3**: This rule says that the RM cells may be transmitted out-of-sequence, but the sequence integrity must be maintained. This rule allows the switch the flexibility to put the RM cells on a priority queue for faster feedback to sources when congested. However, by queuing RM cells separately from the data stream, the correlation between the quantities declared RM cells and the actual values in the data stream may be lost.

- **Switch Rule 4 and 5**: Rule 4 specifies alignment with ITU-T's I.371 draft, and ensures the integrity of the MCR field in the RM cell. Rule 5 allows the optional implementation of a use-it-or-lose-it policy at the switch. We treat the use-it-or-lose-it issue in greater detail later in this dissertation.

Observe that the ABR traffic management framework only specifies that the switch should implement a feedback marking mechanism and gives flexibility on how to handle RM cells. However, the specific schemes to calculate feedback are not standardized. Several other aspects (such as VS/VD, use-it-or-lose-it implementation, switch queuing and buffering architectures, and parameter selection) are implementation specific, and are an area for vendor differentiation. In this dissertation, we address issues in several of these non-standard areas. Towards this direction, the next chapter describes the design goals of switch algorithms.

2.8 Summary

We have presented the source, destination, switch rules, and parameters of the ABR traffic management model. Like any other standard, these rules reflect a compromise between several differing views. As observed, a key component in the traffic management specification is the switch scheme which calculates the feedback to be given to the sources.

The work presented in this dissertation helped develop the source, switch and destination rules of the ATM Forum Traffic Management standard. Specifically, we study and propose designs for switch rate feedback calculation, source rule design (especially SES Rules 3, 5, 9, 11, and 13), and address application performance and switch scheme implementation tradeoffs.

CHAPTER 3

SWITCH SCHEME DESIGN ISSUES

The most important part in ABR traffic management framework is the switch feedback calculation algorithm. The switch algorithm calculates the feedback to be given to the sources. We use the following switch model for further discussion:

3.1 Switch Model

A switch interconnects multiple links and supports multiple ports, typically an input port or/and an output port per-link. Each port may have some buffers associated with it. It is possible to put the buffers exclusively at the input port (an input-buffered architecture), exclusively at the output port (an output-buffered architecture), or at both the input and output ports. Popular switch architectures tend towards being exclusively output buffered [95] due to its superior performance when compared to input buffered switches. We choose to focus on output buffered switch architectures.

Buffers may be logically partitioned into queues, which are scheduled using a specific discipline. Queuing and scheduling at the buffers may be handled in a First In First Out (FIFO) manner where all the cells coming to the port are put into a common buffer (and later serviced) in the order they arrived at the port. On the other hand, a

complex method like per-VC queuing and scheduling (a separate queue for every VC) may be used. Minimally, a switch will have a separate FIFO queue for every traffic class supported (CBR, VBR, ABR, and UBR classes). The rate-based framework defined in the ATM Traffic Management 4.0 standards allows the switch designers total flexibility in choosing the buffer allocation, queuing, and scheduling policy. This was one of the key features that led to its acceptance compared to the credit-based proposal which required per-VC queuing and scheduling to be implemented at every switch. We assume a model of an output buffered switch implementing per-class queues at every output port. The ABR congestion control algorithm runs at every output port's ABR queue.

The capacity of the output link is assumed to be shared between the "higher priority" classes (constant bit rate (CBR), real-time variable bit rate (rt-VBR), and non-real time variable bit rate (nrt-VBR)) and the available bit rate (ABR) class. We bunch the higher priority classes into one conceptual class called "VBR." Link bandwidth is first allocated to the VBR class and the remaining bandwidth, if any, is given to ABR class traffic. The capacity allocated to ABR is called ABR capacity. We study the problem of controlling the ABR capacity and ABR queues of the output port. Note that, it is possible to have a number of separate subclasses within ABR which are queued and serviced separately. In such a case, the switch algorithm applies to each ABR class queue.

3.2 ABR Switch Scheme Goals

The ABR service was initially designed to achieve high throughput with control over cell loss, since early data users reported heavy loss of cells and throughput.

But the development of feedback mechanisms has led to an expansion of these goals. Specifically, switches can give feedback such that the sources are treated in a "fair" manner. Further, switches can control the queuing delays, provide a combination of quick response time, and a stable steady state. Switches today can also compensate efficiently for errors due to variation in network load and capacity. In this section, we will make these goals more concrete, and use these goals as a reference to evaluate switch schemes.

3.2.1 Congestion Avoidance

The goal of congestion avoidance is to bring the network to an operating point of high throughput and low delay [63]. Typically, there is a tradeoff between the link utilization and the switch queuing delay. For low utilization, the switch queue is small, and the delay is small. Once utilization is very high, the queues grow. Finally, when the queue size exceeds the available buffer size, cells are dropped. In this state, though the link utilization may be high (since the queue length is greater than zero), the effective end-to-end throughput is low (since several packets do not reach the destination). In general, we may replace the terms "utilization" and "switch queuing delay" can be replaced by "throughput" and "end-to-end delay" respectively when we consider entire networks.

Figure 3.1 shows the throughput and delay with varying load in the network. The operating point which has a utilization close to 100% and moderate delays is called the *knee* of the delay-throughput curve. Formally, the knee is the point where the ratio of the bottleneck throughput to bottleneck response time (delay) as a function of input load is maximized. In a network which is in a ideal operating point, typical utilization

graphs have a steady state with controlled oscillations close to 100% utilization, and typical queue length graphs have a steady state with controlled oscillations close to zero queue length. This is also illustrated in figure 3.1.

