The University of Kansas

Information and
Telecommunication
Technology Center

Technical Report

# A Framework for Bandwidth Management and Call Admission Control in ATM Networks

Kunyan Liu and David W. Petr

ITTC-FY1997-TR-11230-03

May 1997

Project Sponsor:
Sprint Corporation

# Abstract

One critical issue in traffic management of ATM networks is, to achieve the balance between two seemingly conflicting goals: to guarantee the Quality of Service (QoS) for all service classes, while still allowing enough statistical sharing of bandwidth so that the network is efficiently utilized. Guaranteeing QoS requires traffic isolation, as well as allocation of enough network resource (e.g. buffer space and bandwidth) to each call. However, the statistical bandwidth sharing means the network resource should be occupied on-demand, leading to less traffic isolation and minimal resource allocation.

Due to the complicated nature of this problem, no single measure can possible provide the proper solution. Rather, a comprehensive solution incorporating the schemes dealing with different aspects of the problem is necessary.

In this thesis, we try to address the problem by proposing a framework for bandwidth management and connection admission control, into which further detailed control measures may be added later. In the first part of this thesis, we propose and evaluate a network-wide bandwidth management architecture in which an appropriate compromise between the two conflicting goals is achieved. Specifically, the bandwidth management framework consists of a network model and a network-wide bandwidth allocation and sharing strategy. Implementation issues related to the framework are discussed. For real time applications, we obtain maximum queueing delay and queue length which are important in buffer design and VP (Virtual Path) routing.

Furthermore, we propose a measurement-based CAC strategy based on the proposed architecture. We first discuss how to obtain an accurate description (UPC parameters) of user traffic by using trace-based measurement in conjunction with on-line measurement and dynamic renegotiation. The results show that the proposed strategy is reliable and simple to implement. We then move on to examine the possibility and methodology for exploiting the effect of statistical multiplexing in resource allocation to achieve higher network resource utilization.

# Contents

# List of Figures

# List of Tables

# Section 1

# Introduction

## 1.1 Introduction to ATM Traffic Management

In today's telecommunication and data communication industry, Asynchronous Transfer Mode (ATM) has been generally accepted as the most promising network technology for a wide range of applications, including the future telecommunication networks, Broadband ISDN.

The basic idea of ATM is to segment and multiplex the user traffic into streams of small, fixed-size cells and transfer them through the network. Compared with other technologies such as circuit switching, X.25 packet switching and frame relay, ATM has several important features that makes it suitable to support a wide variety of services which is expected in the future communication world, for example,

- Support high speed switching through the use of small, fixed cell size.
- Support for statistical multiplexing, thus higher network utilization may be achieved compared with traditional circuit switching technology.
- Support Quality of Service (QoS) through connection-oriented technology.
- Support arbitrary, non-hierarchical bandwidth assignment, hence services with a drastically different bitrate range can all be accommodated.

However, there are also many new challenges brought forth by the introduction of ATM in almost every area of telecommunication and networking technology. Among them, traffic management is considered as a fundamental challenge for the success of ATM technology, and hence has been the subject of vigorous research over the past several years [8] [9]. In the ATM environment, the basic missions of traffic management are:

- Protect the network from congestion.
- Provide QoS guarantee for all services.

- Achieve high utilization of network resources (primarily bandwidth and buffer space).

To provide an overall traffic management solution for ATM networks is a highly complicated issue and involves many different technical areas. In this report, we will focus on two particularly important issues within the scope, *bandwidth management* and *call admission control*[1].

## 1.2 ATM Services and Bandwidth Management

Although the amount of available network bandwidth has dramatically increased over the past few decades, it is still not infinite and unlikely to be so in the foreseeable future. Eventually there will be enough users with enough network activity to use up the bandwidth resource. Therefore, bandwidth management is necessary to share bandwidth among all users and maintain the normal operation of any network.

On the other hand, ATM networks are supposed to be able to support a wide variety of services. As defined by the ATM Forum, the different types of service supported by ATM are categorized into four service classes [5]: Constant Bit Rate (CBR), Variable Bit Rate (VBR), Available Bit Rate (ABR), and Unspecified Bit Rate (UBR). Each service class has its own QoS requirement and traffic characteristics and should be treated individually in terms of bandwidth management and control. According to the ATM Forum, CBR and real-time VBR connections have stringent delay and Cell Loss Ratio (CLR) requirements. Moreover, the CBR service class is designed for circuit switching emulation which requires a constant bandwidth capacity for each call. The traffic rate for a VBR connection may fluctuate around its average rate but not exceed its peak cell rate (PCR). The traffic rate for an ABR connection can be adjusted in real time, and its Minimum Cell Rate (MCR) is specified. An UBR source may send as fast as it desires (up to its PCR), but the network does not guarantee any QoS for it.

From the above arguments, we can see that the bandwidth management in ATM networks has two seemingly conflicting goals, i.e., guaranteeing performance for each service class, while still allowing enough statistical multiplexing so that the network is efficiently utilized. In order to achieve these two goals, the ATM research community has proposed numerous control and management schemes [12] [33] [14] [13]. However, schemes for different purposes are often treated independently and lack the capability of co-operating with each other. What is needed, therefore, is a bandwidth management architecture under which the network can be efficiently utilized, and meanwhile, acceptable QoS can be achieved for all service classes.

Although it seems unlikely to optimize bandwidth management by taking into account all aspects of traffic behavior and performance requirements, we believe it is possible to reach a good compromise between these two goals by adopting a bandwidth management architecture which incorporates:

[1] Also called connection admission control sometimes. In this report, these two terms are considered interchangeable.

- A network model designed with special consideration for traffic management.
- The cell level control schemes such as cell scheduling and buffer management algorithms.
- Effective traffic description and policing functions.

Furthermore, the proposed architecture should be simple enough to implement and yet flexible enough to accommodate high-level traffic control schemes such as feed-back flow-control and connection admission control.

## 1.3 Background of Connection Admission Control

Once an effective underlying bandwidth management architecture is established, further traffic control measures can then be developed accordingly to support specific types of service.

It is widely accepted that a significant portion of traffic in future ATM-based B-ISDN will consist of real-time services, such as voice, multimedia and especially video applications. Currently, the service classes defined by the ATM Forum most appropriate for these applications are Constant Bit Rate (CBR) and Variable Bit Rate (VBR). An important characteristic of most applications of this type is that they are either uncontrollable or will suffer unacceptable quality degradation from forced rate-control. Therefore, Connection Admission Control becomes the primary traffic control scheme for them.

The need to accommodate VBR services presents a new challenge for CAC not found in traditional telecommunication networks. First of all, available resources (bandwidth and buffer) are fixed in amount and limited compared with user demand. On the other hand, the user traffic is ever-changing and QoS requirements can be stringent. This problem becomes even more difficult since so far there is no generic theoretical analysis method available to model the behavior of traffic sources (e.g., the long-range dependent video source [35]) and to predict performance accordingly.

So far, many efforts have been made to find an efficient CAC strategy which includes determining resource requirements for VBR traffic [28] [31] [32] [36] [29] [30]. However, the proposed schemes are often limited to a particular aspect of the problem and lack a simple, generic strategy, which will be the second subject of this report.

## 1.4 Report Outline

In this report, instead of concentrating on any individual traffic control schemes, we try to establish an overall framework for both bandwidth management and connection admission control.

The rest of this report is organized as follows. The first three sections discuss the bandwidth management architecture. Section 2 proposes a network model and the corresponding VP assignment policy; Section 3 presents the bandwidth allocation and sharing strategy and discusses corresponding cell-level schemes and switch architecture. Section 4 evaluates the maximum queueing delay and CLR performance for CBR and VBR service classes which are expected to mainly support real time applications.

The next three sections address the CAC strategy used in this context. Section 5 presents an overview of our CAC strategy as well as the way of determining the initial user traffic descriptors. In section 6, we discuss the issue of UPC dynamic renegotiation and its role in the proposed CAC strategy. Section 7 addresses the problem of exploiting statistical multiplexing in CAC.

Finally, section 8 draws a conclusion for this report.

# Section 2

# Network Model

## 2.1 The Partitioning of Core and Edge Networks

We propose a network model in which the ATM-based B-ISDN is partitioned into core and edge networks as shown in Figure 2.1. The primary function of the edge networks is to provide broad-band access to the user through the UNI and to perform cell switching in the local area. The core network functions as the backbone network carrying concentrated traffic between edge networks. The interface between the core and edge network is provided by special edge nodes (gateways). Note that the core and the edge networks are still part of a unified ATM network, and should be able to cooperate in terms of bandwidth management, congestion control, and other administration issues through network-network interfaces [10] [11].

The design of the core network will apply the Virtual Path (VP) concept in which ATM cells are processed based on Virtual Path Identifier (VPI) values (Figure 2.2). The VP concept [4] [6] has been developed to support semi-permanent connections in a large scale backbone network which transports a large number of simultaneous user calls carried by Virtual Circuits (VCs). A VP starts at an edge gateway and terminates at another edge gateway (see Figure 2.2). In the core network, available network resources, such as bandwidth and buffer space, can be managed simply and efficiently on a per-VP basis. On the other hand, the edge networks will carry a smaller number of simultaneous VCs, and will handle traffic on a call by call basis in order to process call arrivals and to setup and tear down individual VCs using Virtual Circuit Identifier (VCI) values (Figure 2.2).

As stated above, a VP is identified by its VPI values in the core network. There have been two kinds of VPI management methods [15]:

- Global VPI assignment, in which VPIs are managed centrally, each VPI has global significance, and each VPI corresponds to a route in the network. No VPI translation is needed at the core switches.

- Local VPI assignment, in which the VPIs have only local significance associated with each physical link and should be translated at each core switch . A VP is therefore

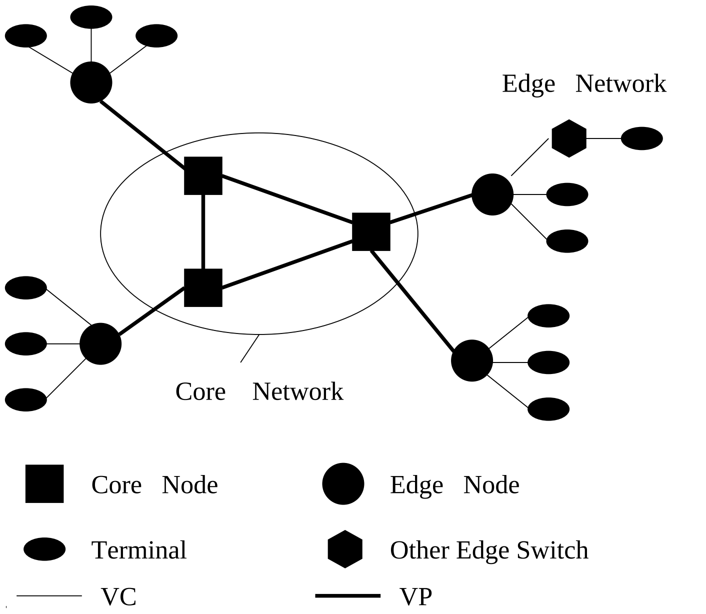


Figure 2.1: Network Model: Core and Edge Concept

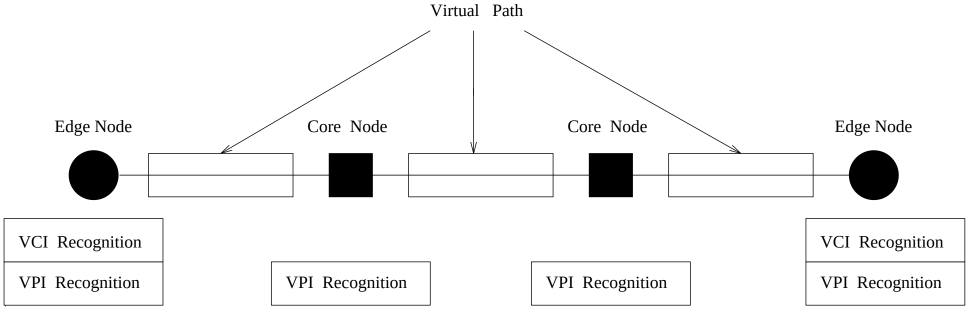


Figure 2.2: Cell Switching

identified by a series of physical link ID and VPI pairs.

Although the first method is simple and easy to implement, it imposes a limit on the total number of VPs within the network. With current 12-bit VPI definition [4], only 4096 VPs can possibly exist. This is far from adequate for a large-scale network. Therefore, we prefer the local VPI assignment method in which, given the ability of identifying each port through its port ID, each core switch may support up to 4096 VPs on each input/output link. Now that VPI has only local importance (to identify a VP from other VPs on the same link), the same VPI can be reused in the network. As a result, the network is able to support any number of VPs, given a route layout such that no more than 4096 VPs exist on a single link. This gives the possibility to support fully-meshed VP network using current fiber network topology.

## 2.2 The VP Assignment Policy

Currently a number of VP layout and assignment schemes have been proposed [16] [17], which differ in the following ways:

- The connectivity of the VP network, i.e., should it be fully-meshed or sparsely connected, such as in a star or ring topology.

- How to map the various services to the VPs. One extreme is to use the same VP for all service classes, thus requiring fewer VPs. However, the task of guaranteeing QoS for all service classes in the VP could be difficult. The opposite is to have a separate VP for each service class, or even for each different QoS requirement within the same service class. Although QoS control is easier in this scheme, the total number of VPs needed can be very large.

We propose a fully-meshed scheme in which there should be at least two VPs assigned between each edge-node-pair (denoted as an Origin-Destination pair, or O-D pair), one for VBR and CBR service, and the other for ABR and UBR service. Other VPs may also exist for alternative routing or other management considerations.

The VP assignment policy described above is based on the following considerations:

- In the fully-meshed VP network, pre-assigned VPs exist between all edge networks, and the core nodes can be easily implemented by ATM cross-connectors. No complicated VC level operations such as add/drop or rerouting are necessary. Meanwhile, even if the number of edge networks grows, the VP network can still scale well given the local VPI management scheme discussed in the last section.

