ADMISSION CONTROL AND ROUTING IN ATM NETWORKS USING INFERENCES FROM MEASURED BUFFERED OCCUPANCY

by

C. Courcoubetis, G. Kesidis, A. Ridder,
J. Walrand, and R. Weber

Memorandum No. UCB/ERL M91/37

26 April 1991

ADMISSION CONTROL AND ROUTING IN ATM NETWORKS USING INFERENCES FROM MEASURED BUFFERED OCCUPANCY

by

C. Courcoubetis, G. Kesidis, A. Ridder,
J. Walrand, and R. Weber

Memorandum No. UCB/ERL M91/37

26 April 1991

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

ADMISSION CONTROL AND ROUTING IN ATM NETWORKS USING INFERENCES FROM MEASURED BUFFERED OCCUPANCY

by

C. Courcoubetis, G. Kesidis, A. Ridder,
J. Walrand, and R. Weber

Memorandum No. UCB/ERL M91/37

26 April 1991

ELECTRONICS RESEARCH LABORATORY

College of Engineering
University of California, Berkeley
94720

Admission Control and Routing in ATM Networks using Inferences from Measured Buffered Occupancy*

C. Courcoubetis[1] G. Kesidis[2] A. Ridder[2] J. Walrand[2]

R. Weber[3]

1. Institute of Computer Science, Research Center of Crete, Heraklion, Crete.

2. EECS Dept., University of California, Berkeley, CA94720.

3. Queen's College, University of Cambridge, Cambridge, U.K.

*Supported by: NSERC of Canada, Pacific Bell, Micro Grant of the State of California, ARO Grant Number DAAL03-89-K-0128

Abstract

We address the issue of call acceptance and routing in ATM networks. Our goal is to design an algorithm that guarantees bounds on the fraction of cells lost by a call. The method we propose for call acceptance and routing does not require models describing the traffic. Each switch estimates the additional fraction of cells that would be lost if new calls were routed through the switch. The routing algorithm uses these estimates. The estimates are obtained by monitoring the switch operations and extrapolating to the situation where more calls are routed through the switch. The extrapolation is justified by a scaling property. To reduce the variance of the estimates, the switches calculate the cell loss that would occur with smaller buffers. A way to choose the sizes of the small buffers in order to minimize the variance is discussed. Thus, the switches constantly estimate their spare capacity. Simulations were performed using Markov fluid sources to test the validity of our approach.

1 Introduction

Asynchronous transfer mode (ATM) is a form of packet switching that is proposed for broadband networks. In ATM, data is divided into 53 byte cells that are multiplexed on a time-slotted channel. When network traffic is bursty, ATM's use of statistical multiplexing results in an efficient use of bandwidth [1].

ATM uses virtual circuits (VC). Every cell of this call will use the same route. When a cell arrives at a switch, the switch determines its output link by looking at the VC number in the header of the cell, and using a lookup table in the switch's memory. The VC number of every call and the lookup tables of every switch are determined by the routing algorithm [2].

Virtual circuits share buffers in switches. The method of call acceptance described in this paper can be used with various switch architectures (e.g. output-buffer, shared-buffer, Batcher-banyan). If, for example, output buffer switches are used, each output link of a switch has an associated buffer. When the traffic offered to a link exceeds the link's capacity, cells begin to accumulate in the buffer. When a cell arrives at a full buffer, it is lost. We wish to minimize the fraction of cells lost due to buffer overflows to the point where it is comparable to losses due to transmission errors (10^{-8}). Since cell losses are rare and delays are small, the statistics of a call does not significantly change along its virtual circuit. Therefore, we assume that calls of the same type (e.g. video, speech, etc.), at a given switch, have the same statistics.

Our goal is to design an algorithm that guarantees bounds on the fraction of cells lost due to buffer overflows. The method we propose does not require models describing the statistics of the traffic. This contrasts with an algorithm based on a parametric model that attempts to estimate the parameters from the traffic. We choose the former approach because realistic models may be complex and slow to fit. We make an analogy with direct vs. indirect adaptive control. In indirect adaptive control, first a parametric model is fitted to the observed traffic. The optimal policy for the estimated parameters is then used. In the direct approach, the quantity to be optimized is measured. The control actions are selected to optimize future values of this quantity.