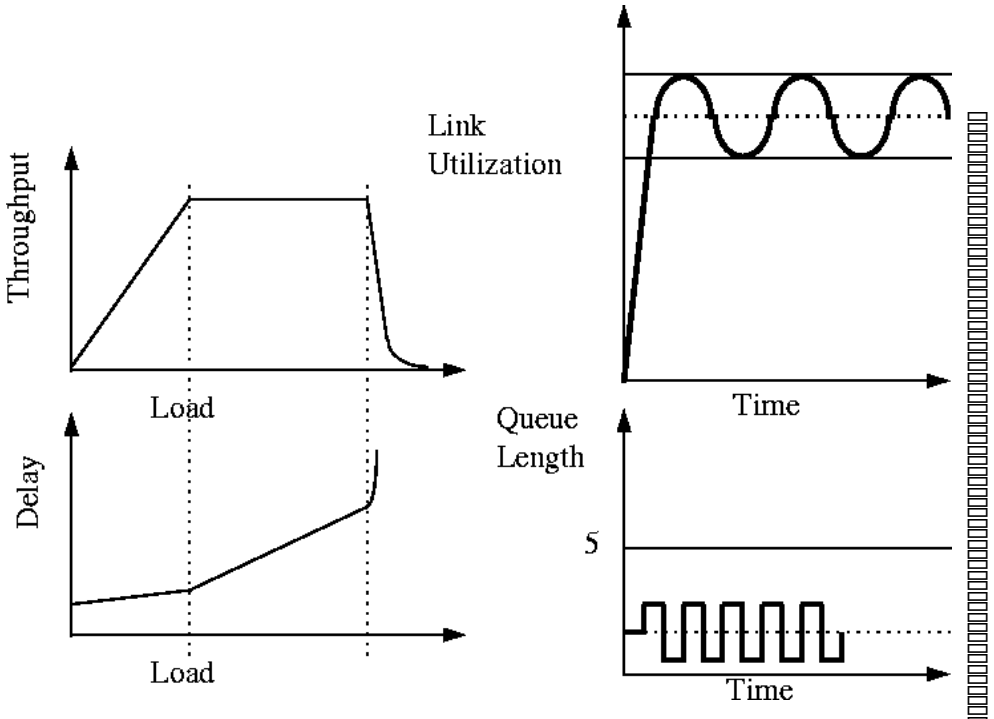

Figure 3.1: Throughput versus delay

The "knee" is a good choice for an operating point for congestion control schemes. Schemes which operate at (or close to) this point are called *congestion avoidance schemes*. Congestion avoidance can be considered as one notion of "efficiency" in the ABR service.

In the terminology of section 1.4.1, the goal could be stated as maximizing the steady state (or average) utilization, $\int \rho_a(t)dt$ while minimizing the average queue length, $\int q_a(t)dt$.

If the load is increased beyond the knee, the delay increases as a function of the load, but there is always a non-zero queuing delay. However, beyond a certain delay, the throughput drops again and the (end-to-end) delays rise sharply (due to higher layer mechanisms like timeout and retransmission). This point is called the "cliff" of the delay-throughput curve. The cliff is a highly unstable operating point, and has the disadvantage of large queuing delays.

Operating points between the knee and the cliff (as shown in figure 3.2) may also be desirable. Such operating points keep the network at 100% utilization in steady state and maintain a "pocket of queues" in the buffer. Further, as the queues grow beyond the desired value, additional capacity is allocated to drain the queues. In other words, the scheme has control over the queuing delay in the steady state, and the queue drain rate under transient conditions. Note that, in general, such an operating point is not very stable for rate-based control, unless the switch uses a function of input load as well in the control. This is because the bottleneck queue length is controlled not by a set of windows (which can at most result in finite queues), but by a set of rates (which can result in infinite queues if not controlled).

Note that this new operating problem poses a new control problem. In the terminology of section 1.4.1, we may state the new goal as maximizing the steady state (average) utilization, $\int \rho_a(t)dt$ while minimizing the difference of the queue length from a desired queue length, $|q_a(t) - q_{desired}|$.

3.2.2 Fairness

In a shared environment, the throughput for a source depends upon the demands by other sources. Ideally, a scheme should equally divide the available bandwidth

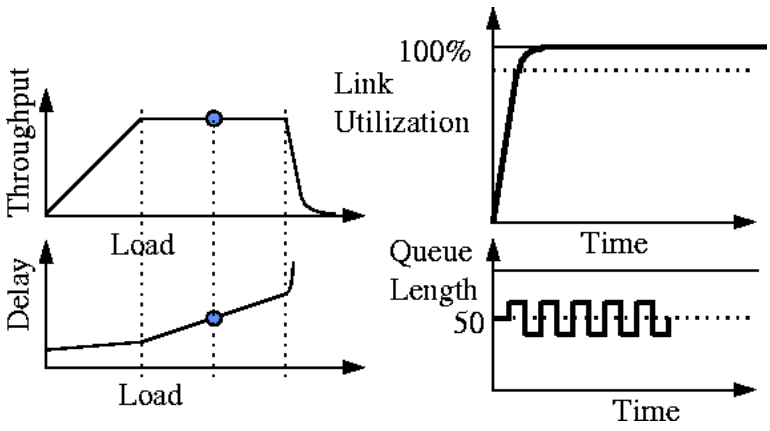

Figure 3.2: Operating point between the "knee" and the "cliff"

among sources which can use the bandwidth (active, unconstrained sources). The most commonly used criterion for what is the correct share of bandwidth for a source in a network environment, is the so called "max-min allocation [46]." It provides the maximum possible bandwidth to the source receiving the least among all contending sources.

Mathematically, the optimality criterion can be written as follows [15]:

Given a configuration with n contending sources, suppose the ith source gets a bandwidth η_i. The allocation vector $\{\eta_1, \ldots, \eta_n\}$ is feasible if all link load levels are less than or equal to 100%.

Consider vector $a = (a_1, \ldots, a_n)$. Let $\hat{a} = (\hat{a_1}, \ldots, \hat{a_n})$ be a permutation of a such that $\hat{a_i} \leq \hat{a_j}$ if $i < j$. Vector b is said to be lexicographically greater than a if either $\hat{a_1} < \hat{b_1}$ or $\exists 1 \leq j \leq n,\ s.t.\ \hat{a_i} = \hat{b_i}\ \forall 1 \leq i < j$ and $\hat{a_j} < \hat{b_j}$.

A vector $\eta = (\eta_1, \ldots, \eta_n)$ is a *max-min* fair vector if it is a feasible vector and it is lexicographically greater than any such feasible vector.