- The mapping of service classes to VPs should be able to achieve a good balance between QoS achievement and complexity. Thus we need to carefully inspect the nature of service classes before determining how to map them into VPs. Real-time VBR [1] and CBR connections have similar performance parameters in terms of delay and CLR. On the other hand, ABR sources are expected to adapt their rates according to network states and do not require stringent delay performance. Separating ABR traffic from the VBR/CBR VP ensures that ABR rate changes do not affect the performance of CBR and VBR service classes. The nature of UBR services indicates that no network resources should be allocated to UBR connections, and consequently, allocating separate VPs to UBR connections is unnecessary. However, the network must provide the necessary isolation (described in the next section) between UBR and other service classes so that the traffic from UBR sources does not affect the performance of other users. Practically, once enough isolation is provided, UBR connections may share the same VP with any other service classes. We choose to integrate UBR with ABR on the same VP because of the similar "best effort" nature for the two service classes.

[1] A Non-real-time VBR connection can be viewed as a Real-time VBR with a large Cell Delay Variation Tolerance (CDVT) parameter. Therefore, Non-real-time VBR VCs can be integrated on VBR/CBR VPs.

# Section 3

# Bandwidth Management Strategy

In order to achieve a successful bandwidth management framework, it is necessary to incorporate efforts at both the cell level and the network design level. In this section we will first introduce the basic concepts of the proposed framework and the cell-level schemes to support them, then look into the network design level issues. Furthermore, we also present a sketch of a possible implementation of the proposed framework.

## 3.1 Basic Ideas: Bandwidth Allocation Vs. Reservation

In traditional telecommunication networks, usually a certain amount of bandwidth is *reserved* for all connections, i.e., each connection will always be given, and only be able to use, the portion of bandwidth explicitly assigned to it. For example, in a TDM system, each connection has (and pays for) its own digital channel and the associated fixed bandwidth. No connection may use the bandwidth on any other channels, even if there is no traffic on them at the time. Since the majority of traffic in those networks is CBR (voice connections), the *reservation-based* scheme is sufficient. However, VBR and "best effort" traffic (ABR and UBR) will play very important roles in ATM networks, and it is difficult, if not impossible, to support these services efficiently by a reservation-based scheme.

In order to achieve both QoS guarantees and high network utilization for all service classes in ATM networks, a new kind of bandwidth management , which we call bandwidth *allocation*, must be introduced. In a bandwidth allocation-based scheme, each connection is allocated a certain amount of bandwidth (which could be zero), and

- Each connection is *guaranteed* to have access to its allocated bandwidth, whenever it has something to send.

- Unused bandwidth is available to other connections.

- Consequently, a connection sometimes can use bandwidth exceeding its allocation, but *only* when other connections are *not* using their allocation.

Note that different services emphasize different aspects of the allocation-based scheme. For example, since the QoS requirements for CBR/VBR connections need to be guaranteed for the duration of the connection, it is necessary to allocate them an amount of bandwidth that will guarantee that the QoS requirement will always be met. On the other hand, ABR connections would be allocated only enough bandwidth to guarantee their minimum cell rate (MCR), since they are supposed to utilize the bandwidth "spared" by CBR and VBR connections. Similarly, UBR connections would likely not be allocated any bandwidth.

## 3.2 Cell-Level Supporting Schemes

### 3.2.1 Cell scheduling schemes

Various cell scheduling schemes, such as Weighted Fair Queueing [18], Round-Robin, and Virtual Clock [19] have been proposed for ATM networks. Among them, Weighted Round Robin(WRR) [20], [21], [22] seems to be the most promising algorithm to support allocation-based bandwidth management.

The basic idea of WRR can be described as follows. There are multiple incoming connections, each of them with a separate queue. One output link is shared among all connections, and the access to it is controlled by a server. The server serves all the queues in the order decided by a circular schedule, in which each queue has a certain number of entries. If the current queue is "inactive", i.e., it does not contain any cell to be served (transmitted), the server will then move to poll the next queue on schedule, until it finds an active connection. Hence the cell slot will not be wasted unless all connections are inactive.

The WRR algorithm has several notable features that make it ideal for our purpose:

**Guaranteed allocated bandwidth:** Given the following parameters:

- $CS$: one cell slot, i.e., the time to serve one cell
- $M$: the total number of cell slot entries in a schedule
- $W$: number of schedule entries (allocated slots) for a particular queue

The allocated bandwidth $BW$ (in Cells/Sec) for the target queue can be obtained as:

$$BW = \frac{W}{M} \times \frac{1}{CS} \tag{3.1}$$

**Automatic sharing of unused bandwidth:** If a connection does not have enough cells in its queue to consume all its schedule entries during a serving cycle, the WRR server will use these excess cell slots to serve other active connections. Thus bandwidth sharing is achieved.

**Intrinsic fairness:** In the WRR algorithm, the excess cell slots will be automatically given to the active connections in proportion to their allocated weight. In this sense, it provides a means to distribute the spared bandwidth fairly.

One version of WRR server is the *distributed WRR server.* Here, *distributed* means the schedule entries for a connection are evenly distributed within the schedule. In addition to the common WRR characteristics, the distributed WRR also helps to smooth the traffic in the multiplexing/demultiplexing procedure. Therefore, we will assume the distributed WRR in the remainder of this paper.

### 3.2.2 Other Cell-Level Schemes

Besides the cell scheduling schemes such as WRR, there are other type of schemes that can be used as either supplementary or alternative means in the bandwidth management framework, especially the following:

**Traffic Policing Schemes:** Policing, or Usage Parameter Control (UPC) has long been recognized as an effective way to enforce the user-network traffic contract. At present the Generic Cell Rate Algorithm (GCRA), which is based on a leaky-bucket algorithm, has been chosen by ATM Forum [4] as the definition of traffic conformance. The policing function at the UNI determines if the individual cells are conforming to the traffic contract and either drops violating cells, or marks them with Cell Loss priority (CLP) = 1 (Conforming cells carry CLP = 0). Since the GCRA includes bandwidth-related parameters such as Peak Cell Rate (PCR) and Sustainable Cell Rate (SCR), it can also be used in bandwidth management.

**Buffer Management Schemes:** In the case that multiple connections share the same physical buffer, buffer management schemes (also known as space priority schemes) [23] are necessary to ensure the proper buffer access priority of different services. The most commonly used space priority schemes are *partial buffer sharing* (also known as *nested threshold cell discarding*) [24] and *push-out queue* [25]. Although the implementation of the two schemes are different, both of them support selective discarding of individual cells. Therefore if the cells are marked as CLP = 0 or CLP = 1 by policing functions, these schemes can be used to protect CLP = 0 traffic from CLP = 1 traffic by giving higher buffer access priority to CLP = 0 cells.

## 3.3 Network Design Level Issues

Given appropriate cell-level schemes, the next question is how to structure the network based on them. Again, the CBR/VBR and ABR/UBR traffic need to be treated differently because of their different nature.

### 3.3.1 Bandwidth Management for CBR/VBR Traffic

As discussed in section 1.2, CBR/VBR services generally require a worst-case QoS guarantee, and the primary way to achieve this is to allocate enough bandwidth to each connection (sufficient buffer space should also be allocated). Thus the admissible traffic load on a VP is determined by the total amount of bandwidth allocated to that VP. From the traffic engineering point of view, this amount should be determined by relatively long-term considerations, such as physical link capacity, traffic forecast and estimation methods, and may be updated in a time scale such as hours or days, rather than on a call-by call basis.

The main advantage of this long-term allocation of VP bandwidth is that it simplifies the VC-level CAC and offers traffic isolation to provide performance guarantees for each VP. The CAC is simplified because the decision of whether to accept a CBR or a VBR call can be made at the corresponding source edge gateway by comparing the bandwidth requirement of the new call and the available amount of allocated bandwidth on the VP which is to carry the new call. For example, an incoming CBR call may be admitted if its PCR can be accommodated by the VP, and an incoming VBR call may be admitted if its SCR can be accommodated by the VP. The detailed CAC strategy will be addressed later in this report. Also notable is that under this strategy, the optimization of VP routes becomes possible using mathematical programming techniques.

Although ABR/UBR VPs should be able to use spare bandwidth from CBR/VBR VPs, bandwidth sharing among CBR/VBR VPs is undesirable. The traffic entering a CBR/VBR VP should be restricted to the allocated VP bandwidth to ensure that the VBR rate fluctuation does not degrade the performance of CBR VCs which are integrated on the same VP. To clearly understand this, note that the spare cell slots from other VPs at one node may not be available at the downstream nodes. Consequently, the extra VBR cells transmitted using spare cell slots from other VPs may be throttled at a downstream node, causing CBR connections sharing the same VP queue to incur more delay variation and even cell loss. Note that since the CAC decision for CBR and VBR should always be based on the allocated VP bandwidth even if there is spare bandwidth in the network, the above restriction will not impact the network capability to accept CBR/VBR calls.

However, we believe that in order to fully exploit the possibility of statistical multiplexing, it is still desirable to have VC-level bandwidth sharing inside each CBR/VBR VP.

### 3.3.2 Bandwidth Management for ABR/UBR Traffic

Through the allocation-based cell scheduling schemes, ABR/UBR VPs will be able to utilize the spare bandwidth from CBR/VBR VPs in the network. As a result, only the small amount of bandwidth corresponding to the MCR of ABR service is necessary for ABR/UBR VPs. The available bandwidth for ABR/UBR VPs beyond that allocated for the MCR is determined by the traffic load on the CBR/VBR VPs in the network, which is changing constantly. Consequently, in order for ABR sources to use this bandwidth and still achieve a low CLR, a mechanism is necessary to feed back the bandwidth information to the ABR traffic sources. The mechanism could be the Resource Management (RM) cell feedback procedure currently being developed by the ATM Forum [5]. Note that the RM cells are needed at both VP and VC level. Generally, the VP-level RM cells carry the available VP bandwidth information collected in the core network to the edge gateways, where the bandwidth is further allocated to individual VCs and sent to the ABR sources by VC-level RM cells. Recent research in this area indicates that efficient algorithms can be developed to control the ABR source rate in order to achieve both low cell loss and good bandwidth utilization [27] [26].

Since the allowable transmission rate of ABR sources is subject to flow control, the ABR UPC parameters need to be updated accordingly. That means *dynamic* UPC functions rather than the *static* UPC in the CBR/VBR case. As to the UBR cells, since the network will not provide any QoS guarantee to them, it is reasonable to mark all of them as $CLP = 1$.

As indicated in section 3.2.2, by utilizing space priority schemes, it is safe to let ABR and UBR connections share the same buffer, and hence the same portion of bandwidth.

### 3.3.3 Summary on Strategy

The conclusions from the above discussion can be summarized as follows:

1. VP bandwidth for CBR/VBR VPs should be determined semi-permanently (update intervals measured in hours or days).

2. Once the VP bandwidth is determined, the traffic entering CBR/VBR VPs should be throttled to the VP bandwidth at the ingress edge gateway.

3. VC-level bandwidth sharing should still be supported within each CBR/VBR VP

4. High network utilization can be achieved by letting ABR/UBR VPs "fill-in" the bandwidth gap left by CBR/VBR VPs on the link, through VP-level allocation-based cell scheduling schemes.

5. Dynamic UPC function is necessary for ABR traffic.

6. By introducing space priority schemes, ABR and UBR traffic may safely share the same buffer in the network. However, since many ABR control algorithms [26] rely on queue-fill information, some kind of mechanism, such as a separate ABR cell counter, might be necessary to keep track of the number of ABR cells in the shared buffer. Another alternative is to use separate VPs for ABR and UBR services.

## 3.4 An Implementation Sketch

To further illustrate how the ideas discussed above can be incorporated into a bandwidth management framework, we here present a sketch of a network implementation based on them. The implementation consists of three parts: *the ingress function of the edge gateway*, *the core switch*, and *the egress function of the edge gateway*.

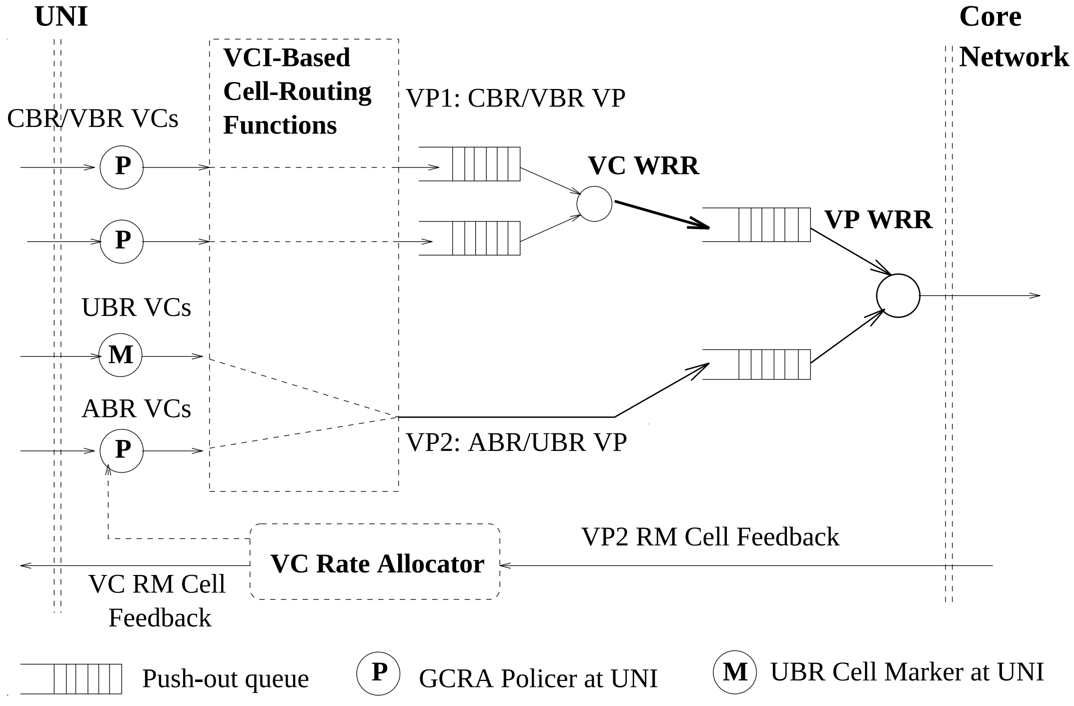


Figure 3.1: Ingress function of the edge gateway

### 3.4.1 The Ingress Function of Edge Gateway

As shown in Figure 3.1, each VC has a GCRA policer [4] at the UNI to ensure that the incoming traffic is conforming. The conforming cells are given high cell loss priority (CLP = 0) and the non-conforming cells are marked as CLP = 1. The GCRA parameters of ABR VCs should be dynamically adjustable to accommodate the fluctuation of available bandwidth in the network. In addition, all UBR cells are marked as CLP = 1 in order to provide necessary isolation for ABR cells.

