ADMISSION CONTROL FOR ATM NETWORKS

by

Ivy Hsu and Jean Walrand

Memorandum No. UCB/ERL M94/44

20 June 1994

ADMISSION CONTROL FOR ATM NETWORKS

by

Ivy Hsu and Jean Walrand

Memorandum No. UCB/ERL M94/44

20 June 1994

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

ADMISSION CONTROL FOR ATM NETWORKS

by

Ivy Hsu and Jean Walrand

Memorandum No. UCB/ERL M94/44

20 June 1994

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

Admission Control for ATM Networks *

Ivy Hsu & Jean Walrand
Department of EECS
University of California
Berkeley, CA 94720
WLR@eecs.berkeley.edu

Abstract

We start with a brief review of ATM (Asynchronous Transfer Mode) technology and the control actions that networks can take. We then present methods to estimate the small loss probabilities in ATM networks. Small losses are achieved by averaging traffic over time in a large buffer, or by multiplexing a large number of sources when the buffer is small or negligible. In the first case (large buffer), we discuss the notions of effective and decoupling bandwidths. In the second case (many sources), we show that a good estimate of the loss probability can be obtained using a theorem of Bahadur and Rao.

We propose a call admission strategy in which connections with tight delay constraints, such as interactive audio/video, are given service priority. The delay constraints limit the acceptable buffer size. Thus the many sources asymptotic is appropriate in estimating the loss rate of such service. Effective and decoupling bandwidths are then used for the control of connections that can tolerate longer delays due to large buffers.

1 Introduction

One objective of ATM (Asynchronous Transfer Mode) networks is to transport a wide range of information, such as voice, video, and data, with diverse characteristics and quality of service requirements.

Internet transports messages without guarantee nor control on the delay and the throughput of transmissions. In fact, Internet transports messages without being aware that these messages are part of a connection. Consequently, Internet does not use information about the temporal characteristics of the connections to control the quality of service that it offers to users. Modifications of the Internet protocols are being developed to try to fix those limitations. Although some networking experts believe that Internet can be upgraded to gradually offer the benefits expected from future networks, many researchers see the advantages of the ATM technology and understand that these advantages can be gained without making the Internet applications or most of its infrastructure obsolete. This paper is concerned with ATM networks.

The basic components of ATM networks include transmission links, ATM switches, ATM interface boards installed in user equipment, and specialized servers. We will review some of these basic components in section 2. Today, you can buy local ATM switches and ATM interface boards from a dozen vendors and use them to set up local ATM networks. The cost per connected workstation is rapidly dropping to a level comparable to that of FDDI even though the ATM network can have a much larger throughput and smaller delays. A few vendors sell large ATM switches that are used by public carriers to build metropolitan and wide area networks.

*This research is supported in part by a grant of the Networking and Communications Research Program of the National Science Foundation.

The services of ATM networks, in addition to the usual Internet applications, will include interactive TV, video conferencing, high-speed bridging of LANs, video and high-fidelity broadcasts. Many of these services benefit from higher rates (e.g., Mosaic) while others necessitate them and require low delays (e.g., video conferences).

After reviewing the basic features of ATM networks in section 2, we turn our attention to control methods in section 3. Our next topic is the analysis of small overflow probabilities. In section 4, we focus on the case of large buffers. In section 5, we analyze the case of many sources. In section 6, we propose a admission control strategy based on the results of sections 4 and 5. We conclude the paper in section 6 with a few remarks.

See [13] for a detailed presentation of the technology of high-speed networks and of their analysis, design, and control. That text is complemented by an interactive CD-ROM that contains illustrations and animations of the main algorithms, protocols, and devices.

2 ATM Networks

ATM networks transport information in fixed-sized packets of 53 bytes, called *cells*, along *virtual circuits*. Virtual circuit transport means that all the cells of the same connection are identified as such and follow the same path in the network. Examples of connections include transmissions of video or audio signals and file transfers. The fixed size of cells simplifies the hardware. The small size was selected to limit the time needed to assemble cells that transport voice.

Physically, the ATM network consists of switches and user equipment connected by optical fibers and by high-speed links. An ATM switch has a number of input and output lines. When a cell arrives, the switch reads its virtual circuit number and sends the cell to the corresponding output link. The switch is equipped with buffers that can store cells when they occasionally arrive faster than they can be transmitted. ATM switches differ in how their buffers and routing fabric are arranged.

A local ATM network may consist of a single ATM switch located in some wiring closet to which a number of workstations are attached via spare telephone twisted pairs. Pairs may be used in groups of 3 or 4 to achieve a transmission rate of 100 Mbps. Alternatively, optical fibers may be installed to connect the devices. Each workstation is equipped with an ATM interface board. The interface board packages the messages or video or audio signals to be transmitted into ATM cells and reassembles the ATM cells that it receives into messages or signals. To start a connection, the workstation must exchange some signalling information with the switch. That information, also transported in cells, specifies the characteristics of the connections (e.g., average rate and burstiness) and the desired quality of service (e.g., maximum delay and acceptable loss rate). The switch, by monitoring the ongoing connections, decides whether it can accept the new connection and offer the desired quality of service while maintaining the quality of service it promised to the ongoing connections. We try to understand how the switch can make such a determination in sections 4-5. In a typical installation, 64 workstations can be connected to the central switch by links at 100 Mbps (or higher rates) each. Potentially, hundreds of simultaneous connections are in progress. Some of the connections are video conferences at 1.5 Mbps each, others are real-time transfers of animations produced by simulations on high-performance workstations, others are transfers of high-resolution medical images, and other still are connections between bridges attached to FDDIs or other LANs.

A metropolitan area ATM network can consist of a single ATM switch that belongs to some public carrier (e.g., a local phone company). Users are attached to the switch by 1.5 Mpbs, 45 Mbps, 155 Mbps, or possibly higher-rate lines, either twisted pairs, coaxial cable, or optical fiber. The local access between the users and the public carrier can be provided by a cable TV operator, by the phone company, or by a third party. The metropolitan network connects local ATM networks, other LANs, specialized machines (e.g., video-servers to set-top boxes).

A wide area ATM network consists of a mesh-like network of ATM switches. The backbone consists of optical fibers at 155 Mbps, 622 Mbps, or multiples. This backbone connects the metropolitan networks we discussed earlier.