Observe that this definition means that the optimal vector is such that its smallest component is maximized over all feasible vectors, then, given the value of the smallest component, the next smallest component is maximized, etc.

In other words, we know that the total number of feasible vectors is infinite. For each allocation vector, the source that is getting the least allocation is in some sense, the "unhappiest source." Given the set of all feasible vectors, find the vector that gives the maximum allocation to this unhappiest source. Actually, the number of such vectors is also infinite although we have narrowed down the search region considerably. Now we take this "unhappiest source" out and reduce the problem to that of remaining $n-1$ sources operating on a network with reduced link capacities. Again, we find the unhappiest source among these $n-1$ sources, give that source the maximum allocation and reduce the problem by one source. We keep repeating this process until all sources have been given the maximum that they could get. In summary, a network is considered to be in a state of max-min fairness if it is impossible to increase the rate of any session without decreasing the rate of sessions whose rate is equal or smaller.

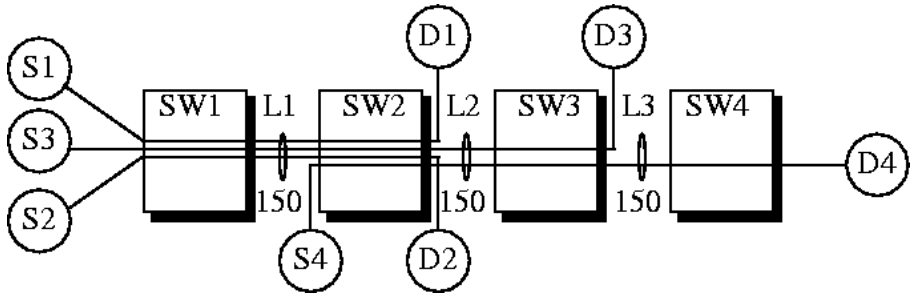

Figure 3.3: Sample configuration for max-min fairness

The following example illustrates the above concept of max-min fairness. Figure 3.3 shows a network with four switches connected via three 150 Mbps links. Four VCs are setup such that the first link L1 is shared by sources S1, S2, and S3. The second link is shared by S3 and S4. The third link is used only by S4. Let us divide the link bandwidths fairly among contending sources. On link L1, we can give 50 Mbps to each of the three contending sources S1, S2, and S3. On link L2, we would give 75 Mbps to each of the sources S3 and S4. On link L3, we would give all 155 Mbps to source S4. However, source S3 cannot use its 75 Mbps share at link L2 since it is allowed to use only 50 Mbps at link L1. Therefore, we give 50 Mbps to source S3 and construct a new configuration shown in Figure 3.4, where Source S3 has been removed and the link capacities have been reduced accordingly. Now we give 1/2 of the link L1's remaining capacity to each of the two contending sources: S1 and S2; each gets 50 Mbps. Source S4 gets the entire remaining bandwidth (100 Mbps) of link L2. Thus, the fair allocation vector for this configuration is (50, 50, 50, 100). This is the max-min allocation.

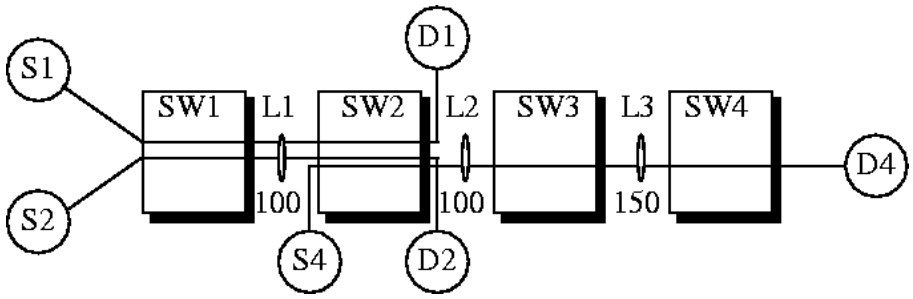

Figure 3.4: Configuration after removing VC 3

Notice that max-min allocation is both fair and efficient. It is fair in the sense that all sources get an equal share on every link provided that they can use it. It is efficient in the sense that each link is utilized to the maximum load possible.

When we take the minimum cell rate (MCR) of sources into account, there are several possible optimality criteria. Other criterion such as weighted fairness have been proposed to determine optimal allocation of resources over and above MCR [35].

Abraham and Kumar [1] develop a natural extension of the concept of max-min fair rate allocation to the case of ABR sessions with non-zero MCRs. Specifically, the feasibility condition includes the fact that *every VC's rate is at least its MCR*; the max-min criteria is the same: the network is considered to be in a state of max-min fairness if it is impossible to increase the rate of any session, while maintaining feasibility, without decreasing the rate of sessions whose rate is equal or smaller. The characterization in terms of rate vectors is also the same, i.e., a rate vector is max-min fair is it is lexicographically the largest among all feasible rate vectors. The authors also develop centralized and distributed algorithms to achieve this max-min allocation.

Finally, it should be pointed out that all definitions of fairness assume that the traffic sources always have data to send (i.e., are infinite sources). For traffic which is "bursty" (i.e., contains active and idle periods), the concept of fairness is ill-defined. As a heuristic, the definitions should be rephrased in terms of the throughputs achieved by sources. Source throughput is measured over a long time interval (covering many idle and active intervals) and not approximated as a series of instantaneous rate allocations. In other words, "fairness" is a long-term goal. While we

mention this concept of fairness for bursty traffic, we do not use the concept further in this dissertation.

3.3 Stable Steady State

The steady state of the system is a state where the goals of efficiency (congestion avoidance) and fairness (max-min) have been achieved. The scheme should first be able to converge to a steady state from any set of initial conditions, provided that the demand and capacity remain constant. Secondly, once the scheme reaches optimal steady state operation, it should stay close to the optimal operation in spite of asynchrony in feedback/response characteristics of the network. In other words, the steady state oscillations between overloaded and underloaded states should be minimal and bounded. An example of a desirable steady state operation is shown (with respect to congestion avoidance, or efficiency alone) in Figure 3.1. Typical steady states have a small amount of residual bandwidth to drain out transient queues and reach the steady state operation of near-zero queuing delay and high throughput.