At the ingress of the edge gateway, there are two stages of distributed WRR servers. At the first stage, CBR/VBR VCs are multiplexed into CBR/VBR VPs by using a VC-based distributed WRR server for each CBR/VBR VP. Each CBR (VBR) VC has a separate queue and is allocated a $W$ corresponding to its PCR (SCR) bandwidth. The output rate of the distributed WRR is fixed at the allocated VP bandwidth; thus the traffic entering a CBR/VBR VP is limited according to the policy specified previously. Since each CBR/VBR VP is throttled to the allocated VP bandwidth at ingress, and the core switches provide at least the allocated VP bandwidth, core switch buffers for VBR/CBR VPs can be very small (see numerical example in section 4). The ABR/UBR VCs do not have per-VC queueing and VC-based WRRs at the first stage. The ABR and UBR VCs on the same VP are simply mixed into a single ABR/UBR VP queue. At the second stage, the VPs are routed into the core network.

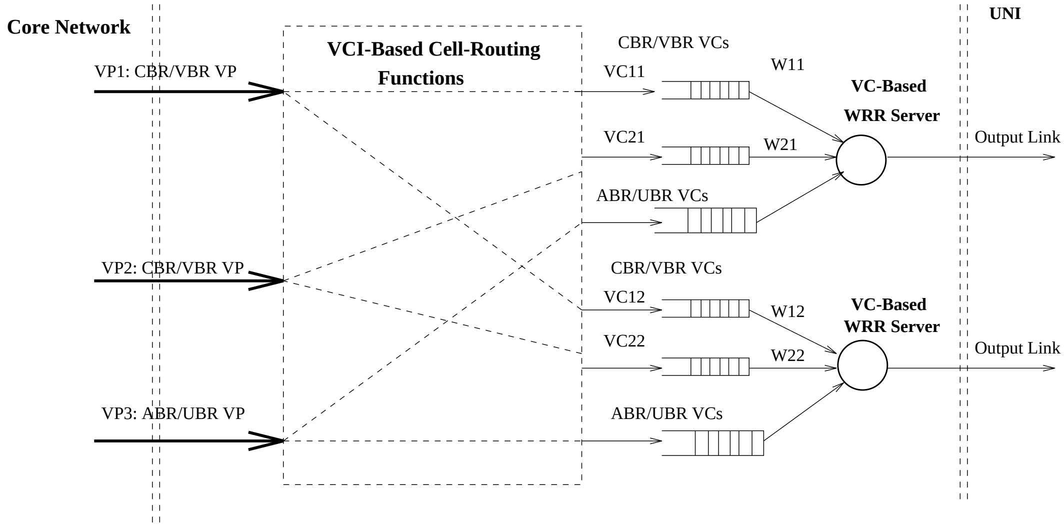


Figure 3.2: Egress function of the edge gateway

### 3.4.2 The Core Switch

The core switches are now simply ATM cross-connectors performing VP multiplexing/switching by using allocation-based cell scheduling schemes such as WRR.

### 3.4.3 The Egress Function of Edge Gateway

As shown in Figure 3.2, the VCs are demultiplexed from VPs at the egress edge switch, and routed to their destination UNI. There is a VC-based WRR server at each UNI. Again, each CBR (VBR) VC has a separate queue and is allocated a $W$ corresponding to its PCR (SCR). The ABR/UBR VCs on the same UNI share the same queue.

All queues are implemented as push-out queues ([25]), i.e, the arriving CLP=0 cells can "push out" those CLP = 1 cells in the queue if the queue is full. Therefore, the throughput of CLP=0 cells will be essentially unaffected by CLP=1 cells. Note, the CLP=1 cells may still enter the network given any available bandwidth, but without any performance guarantee.

# Section 4

# Queueing Delay and Queue Length Analysis for CBR/VBR Services

Real-time applications carried by CBR/VBR VCs will have stringent delay and cell loss requirements. We evaluate maximum queueing delay and maximum queue length for CBR/VBR services under the proposed framework and implementation. These worst case performance evaluations will have important impacts in buffer size design and VP routing decisions.

On the other hand, ABR services only have a CLR objective which will be determined by the specific rate-based flow control method deployed. The UBR service does not have any QoS requirement. For these reasons, a detailed performance evaluation for ABR/UBR services will not be included here.

## 4.1 Maximum Queueing Delay

According to the proposed network model, the cell transfer delay consists of three elements: the propagation delay on transmission links, cell routing delay in switches, and queueing delay at WRR servers and VP multiplexers/switches.

However, the first two will remain relatively constant after the connection is established. Therefore we only focus on *maximum queueing delay* performance.

As shown in Figure 4.1, the end-to-end connection (A-E) consists of a VC-based WRR between (A-B) and a VP Multiplexer (VP MUX) between (B-C) at the ingress edge, $N$ VP MUXes between (C-D) in the core , and a VC-based WRR between (D-E) at the egress edge. The end-to-end queueing delay $Delay_{A-E}$ is the sum of queueing delays incurred at ingress edge VC-based WRR, egress edge VC-based WRR, and all VP MUXes:

$$Delay_{A-E} = Delay_{A-B} + Delay_{B-D} + Delay_{D-E} \tag{4.1}$$

For a particular CBR or VBR VC between {A,E}, the worst case end-to-end delay performance occurs under the following conditions:

1. At both ingress edge and egress edge VC-based WRRs, the target VC can only be served according to its allocated bandwidth, PCR for CBR VC and SCR for VBR VC.

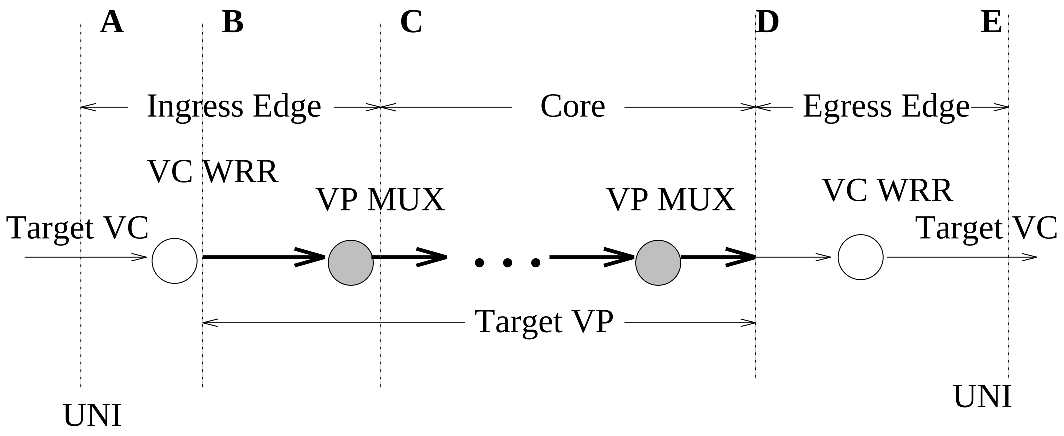


Figure 4.1: End-to-end Connection

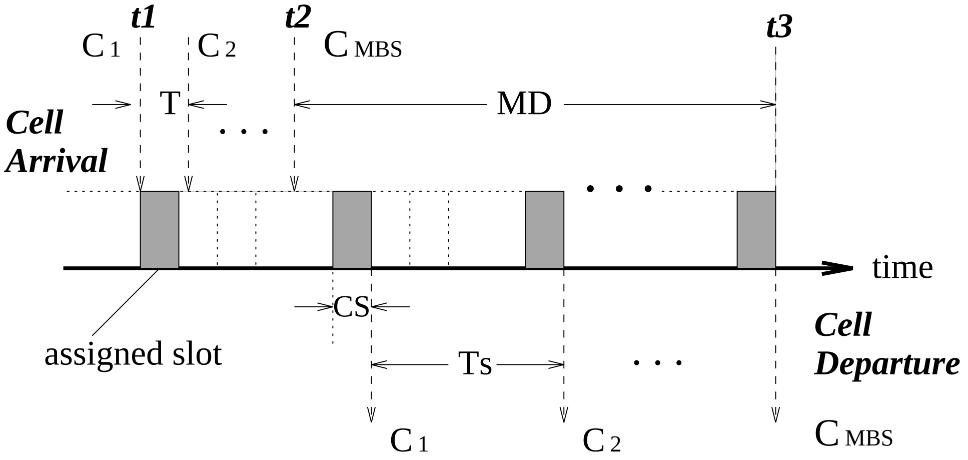


Figure 4.2: Maximum queueing delay under distributed WRR

2. All cell slots of the ingress edge VC-based WRR are used, i.e., the rate of traffic entering a target VP reaches the upper limit.

3. The target VP can obtain only its allocated bandwidth at each VP MUX it passes.

With these conditions in mind, the maximum queueing delay for a VBR connection is evaluated as follows.

Suppose a target VBR VC conforms to $GCRA(T, 0)$ (T is $\frac{1}{PCR}$, CDVT is zero) and $GCRA(T_s, \tau_s)$ ($T_s$ is $\frac{1}{SCR}$, $\tau_s$ is the burst tolerance) [4]. Accordingly, the size of a maximum conforming burst [4] is: $MBS = 1 + \frac{\tau_s}{T_s - T}$.

At the ingress edge VC-based WRR server, the $W$ for the target VBR VC is, according to its SCR, $W = M \times CS \times \frac{1}{T_s}$. As shown in Figure 4.2, when the last cell ($C_{MBS}$) of a maximum burst arrives, the queue will reach its maximum length; consequently, this cell will incur the longest queueing delay. The maximum queueing delay at the ingress edge is therefore:

$$\begin{aligned} MD_{VBR}^{A-B} &= t_3 - t_2 \\ &= (MBS \times T_s + CS_{in}) - [(MBS-1) \times T] \\ &= T_s + \tau_s + CS_{in} \end{aligned} \tag{4.2}$$

where $CS_{in}$ is the cell slot at the ingress edge WRR. Given condition 3, the traffic on the corresponding VP conforms to $GCRA(\frac{1}{BW_{VP}}, 0)$. Accordingly, the maximum queueing delay at the $i_{th}$ VP MUX is: $MD_i = \frac{1}{BW_{VP}} + CS_i, \;\; where \;\; i = 1, \ldots, N+1$, and $CS_i$ is the cell slot at the $i_{th}$ VP MUX. Thus the maximum queueing delay between {B,D} is:

$$\begin{aligned} MD_{VBR}^{B-D} &= \sum_{i=1}^{N+1} MD_i \\ &= \frac{N+1}{BW_{VP}} + \sum_{i=1}^{N+1} CS_i \end{aligned} \tag{4.3}$$

Viewed by the egress edge VC-based WRR given condition 1, 2, and 3, the target VBR VC is $GCRA(T_s, 0)$ conforming, i.e., the "bursty" VBR VC becomes constant-bit-rate after it passes the ingress edge VC-based WRR. Therefore, the maximum queueing delay at the egress edge is:

$$MD_{VBR}^{D-E} = T_s + CS_{out}, \tag{4.4}$$

where $CS_{out}$ is the cell slot at the egress edge WRR. Therefore, the maximum end-to-end queueing delay for the target VBR VC is:

$$\begin{aligned} MD_{VBR}^{A-E} &= MD_{A-B} + MD_{B-D} + MD_{D-E} \\ &= 2T_s + \tau_s + CS_{in} + CS_{out} + \frac{N+1}{BW_{VP}} + \sum_{i=1}^{N+1} CS_i \end{aligned} \tag{4.5}$$

A CBR VC can be thought as a VBR VC with $T_s = T$ ,$\tau_s = 0$. So for CBR VCs, the end-to-end maximum queueing delay can be obtained by simplifying equation (4.5):

$$MD_{CBR}^{A-E} = 2T + CS_{in} + CS_{out} + \frac{N+1}{BW_{VP}} + \sum_{i=1}^{N+1} CS_i \tag{4.6}$$

## 4.2 Maximum Queue Length

The following evaluation concentrates on the maximum queue length for CBR and VBR connections at each WRR server. If each queue size is designed to be at least the maximum queue length, there will be no buffer overflow for conforming CBR/VBR traffic.

For a $GCRA(T, \tau)$ conforming cell stream served by a distributed WRR with allocated $W = \frac{1}{T} \times M \times CS$, the relationship between the maximum queueing delay $D_{max}$ and the maximum queue length $MQL$ is: $D_{max} = MQL \times T + CS$. Combined with the result of equation 4.2, the maximum queueing length can be obtained as:

$$MQL = 1 + \frac{\tau}{T} \tag{4.7}$$

Using Equation 4.7, the maximum queue length at each stage of a CBR or VBR connection can be obtained as follows:

**At ingress VC-based WRR:** $MQL$ for CBR VCs is 1, while $MQL$ for VBR VCs is $1 + \frac{\tau_s}{T_s}$.

**At *i*th VP MUX:** Because the output rate of the ingress edge VC-based WRR is limited at the corresponding allocated VP bandwidth, the traffic entering the VP at reference point B must be $GCRA(\frac{1}{BW_{VP}}, 0)$ conforming. Note between B and the $i$th VP MUX, the maximum queueing delay is $\sum_{j=1}^{i-1} CS_j + \frac{i-1}{BW_{VP}}$, and the minimum queueing delay is $\sum_{j=1}^{i-1} CS_j$. Therefore, the cell delay variation (CDV) [1] introduced between B and the $i$th VP MUX is:

$$CDV_{B-i} = \frac{i-1}{BW_{VP}} \qquad i = 1, \ldots, N+1 \tag{4.8}$$

Therefore the total CDV at the $i$th VP MUX is:

$$\begin{aligned} CDV_i &= CDVT_B + CDV_{B-i} \\ &= 0 + CDV_{B-i} \\ &= \frac{i-1}{BW_{VP}} \qquad i = 1, \ldots, N+1 \end{aligned} \tag{4.9}$$

Consequently, the maximum queue length at the $i$th VP MUX is obtained as:

$$\begin{aligned} MQL_i &= 1 + CDV_i \times BW_{VP} \\ &= i \qquad i = 1, \ldots, N+1 \end{aligned} \tag{4.10}$$

Note that equation 4.11 is true for both CBR and VBR VCs.