Thus, by monitoring the traffic through its buffers, each switch constantly estimates its spare capacity to accept new calls. The algorithm then routes a call by using these estimates.

The paper is organized as follows. Section 2 describes the call acceptance and routing algorithm in some detail. In section 3 we assume the input sources are of the same type, and prove a predictive scaling property of the probability of buffer overflow in a busy cycle. We relate this quantity to the fraction of cells lost. In section 4, we describe two methods of variance reduction. Our simulations are described in section 5. Finally, in section 6, we discuss a way to handle multiple types of calls with this method and draw conclusions.

2 Monitor to Infer Network Overflow Statistics (MINOS)

As explained above, we want to estimate the loss probability in the switch buffers. Consider a given buffer of size B cells, with $N > 0$ virtual circuits sharing the buffer, and served by a fiber with transmission rate c cells/s. Let $F(N, B, c)$ be the fraction of cells lost due to buffer overflows. We want to estimate the fraction $F(N(1+\epsilon), B, c)$ of cells lost when a fraction ϵ more calls are added. From this, we can decide whether or not to accept a fraction ϵ more calls. In section 3, we show the probability of buffer overflow in a busy cycle, Φ, has the following property: for large B,

$$\Phi(N(1+\epsilon), B, c) \approx \Phi(N, B, \frac{c}{1+\epsilon}).$$

We express F in terms of Φ conclude that F has this property as well. Thus, we can estimate F when a fraction ϵ more calls are added by estimating F with the current number of calls, N, and the service rate reduced by the same fraction. To estimate $F(N(1+\epsilon), B, c)$, a device is added to the switch that calculates the buffer occupancy, $X(t)$, when the service rate is $\frac{c}{1+\epsilon}$. Thus, when a cell arrives at the buffer, $X(t)$ is incremented by one. Also, $X(t)$ is decremented by one every $\frac{1+\epsilon}{c}$ seconds when $X(t) > 0$. This function could be realized by a chip. The problem now is to estimate $F(N, B, \frac{c}{1+\epsilon})$–which is very small–by monitoring the buffer.

To improve the estimator, the device will estimate the losses for smaller buffers so as to increase the frequency of buffer overflows, and therefore speed-up the collection of "important" samples. There is a tradeoff in choosing the size of the small buffers. If these small buffers are still too large, our estimates will be too slow. However, if these small buffers are too small, the original system system is over-distorted and we have a large error when we extrapolate back to B. Let B/k be the size of a small buffer for some $k > 1$.

The small buffer estimate, $F(N, \frac{B}{k}, \frac{c}{1+\epsilon})$, is related to $F(N, B, \frac{c}{1+\epsilon})$, in the following way. We can show (appendix A) that F has the following form:

$$F(N, B, \frac{c}{1+\epsilon}) = \exp(-BI(N, \frac{c}{1+\epsilon}) + o(B)).$$

Taking the $e^{o(B)}$ term to be AB^{ξ}, we obtain estimates of

$$F(N, \frac{B}{k}, \frac{c}{1+\epsilon}) = A\left(\frac{B}{k}\right)^{\xi} \exp\left(-\frac{B}{k} I(N, \frac{c}{1+\epsilon})\right).$$

Because we have three unknowns (A, ξ, I), we will carry out this estimate for for three values of k: $k_0 > k_1 > k_2 > 1$. These three equations can be solved for A, ξ and $I(N, \frac{c}{1+\epsilon})$. We can then plug in A, ξ and $I(N, \frac{c}{1+\epsilon})$ into the expression for $F(N, B, \frac{c}{1+\epsilon})$, and thus compute the desired quantity $F(N(1+\epsilon), B, c) \approx F(N, B, \frac{c}{1+\epsilon})$.

To summarize the above, the estimation algorithm in the device keeps track of three "small buffer" occupancy processes with buffers of size B/k_i, $i = 1, 2, 3$, and service rate $\frac{c}{1+\epsilon}$. Note that these computations can be done in parallel with the normal operation of the switch so that the estimates of $F(N, \frac{B}{k_i}, \frac{c}{1+\epsilon})$ are constantly available to the routing algorithm. In section 4 we also describe a way to further reduce the variance of an estimate of Φ.