A few scenarios are plausible for the implementation of ATM networks. Some installations upgrade their LANs from Ethernets to local ATM networks to provide video conferencing or high-speed access to specialized file- and compute-servers or for implementing parallel processing across their networks. Other users subscribe to high-speed services such as the switched multi-megabit data service (SMDS) of a public carrier for connecting LANs. Residential customers subscribe to interactive TV and to videophone services from the cable TV or phone company. Eventually, these services are transported by an ATM wide area network that piggybacks improved Internet applications. We can barely imagine the impact of a truly multimedia version of Internet.

There is not yet a universal agreement on quality of service parameters. Values that are often cited are a maximum delay of about 300 ms for interactive voice/video and of a few seconds for interactive database services. The acceptable loss rate ranges from 10^{-8} for data to a few percent for video and audio. To be specific, let us choose a delay of about 200 ms for interactive audio/video with a loss rate of 10^{-5}. Let us call this application "video." For remote access to databases, let us choose a delay of 1 second and a loss rate of 10^{-7}. We call this application "database." Keep in mind that these sets of requirements are our working hypotheses and that they are not standardized values.

To place the delay values in some perspective, note that a 53-byte cell takes about 3 microseconds to be transmitted by a 155 Mbps link so that a delay of 30 ms corresponds to 10^4 cell transmission times. If the connection goes through 10 links at 155 Mbps each, then the cell can be queued behind 1,000 other cells at each node and still face a queueing delay of less than 30 ms through the network. This should guarantee a maximum total end-to-end delay less than our target 200 ms. In addition, the cell faces a propagation delay (about 5 μs per km) and processing delays (a few 100 μs per node). The delay jitter (i.e., fluctuations) is caused by variations in the queueing delays. Another important cause of delay is the packetization delay: the time it takes to assemble a cell. If the source produces bits at the rate of R bps, then it takes $48 \times 8/R$ seconds to assemble the 48 bytes that are in a 53-byte cell. (The remaining 5 bytes of the cell are occupied by the header.) For a standard (64-kbps) voice stream, this packetization delay is 6 ms. For higher bit rate streams, the packetization delay is essentially negligible.

These elementary considerations show that the delay requirement for our *video* application is satisfied if the queueing does not exceed 1,000 cells per node. For our *database* application, the queueing should not exceed 20,000 cells per node.

Cells are lost because of transmission errors and when buffers overflow. The cell loss rate due to transmission errors is roughly equal to the bit error rate multiplied by 424 (= 53 bytes). Thus, if the BER is 10^{-12}, then the cell error rate is about 4×10^{-10}. The probability that a buffer overflows can be made negligible by using a very large buffer. However, this defeats the objective of keeping the delays small.

Remembering our previous analysis, we see that a reasonable objective for *video* is to control the network so that buffers that can hold about 1,000 cells overflow with a probability less than 10^{-6} (assuming 10 nodes). For *database* the objective is for buffers that can hold 20,000 cells to overflow with a probability less than 10^{-8}. We will refer to these values to check the conclusions of our analysis. In this preliminary discussion, we ignore the interactions of video and database services. Presumably, we give priority to video when both services share a buffer. Another possibility is to segregate buffers and serve them in parallel with different fractions of the bandwidth. We discuss that point later.

3 Control Actions

What can the network do to control its operations? The basic control actions are admission control, routing, flow control, traffic shaping, bandwidth and buffer allocation.

Admission control: When a connection is requested with its traffic descriptors and quality of service requirements, the network decides whether to accept or reject the connection. The network determines if it has the necessary resources available to meet the requirements of the new connection while maintaining those of the ongoing connections.

Routing: The connections are transported as virtual circuits. The network selects a path for the new connection. Note that the admission control and routing are coupled problems since the network accepts a call only if it can find a suitable path.

Flow control: While transporting a connection, a network node may decide to postpone the transmission of cells. The objective of the node is to regulate its output to avoid swamping a neighbor.

Traffic shaping: The source may smooth out its cell stream before sending them to the network. This traffic shaping makes the stream easier to transmit because it reduces the amount of storage the stream requires in the network.

Bandwidth allocation: A switch can decide which cell it transmits next on a given fiber. For instance, the switch may give priority to *video* traffic over *database* traffic.

Buffer allocation: When a buffer shared by *video* and *database* cells is full and a new database cell arrives, the switch may discard a video cell to make room for a database cell.

A key question when designing control mechanisms is the information available to the agent choosing the control actions. A related question is the potential obsolescence of the available information: dated information may not be useful. To appreciate this question, consider two possible strategies for traffic shaping: open-loop and link-feedback. One open-loop traffic shaping technique for video signal uses a multi-resolution coding that separates coarse resolution information and finer resolution information in different cells. The mechanism marks the fine resolution cells so that the network switch can discard them when it becomes congested. The possibility of discarding a significant fraction of cells when the switch becomes congested substantially reduces the likelihood of loosing coarse resolution cells because of overflows. A link-feedback strategy uses two different types of video compression: one that produces a coarse output with a small bit rate and a high-quality compression with a large bit rate. The network switch indicates to the user when it becomes congested and the algorithm switches to the low resolution mode. This feedback method is potentially more responsive to the network congestion and avoids sending cells that have to be flushed anyway. However, if the propagation time between the switch and the user is very long, then the feedback mechanism may be too sluggish to be effective.

Similar considerations apply to flow control. Say that the source throttles its output to avoid congesting the network. Two extreme methods can be used: window flow control and rate control. When using window flow control,

the destination signals back to the source when it receives cells. This information enables the source to limit the number of its cells inside the network to a specified value. The rate control is open-loop and limits the burstiness of the stream. A simple mechanism for doing this is the so-called *leaky bucket.* (A variation is the generic cell rate algorithm GCRA recommended by the ATM forum. See [2].) Essentially, the algorithm grants permits at a fixed rate but only a specified number of permits can be accumulated. To transmit a cell, the stream must use up one of its permits. The algorithm limits the average rate of cells to the rate at which it provides permits and it also limits the size of a burst of cells to the number of permits that can be accumulated. Intuition suggests that window flow control becomes ineffective when the buffer-delay product of the stream is large. For instance, consider a source that produces cells at 1 Gbps. Assume that the connection goes coast-to-coast with a total delay of about 25 ms. The bandwidth-delay product is then about 25 Mbits. If it limits the number of cells to fewer than the equivalent of 25 Mbits, the window flow control slows down the connection. If it limits that number to 25 Mbits, then there is no guarantee that the cells will not pile up in a few nodes, so that the control is not effective. For such sources, a leaky bucket control may prove more effective. In practice, a 45-Mbps source (HDTV) is already quite fast and it is not sure that window flow control cannot be made effective for such rates. However, common wisdom seems to currently favor open-loop rate control.