**At egress edge:** For a particular VBR VC, the maximum queueing delay at the egress edge denoted by $D_{max}^{egress}$ is the end-to-end maximum queueing delay (equation 4.5) subtracting the minimum ingress edge delay and minimum core delay. Therefore:

$$D_{max}^{egress} = 2T_s + \tau_s + CS_{out} \tag{4.11}$$

Noting that $D_{max}^{egress} = MQL_{VBR}^{egress} \times T_s + CS_{out}$, the maximum queue length for VBR VCs at egress edge should be:

$$MQL_{VBR}^{egress} = 2 + \frac{\tau_s}{T_s} \tag{4.12}$$

Consequently, for CBR VCs, $MQL_{CBR}^{egress} = 2$.

[1] This notion of CDV is defined by the ATM Forum [5] as peak-to-peak CDV

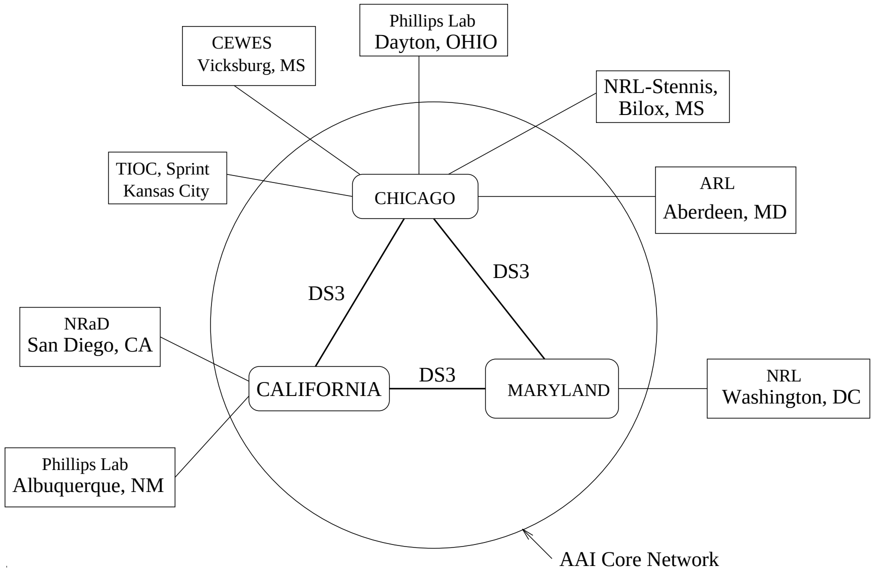


Figure 4.3: AAI Network Topology

## 4.3 Numerical Example

The ACTS (Advanced Communications Technology Satellite) ATM Internetwork (AAI) is an ARPA research network providing wide area ATM connectivity . Initially, AAI consists of three core switches and seven edge networks (see Figure 4.3). All seven edge networks use a FORE System's local area ATM switch as edge gateways. Initially, all links (including access links) have DS3 (nominally 45Mb/s) capacity. The AAI is supporting research in the areas of network signaling, congestion management, multicast, gateways to non-ATM LANs, etc.

Taking the AAI network configuration of figure 4.3 as an example, consider a hypothetical VP traversing two core switches from the Naval Research Lab (NRL) at Washington, D.C. to Technology Integration and Operation Center (TIOC) at Sprint Corporation, Overland Park, Kansas. Suppose at a certain moment, the VP is carrying ten 64kb/s CBR voice channels, one VBR MPEG video channel, and one VBR non-MPEG video channel. Table 4.1 shows the PCR, SCR and $\tau_s$ parameters for each type of call, where all parameters were selected based on by measurements from real traffic trace data. It is assumed that no traffic shaping function is used by the video sources, so the PCR of video sources is the access link rate. The SCR and $\tau_s$ for video calls are obtained from real trace data by using the method that will be discussed later in section 5. Also, assume the VP is allocated a bandwidth of 19.31Mb/s (45544 cells/sec) which is the sum of ten voice PCRs and two video SCRs (see Table 4.1).

Based on the analysis conducted in the previous section, the maximum queueing delay and queue length performance is presented in Table 4.2. The results show that under the proposed bandwidth management framework, these services require reasonably small buffer

| VC type | PCR<br>(cells/sec) | SCR<br>(cells/sec) | $\tau_s$<br>(ms) | Mean Rate<br>(cells/sec) |
|---|---|---|---|---|
| CBR (64kbps voice) | 167 | N/A | 0 | 167 |
| VBR (MPEG) | $1.06 \times 10^5$ | 9434 | 49.8 | 974 |
| VBR (Non-MPEG) | $1.06 \times 10^5$ | 34433 | 49.8 | 13907 |

Table 4.1: Traffic parameters

|  | CBR | MPEG<br>VBR | Non-MPEG<br>VBR |
|---|---|---|---|
| Max queueing delay (ms) | 12.14 | 50.0 | 50.0 |
| MQL at ingress edge (cells) | 1 | 471 | 1716 |
| MQL at ingress VP MUX (cells) | 1 | 1 | 1 |
| MQL at 1st core switch (cells) | 2 | 2 | 2 |
| MQL at 2nd core switch (cells) | 3 | 3 | 3 |
| MQL at egress edge (cells) | 2 | 472 | 1717 |

Table 4.2: Maximum delay and queue length

sizes and can obtain satisfactory queueing delay performance on the current AAI network topology. It should be noted that all these figures are derived from worst-case analysis; in reality the performance could be better due to VC-level bandwidth sharing inside the target VP.

# Section 5

# Traffic Description and CAC

As defined by ATM Forum [5], Connection Admission Control (CAC) is the set of actions taken by the network at virtual connection establishment in order to determine whether a connection can be accepted or should be rejected. However, any CAC strategy must be supported and limited by the bandwidth management architecture on which it resides. Beginning with this section, we will investigate the CAC strategy suitable for the bandwidth management solution discussed in the previous sections.

## 5.1 CAC Strategy Overview

In general, CAC has to make the decision based on whether or not all connections (including both the existing ones and the new connection) will be able to achieve their QoS, given limited network resources. A successful CAC strategy should achieve a good balance between the users' desire for QoS guarantees (conservative resource allocation) and the network provider's desire for maximum revenue (aggressive resource allocation). Furthermore, it should be relatively simple to implement, suitable to a wide range of traffic types, and able to deal with time-varying traffic.

Another important issue in any CAC strategy is the pricing policy. Usually, price can be based on either or both of the following:

- Resource allocation, which may be measured in terms of declared traffic parameters.

- Actual usage (cell counts).

We take the position that pricing based on both factors is necessary to satisfy both users and network providers. Such policies provide incentives for both users and network providers to maintain consistency between resource allocation and actual usage.

As part of a comprehensive traffic management solution, CAC needs support from the following two aspects:

- A traffic description method accepted by both user and network. Currently ATM forum has chosen the GCRA for this purpose. The basic traffic parameters for VBR services are Sustainable Cell Rate (SCR) and Burst Tolerance (BT). [1]
- An efficient underlying resource management scheme, which we have already discussed in the previous sections.

It should be noted that the $(SCR, BT)$ traffic description that will yield a certain violation ratio for a given type of traffic is not unique. For example, given an SCR value, there exists a $BT_{min}$ such that for any $BT \geq BT_{min}$, the policer based on $(SCR, BT)$ will give zero cell-tagging. More important, $BT_{min}$ itself will vary with SCR. Clearly, the total number of admissible $(SCR, BT)$ pairs is infinite.

The choice of $(SCR, BT)$ is important since it is directly related to resource allocation (bandwidth and buffer) and QoS (delay and loss ratio). Throughout this report, we have assumed the allocated bandwidth of WRR equals to the SCR value. As shown in section 4, this allocation will result in zero cell loss and a delay bound close to $BT$. The following parts of this report will discuss how to choose proper $SCR$ and $BT$ so that $BT$ matches the user's delay requirement. Note that this method will always be more efficient in bandwidth allocation compared to specifying a $BT$ that does not match the delay bound and trying to meet the delay requirement by allocating WRR bandwidth larger than SCR. The reason is that the latter method essentially has to assume the worst-case traffic pattern that fits the specified $SCR$ and $BT$ parameters, and thus results in conservative bandwidth allocation.

The problem then becomes: Given a certain delay bound and CLR requirement, determine the corresponding $(SCR, BT)$ value that will satisfy the requirements and yet minimize the amount of resource allocation.

In this context, we propose a two-part CAC strategy:

1. Choose proper traffic descriptors (SCR and BT values) for each incoming connection and maintain accurate values using a combination of off-line measurement and on-line measurement-based dynamic renegotiation.

   The baseline VBR CAC strategy can then be expressed as: *allocate network bandwidth and buffer resources according to SCR and BT for each VC connection. If sufficient spare resources are available, the call can be accepted. Otherwise it should be rejected.*

2. Since the baseline strategy is likely to be quite conservative, the second part of the strategy is to enhance the CAC performance (number of admissible connections) by taking the statistical bandwidth multiplexing (enabled by WRR) into consideration. Note that proper traffic descriptors will still be essential for the success of this enhanced strategy.

[1] Peak Cell Rate (PCR) and Cell Delay Variation Tolerance (CDVT) are also defined but the values for these are usually determined by equipment configurations.

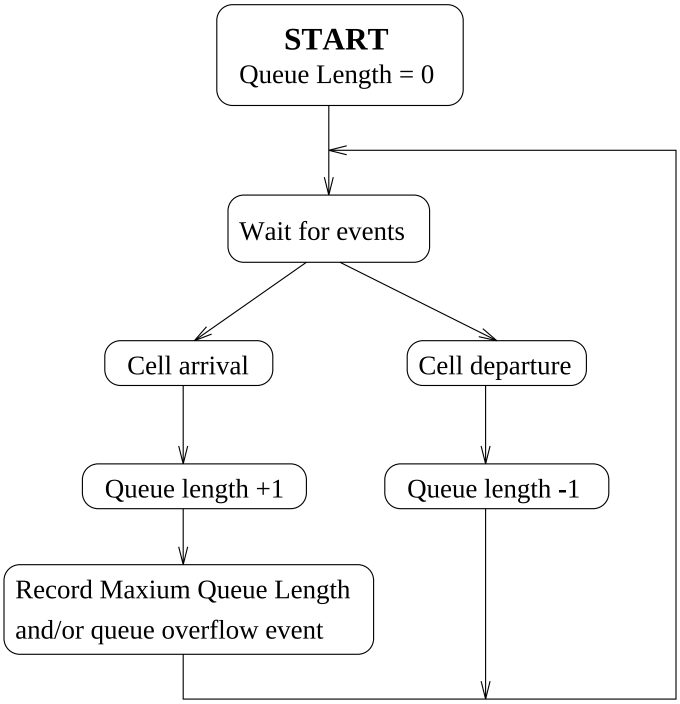


Figure 5.1: Virtual Buffer Measurement Mechanism

## 5.2 Virtual Buffer Measurement

Since at present there is no generic analytic model available to obtain the traffic description for VBR traffic, the best we can do is to resort to some operational (measurement-based method). In this report, we adopt a measurement scheme called *virtual buffer (VB) measurement*.

In virtual buffer measurement [3], the virtual buffer is actually a cell counter, which increases by one on cell arrival and decreases at a preset drain rate, as shown in figure 5.1. The volume of VB is defined as the upper bound of the counter value. If on a cell arrival the current counter value has already reached the VB volume, a VB overflow event is then recorded.

The counter value is sampled (perhaps on every cell arrival) for measurement processing. Depending on the processing techniques, various kind of results can be obtained, such as the maximum VB value and probability of VB overflow.

The importance of the above measurement is two-fold:

1. Simulates a FIFO with fixed serving rate equal to WRR allocated bandwidth, providing worst-case delay and loss estimates. There estimates are worst-case since the actual bandwidth available to a connection by the WRR server is always greater than or equal to the allocated bandwidth.

    - The VB counter value is equal to the worst-case FIFO queue length. The VB overflow events are equivalent to worst-case cell loss events. If no VB overflow event happens, the maximum observed VB counter value corresponds to the WRR buffer size that will yield zero cell loss.

- The queue length in the VB observed at cell arrival is directly proportional to the worst-case delay experienced by the arrived cell.

2. Simulates a GCRA (leaky bucket) policer, allowing UPC adjustments.
   - SCR corresponds to the VB drain rate, and BT to the volume of VB.
   - The VB overflow event is equivalent to cell-tagging in the corresponding UPC function.
   - The desired BT value for different CLR/tagging ratio requirements for a given SCR can be acquired through some simple manipulation of measurement results.

VB-based measurement has a great advantage that it is very simple and low-cost. The network provider can easily put it at the UNI or any other place to monitor traffic. Also it is very flexible and can be applied for many different purposes, which we will discuss in the rest of this report.

One interesting observation is that the QoS guarantee obtained as above is not explicitly related with the $PCR$ and $CDVT$, i.e., all sources with the same measured $SCR$ and $BT$ can achieve the same QoS regardless of their $PCR$ and $CDVT$ value. Actually, this has already been shown from the results we obtained in the maximum delay and queue length analysis in the previous section.

## 5.3 Off-Line Traffic Characterization

Given a certain kind of incoming traffic, if we use a number of VBs with different drain rates (SCR) in parallel and record the maximum delay for each SCR, the result will be a delay-bound vs. SCR curve, which is an important characteristic of this traffic. If the curve is already known, the network provider can then easily determine GCRA parameters and resource allocation by picking an SCR/BT pair that satisfies the user's delay requirement.