3.4 Transient Response

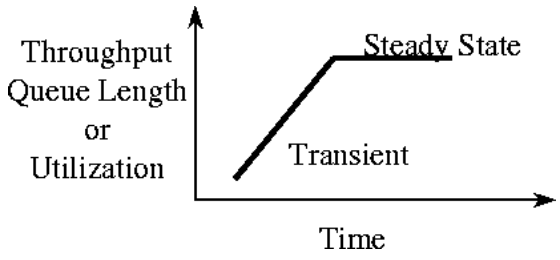

Figure 3.5: Transient vs Steady State Performance

In the real world, the transient response is almost as important as steady state performance. Jain and Routhier [64] point out that most real world traffic is bursty because most sources are transient. Transient response can be tested using transient sources which start after other sources have started and/or stop before the other sources have stopped. Good schemes must be able to respond rapidly to these load transients and achieve optimal performance in the steady state. The difference between steady state and transient performance is shown in figure 3.5.

The transient performance implies that the scheme should converge quickly from overloaded or underloaded states to the steady state, given that the sources can respond to the feedback. Typically, in linear control systems, the transient can either be short and sharp (i.e., resulting in large transient queues), or slow and smooth (underutilization and near-zero queues). On the other hand, if the scheme is optimized just for transients, the steady state may exhibit oscillations. We desire a scheme which optimizes both the transient and the steady state performance, i.e., quickly converges to a solid steady state from any initial conditions, and drains the queues produced in the transient phase rapidly. The "congestion avoidance" steady state is finally reached when both the rates and the queues stabilize.

3.5 Miscellaneous Goals

Robustness and Handling High Variation Workloads: Another complexity issue for the ABR service (and in general for data service classes in high speed integrated service networks) is the fact the available capacity for data traffic is variable. Older networks typically had constant capacity links dedicated for data transmission and the capacity was also lower. As a result, the congestion

control problem was also simpler. Specifically, the ABR service needs to provide good performance given that the variation in both the demand and capacity is high. Demand is usually variable because of the bursty nature of data traffic. Capacity varies because of the presence of higher priority CBR or VBR traffic classes. Further, the scheme must be robust to adapt to conditions like delayed or lost feedback.

Implementation Cost/Performance tradeoffs: The scheme should provide several tradeoff points for implementation incurring different costs. The basic version of the scheme should not require expensive implementation (for example, per-VC queuing as in the case of credit-based schemes). The scheme should be flexible enough to perform well for the target workload scenario, at an acceptable cost.

Scalability: The scheme should not be limited to a particular range of speed, distance, number of switches, or number of VCs. Typical parameters which have scalability implications are: the amount of buffers required, the buffer allocation, queuing and scheduling policy required, the number of switch algorithm operations required per-control cell, and the convergence time (time taken to reach a steady state) of the switch algorithm.

Implementation, Space and Time Complexity: The scheme should be simple to implement - it should not require measurements or logic which are expensive. Further, the amount of memory required for the scheme should be minimal. The best possibility is to have a constant space (or O(1) space complexity) algorithm which utilizes constant space irrespective of the number of VCs setup

or active. The Phantom algorithm [3] is an example of an O(1) space complexity algorithm. If the algorithm is not O(1) in space, it should utilize no more than a constant amount of space per connection (extra space in the VC table) to store per-VC parameters.

The next requirement is for the algorithm to execute a constant number of steps to calculate feedback, especially when an RM cell is being processed. Such an algorithm is said to have an O(1) time complexity. Some algorithms like those developed in this dissertation may do some simple O(N) operations in the background (like the averaging of measured quantities, clearing of bits etc). Such operations can also be efficiently compiled on parallel architectures to have a lower execution complexity like O(log N). However, the algorithm should not require complex implementations like per-VC queuing and scheduling.

3.6 ABR Switch Scheme Limitations

ABR switch algorithms are limited in the following respects:

- **Initial Cell Rate of Sources**: Sources negotiate the Initial Cell Rate (ICR) parameter from the switches along the path during connection setup. Switch algorithms attempt to allocate the available capacity among the currently active sources. In practice, though a large number of VCs are setup, only a few are active at any time. If a number of inactive VCs suddenly become active, and they send data at the negotiated ICR. This may cause a buffer overflow at the switch before the feedback reaches the sources, and the zero loss goal is not met. It is possible that the negotiated Cell Loss Ratio (CLR) may be violated under such conditions. This is an inherent tradeoff in ABR, but the probability

of such an occurrence is expected to be low. For high latency paths, the cell loss can be controlled using source end system parameters (open loop control) during the first round trip. ICR negotiation for ABR is a connection admission control (CAC) problem.

- **Time lag for feedback to be effective**: Observe that a switch does not give rate feedback with every cell. In particular, a feedback may be given in an RM cell traversing in the forward or reverse direction. Sometimes the switch experiences a load, but may not find RM cells to give the feedback. Again, the result is that the feedback is delayed. It is also possible that the load disappears when the RM cell actually arrives. In summary, there is a time lag between the switch experiencing a load, giving feedback, and experiencing the new load due to the feedback. There are three components which affect this time lag:

  Round trip time and Feedback delay: There is a non-zero delay between the switch giving feedback, the sources receiving the feedback, and the switch experiencing the new load.

  The *round trip time (RTT)* is the time taken by a cell to traverse the path from the source to the destination and back. It includes the transmission, propagation and the queuing delays. The sum of the transmission and propagation delays is called fixed round trip time (FRTT).

  The time between the switch giving the feedback and it experiencing the load due to the feedback is called *feedback delay (FD)*. In general, the feedback delay is less than or equal to the round trip time. The shortest feedback delay (SFD) is twice the propagation and transmission delay from

the source to the switch. The feedback delay equals RTT for sources sending data after idle periods, because that is when these sources get their first feedback for the new burst. The feedback delay equals SFD for sources which are already active, and new sources overload the switch.