There is a compromise between quality of service and number of connections that can be accommodated. One method for reaching a satisfactory compromise is through pricing mechanisms. Various pricing mechanisms are being explored. We do not discuss this important issue here for lack of space.

With this general overview behind us, we can begin our exploration of analytical methods. Before turning to this exploration, let us make a few general observations. Our objective will be to develop methods that network engineers can use to improve networks. Specifically, we focus on methods that estimate cell loss rates. In the paper, we explain how to estimate the loss rate if one has a good model of the traffic. In practice, such good models may be obtained by running an estimation algorithm as the network operates. One can also design control algorithms that learn desirable control actions without attempting to estimate parameters of traffic models. We believe that the analysis that we present can guide the development of such adaptive control algorithms. Finally, keep in mind that the performance evaluation of ATM networks is a relatively new subject and although some useful results are available much remains to be done.

Most sources produce *bursty* cell streams. That is, the peak instantaneous rate of such streams is much larger than their long-term average rate. Allocating bandwidth according to the peak rate results in low resource utilization. For connections that can tolerate some amount of loss, a significant saving in bandwidth can be achieved via *time-averaging* of fluctuations of such streams and by *statistical multiplexing* of many traffic streams. Such averaging smoothes out the fluctuations of the streams. Time averaging occurs in large buffers and is the subject of section 4. Statistical multiplexing is discussed in section 5.

4 Large Buffers

In this section, we consider a situation where the loss rate is kept small by using a large buffer. The buffer stores bursts of cells that arrive faster that they can be transmitted. It is unlikely that the bursts are frequent enough to make the buffer overflow.

Except for very simple source models (e.g., Poisson or a Markov modulate process with few states), it is difficult to analyze exactly the small loss rate at a large buffer. The cause of the difficulty is that the state space of a Markov model of the source and buffer system is large and this fact makes the numerical analysis complex. Because of that complexity, and with the objective of deriving tractable results, we turn to an asymptotic analysis of the loss rate as the buffer becomes large. Not surprisingly, the loss rate becomes smaller as the buffer increases. When the buffer is large, the loss rate is well approximated by an exponential function of the buffer size. Roughly, the loss rate is approximately $\exp\{-BI(C)\}$ where B is the buffer size (in cells, say) and $I(C)$ is some increasing function of the transmitter rate C and, obviously, of the statistics of the traffic. We argue that we should choose C large enough so that $\exp\{-BI(C)\}$ is small enough. For our video source, we want $\exp\{-BI(C)\} \approx 10^{-6}$ when $B = 1,000$. (See section 2.) Thus, we want $I(C) \approx 1\%$. For our database source, we want $\exp\{-BI(C)\} \approx 10^{-8}$ for $B = 20,000$, so that $I(C) \approx 0.1\%$. We designate this target value of $I(C)$ by δ. Thus, $\delta = 1\%$ for video and $\delta = 0.1\%$ for database. (Once again, recall that these are working hypotheses and not standards.)

Now suppose that there are J types of traffic, and n_j sources of type j are multiplexed onto an output link. We want

$$\lim_{B\to\infty} \frac{1}{B} \log P(W \geq B) \leq -\delta,$$

where W is the stationary buffer occupancy. It is shown [12, 7, 9, 11] that under appropriate assumptions, this constraint can be satisfied when

$$\sum_{j\in J} n_j \alpha_j(\delta) \leq C, \tag{1}$$

where C is the total output link rate, and $\alpha_j(\delta)$ is the *effective bandwidth* for the type j source corresponding to δ. The most general result is in [12, 7] where the above conclusion is shown to hold for a very large class of traffic sources.

Equation (1) allows a simple policy for call acceptance that is analogous to that of the traditional circuit-switch networks, since the effective bandwidth for each call can be determined independently of the other types of calls. Furthermore, since $\alpha_j(\delta)$ lies between the mean and peak rates of the source, the difference between the peak rate and $\alpha_j(\delta)$ is the bandwidth saving through multiplexing.

The effective bandwidth $\alpha(\delta)$ of a source that produces a random number $A(t)$ of cells in t seconds can be calculated as

$$\alpha(\delta) = \frac{\Lambda(\delta)}{\delta} \tag{2}$$

where

$$\Lambda(\delta) = \lim_{t\to\infty} \frac{1}{t} \log E\{e^{\delta A(t)}\}. \tag{3}$$

When δ is small enough, one may be able to justify the following approximation:

$$\log E\{e^{\delta A(t)}\} \approx \log[1 + \delta E\{A(t)\} + \frac{\delta^2}{2} E\{A(t)^2\}] \approx \delta E\{A(t)\} + \frac{\delta^2}{2} E\{A(t)^2\},$$

which results in

$$\alpha(\delta) \approx \lambda + \frac{1}{2}\delta D^{\,2} \tag{4}$$

where

$$\lambda := \lim_{t\to\infty} \frac{1}{t} E\{A(t)\} \text{ and } D^2 := \lim_{t\to\infty} \frac{1}{t} E\{A(t)^2\}.$$

In the above definitions and in (4), λ is the average rate of the stream and D^2 is called its *dispersion*. This simple approximation for $\delta \ll 1$ shows that the effective bandwidth increases with the burstiness of the stream and indicates that an approximate measure of burstiness (for large buffers and small δ) is the dispersion. Note that $\delta \ll 1$ means that we are willing to lose quite a few cells so that the second moments of the stream are good predictors of the loss rate, as one might guess from a functional central limit theorem. When δ is larger, then the losses are determined by the tail behavior and the higher moments cannot be neglected in the calculation of the effective bandwidth.