Unfortunately, the exact curve generally can be not obtained until a connection is admitted to the network. However, since the curve itself is an important characteristic of the particular traffic type, the result measured from pre-sampled trace files can serve as a guideline for the user-network traffic contract during the initial CAC.

Since it is commonly believed that digital video traffic will be a significant portion of VBR traffic in B-ISDN, we have examined a number of video samples (each on the order of 100 minutes long), including both MPEG-I encoded and JPEG encoded traces, for the above purpose. The results are shown in figures 5.2 through 5.4. In all these figures, we have assumed a frame-level "bursty" source, i.e., the source segments a whole video frame and transmits the resulting cells at a PCR corresponding to a very high link rate (OC-3, or 155 Mb/s).

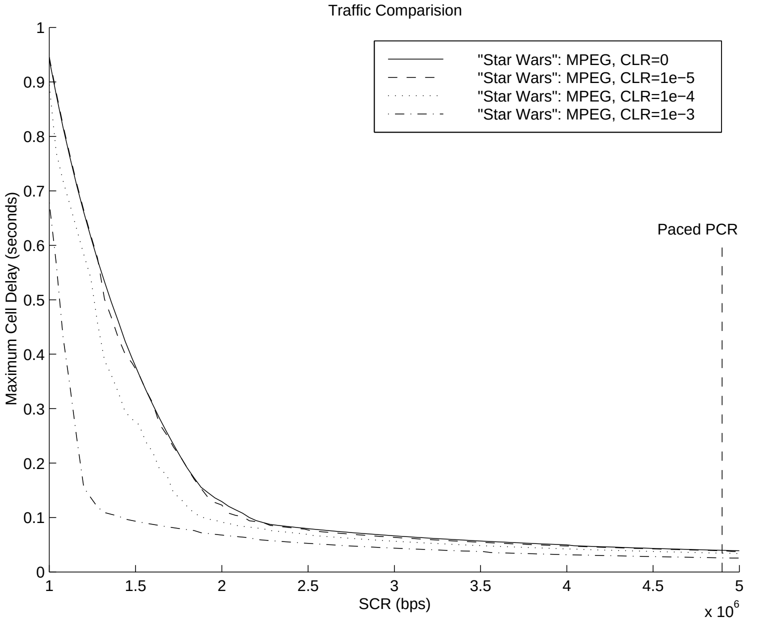


Figure 5.2: SCR-Delay relationship: different CLR

The result as shown in figure 5.2 is acquired by measuring a 2-hour MPEG-I video trace (the movie "Star Wars", from BELLCORE). There are four curves, representing CLR requirement of 0 (no loss/tagging happens), $10^{-5}$, $10^{-4}$ and $10^{-3}$ respectively. The most striking feature is that a sharp knee is present in all four curves. This seems to be a universal characteristic, since we observed the same feature in all our experiments. The implication is that there exists a bandwidth threshold operation point below which the delay and buffer requirement are very sensitive to bandwidth, but above which the delay and buffer requirement changes little. The natural choice of operational bandwidth is then somewhere close to, yet "safely" above the threshold. Though the exact threshold value will depend on the type of traffic, note that the threshold bandwidth for this video trace is much lower than the often-suggested paced peak rate (maximum frame size divided by frame interval) of $4.9Mb/s$ despite the fact that we have used a very bursty source. However, the burstiness of the traffic source shows its impact in the fact that the operational bandwidth is in general still much higher than the mean rate ($413kbit/s$).

We can also see that the bandwidth requirement varies, sometimes considerably, with different CLR requirements. For example, the threshold bandwidth drops almost 30% when the CLR requirement changes from $10^{-4}$ to $10^{-3}$.

Figure 5.3 and 5.4 are based on sequential JPEG encoded video. All video samples are obtained by recording 100 minutes of broadcast TV programs. The sampling and encoding is done using the SunVideo video system on SUN SPARC workstations.

In figure 5.3, we examine the sensitivity to different encoding quality (Q40 is lower quality than Q50). The result is just as expected: the bandwidth requirement for the same QoS

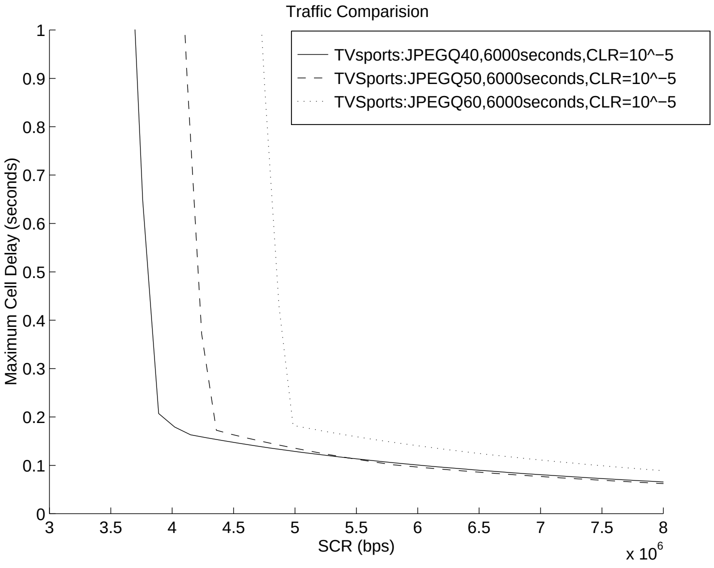


Figure 5.3: SCR-Delay relationship: different quality

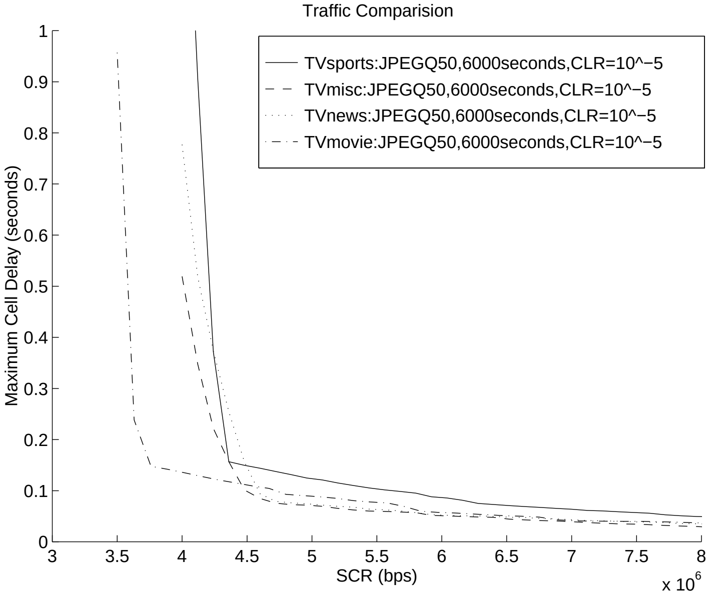


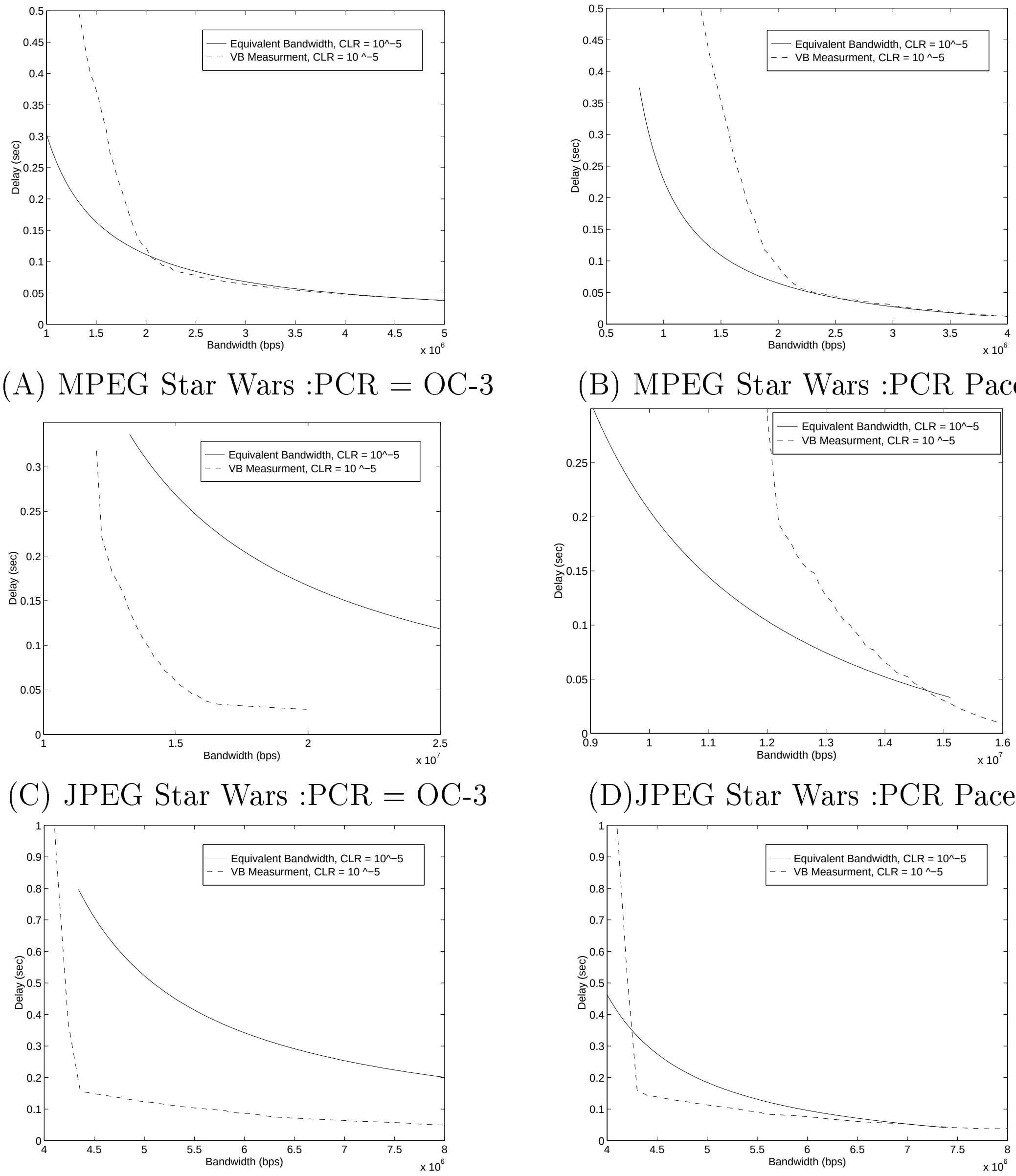
Figure 5.4: SCR-Delay relationship: different material

rises as the encoding quality gets better. However, the shape of curve remains the same, which means the effect of varying encoding quality is generally predictable.

The sensitivity to the program content is illustrated in figure 5.4 . Generally, there can be considerable variation caused by program content. As a result, the network operator and user may have to choose the worst-case curve (the rightmost one) for initial CAC decisions, since neither of them are likely to have an accurate estimation of the precise content of traffic. However, it is possible that more in-depth and systematic study will reveal some general principle regarding this case.

To further investigate the bandwidth requirement for the video sources, we compare the results obtained from VB measurement with those obtained by the well-known Equivalent Bandwidth (EBW) [33] [34] method. Based on an on-off fluid flow model, the EBW method estimates the bandwidth requirement from mean burst length, mean bit rate, peak rate, buffer size, and CLR requirement. In our experiments, burst lengths and rates are measured from the trace files, and the delay bound is considered as the buffer size divided by the resulting bandwidth. Both MPEG and JPEG encoded trace files are used in the experiment, the PCR of the source is set at either access link rate (OC-3) or paced PCR (maximum frame size/ frame interval). The comparison results are shown in figure 5.5

Generally, we observed that for different traffic characteristics, EBW can either considerably over-estimate the bandwidth requirement (as in figure 5.5 (C)(E)) or under-estimate it (as in figure 5.5 (A)(B)(D)). Even for the same traffic source, the EBW may also either over-estimate or under-estimate the bandwidth, depending on different delay requirement, as shown in figure 5.5 (F). These results are further evidence that the bandwidth requirement for VBR traffic generally can not be obtained from currently existing traffic models.



(A) MPEG Star Wars :PCR = OC-3

(B) MPEG Star Wars :PCR Paced

(C) JPEG Star Wars :PCR = OC-3

(D)JPEG Star Wars :PCR Paced

(E) JPEG TV program: PCR = OC-3

(F) JPEG TV program: PCR Paced

Figure 5.5: Comparison with Equivalent Bandwidth

# Section 6

# UPC Dynamic Renegotiation

Although results acquired from representative trace files can provide a starting point for the user-network traffic contract, more accurate knowledge can only be obtained *during* the connection by on-line measurement and estimation. Once new reference UPC parameters are acquired, a renegotiation procedure is then needed to adjust the traffic contract accordingly.

## 6.1 Basic Ideas and Schemes

The goal of on-line measurement and estimation is to find the appropriate UPC parameters (i.e., the corresponding bandwidth and buffer allocation) that can satisfy both the delay constraint and the CLR objective for a particular traffic source.

Within our network architecture, this two-dimensional problem can be transformed into one-dimension by setting buffer space to the product of bandwidth and delay constraints, and then finding the appropriate bandwidth value that will satisfy the loss constraints under this condition.

The most important information is the relationship between the CLR and SCR settings for a particular traffic source, which can be obtained by using VB measurement. However, in many cases the CLR objective can be very low (e.g., $10^{-9}$) and a direct measurement will take an unrealistic amount of time to perform. To address this problem, we propose the following scheme:

1. Set up a number of Virtual Buffers (VBs) with different drain rates (within the SCR range of interest), and set the volume of each VB (which is related to BT) to the product of drain rate and delay constraint.

2. For each VB, measure the CLR associated with it. The result is then a number of data samples indicating the CLR vs. SCR relationship.