Inter-RM Cell Time (IRCT): The switches can give feedback in every RM cell they see, and no faster. Hence, the time between successive RM cells determines the rate of feedback. This is true once there is a continuous flow of RM cells (the control loop is set up). The inter RM cell time (IRCT) is not constant, but a function of the source rate. Specifically, it is the inverse of the rate of RM cells, which in turn is a small fraction (1/32) of the source's rate. The IRCT is large when the source rates are small. The IRCT and the switch averaging interval (see the next item) is usually the dominant factor in local area networks (LANs) in determining the time lag for feedback to be effective (note that RTTs and SFDs are small in LANs). Further, in LANs, due to asynchrony sources with smaller IRCTs get feedback faster.

Switch Averaging Interval (AI): The switches usually measure certain quantities which are then used to calculate rate feedback. These quantities are called "metrics." For example, switches typically measure the ABR capacity to be shared among contending sources. Some switch algorithms may measure other quantities like the number of active ABR sources, the input rate and the individual rates of the sources. One important concern of the switch algorithm is to maintain the correlation of the measured quantities ("control") and the "feedback" given. Our algorithm, ERICA achieves this

by giving only one feedback per measurement (i.e., per averaging interval). As a result, the length of the averaging interval determines how often new feedback is given to sources. Shorter averaging intervals result in more feedback, but also more variation in feedback. Longer intervals impact the response time to load changes, but provides more stable feedback in the presence of asynchrony, and heterogeneous RTTs and FDs. The averaging interval length and the IRCT are the dominant factors in LANs (and for sources having short SFDs and RTTs).

In general, the time required for convergence due to a change in load, assuming no further changes in load is depends upon the RTTs of the newly active connections, the feedback delays of already active connections, the length of the averaging interval and the inter-RM Cell time of all connections.

- **ACR Retention** by sources: It is possible that a source gets a high allocation, becomes idle and uses the (now stale) high allocation later when the network conditions have changed. This is called ACR Retention. When multiple sources use their stale allocations simultaneously, buffers may overflow at the switch before feedback is effective. While there are some source policies which can control this, the problem is not eliminated. The switches can gradually decrease a source's allocation (down to ICR) if it detects inactivity, but such policies can reduce the average response time and throughput of bursty sources over any time scale. There are several solutions to this problem (described in chapter 7, each with a set of side effects.

- **Measurement Inaccuracies**: Measurement and the use of metrics is sometimes preferred over the use of parameters to characterize the system and calculate the feedback. This is because, measurement gives real data as opposed to assumed values due to parameters. However, measurement can introduce variance in the system because of inaccuracies during measurement. The more the number of metrics, the more the effect of variance. Variance can be reduced by taking averages of quantities rather than instantaneous values. Averages taken over an interval may still not capture certain conditions in the input workload. This may cause unnecessary queues or rate fluctuations and limit the accuracy with which the goals are achieved. Compensation for these measurement errors must be provided in the algorithm. Interestingly enough, such compensation requires the use of parameters. For example, the drain capacity may be parametrically increased when queues increase without control, due to measurement errors.

- **Limitations of Parametric Control**: The main problem with parameters is to find optimal sets of parameters for the different workload conditions. Even with optimal sets, it may not be possible to achieve optimal performance, because of the tradeoffs in the design of the parameters. For example, a parameter may control the maximum rate increase per feedback. This parameter limits the convergence time from underload. When the input traffic pattern is not known and the wrong parameter sets are chosen, the performance can degrade drastically. It is hence important to minimize the set of parameters, understand their effects and make the parameters easily settable, and design the scheme to provide acceptable performance even for slightly mistuned parameters.

BIBLIOGRAPHY

[1] Santosh P. Abraham and Anurag Kumar. Max-Min Fair Rate Control of ABR Connections with Nonzero MCRs. *IISc Technical Report*, 1997.

[2] Yehuda Afek, Yishay Mansour, and Zvi Ostfeld. Convergence Complexity of Optimistic Rate Based Flow Control Algorithms. In *28th Annual Symposium on Theory of Computing (STOC)*, pages 89–98, 1996.

[3] Yehuda Afek, Yishay Mansour, and Zvi Ostfeld. Phantom: A Simple and Effective Flow Control Scheme. In *Proceedings of the ACM SIGCOMM*, pages 169–182, August 1996.

[4] Anthony Alles. ATM Internetworking. White paper, Cisco Systems, http://www.cisco.com, May 1995.

[5] G.J. Armitage and K.M. Adams. ATM Adaptation Layer Packet Reassembly during Cell Loss. *IEEE Network Magazine*, September 1993.

[6] Ambalavanar Arulambalam, Xiaoqiang Chen, and Nirwan Ansari. Allocating Fair Rates for Available Bit Rate Service in ATM Networks. *IEEE Communications Magazine*, 34(11):92–100, November 1996.

[7] A.W.Barnhart. Changes Required to the Specification of Source Behavior. ATM Forum 95-0193, February 1995.

[8] A.W.Barnhart. Evaluation and Proposed Solutions for Source Behavior # 5. ATM Forum 95-1614, December 1995.

[9] A. W. Barnhart. Use of the Extended PRCA with Various Switch Mechanisms. ATM Forum 94-0898, 1994.

[10] A. W. Barnhart. Example Switch Algorithm for TM Spec. ATM Forum 95-0195, February 1995.

[11] J. Bennett, K. Fendick, K.K. Ramakrishnan, and F. Bonomi. RPC Behavior as it Relates to Source Behavior 5. ATM Forum 95-0568R1, May 1995.

[12] J. Bennett and G. Tom Des Jardins. Comments on the July PRCA Rate Control Baseline. ATM Forum 94-0682, July 1994.

[13] J. Beran, R. Sherman, M. Taqqu, and W. Willinger. Long-Range Dependence in Variable-Bit-Rate Video Traffic. *IEEE Transactions on Communications*, 43(2/3/4), February/March/April 1995.

[14] U. Black. *ATM: Foundation for Broadband Networks*. Prentice Hall, New York, 1995.

[15] P. E. Boyer and D. P. Tranchier. A reservation principle with applications to the atm traffic control. *Computer Networks and ISDN Systems*, 1992.