Formulas or algorithms are available for calculating the effective bandwidth of a large class of models. See e.g. [12]. Methods for on-line estimation of the effective bandwidth are the subject of current research and so are adaptive techniques for selecting a suitable value of C.

The above result deals with a single buffer and may be usable for a local ATM network. When traffic goes through multiple buffers, the situation becomes more complex. When streams share a buffer, they interact and modify each other's statistics and effective bandwidth. At first the problem appears intractable: the statistics of a stream depend on those of all the streams it interacted with and the same is true for the latter streams. Fortunately, a simplification occurs. One can show that if the transmitter rate C of a buffer is large enough, then a stream preserves its effective bandwidth as it goes through the buffer. Specifically, stream j preserves its effective bandwidth if C is larger than the sum of $\alpha_j^*(\delta)$ and the average rate of all the other streams that share the buffer. Here, $\alpha_j^*(\delta)$ is the *decoupling bandwidth* of stream j. Formulas for calculating that decoupling bandwidth are given in [7] where the applications of that result to call admissions are discussed.

The above approach works well only if the buffer is large. Numerical and simulation experiments show that admission control based on the notions of effective and decoupling bandwidth may be too conservative. What is happening is that the method is based on the estimate of the exponential rate of decay of the loss probability and it ignores the pre-exponential factor which may be very small.

In the next section, we analyze the case of many sources.

5 Many Sources

Many real-time applications, such as voice and video, are subject to tight delay constraints and cannot allow large buffers. Over the small buffer region, the effective bandwidths are significantly overestimated, leading to inefficient bandwidth utilization. Thus it is necessary to consider the small buffer case separately.

The precise boundary that separates the large buffer case from the small buffer case is not easy to determine. Our experiments suggest the following rule of thumb for ON-OFF Markov sources. The buffer is large if it can store at least a few bursts from a source. Here, a burst is understood as the average number of cells that a source produces when it is ON. For instance, if we model a variable bit rate video source (say MPEG 2) as being ON-OFF, then the average duration of the ON period corresponds to the average duration of a scene of the movie where the data compression is not very effective. The average duration of a typical movie scene is a few seconds, say 6 seconds. During that time, the source will produce thousands of cells. Indeed, the active period produces bits at a rate that is a few times the average rate, say 4 Mbps. During these 6 active seconds one source produces about 24 Mbits, i.e., about 6×10^4 cells. In section 2, we argued that a reasonable buffer for video was about 1,000 cells, shared by a number of video sources. Thus, we are led to conclude that the video buffer is a small buffer. Intuitively, the fluctuations of the video signal are slow and we cannot smooth them out with a buffer of an acceptable size. To avoid having to design the network for the peak rate of traffic, we can multiplex many sources.

In this section we analyze the case of small buffers. If the number of virtual circuits routed through a link is large, the combined input rate from all the virtual circuits rarely exceeds a value larger than the mean. Indeed, as we will see, the buffer overflow probability is roughly inversely exponential in the number of virtual circuits N. We refer to this multiplexing approach as the *many sources asymptotic*, as opposed to the large buffer asymptotic.

Consider then a large number N of sources that share a buffer served by a transmitter with rate Nc. Thus, c is the service rate per source. The buffer can store Nb cells, i.e., b cells per source. We want to analyze the buffer overflow probability for large N.

We decompose our analysis into two parts. In the first part, we consider the case $b = 0$ (zero buffer). In the second part, we examine the effect of $b > 0$ (small buffer).

5.1 Zero Buffer

The idea of not using a buffer may seem absurd, specially in view of the relatively low cost of memories. However, we will see that this case is most interesting and leads to somewhat counter-intuitive results.

We denote by Y_j the stationary rate of source j, for $j = 1, \ldots, N$. We assume that the sources are independent and identically distributed, so that the random variables $\{Y_j, j = 1, \ldots, N\}$ are i.i.d. Note that we are ignoring the dynamics of the rates and are focusing instead on their instantaneous distribution. The transmitter is unable to keep up with the traffic produced by the N sources as soon as $Y_1 + \cdots + Y_N > Nc$. We can estimate the fraction of time that this situation occurs by using Cramer's theorem [4],

$$P(Y_1 + \cdots + Y_N > Nc) \approx e^{-NI(c)}, \tag{5}$$

where $I(c) \equiv \sup_\theta[\theta c - \varphi(\theta)]$, and $\varphi(\theta) \equiv \log E[\exp(\theta Y_1)]$.

For ON-OFF Markov sources with birth rate λ, death rate μ, and ON-rate a, we note that Y_j takes values in $\{0, a\}$ with $P(Y_j = a) = 1 - P(Y_j = 0) = \lambda/(\lambda + \mu) \equiv \rho$. Consequently,

$$\varphi(\theta) = \log(\frac{\mu}{\lambda + \mu} + \frac{\lambda}{\lambda + \mu} e^{a\theta}).$$

The value of θ that maximizes $I(c)$ is $\theta_c = \frac{1}{a} \log(\frac{c\mu}{\lambda(a-c)})$, at which $I(c) = \frac{c}{a} \log(\frac{c/a}{\rho}) + (1 - \frac{c}{a}) \log(\frac{1-c/a}{1-\rho})$.

We will use a refinement of Cramer's theorem due to Bahadur and Rao [3] (see also [6], section 3.7). Numerical experiments show that the increase in complexity of the algebra results in much improved estimates. Using the Bahadur and Rao theorem, we find (see Appendix)

$$P(Y_1 + \cdots + Y_N > Nc) \approx \frac{1}{\sqrt{2\pi}\sigma\theta_c\sqrt{N}} e^{-NI(c)}. \tag{6}$$

In the above expression,

$$\sigma^2 = \frac{M''(\theta_c)}{M(\theta_c)} - c^2$$

where

$$M(\theta) := E[\exp(\theta Y_1)]$$

and θ_c achieves the maximum in

$$I(c) = \sup_{\theta}[\theta c - \varphi(\theta)].$$

As a numerical example, we use the following values for two-state sources, each with states 1 and 0: $1/\lambda = 45$ sec, $1/\mu = 5$ sec, $a_1 = 3$ Mbits/sec, $a_0 = 1$ Mbits/sec, and $c = 1.55$ Mbits/sec. This two-state source is a constant source with rate a_0 plus the ON-OFF source define earlier with $a = a_1 - a_0$.