Let us now describe how the routing algorithm can use the above estimates. Denote by $F_n = F(N_n, B_n, c_n)$ the current fraction of cells lost at buffer n, for all buffers n in the network. Assuming a first-come-first-serve queuing discipline in each buffer, F_n is the fraction of cells lost at buffer n by *each* call that uses buffer n. If call i uses buffers $1, 2, ..., m$, the fraction of cells lost by that call is $1 - \prod_{n=1}^{m}(1 - F_n) \approx \sum_{n=1}^{m} F_n$.

Now say we are trying to route a new call. Using the above method, buffer n estimates $F'_n = F(N_n(1+\epsilon), B_n, c_n)$. We attempt to find a path for the new call that satisfies

$$G^{\text{new}} \geq \sum_{n \in \text{path}} F'_n$$

where G^{new} is the fraction of lost cells acceptable to the new call. Moreover, the router must ensure that, by choosing a particular path for the new call, the above constraint is not violated for any existing, previously routed call i (with guarantee G^i) which uses all or part of that path. If no path is found that satisfies these constraints, the new call is refused.

The routing policy just described is myopic. More sophisticated strategies should be investigated. For instance, given statistics about the generation of new calls, one can formulate a dynamic programming problem which is solved by the optimal policy.

3 A Predictive Scaling Property

Markov fluids are commonly used to model bursty ATM traffic [3]. In this paper we analyze the two-rate Markov fluid model for simplicity: each call generates cells at two different deterministic rates. A call will fire cells at one rate for a random exponentially distributed amount of time after which it will begin to fire at the other rate. The results given in this section can be extended to m-rate Markov fluid models, with $m > 2$, as well as batch Poisson models. For the details of these extensions we refer to a future paper [4].

In this section we consider calls of the same type to derive an expression for the probability of buffer overflow in a busy cycle. We want to find expressions for probabilities of rare events which leads us to employ the theory of large deviations below. We consider a given buffer of size B cells, with N two-rate Markov fluid calls sharing the buffer, and served by a fiber with transmission rate c cells/s.

3.1 The Model

Define the process Y with

$$\frac{dY}{dt}(t) = \sum_{k=1}^{N} r_k(t) - c$$

where the r_k are i.i.d. Markov chains with rate matrix

$$Q_r = \begin{bmatrix} -q_0 & q_0 \\ q_1 & -q_1 \end{bmatrix}$$

and stationary probabilities $\pi_i = q_{1-i}/(q_0+q_1)$, $i = 0, 1$. The state space of the r_k's is $\Lambda = \{\Lambda_0, \Lambda_1\}$ where $\Lambda_0 - \frac{c}{N} < 0$ and $\Lambda_1 - \frac{c}{N} > 0$. Thus dY/dt is a Markov chain in the set $\{z_i \equiv i(\Lambda_1 - \frac{c}{N}) + (N-i)(\Lambda_0 - \frac{c}{N}) \ : \ i \in \{0, 1, ..., N\}\}$, and its rate matrix has components $q(i, i+1) = (N-i)q_0$, for $0 \leq i < N$, and $q(i, i-1) = iq_1$, for $0 < i \leq N$. Therefore Y has piecewise linear trajectories with Markov slope.

The traffic intensity of the buffer is $\rho\left(\frac{N}{c}\right) \equiv \frac{N}{c}(\Lambda_0\pi_0 + \Lambda_1\pi_1) < 1$. We are focusing on the case $\rho \approx 1$: the buffer is experiencing *heavy traffic.*

In appendix A, we show that, in heavy traffic,

$$\Phi(N, B, c) \ \leq \ \exp\left(-B\frac{(q_0+q_1)\frac{c}{N}(1-\rho(\frac{N}{c}))}{(\Lambda_1 - \Lambda_0)(\Lambda_1 - \frac{c}{N})} + o(B)\right).$$

This result can be generalized to a case where the state space of the r_k is $\{\Lambda_0, ..., \Lambda_{m-1}\}$ with $m > 2$ [4].