[16] D. Cavendish, S. Mascolo, and M. Gerla. SP-EPRCA: an ATM Rate Based Congestion Control Scheme basedon a Smith Predictor. Technical report, UCLA, 1997.

[17] Y. Chang, N. Golmie, L. Benmohamed, and D. Siu. Simulation study of the new rate-based eprca traffic management mechanism. *ATM Forum 94-0809*, 1994.

[18] A. Charny, G. Leeb, and M. Clarke. Some Observations on Source Behavior 5 of the Traffic Management Specification. ATM Forum 95-0976R1, August 1995.

[19] Anna Charny. An Algorithm for Rate Allocation in a Cell-Switching Network with Feedback. Master's thesis, Massachusetts Institute of Technology, May 1994.

[20] Anna Charny, David D. Clark, and Raj Jain. Congestion control with explicit rate indication. In *Proceedings of the IEEE International Communications Conference (ICC)*, June 1995.

[21] D. Chiu and R. Jain. Analysis of the Increase/Decrease Algorithms for Congestion Avoidance in Computer Networks. *Journal of Computer Networks and ISDN Systems*, 1989.

[22] Fabio M. Chiussi, Ye Xia, and Vijay P. Kumar. Dynamic max rate control algorithm for available bit rate service in atm networks. In *Proceedings of the IEEE GLOBECOM*, volume 3, pages 2108–2117, November 1996.

[23] D.P.Heyman and T.V. Lakshman. What are the implications of Long-Range Dependence for VBR-Video Traffic Engineering ? *ACM/IEEE Transactions on Networking*, 4(3):101–113, June 1996.

[24] Harry J.R. Dutton and Peter Lenhard. *Asynchronous Transfer Mode (ATM) Technical Overview*. Prentice Hall, New York, 2nd edition, 1995.

[25] H. Eriksson. MBONE: the multicast backbone. *Communications of the ACM*, 37(8):54–60, August 1994.

[26] J. Scott et al. Link by Link, Per VC Credit Based Flow Control. ATM Forum 94-0168, 1994.

[27] L. Roberts et al. New pseudocode for explicit rate plus efci support. *ATM Forum 94-0974*, 1994.

[28] M. Hluchyj et al. Closed-loop rate-based traffic management. *ATM Forum 94-0438R2*, 1994.

[29] M. Hluchyj et al. Closed-Loop Rate-Based Traffic Management. ATM Forum 94-0211R3, April 1994.

[30] S. Fahmy, R. Jain, S. Kalyanaraman, R. Goyal, and F. Lu. On source rules for abr service on atm networks with satellite links. In *Proceedings of First International Workshop on Satellite-based Information Services (WOSBIS)*, November 1996.

[31] Chien Fang and Arthur Lin. A Simulation Study of ABR Robustness with Binary-Mode Switches: Part II. ATM Forum 95-1328R1, October 1995.

[32] Chien Fang and Arthur Lin. On TCP Performance of UBR with EPD and UBR-EPD with a Fair Buffer Allocation Scheme. ATM Forum 95-1645, December 1995.

[33] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, and T. Berners-Lee. Hypertext Transfer Protocol – HTTP/1.1. Request For Comments, RFC 2068, January 1997.

[34] ATM Forum. http://www.atmforum.com.

[35] ATM Forum. The ATM Forum Traffic Management Specification Version 4.0. ftp://ftp.atmforum.com/pub/approved-specs/af-tm-0056.000.ps, April 1996.

[36] M. Garrett and W. Willinger. Analysis, modeling, and generation of self-similar vbr video traffic. In *Proceedings of the ACM SIGCOMM*, August 1994.

[37] Matthew S. Goldman. Variable Bit Rate MPEG-2 over ATM: Definitions and Recommendations. ATM Forum 96-1433, October 1996.

[38] Rohit Goyal, Raj Jain, Shiv Kalyanaraman, Sonia Fahmy, and Seong-Cheol Kim. Performance of TCP over UBR+. ATM Forum 96-1269, October 1996.

[39] Rohit Goyal, Raj Jain, Shiv Kalyanaraman, Sonia Fahmy, Bobby Vandalore, Xiangrong Cai, and Seong-Cheol Kim. Selective Acknowledgements and UBR+ Drop Policies to Improve TCP/UBR Performance over Terrestrial and Satellite Networks. ATM Forum 97-0423, April 1997.

[40] Rohit Goyal, Raj Jain, Shiv Kalyanaraman, Sonia Fahmy, Bobby Vandalore, Xiangrong Cai, and Seong-Cheol Kim. Selective Acknowledgements and UBR+ Drop Policies to Improve TCP/UBR Performance over Terrestrial and Satellite Networks. ATM Forum 97-0423, April 1997.

[41] M. Grossglauser, S.Keshav, and D.Tse. RCBR: a simple and efficient service for multiple time-scale traffic. In *Proceedings of the ACM SIGCOMM*, August 1995.

[42] S. Hrastar, H. Uzunalioglu, and W. Yen. Synchronization and de-jitter of mpeg-2 transport streams encapsulated in aal5/atm. In *Proceedings of the IEEE International Communications Conference (ICC)*, volume 3, pages 1411–1415, June 1996.

[43] D. Hughes and P. Daley. More abr simulation results. *ATM Forum 94-0777*, 1994.

[44] D. Hunt, Shirish Sathaye, and K. Brinkerhoff. The realities of flow control for abr service. *ATM Forum 94-0871*, 1994.

[45] Van Jacobson. Congestion avoidance and control. In *Proceedings of the ACM SIGCOMM*, pages 314–329, August 1988.

[46] J. Jaffe. Bottleneck Flow Control. *IEEE Transactions on Communications*, COM-29(7):954–962, 1980.

[47] R. Jain. A delay-based approach for congestion avoidance in interconnected heterogeneous computer networks. *Computer Communications Review*, 19.

[48] R. Jain. A timeout-based congestion control scheme for window flow-controlled networks. *IEEE Journal on Selected Areas in Communications*, 1986.