Note that the number of sources that are in state 1 has a binomial distribution. Thus the probability that the aggregate input rate exceeds the output rate can be represented exactly as

$$\sum_{k \geq N(c-a_0)/(a_1-a_0)}^{N} \left(\begin{array}{c} N \\ k \end{array} \right) \rho^k (1-\rho)^{N-k}.$$

For large N, we approximate $N!$ by Stirling's formula: $N! \sim N^N e^{-N} \sqrt{2\pi N}$.

This is compared with the approximation using Cramer's theorem given in Eqn. 5 and the refined approximation in Eqn. 6. The result is plotted over a range of values for N in Figure 1. Note that Eqn. 6 not only is much closer to the exact value than Eqn. 5, but also approximates the exact value well even for very small N.

The Bahadur-Rao theorem enables us to analyze the case of multi-rate sources. That theorem can also be used to analyze the overflows of mixtures of different types of sources. To do this, say that we have 50% of sources of type Y and 50% of sources of type Z. We can then construct an hybrid source W that is of type $Y + Z$. The analysis of such a situation reveals that the value of c required to achieve a small loss probability cannot be written as the sum of the necessary rates for the Y-sources and the Z-sources. Thus, unfortunately, for small buffers there is no additive result similar to the effective bandwidth. Our preliminary work indicates that the boundary of the acceptable region is convex. It appears that the "closer" Y and Z are in terms of the peak and average rates, the closer this boundary is to a hyperplane.

5.2 Small Buffer

In this section we explore the effect of the small buffer b per source on the loss probability. We combine the analysis of Weiss [14] with our application of the Bahadur-Rao theorem.

Recall the model. There are N i.i.d. sources that share a buffer with capacity Nb and served at rate Nc.

Following Weiss, we argue that the buffer overflows because of the conjunction of two events. First, the sources become active at the same time, so that their total rate reaches Nc. Second, their total rate continues to exceed Nc long enough so that the buffer accumulates Nb cells and starts overflowing. We calculated the probability of the first event in the previous section, using the Bahadur-Rao theorem. In [14], Weiss calculates that, for ON-OFF sources, the likelihood of the second event is approximately given by

$$\exp(-NC_2\sqrt{b})$$

when N is large. The value of C_2 depends on the parameters of the sources. For the ON-OFF sources defined earlier, $C_2 = \sqrt{\frac{1}{a}(\lambda(1-\frac{c}{a}) + \mu\frac{c}{a}\log(\frac{\mu c}{\lambda(a-c)}) - 2(\mu\frac{c}{a} - \lambda(1-\frac{c}{a})))}$.

Combining this calculation with the result of the previous section, we conclude that the loss probability is approximately given by

$$\frac{1}{\sqrt{2\pi}\sigma\theta_c\sqrt{N}} \exp\{-N[I(c) + C_2\sqrt{b}]\}.$$

We compare the approximation with Anick's exact derivation [1]. For the same two-state sources as above, we fix $N = 100$, and plot the results over a range of values for b in Figure 2.

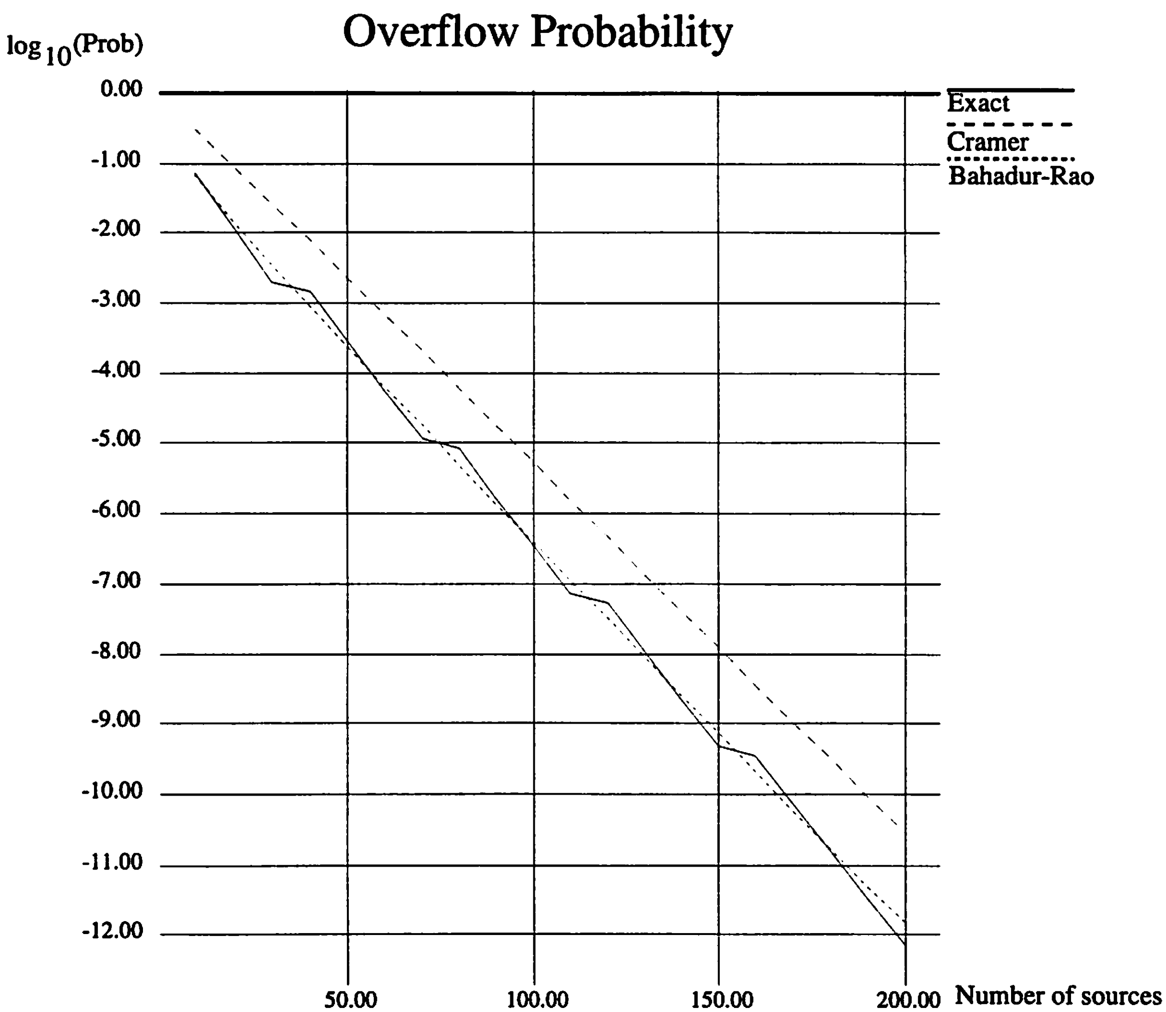

Figure 1: Exact and approximate values of the probability that the aggregate input rate exceeds the output link rate.