Asymptotics of Markov fluids in queuing systems are also discussed in [5]. From their result, we obtain a lower bound for Φ (appendix A):

$$\Phi(N, B, c) \ \geq \ \exp\left(-B\frac{(q_0+q_1)\frac{c}{N}(1-\rho(\frac{N}{c}))}{(\frac{c}{N} - \Lambda_0)(\Lambda_1 - \frac{c}{N})} + o(B)\right).$$

From these two inequalities we can conclude that, in heavy traffic,

$$\Phi(N(1+\epsilon), B, c) \approx \Phi(N, B, \frac{c}{1+\epsilon}).$$

For batch Poisson sources, using the results in [6], [7], and [8], we can show $\Phi(N,B,c) = \exp(-BI(N,c) + o(B))$ where $I(N,c) = I(1, \frac{c}{N})$ [4].

The fraction of cells lost due to buffer overflows, $F(N,B,c)$, can be expressed in terms of $\Phi(N,B,c)$, the average number of cells lost in an overflowing cycle, $D(N,B,c)$, and the average number of cells arriving in a cycle, $C(N,B,c)$:

$$F(N,B,c) = \frac{D(N,B,c)\Phi(N,B,c)}{C(N,B,c)}.$$

Note that the numerator, $D\Phi$, is the average number of cells lost in a cycle. We assume that the rate processes are stationary, and that cycles in which a single cell arrives to an empty buffer and departs before the next cell arrives are counted.

In finding an expression for C, we ignore the buffer size B because the number of cells arriving in an overflowing cycle is of the order B and the probability of an overflowing cycle is of the order $\exp(-BI)$. This gives us a contribution to C of approximately $B\exp(-BI)$ which is negligible for large B. In finding an expression for D we argue that the value of dY/dt when an overflow occurs is the maximum $N\Lambda_1 - c$. Also, we assume a negligible number of cells will be lost after the rate returns to zero. Under these assumptions, D will have a negligible dependence on B as well (for large B). Thus,

$C/D = o(B)$. One can heuristically show that $C(N,c) \approx C(1,c/N)$ and $D(N,c) \approx D(1,c/N)$ [4]. We conclude that the fraction of cells lost, F, has the same (N,c) scaling properties as the probability of buffer overflow, Φ.

4 Variance Reduction

4.1 Small Buffers

Recall that we proposed using small buffers to increase the frequency of "important" samples (buffer overflows) in order to reduce the variance of the estimate of F. The small buffers have sizes B/k_i, $i = 1,2,3$, with $k_0 > k_1 > k_2 > 1$. We will now explain a way to express the quantity we want to estimate

$$F \equiv F(N(1+\epsilon), B, c) \approx F(N, B, \frac{c}{1+\epsilon}) \approx AB^{-\xi}e^{-BI}$$

in terms of the small buffer estimates

$$F_i \equiv F(N, \frac{B}{k_i}, \frac{c}{1+\epsilon}) \approx A\left(\frac{B}{k_i}\right)^{-\xi} \exp(-\frac{B}{k_i}I)$$

$i = 0,1,2$. Let $a_{i,j} = \frac{1}{k_i} - \frac{1}{k_j}$, $e_0 = \frac{-k_0(k_2-1)}{k_0-k_2}$, $e_2 = \frac{k_2(k_0-1)}{k_0-k_2}$, and $\gamma = \log(k_0^{e_0}k_2^{e_2})/\log(k_0^{a_{1,2}}k_1^{a_{2,0}}k_2^{a_{0,1}})$. Solving for the three unknowns, A, ξ, and I, in terms of the F_i, and substituting into the expression for F we get,

$$\log F = l_0 \log F_0 + l_1 \log F_1 + l_2 \log F_2$$

where $l_0 = e_0 + \gamma a_{2,1}$, $l_1 = \gamma a_{0,2}$, and $l_2 = e_2 + \gamma a_{1,0}$.