[49] R. Jain. A comparison of hashing schemes for address lookup in computer networks. *IEEE Transactions on Communications*, 1992.

[50] R. Jain. The eprca+ scheme. *ATM Forum 94-0988*, 1994.

[51] R. Jain. The osu scheme for congestion avoidance using explicit rate indication. *ATM Forum 94-0883*, 1994.

[52] R. Jain. Atm networking: Issues and challenges ahead. *Engineers Conference, InterOp+Network World*, 1995.

[53] R. Jain. Congestion control and traffic management in atm networks: Recent advances and a survey. *Computer Networks and ISDN Systems*, 1995.

[54] R. Jain, D. Chiu, and W. Hawe. A quantitative measure of fairness and discrimination for resource allocation in shared systems. *DEC TR-301*, 1984.

[55] R. Jain, D. Chiu, and W. Hawe. A quantitative measure of fairness and discrimination for resource allocation in shared systems. *DEC TR-301*, 1984.

[56] R. Jain, S. Kalyanaraman, R. Goyal, S. Fahmy, and F. Lu. A Fix for Source End System Rule 5. ATM Forum 95-1660, December 1995.

[57] R. Jain, S. Kalyanaraman, R. Goyal, S. Fahmy, and F. Lu. Erica+: Extensions to the erica switch algorithm. *ATM Forum 95-1145R1*, 1995.

[58] R. Jain, S. Kalyanaraman, and R. Viswanathan. Method and apparatus for congestion management in computer networks using explicit rate indication. *U. S. Patent application filed (S/N 307, 375),*, 1994.

[59] R. Jain, S. Kalyanaraman, and R. Viswanathan. The transient performance: Eprca vs eprca++. *ATM Forum 94-1173*, 1994.

[60] R. Jain, S. Kalyanaraman, R. Viswanathan, and R. Goyal. A sample switch algorithm. *ATM Forum 95-0178R1*, 1995.

[61] R. Jain, K. Ramakrishnan, and D Chiu. Congestion avoidance scheme for computer networks. *U.S. Patent #5377322,*, 1994.

[62] R. Jain and K. K. Ramakrishnan. Congestion avoidance in computer networks with a connectionless network layer: Concepts, goals, and methodology. *Proc. IEEE Computer Networking Symposium*, 1988.

[63] R. Jain, K. K. Ramakrishnan, and D. M. Chiu. Congestion Avoidance in Computer Networks with a Connectionless Network Layer. Technical Report DEC-TR-506, Digital Equipment Corporation, August 1987.

[64] R. Jain and S. Routhier. Packet Trains - Measurement and a new model for computer network trafic. *IEEE Journal of Selected Areas in Communications,*, 1986.

[65] Raj Jain. Congestion Control in Computer Networks: Issues and Trends. *IEEE Network Magazine*, pages 24–30, May 1990.

[66] Raj Jain. *The Art of Computer Systems Performance Analysis*. John Wiley & Sons, 1991.

[67] Raj Jain. Myths about Congestion Management in High-speed Networks. *Internetworking: Research and Experience*, 3:101–113, 1992.

[68] Raj Jain. ABR Service on ATM Networks: What is it? Network World, 1995.

[69] Raj Jain. Congestion Control and Traffic Management in ATM Networks: Recent advances and a survey. *Computer Networks and ISDN Systems Journal*, October 1996.

[70] Raj Jain, Sonia Fahmy, Shivkumar Kalyanaraman, Rohit Goyal, and Fang Lu. More Straw-Vote Comments: TBE vs Queue sizes. ATM Forum 95-1661, December 1995.

[71] Raj Jain, Shiv Kalyanaraman, Rohit GOyal, and Sonia Fahmy. Source Behavior for ATM ABR Traffic Management: An Explanation. *IEEE Communications Magazine*, 34(11), November 1996.

[72] Raj Jain, Shivkumar Kalyanaraman, Sonia Fahmy, and Fang Lu. Bursty ABR Sources. ATM Forum 95-1345, October 1995.

[73] Raj Jain, Shivkumar Kalyanaraman, Sonia Fahmy, and Fang Lu. Out-of-Rate RM Cell Issues and Effect of Trm, TOF, and TCR. ATM Forum 95-973R1, August 1995.

[74] Raj Jain, Shivkumar Kalyanaraman, Sonia Fahmy, and Fang Lu. Straw-Vote comments on TM 4.0 R8. ATM Forum 95-1343, October 1995.

[75] Raj Jain, Shivkumar Kalyanaraman, Rohit Goyal, Ram Viswanathan, and Sonia Fahmy. Erica: Explicit rate indication for congestion avoidance in atm networks. U.S. Patent Application (S/N 08/683,871), July 1996.

[76] Raj Jain, Shivkumar Kalyanaraman, and Ram Viswanathan. The osu scheme for congestion avoidance in atm networks: Lessons learnt and extensions. *Performance Evaluation Journal*, October 1997. to appear.

[77] Raj Jain and Shivkumar Kalyanaraman Ram Viswanathan. 'method and apparatus for congestion management in computer networks using explicit rate indication. U. S. Patent application (S/N 307,375), SepJuly 1994.

[78] H. Tzeng K. Siu. Intelligent congestion control for abr service in atm networks. *Computer Communication Review*, 24(5):81–106, October 1995.

[79] Lampros Kalampoukas, Anujan Varma, and K.K. Ramakrishnan. An efficient rate allocation algorithm for atm networks providing max-min fairness. In *6th IFIP International Conference on High Performance Networking (HPN)*, September 1995.

[80] Shivkumar Kalyanaraman, Raj Jain, Sonia Fahmy, Rohit Goyal, and Jianping Jiang. Performance of TCP over ABR on ATM backbone and with various VBR traffic patterns. In *Proceedings of the IEEE International Communications Conference (ICC)*, June 1997.

[81] Shivkumar Kalyanaraman, Raj Jain, Rohit Goyal, and Sonia Fahmy. A Survey of the Use-It-Or-Lose-It Policies for the ABR Service in ATM Networks. Technical Report OSU-CISRC-1/97-TR02, Dept of CIS, The Ohio State University, 1997.