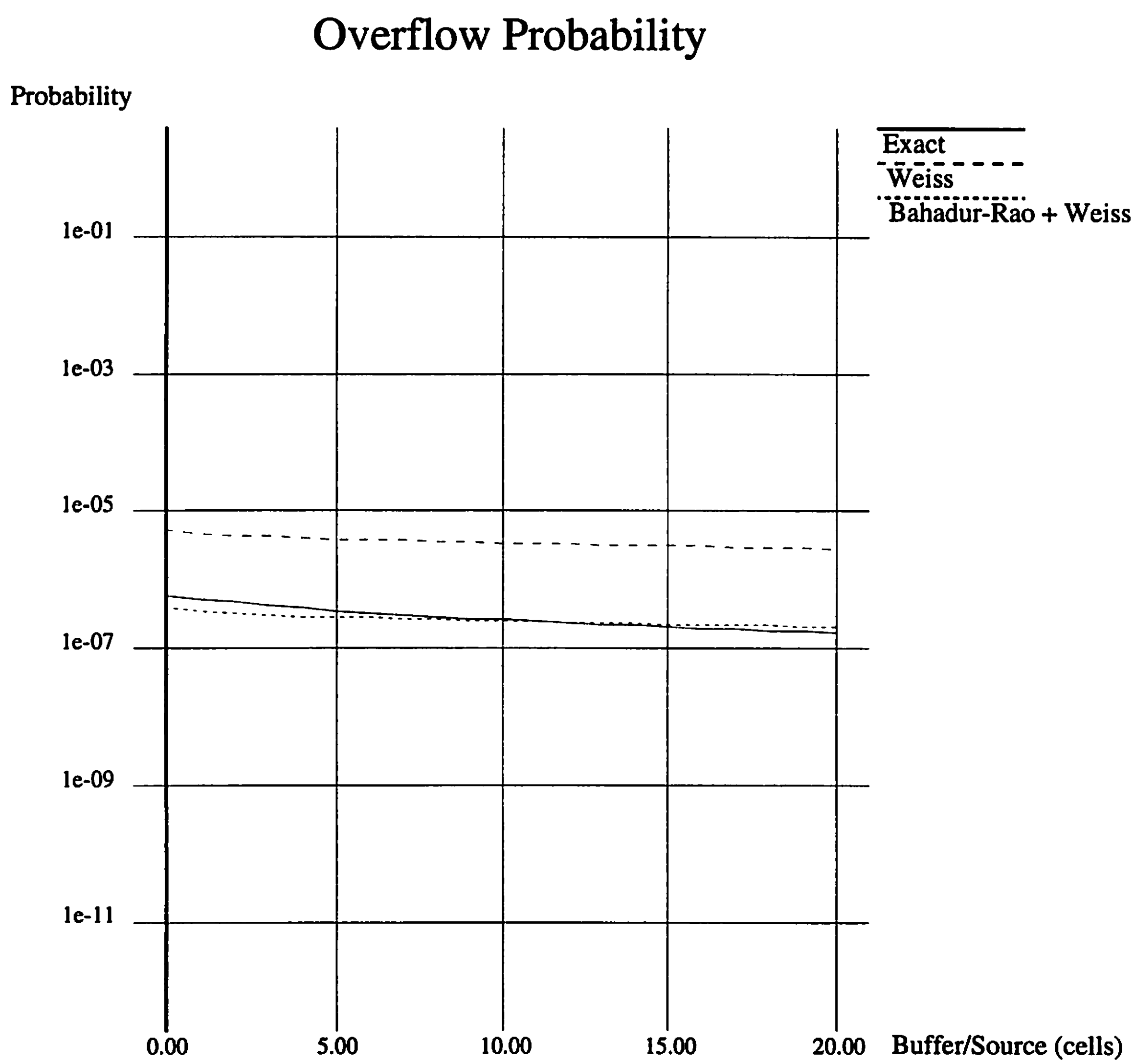

Figure 2: Exact and approximate values of the buffer overflow probability.