3. Do curve-fitting on the data samples, and find the desired SCR value by extrapolation or interpolation.

## 6.2 Underlying Traffic Model: Stationary Vs. Non-Stationary

The estimation/renegotiation technique may vary with the choice of underlying traffic arrival process model (stationary or non-stationary), which we will discuss in this section.

### 6.2.1 Stationary model

In this case, the traffic arrival process is viewed as a stationary process (the statistical characteristics do not vary over time). Therefore, there exists a global optimal bandwidth value for the traffic source. The problem then becomes how to find this value asymptotically.

Since the arrival process is assumed stationary and ergodic, the characteristics of the process can be estimated through part of the process, i.e., it is possible to predict the optimal bandwidth value for the future using the information from its history. The accuracy of prediction depends on the amount of available data. Consequently, the *accumulative* measurement method should be used for this model, i.e., the CLR for all VBs should be measured cumulatively from the very start of the connection, including all loss/arrival events that are observed.

A successful bandwidth estimation method based on a stationary model should converge rapidly (perhaps in the scale of minutes) to the optimal value and should vary little over time.

One advantage of stationary model is that it can utilize all data collected in the past. Hence it is possible to get the data samples for lower CLRs if the time of observation is long enough.

### 6.2.2 Non-Stationary model

In some cases (like the LRD video model), it might be desirable to model the traffic arrival process as non-stationary, i.e., having different statistical characteristics over different periods of time. Accordingly, the bandwidth requirement also varies over time.

From this point of view, the network and user may choose to adjust UPC parameters and bandwidth allocation according to the change in the statistical characteristics of the traffic source. Note there is no longer a global optimal bandwidth estimation in this case, instead, we seek a series of estimations that is optimal locally (with respect to its corresponding time period). For this purpose, a *window-based* measurement method should be used, in which all CLR values are only measured within a certain *measurement window* time. Cell loss and arrival events outside this window are not counted.

The following are the possible implications introduced by this model:

- In the reservation-based schemes or even in the "baseline" CAC strategy we proposed in last section, this could help to accommodate more traffic (especially ABR/UBR) by indicating the current bandwidth usage explicitly to the CAC decision maker.

- If bandwidth estimation is applied to aggregated traffic such as an entire VP, the accumulative estimation may become meaningless because of its slow reaction to traffic change due to call arrival or departure. Here a non-stationary model becomes the only practical choice.
- Since this is basicly a reactive method, the reaction speed is crucial. For example, large blocks of cells may come in and get lost because of the failure to respond to this sudden change in bandwidth requirement. Therefore, a dilemma exists in choosing the measurement window. In order to improve the reaction speed, the window should be small (on the scale of seconds). On the other hand, since the size of data sample set is limited by the length of window, in order to improve the estimation inaccuracy and minimize estimation noise, a large window is desired (in the scale of $10^3 - 10^4$ seconds in most cases).

Note that it may be possible to combine these two models in practice. For example, it may be possible to use application level information to determine if there is a change in process characteristics. Between those changes, the process could be modeled as stationary and the accumulative measurement could be used to estimate bandwidth.

## 6.3 Renegotiation Procedure

After the initial UPC parameters are selected and the connection is established, the following procedure can be used to monitor the traffic and initiate renegotiation if necessary:

**Network initiated** renegotiation procedure:

1. Setup a number of VBs and start collecting data
2. Observe the CLR results of all VBs periodically, find the fitting function for the CLR-SCR curve, and then estimate the bandwidth requirement by extrapolation or interpolation.
3. Decide if a renegotiation is appropriate (e.g., whether the network has enough resource to guarantee an increase in SCR). If the answer is positive, send the corresponding UPC parameters back to the user in a **RENEG_REQ** message and wait for the user's response.
4. The user then evaluates the possible gain vs. possible risk brought by this new set of UPC parameters. Two cases may occur:

   **A.** Less network resource is associated with the new UPC parameters. In this case, the user may get lower cost by giving up part of the resources allocated, but there is a risk: since the network does not guarantee to give these resources back, the user may suffer a performance penalty in case he needs them in the future.

**B.** More network resource is associated with the new UPC parameters. This implies that the currently allocated resources are not sufficient to meet the QoS objective, that is, the traffic sent by the user no longer conforms to the current UPC parameters. If the user does not accept the new UPC parameters and the associated higher cost, he is likely to suffer QoS degradation.

If the user approves the new UPC parameters, he then sends back a **RENEG_ACK** message to the network and commits to the new traffic contract. Otherwise he should send a **RENEG_FAIL_ACK** message to the network, and the previous traffic contract is retained.

**User initiated** renegotiation procedure: there are two possibilities as follows.

**Possibility 1** The user monitors the traffic and initiates renegotiation just as the network does in previous case, only now the VBs are put at the user side.

**Possibility 2** The user can also initiate a renegotiation based on high-level knowledge about the traffic source, for example, a switch from video clip transmission to voice transmission in a multimedia application environment.

Note that in both cases of user-initiated renegotiation, the network should always accept the request indicating less resource allocation. However, the user request indicating more resource allocation is subject to CAC-like approval.

## 6.4 Discussion on bandwidth estimation technique

Although the basic idea for on-line bandwidth estimation is quite simple and straightforward, the details involved in getting an accurate estimation can be rather complicated. This section is a summary of our recent efforts which, though somewhat rudimentary, should serve as a starting point for further research work.

The fundamental problem in our bandwidth estimation methodology is to determine an appropriate fitting function. Generally, since the fitting function is based on collected data samples, it should minimize the error between fitting function value and data samples according to some criteria. Meanwhile, it should be obvious that the fitting function must be non-increasing in the SCR range of interest. Most importantly, we desire the fitting function to be a good approximation of the CLR-SCR relationship over a wide range, even though it is based on a limited data sample set.

Usually, the fitting function can be found by selecting a generic function and finding the coefficients using some optimization criteria. It is possible to formulate this problem as a constrained optimization problem and then solve it analytically or numerically. However, for the sake of simplicity, here we use a simplified method to demonstrate the feasibility and effectiveness of this approach.

After observing the data samples, We have chosen the following form of the fitting function:

$$log(CLR) = c_1(SCR)^{p_1} + c_2(SCR)^{p_2}$$

where $p_1$, $p_2$ are heuristically chosen and $c_1$, $c_2$ are obtained through a least-square fitting procedure on the available data samples.

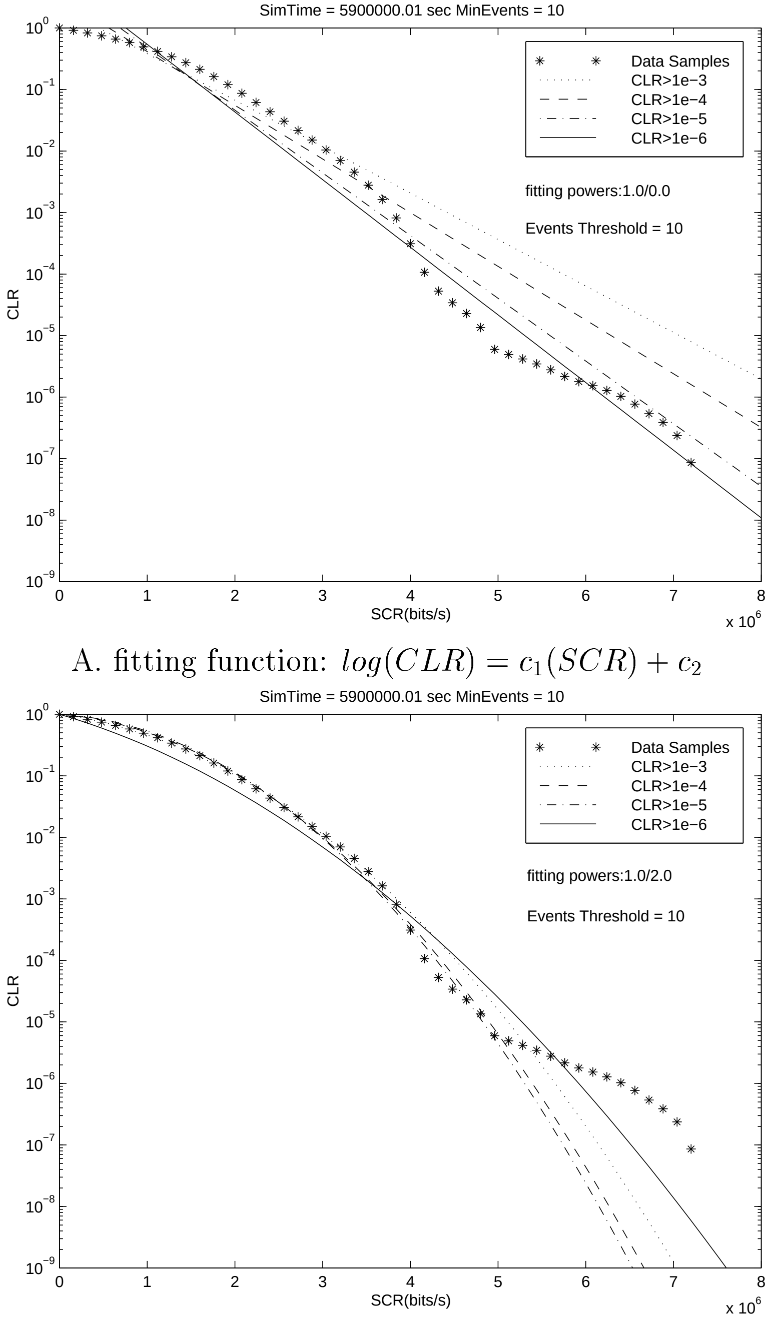
Having chosen the generic form for the fitting function, we now examine the outcome of this estimation method by running simulations on designated trace files. In all simulations in this report, we set the delay constraint at 100 milliseconds and set the VB volumes accordingly.

As we have stated, a good fitting function should be able to predict the CLR-SCR relationship based on limited data. To obtain a large data sample set, We have built a very long JPEG video trace (approximately $5.9 \times 10^6$ seconds) by concatenating shorter video traces (approximately 6000 seconds) randomly chosen from a small video trace library of 20 independent files sampled from broadcast TV programs. Using this long trace, the prediction ability of the fitting function can be examined by comparing the fitting results based on part of the data samples with the actual measured data samples.

As shown in figure 6.1 and 6.2, the choice of $p_1$ and $p_2$ has great significance. For example, by choosing $p_1 = 1.0$, $p_2 = 0.0$ (figure 6.1(A)), the estimation using only data samples with $CLR < 10^{-3}$ tends to greatly over-estimate the bandwidth requirement for low CLR objective, e.g., $10^{-9}$. On the other hand, if we use $p_1 = 1.0$, $p_2 = 2.0$ (figure 6.1(B)), the estimation based on the same data set then underestimates the bandwidth requirement for low CLR objective. After some experiments we have found that the setting of $p_1 = 0.5$, $p_2 = 1.5$ might be the best choice for the type of traffic that we use. To practically verify this hypothesis, we have built two more long traces using the same method but based on different sample trace sets (actually different parts of the same video trace library). The result is shown in figure 6.3. It can be seen that the fitting function using this setting works very well in both cases.

Figure 6.4 shows the bandwidth estimation on two different video traces using the accumulative measurement method. The first one is a 6000-second JPEG video sampled from a TV broadcast sports program. The other is a long trace concatenated by all 20 traces from the video library we have built. It can be seen that in general the bandwidth estimation converges after a reasonable amount of time, and the fluctuation in estimation is relatively small.

In figure 6.5, we compare the performance of the bandwidth estimation using the accumulative measurement and the window measurement with different window sizes. More results are shown in table 6.1 and 6.2. We note that though the smaller measurement window size generally introduces larger fluctuations in the estimation, the basic shape of the curve as well as the mean of the estimated bandwidth is very close in all three cases. This suggests that a usable bandwidth estimation might be acquired by "smoothing" the results from window-based measurements, even if the window width is relatively small. Consequently, a window-based estimation on VP level traffic might be feasible. It is also notable that, although it may be expected that window-based estimation can lead to a lower average