[82] D. Kataria. Comments on rate-based proposal. *ATM Forum 94-0384*, 1994.

[83] J.B. Kenney. Problems and Suggested Solutions in Core Behavior. ATM Forum 95-0564R1, May 1995.

[84] Bo-Kyoung Kim, Byung G. Kim, and Ilyoung Chong. Dynamic Averaging Interval Algorithm for ERICA ABR Control Scheme. ATM Forum 96-0062, February 1996.

[85] H. T. Kung. Adaptive Credit Allocation for Flow-Controlled VCs. ATM Forum 94-0282, 1994.

[86] H. T. Kung. Flow Controlled Virtual Connections Proposal for ATM Traffic Management. ATM Forum 94-0632R2, September 1994.

[87] T.V. Lakshman, P.P. Mishra, and K.K. Ramakrishnan. Transporting compressed video over atm networks with explicit rate feedback control. In *Proceedings of the IEEE INFOCOM*, April 1997.

[88] L.G.Roberts. Operation of Source Behavior # 5. ATM Forum 95-1641, December 1995.

[89] Hongqing Li, Kai-Yeung Siu, Hong-Ti Tzeng, Chinatsu Ikeda, and Hiroshi Suzuki. Tcp over abr and ubr services in atm. In *Proceedings of IPCCC'96*, March 1996.

[90] S. Liu, M. Procanik, T. Chen, V.K. Samalam, and J. Ormond. An analysis of source rule # 5. ATM Forum 95-1545, December 1995.

[91] B. Lyles and A. Lin. Definition and preliminary simulation f a rate-based congestion control mechanism with explicit feedback of bottleneck rates. *ATM Forum 94-0708*, 1994.

[92] P. Newman. Traffic Management for ATM Local Area Networks. *IEEE Communications Magazine*, 1994.

[93] P. Newman and G. Marshall. Becn congestion control. *ATM Forum 94-789R1*, 1993.

[94] P. Newman and G. Marshall. Update on becn congestion control. *ATM Forum 94-855R1*, 1993.

[95] Craig Partridge. *Gigabit Networking*. Addison-Wesley, Reading, MA, 1993.

[96] Vern Paxson. Fast Approximation of Self-Similar Network Traffic. Technical Report LBL-36750, Lawrence Berkeley Labs, April 1995.

[97] K. Ramakrishnan and R. Jain. A binary feedback scheme for congestion avoidance in computer networks with connectionless network layer. *ACM Transactions on Computers*, 1990.

[98] K. K. Ramakrishnan, D. M. Chiu, and R. Jain. Congestion Avoidance in Computer Networks with a Connectionless Network Layer. Part IV: A Selective Binary Feedback Scheme for General Topologies. Technical report, Digital Equipment Corporation, 1987.

[99] K. K. Ramakrishnan and P. Newman. Credit where credit is due. *ATM Forum 94-0916*, 1994.

[100] K. K. Ramakrishnan and "Issues with Backward Explicit Congestion Notification based Congestion Control. Issues with backward explicit congestion notification based congestion control. *ATM Forum 94-0231*, 1993.

[101] K. K. Ramakrishnan and J. Zavgren. Preliminary simulation results of hop-by-hop/vc flow control and early packet discard. *ATM Forum 94-0231*, 1994.

[102] K.K. Ramakrishnan, P. P. Mishra, and K. W. Fendick. Examination of Alternative Mechanisms for Use-it-or-Lose-it. ATM Forum 95-1599, December 1995.

[103] L. Roberts. The benefits of rate-based flow control for abr service. *ATM Forum 94-0796*, 1994.

[104] L. Roberts. Enhanced prca (proportional rate-control algorithm). *ATM Forum 94-0735R1*, 1994.

[105] L. Roberts. Rate-based algorithm for point to multipoint abr service. *ATM Forum 94-0772R1*, 1994.

[106] Larry Roberts. Enhanced PRCA (Proportional Rate-Control Algorithm). ATM Forum 94-0735R1, August 1994.

[107] A. Romanov. A performance enhancement for packetized abr and vbr+ data. *ATM Forum 94-0295*, 1994.

[108] Allyn Romanov and Sally Floyd. Dynamics of TCP Traffic over ATM Networks. *IEEE Journal on Selected Areas in Communications*, May 1995.

[109] W. Stallings. Isdn and broadband isdn with frame relay and atm. *ATM Forum 94-0888*, 1995.

[110] Lucent Technologies. Atlanta chip set, microelectronics group news announcement, http://www.lucent.com/micro/news/032497.html.

[111] Christos Tryfonas. MPEG-2 Transport over ATM Networks. Master's thesis, University of California at Santa Cruz, September 1996.

[112] H. Tzeng and K. Siu. A class of proportional rate control schemes and simulation results. *ATM Forum 94-0888*, 1994.

[113] H. Tzeng and K. Siu. Enhanced credit-based congestion notification (eccn) flow control for atm networks. *ATM Forum 94-0450*, 1994.

[114] International Telecommunications Union. http://www.itu.ch.

[115] Bobby Vandalore, Shiv Kalyanaraman, Raj Jain, Rohit Goyal, Sonia Fahmy, Xiangrong Cai, and Seong-Cheol Kim. Performance of Bursty World Wide Web (WWW) Sources over ABR. ATM Forum 97-0425, April 1997.

[116] Bobby Vandalore, Shiv Kalyanaraman, Raj Jain, Rohit Goyal, Sonia Fahmy, and Pradeep Samudra. Worst case TCP behavior over ABR and buffer requirements. ATM Forum 97-0617, July 1997.

[117] L. Wojnaroski. Baseline text for traffic management sub-working group. *ATM Forum 94-0394R4*, 1994.

[118] Gary R. Wright and W. Richard Stevens. *TCP/IP Illustrated, Volume 2.* Addison-Wesley, Reading, MA, 1995.

[119] Lixia Zhang, Scott Shenker, and D.D.Clark. Observations on the dynamics of a congestion control algorithm: The effects of two-way traffic. In *Proceedings of the ACM SIGCOMM*, August 1991.