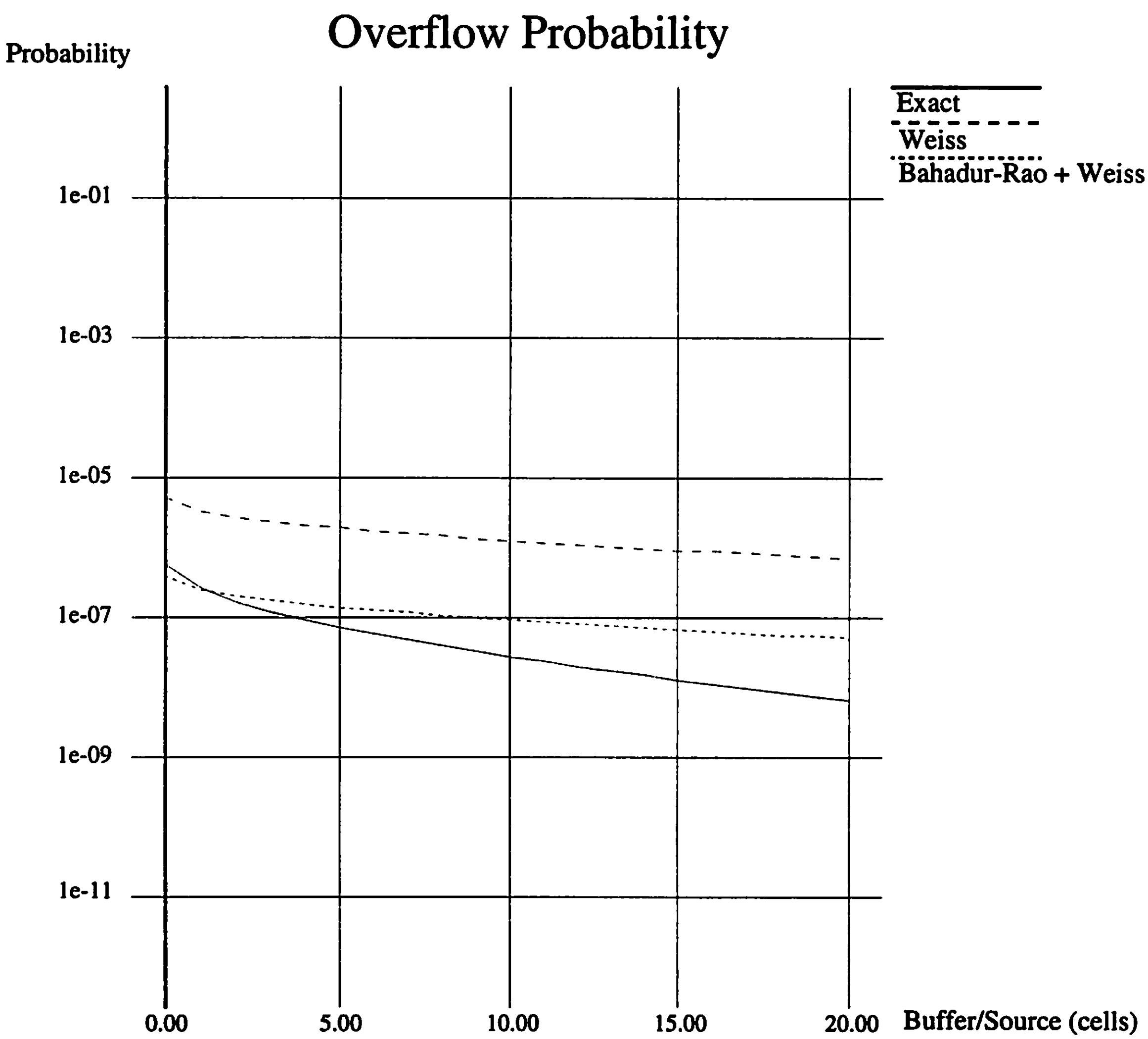

Figure 3: Buffer overflow probability for sources with $1/\lambda = 4.5$ sec and $1/\mu = 0.5$ sec.

Note that in this case the overflow probability does not improve very much by the presence of a small buffer.

For comparison, Figure 3 shows the buffer overflow probability when the transition rates of the Markov sources are ten times faster ($1/\lambda = 4.5$ sec, $1/\mu = 0.5$ sec). Note two observations:

(1) The benefit of placing a buffer at the output line is much greater in this case. This is because when the source alternates between the two states at a faster rate, the expected duration over which the aggregate input rate exceeds the output rate is much shorter. Indeed, define $K \equiv \lceil N(c - a_0)/(a_1 - a_0) \rceil$ and $A \equiv \{K, K+1, \ldots, N\}$. Then by solving the first-step equations, the expected sojourn time in A can be expressed as

$$E[\text{sojourn time in } A] = \sum_{i=0}^{N-K} \left[\frac{1}{(N-i)\mu} \prod_{j=i+1}^{N-K} \frac{j\lambda}{(N-j)\mu} \right],$$

which is inversely proportional to μ.

(2) The approximation $\exp(-N\sqrt{b}C_2)$ diverges more quickly from the exact values in this case.

These two observations point out that one must normalize the time scale in order to determine if the buffer can be considered "small." Take the time unit to be the average time that a source spends in the ON state ($= 1/\mu$), and the data unit the number of bits generated in one time unit in the ON state ($=a/\mu$). Then the normalized buffer size is $b\mu/a$. Therefore the system in Figure 3 has ten times the normalized buffer than that in Figure 2.

6 Admission Control

In this section, we propose guidelines for admission control in an ATM network. We are aware that these guidelines need to be refined and we propose them mainly as a way of summarizing the results of the paper.

Consider the problem of designing and controlling a wide area ATM network that supports video and database services in addition to best-effort traffic (e.g., email). Recall our definitions of video and database services in section 2.

We simplify the design by assigning priority to video traffic. We design the video traffic assuming zero buffers. The motivation for this assumption is that buffers would only reduce the loss rate by a small amount, as we saw in section 5.2. Say that we use links at 155 Mbps. We calculate values of N and c so that $Nc = 155Mbps$ and so that

$$\frac{1}{\sqrt{2\pi}\sigma\theta_c\sqrt{N}} e^{-NI(c)} \approx 10^{-6}.$$

Of course, we need a satisfactory model of the video sources to calculate these values of N and c. Once this calculation is performed, we know how many video sources can be transmitted through each one of our switches.

The admission and routing of video calls is then reduced to the corresponding problems for circuit-switched networks. Indeed, we know the capacity N of each link. A call is blocked if a path cannot be found that has spare capacity. In fact, we can benefit from the lessons learned for the telephone network and implement an alternate routing strategy with trunk reservations.

We then propose to analyze the database traffic as follows. If N video sources can go through the switch, the average available rate for the non-video traffic is $C' := N(c - m)$, where m is the average rate per video connection ($m < c$). In a first approximation, we consider this rate as fixed. We propose to solve the admission and routing problem for the database traffic by using the effective bandwidth and decoupling bandwidth heuristic. The justification for this approach is that the acceptable delays are much larger and therefore, so are the queue lengths.

Finally, the best effort traffic is carried with the left-over capacity, at the lowest priority.

7 Conclusion

In this paper we reviewed methods that estimate the small loss probabilities in ATM networks.

We explained that these small losses are achieved by averaging traffic over time in a large buffer or by multiplexing a large number of sources.

In the first case (large buffer), we discussed the notions of effective and decoupling bandwidths.

In the second case (many sources), we analyzed the zero buffer and the small buffer cases separately. We explained that the loss rate can be estimated by assuming zero buffers and that a good estimate can be derived using a theorem of Bahadur and Rao.