A. fitting function: $log(CLR) = c_1(SCR) + c_2$

B. fitting function: $log(CLR) = c_1(SCR) + c_2(SCR)^2$

Figure 6.1: Different fitting functions

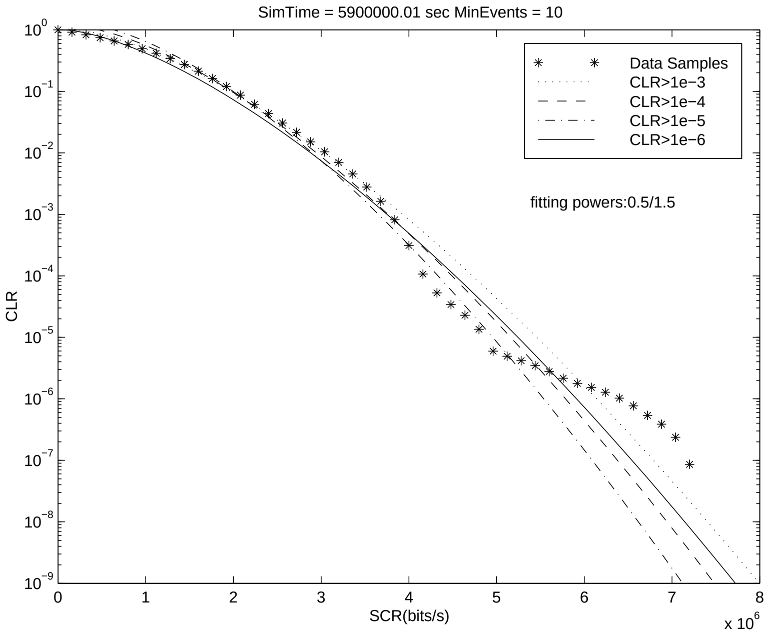
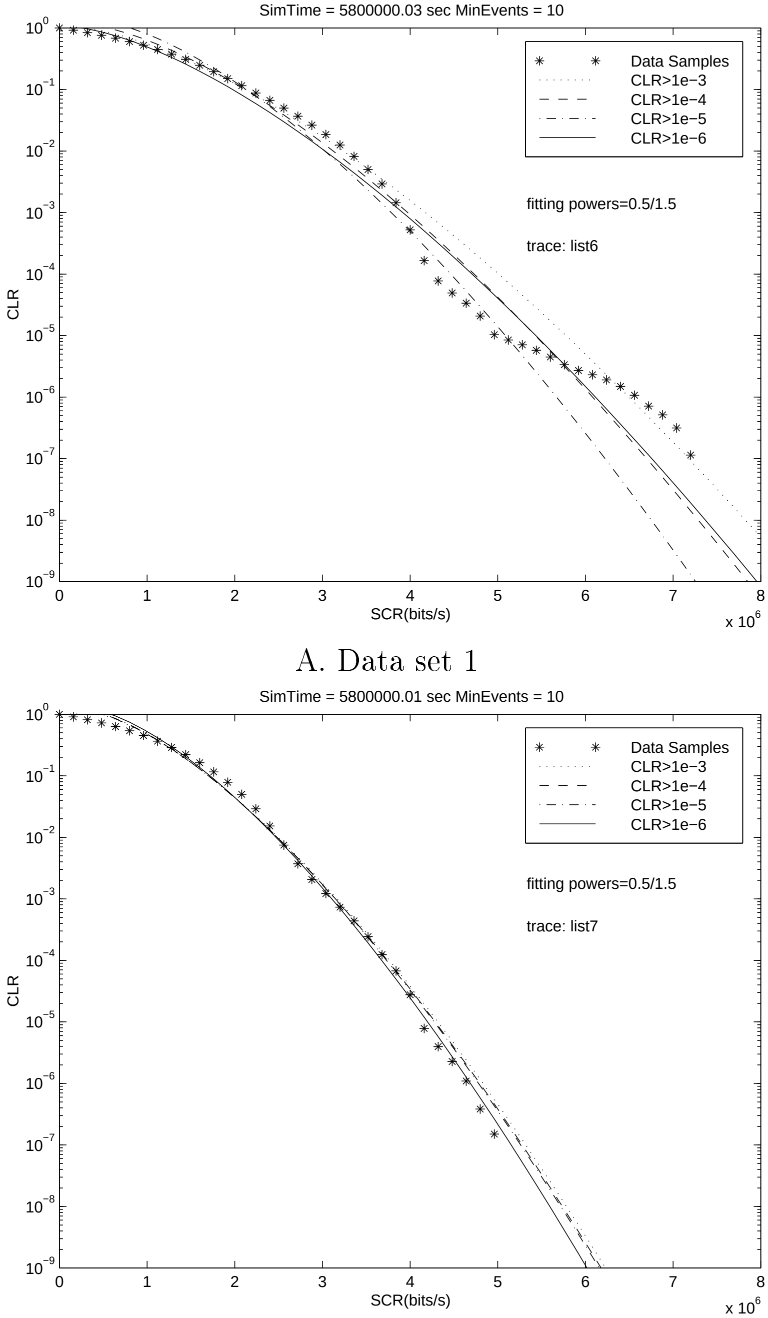


Figure 6.2: A good fitting function: $log(CLR) = c_1(SCR)^{0.5} + c_2(SCR)^{1.5}$

bandwidth requirement, it is not necessarily the case.

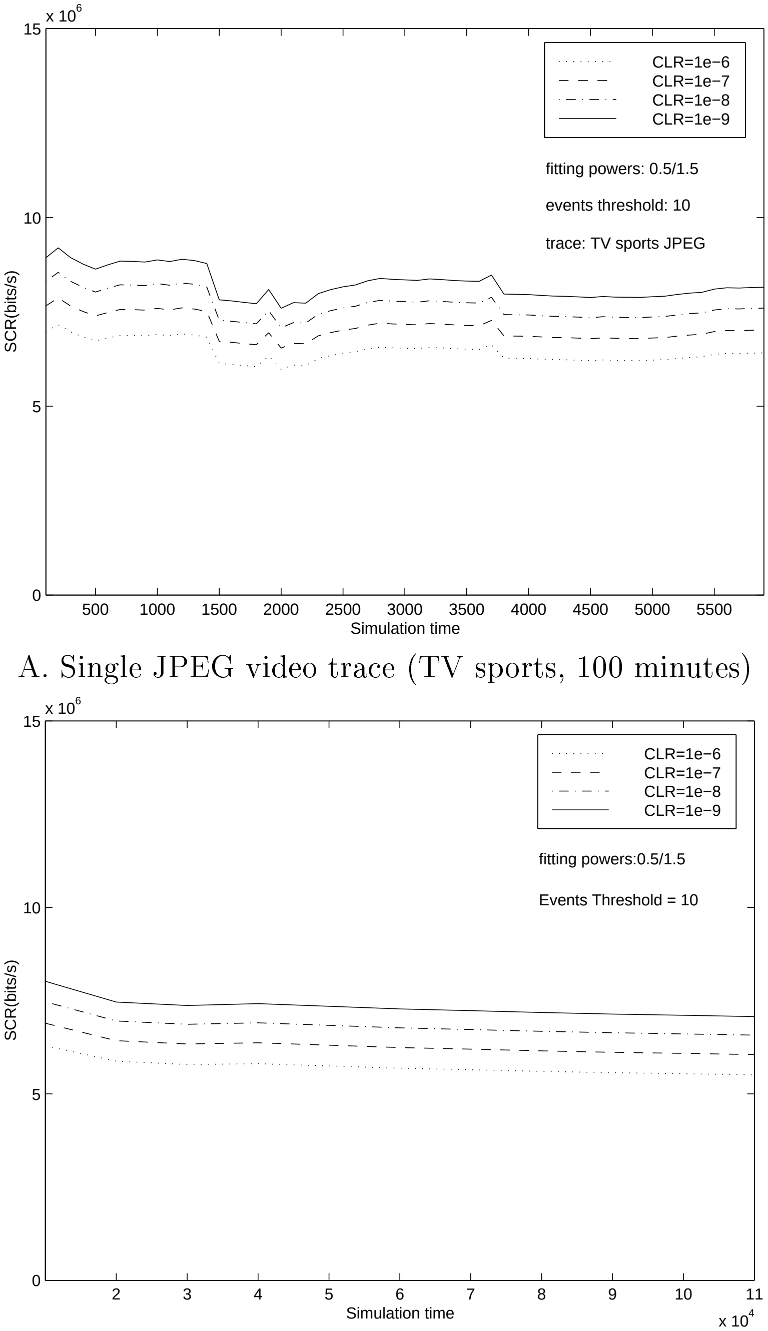
It should be noted that the approach presented in this section is still a very simple and immature one. There are many improvements that could be incorporated, such as adjusting the $p_1$,$p_2$ parameters adaptively, using more complicated fitting criteria than least-square fitting, and using an adaptive measurement window. These could all enhance the performance of bandwidth estimation.



A. Data set 1

B. Data set 2

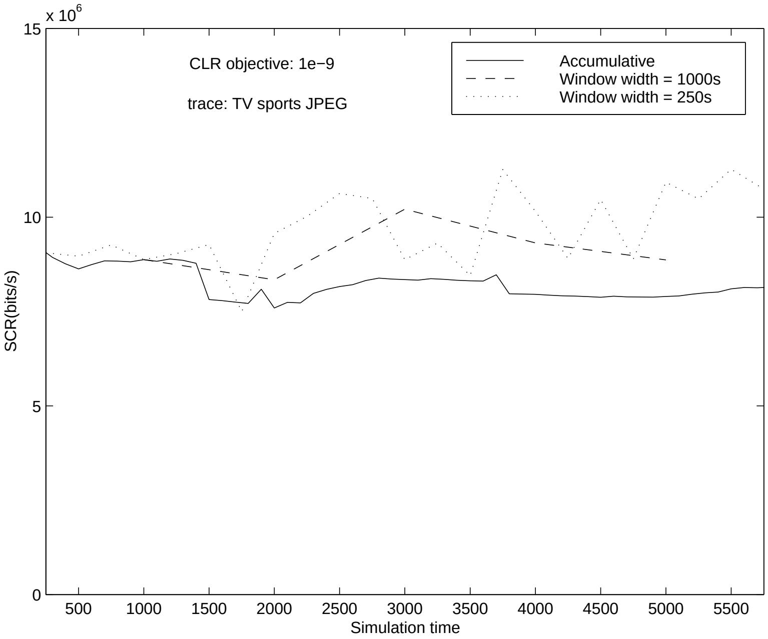
Figure 6.3: Fitting results for different trace data



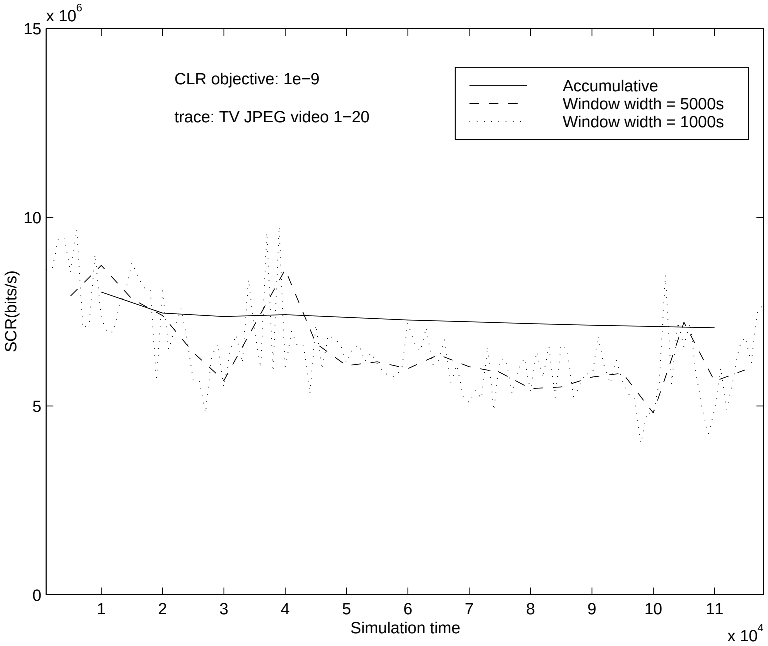
A. Single JPEG video trace (TV sports, 100 minutes)

B. Multiple JPEG video trace (20 TV traces, 100 min. each)

Figure 6.4: Bandwidth estimation using accumulative method



A. Single JPEG video trace (TV sports, 100 minutes)



B. Multiple JPEG video trace (20 TV traces, 100 min. each)

Figure 6.5: Comparison between estimation methods

|  | CLR=$10^{-6}$ | CLR=$10^{-7}$ | CLR=$10^{-8}$ | CLR=$10^{-9}$ |
|---|---|---|---|---|
| Mean Bandwidth (Mbits/s) (Accumulative) | 6.454845 | 7.075649 | 7.667772 | 8.235869 |
| Final Bandwidth (Mbits/s) (Accumulative) | 6.413153 | 7.019180 | 7.597287 | 8.151996 |
| Mean Bandwidth (Mbits/s) (Window width = 1000s) | 7.097494 | 7.803381 | 8.476494 | 9.122180 |
| Mean Bandwidth (Mbits/s) (Window width = 250s) | 7.494803 | 8.253554 | 8.977010 | 9.670943 |

Table 6.1: Bandwidth estimation: single JPEG video trace

|  | CLR=$10^{-6}$ | CLR=$10^{-7}$ | CLR=$10^{-8}$ | CLR=$10^{-9}$ |
|---|---|---|---|---|
| Mean Bandwidth (Mbits/s) (Accumulative) | 5.733688 | 6.290358 | 6.821268 | 7.330609 |
| Final Bandwidth (Mbits/s) (Accumulative) | 5.510176 | 6.055160 | 6.574854 | 7.073385 |
| Mean Bandwidth (Mbits/s) (Window width = 5000s) | 5.036974 | 5.542051 | 6.023681 | 6.485691 |
| Mean Bandwidth (Mbits/s) (Window width = 1000s) | 5.047047 | 5.558308 | 6.045802 | 6.513415 |

Table 6.2: Bandwidth estimation: 20 JPEG video traces

# Section 7

# CAC Based on Statistical Multiplexing Effect

While the CAC approach that relates the individual connection traffic descriptors directly to resource requirements does provide QoS guarantees for admitted connections, it ignores a great advantage of ATM networks, *statistical multiplexing.* Actually, supported by a bandwidth-sharing scheme such as WRR, it is possible to reduce the total amount of required bandwidth considerably.

Table 7.1 shows some results of video source multiplexing. The sources used here are six 20 minute segments taken from the movie "Star Wars", either MPEG-I or JPEG encoded. The fourth column is the sum of SCRs corresponding to the required delay bound (which corresponds to the convervative CAC), the fifth column is the total bandwidth required to achieve the same delay bound after the traffic from the sources is multiplexed using WRR. The multiplexing gain is defined as $(\sum SCR - MultiplexedBW)/\sum SCR$. Clearly, there are significant multiplexing gains regardless of delay requirements, even if the total number of sources is relatively small. Furthermore, many studies [29] [30] show that, for MPEG video sources, when the number of multiplexed sources increases, the aggregated bitrate distribution becomes more Gaussian and narrow. As a result, the aggregated peak rate tends to get closer to aggregated mean rate, and the effect of statistical multiplexing becomes even more significant.

However, in real life a VP will probably carry many kinds of traffic with greatly-varying

| Traffic Type | Number of Sources | Delay Bound (ms) | $\sum SCR$ (Mbps) | Multiplexed Bandwidth ($Mbps$) | Multiplexing Gain |
|---|---|---|---|---|---|
| MPEG-I video | 6 | 100 | 10.2 | 5.72 | 44% |
| MPEG-I video | 6 | 50 | 19.9 | 11.5 | 42% |
| JPEG video | 6 | 100 | 69.5 | 44.4 | 36% |
| JPEG video | 6 | 50 | 73 | 46.3 | 37% |

Table 7.1: Statistical Multiplexing Gain on Video Sources

characteristics, and the statistics such as mean rate, peak rate and burst size, which are required in many previous studies, can only be obtained *during* the lifetime of the connection. Meanwhile, when a network operator decides to take advantage of statistical multiplexing to increase total admission, he also takes the risk of possible over-admission and the resulting QoS degradation. For example, there is no guarantee that all users will not transmit at SCR simultaneously for a significant period of time. To avoid this situation as much as possible, it is necessary to constantly monitor current resource usage. Therefore, a practical CAC strategy considering the multiplexing effect should generally employ some kind of on-line measurement.

To deal with the above problem, we now propose a CAC strategy based on estimation of actual usage of bandwidth. The strategy can be expressed as follows:

Let $T_d$ be a pre-defined "qualification period", and for each VC, let $T_s$ be the time elapsed since admission. Define two sets $S_1 = \{VCs : T_s > T_d\}$, $S_2 = \{VCs : T_s < T_d\}$, so that $S_2$ contains VCs for which the available data is insufficient to make a valid estimation.

As shown in figure 7.1, The network estimates the bandwidth requirement for the entire VP as the sum of the following elements:

- The bandwidth value resulting from VB measurement/estimation on the aggregated traffic in the VP (including VCs in both $S_1$ and $S_2$). This value is chosen so as to satisfy the most stringent delay and CLR requirement of all the VCs in the VP.

- The sum of SCRs of all the VCs belonging to $S_2$. Here the network behaves conservatively by estimating their bandwidth requirement as the claimed SCR.

Note that the estimation is still somewhat conservative since the VC's in $S_2$ are actually evaluated twice in the estimation. By using more complicated measurements, it may be possible to eliminate this effect.

The admission criteria then becomes:

*If the sum of the estimated bandwidth for current traffic in the VP and the SCR for incoming VC is greater than the allocated VP bandwidth, reject the new call; otherwise the call can be accepted.*

As we stated before, this kind of CAC approach can be risky and should be applied with caution. For example, in practice it is probably desirable for the network operator to set a high water-mark of bandwidth usage (e.g., 90% of physical VP bandwidth), and use that as VP bandwidth in the above CAC procedure.

Another open issue is the choice of $T_d$. Generally, a larger $T_d$ means a more conservative strategy, and a smaller $T_d$ means more aggressive. Furthermore, it may be desirable to combine results from several measurement windows. We hope a good rule can be found through further experiments and analysis.

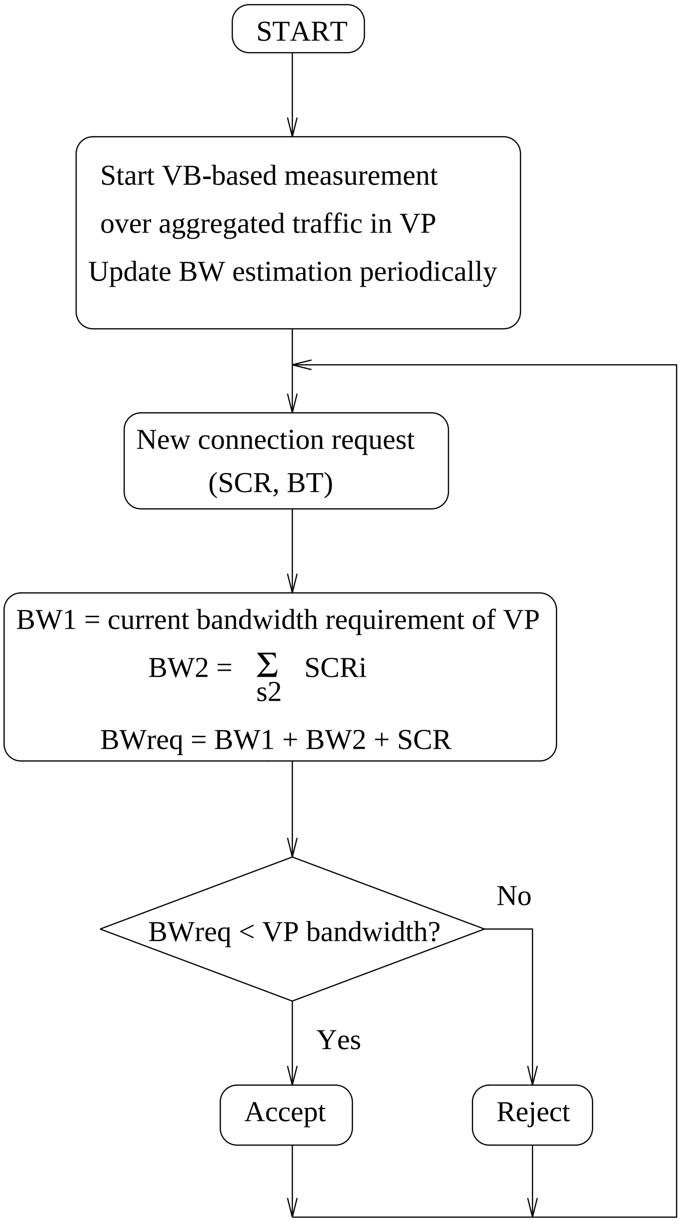

START
Start VB-based measurement
over aggregated traffic in VP
Update BW estimation periodically
New connection request
(SCR, BT)
BW1 = current bandwidth requirement of VP
BW2 = $\sum_{s2}$ SCRi
BWreq = BW1 + BW2 + SCR
BWreq < VP bandwidth?
No
Yes
Accept
Reject

Figure 7.1: CAC algorithm based on actual usage

# Section 8

# Conclusion

In this report, we have proposed a framework for bandwidth management and CAC in ATM networks, and examined some important related issues. First, we defined a bandwidth management architecture for ATM-based B-ISDN. The architecture consists of a network model and a bandwidth allocation strategy. Here the network is partitioned into core and edge networks. The advantage of this partitioning has been discussed. The network bandwidth is allocated in such a way that each VP is semi-permanently allocated a certain amount of bandwidth, while statistical bandwidth sharing may still be allowed among different VPs and VCs. The VP routes can be optimized using existing optimization techniques.

Cell scheduling and queueing implementations were discussed. The major elements of our framework related to implementation are the use of distributed WRR servers, push-out queues, and GCRA policers. Under the proposed implementation, maximum end-to-end queueing delay and cell loss performance have been evaluated for CBR and VBR connections.

Based on this architecture, we proposed a new CAC strategy for real-time VBR services in ATM networks. After introducing the idea of Virtual Buffer measurement for resource usage and UPC parameters, we discussed and illustrated how to obtain accurate UPC parameters for user traffic, by employing Virtual Buffer measurement and dynamic renegotiation. The baseline (conservative) CAC strategy tightly couples resource allocation and UPC parameters. From this basis, we move further to examine the possible resource gain from statistical multiplexing effects, and propose a more aggressive CAC strategy to exploit these effects. However, since the work here is more of a framework nature, further work is necessary to work out the details, such as the performance evaluation and implementation issues.

# Bibliography

[1] K. Liu, D. W. Petr, and C. Braun, "A Measurement-Based CAC Strategy for ATM Networks," *Proc. IEEE ICC'97*, 1997

[2] K. Liu, H. Zhu, D. W. Petr, V. S. Frost, C. Braun, and W. Edwards, "Design and Analysis of a Bandwidth Management Framework for ATM-Based Broadband ISDN," *Proc. IEEE ICC'96*, pp. 1712-1716, 1996.

[3] H. Zhu and V. S. Frost, "In-Service Monitoring and Estimation of Cell Loss Ratio QoS in ATM Networks," *IEEE/ACM Transactions on Networking*, Vol. 4, No. 2, pp. 240-248, April, 1996.

[4] The ATM Forum Technical Committee, *User-Network Interface (UNI) Specification Version 3.1*, 1994.

[5] The ATM Forum Technical Committee, *Traffic Management Specification Version 4.0*, AF-TM 0056.000, April, 1996.

[6] Martin de Prycker, *Asynchronous Transfer Mode - Solution for Broadband ISDN*, 2nd edition, Ellis Horwood, 1993.

[7] D. V. Batorsky, D. R. Spears and A. R. Tedesco, "The Evolution of Broadband Network Architectures," *Proc. IEEE Globecom'88* pp. 367-373, 1988.

[8] A. E. Eckberg, "B-ISDN/ATM Traffic and Congestion Control," *IEEE Network Magazine*, Vol. 6 No. 5, pp. 28, September 1992.

[9] Raj Jain, "Congestion Control and Traffic Management in ATM Networks: Recent Advances and A Survey," *Invited submission to Computer Networks and ISDN Systems*, Draft Version, January 26, 1995.

[10] ATM Forum/95-0221R2, *Draft PNNI Signaling*, 1995.

[11] J. Totzke and J. Welscher, "A prototyped Implementation of B-ISDN Signaling at the Network Node Interface," *Proc. IEEE Globecom'95*, Vol. 1, pp. 252-257, 1995.

[12] C. J. Chang, R. G. Cheng, "Traffic Control in an ATM Network Using Fuzzy Set Theory," *Proceedings of IEEE Infocom'94*, Vol. 3, pp. 1200, 1994.

[13] A. Charney, D.D. Clark, R. Jain, "Congestion Control with Explicit Rate Indication," *Proc. IEEE ICC'95*, June 1995.

[14] The ATM Forum Technical Committee, *Flow Controlled Virtual Connections - Proposal for ATM Traffic Management*, September 1994.

[15] S. OHTA, K. SATO, and I. TOKIZAWA, "A Dynamically Controllable ATM Transport Network Based on the Virtual Path Concept," *Proc. IEEE Globecom'88*, pp. 1272-1276, 1988.

[16] K. R. Krishnan and R. H. Cardwell, "Routing and Virtual-Path Design in ATM Networks," *Proc. IEEE Globecom'94*, Vol. 2, pp. 765-769, 1994.

[17] F. Y-S. Lin and K-T. Cheng, "Virtual Path Assignment and Virtual Circuit Routing in ATM Networks," *Proc. IEEE Globecom'93*, Vol. 1, pp 436-441, 1993

[18] K. Mezger, D. W. Petr and T. Kelley, "Weighted Fair Queueing vs. Weighted Round Robin: A comparative Analysis", *IEEE Wichita Conference on Communications, Networking and Signal Processing*, April 1994.

[19] Lixia Zhang, "Virtual Clock: A New Traffic Control Algorithm for Packet-Switched Networks," *ACM Transactions on Computer Systems*, Vol. 9, No. 2, pp. 101-124, May 1991.

[20] K. Sriram, "Dynamic Bandwidth Allocation and Congestion Control Schemes for Voice and Data Multiplexing in Wideband Packet Technology," *Proc. of IEEE ICC'90*, pp. 1003-1009, April 1990.

[21] M. Katevenis, S. Sidiropoulos and C. Courcoubetis, "Weighted Round-Robin Cell Multiplexing in a General-Purpose ATM Switch Chip," *IEEE JSAC*, Vol. 9, No. 8, pp. 1265-1279, October 1991.

[22] Y. Wang, T. Lin and K. Gan, "An Improved Scheduling Algorithm for Weighted Round-Robin Cell Multiplexing in an ATM Switch," *Proc. IEEE ICC'94*, pp. 1032-1037, 1994.

[23] H. Kroner, "Comparative Performance Study of Space Priority Mechanisms for ATM Networks," *Proc. IEEE Infocom'90*, pp. 1136-1143, 1990.

[24] D. W. Petr, V. S. Frost, "Nested Threshold Cell Discarding for ATM Overload Control Optimization Under Cell Loss Constraints," *Proc. IEEE Infocom'91*, pp. 1403-1412, 1991.

[25] G. Hebuterne and A. Gravey, "A Space Priority Queueing Mechanism for Multiplexing ATM Channels," *Computer Networks and ISDN Systems*, pp. 37-43, December 1990.

[26] Q. Hu, D. W. Petr, and C. Braun, "Self-tuning Fuzzy Traffic Rate Control for ATM Networks," *Proc. IEEE ICC'96*, pp.424-427, 1996.

[27] A. Kolarov, G. Ramamurthy, " A Control Theoretic Approach to the Design of Closed Loop Rate Based Flow Control for High Speed ATM Networks," *Proc. IEEE Infocom'97*, 1997.

[28] S. Jamin, P. B. Danzig, S. Shenker and L. Zhang, "A Measurement-based Admission Control Algorithm for Integrated Services Packet Networks," *Proc. ACM SIGCOMM'95*, pp. 2-13, 1995.

[29] J. Mata, G. Pagan and S. Sallent, "Multiplexing and Resource Allocation of VBR MPEG Video Traffic on ATM Networks," *Proc. IEEE ICC'96*, pp. 1401-1405, 1996.

[30] M. Krunz, R. Sass and H. Hughes, "Statistical Characteristics and Multiplexing of MPEG Streams," *Proc. IEEE INFOCOM'95*, pp. 455-462, 1995.

[31] F. Guillemin , C. Rosenberg and J. Mignault, "On Characterizing an ATM Source via the Sustainable Cell Rate Traffic Descriptor," *Proc. IEEE INFOCOM'95*, pp. 1129-1136, 1995.

[32] A. R. Reibman and A. W. Berger, "Traffic Descriptors for VBR Video Teleconferencing Over ATM networks," *IEEE/ACM Transactions on Networking*, Vol. 3, No. 3, pp.329-339, June 1995.

[33] R. Guérin, H. Ahmadi, and M. Naghshineh, "Equivalent Capacity and Its Application to Bandwidth Allocation in High-Speed Networks" *IEEE Journal on Selected Areas in Communications,* Vol. 9, No. 7, September 1991.

[34] C. Braun, D. W. Petr and T. G. Keley, "Performance Evaluation of Equivalent Capacity for Admission Control," *Proc. IEEE Wichita Conference on Communications, Networking and Signal Processing*, April 1994

[35] J. Beran, R. Sherman, M. S. Taqqu and W. Willinger, "Long-Range Dependence in Variable-Bit-Rate Video Traffic," *IEEE Transactions on Communications*, Vol. 43, No. 2/3/4, pp. 1566-1579, Feb./Mar./Apr. 1995.

[36] D. J. Reininger, D. Raychaudhuri and J. Y. Hui, "Bandwidth Renegotiation for VBR Video over ATM Networks," *IEEE JSAC*, Vol. 14, No. 6, pp. 1076-1085, August 1996.