Finally, we proposed a strategy for call admission control based on the previous results. The network is first designed for video traffic by using the zero-buffer estimates. That traffic is admitted and routed using alternate routing with trunk reservations. The next priority is for database traffic and the network is designed and controlled using the effective and decoupling bandwidth ideas. This control amounts to a second level of alternate routing and trunk reservation. Finally, the best effort traffic is allocated the left over capacity.

8 Appendix: Bahadur-Rao Theorem

Let $\nu(\cdot)$ be the distribution of the i.i.d. random variables $Y_1, Y_2, \ldots$. Define $M(\theta) = E[\exp(\theta Y_1)]$, and the "tilted" probability distribution

$$\nu_\gamma(dx) \equiv \frac{\exp(\gamma x)\nu(dx)}{\int \exp(\gamma x)\nu(dx)} = \frac{\exp(\gamma x)\nu(dx)}{M(\gamma)}.$$

Thus the moment generating function of this distribution is

$$\int \exp(\theta x)\nu_\gamma(dx) = \frac{M(\theta+\gamma)}{M(\gamma)}.$$

Taking the inverse transform of $[\frac{M(\theta+\gamma)}{M(\gamma)}]^N$,

$$\nu_\gamma^{*N}(dx) \equiv \text{convolution of } N \text{copies of } \nu_\gamma(dx) = \frac{\exp(\gamma x)\nu^{*N}(dx)}{M(\gamma)^N}.$$

Thus

$$\begin{aligned} P(\frac{1}{N}(Y_1 + \cdots + Y_N) \geq c) &= \int_{Nc}^{\infty} \nu^{*N}(dx) \\ &= M(\gamma)^N \int_0^\infty e^{-\gamma(x+Nc)}\nu_\gamma^{*N}(dx+Nc). \end{aligned}$$

Substitute $\gamma = \theta_c$,

$$P(\frac{1}{N}(Y_1 + \cdots + Y_N) \geq c) = e^{-NI(c)} \int_0^\infty e^{-\theta_c x}\nu_{\theta_c}^{*N}(dx+Nc).$$

Note that the mean of $\nu_{\theta_c}(\cdot)$ is $\frac{M'(\theta_c)}{M(\theta_c)} = c$, and its variance is $\sigma^2 = \frac{M''(\theta_c)}{M(\theta_c)} - c^2$. Therefore $\nu_{\theta_c}^{*N}(\cdot + Nc)$ has zero mean and variance $N\sigma^2$. Approximate it by $\mathcal{N}(0, N\sigma^2)$ and let $y = x/\sqrt{N}\sigma$,

$$\begin{aligned} \int_0^\infty e^{-\theta_c x}\nu_{\theta_c}^{*N}(dx+Nc) &\approx \frac{1}{\sqrt{2\pi}\sqrt{N}\sigma}\int_0^\infty e^{-\theta_c x}e^{-\frac{x^2}{2N\sigma^2}}dx \\ &= \frac{1}{\sqrt{2\pi}}e^{\frac{1}{2}(\theta_c\sqrt{N}\sigma)^2}\int_0^\infty e^{-\frac{1}{2}(y+\theta_c\sqrt{N}\sigma)^2}dy \\ &= \frac{1}{\sqrt{2\pi}}e^{\frac{1}{2}(\theta_c\sqrt{N}\sigma)^2}\int_{\theta_c\sqrt{N}\sigma}^\infty e^{-y^2/2}dy. \end{aligned}$$

Since $\int_u^\infty \exp(-y^2/2)dy \sim \frac{1}{u}\exp(-u^2/2)$ as $u \to \infty$ ([8], Theorem 1.3), we further obtain the following approximation:

$$P(\frac{1}{N}(Y_1 + \cdots + Y_N) \geq c) \approx \frac{1}{\sqrt{2\pi}\sigma\theta_c\sqrt{N}} e^{-NI(c)}. \tag{7}$$

Acknowledgments

This paper summarizes the work of many researchers. The authors are particularly grateful to Professors Courcoubetis, de Veciana, Kesidis, Varaiya, and Weber for their contributions and suggestions.

References

[1] D. Anick, D. Mitra, and M. M. Sondhi (1982)
"Stochastic Theory of a Data-Handling System with Multiple Source," *Bell System Technical Journal*, vol. 61, no. 8, Oct 1982, pp. 1871-1894.

[2] The ATM Forum (1993)
ATM User-Network Interface Specification: Version 3.0, PTR Prentice-Hall.

[3] R. R. Bahadur and R. R. Rao (1960)
"On Deviations of the Sample Mean," *Ann. Math. Statis.*, vol. 31, 1960, pp. 1015-1027.

[4] J. A. Bucklew (1990)
Large Deviation Techniques in Decision, Simulation, and Estimation, Wiley.

[5] C. Courcoubetis and R. Weber (1993)
"Buffer Overflow Asymptotics for a Switch Handling Many Traffic Sources," *preprint.*

[6] A. Dembo and O. Zeitouni (1993)
Large Deviations Techniques and Applications, Jones and Bartlett.

[7] G. de Veciana (1993)
Design Issues in ATM Networks: Traffic Shaping and Congestion Control, PhD thesis, Dept. of EECS, Univ. of California, Berkeley.

[8] R. Durrett (1991)
Probability: Theory and Examples, Wadsworth and Brooks.

[9] R. J. Gibbens and P. J. Hunt
"Effective Bandwidths for the Multi-Type UAS Channel," *Queueing Systems*, no. 9, pp. 17-28.

[10] J. Y. Hui (1988)
"Resource Allocation for Broadband Networks," *IEEE Journal on Selected Areas in Communications*, vol. 6, no. 9, pp. 1598-1608.

[11] F. P. Kelly (1991)
"Effective Bandwidths at Multi-Class Queues," *Queueing Systems*, no. 9, pp. 5-16.

[12] G. Kesidis, J. Walrand, and C.S. Chang (1993)
"Effective Bandwidths for Multiclass Markov Fluids and other ATM Sources," *IEEE/ACM Trans. Networking*, August 1993.

[13] P. Varaiya and J. Walrand (1995)
High-Speed Networks - Building the Information Superhighway, to be published by Morgan Kaufman, 1995.

[14] A. Weiss (1986)
"A New Technique for Analyzing Large Traffic Systems," *Adv. Appl. Prob.*, pp. 506-532.