4.2 Analysis of Variance Reduction using Small Buffers

For simplicity we take $\xi = 0$ and consider the variance reduction achieved by using two small buffers (instead of three) to estimate Φ (instead of F). Thus, we estimate

$$\Phi \equiv \Phi(N(1+\epsilon), B, c) \approx \Phi(N, B, \tfrac{c}{1+\epsilon}) \approx Ae^{-BI}$$

from the small buffer estimates

$$\Phi_i \equiv \Phi(N, \frac{B}{k_i}, \frac{c}{1+\epsilon}) \approx A \exp(-\frac{B}{k_i} I)$$

$i = 0, 2$. Substituting for A and I we get $\Phi = \Phi_0^{e_0} \Phi_2^{e_2}$.

Assume the time to estimate, n, in *busy cycles* is fixed and the same for both small buffers. Let σ_i be the standard deviation of the estimate of Φ_i so that $\sigma_i = \sqrt{\Phi_i(1-\Phi_i)/n}$ $\approx \sqrt{\Phi_i/n}$, $i = 0, 2$. Thus, the relative error of the estimate of Φ, σ/Φ, satisfies

$$\frac{\sigma}{\Phi} \leq -\frac{\sigma_0}{\Phi_0} e_0 + \frac{\sigma_2}{\Phi_2} e_2 =: f(k_0, k_2)$$

for σ_i sufficiently small. Note that $e_0 < 0$ and f is an upper bound for σ/Φ because we have ignored the fact that the Φ_i are positively correlated.

Minimizing f over (k_0, k_2) we get that the optimal k_0 is very large and the optimal k_2 minimizes $g(k) \equiv (k-1)\sqrt{1-A} + k\sqrt{\exp(BI/k) - A}$. Let n_k and n_{TA} be the number of cycles required to achieve $\epsilon \times 100\%$ relative error with 95% confidence [9] using two small buffers and direct time averaging respectively. A simple computation yields: $n_k/n_{TA} = g^2(k_2)\exp(-BI)$. In our simulations, we found $A << 1$ (which implies the optimal $k_2 \approx 0.4BI$), and $BI \approx 8$, so that $n_k/n_{TA} \approx 1/17$. The speed up factor is actually larger than 17 because $\sigma/\Phi < f$; using sample standard deviations, we found a speed up was about 100. Unfortunately, fixing $\xi = 0$ results in estimates of Φ that are consistently one order of magnitude too small. These calculations give us a rule of thumb for choosing the k_i, $i = 0, 1, 2$: choose k_0 large and k_2 small (the tradeoff discussed in section 2).

4.3 Variance Reduction using the Kullback-Leibler Distance

We now describe a method to estimate the probability of buffer overflow in a cycle. Instead of using three small buffers etc., we monitor the *peak buffer occupancy* in every cycle (call it Z_i for the i^{th} cycle). Let n be a given number of cycles, and $B^* = B/k_0$. For integers $b \geq \bar{B} - 1$, define the empirical tail distribution of Z_i:

$$p(b) = \frac{1}{n} \times \begin{cases} \sum_{i=1}^{n} 1\{Z_i < B^*\} & \text{if } b = B^* - 1 \\ \sum_{i=1}^{n} 1\{Z_i = b\} & \text{if } b \geq B^*. \end{cases}$$

Also define

$$\phi(A,I,b) = \begin{cases} 1 - A\exp(-B^*I) & \text{if } b = B^* - 1 \\ A\exp(-bI) - A\exp(-(b+1)I) & \text{if } b \geq B^* \end{cases}$$

where we have taken $\xi = 0$. The Kullback-Leibler distance [10] between ϕ and p is

$$K(A,I) = \sum_{b=B^*-1}^{\infty} p(b)\log\left(\frac{p(b)}{\phi(A,I,b)}\right).$$

The values of A and I that minimize K are given by

$$\begin{aligned} I &= \log\left(1 + \frac{1-p(B^*-1)}{\sum_{b=B^*}^{\infty} bp(b) - B^*(1-p(B^*-1))}\right) \quad \text{and} \\ A &= (1-p(B^*-1))\exp(B^*I). \end{aligned}$$

These expressions for A and I can be easily updated at the end of every cycle. Taking $\xi \neq 0$ so that K is a function of three parameters, we found no simple closed form solution to $\partial K/\partial\xi = \partial K/\partial I = \partial K/\partial A = 0$.

5 Simulations

We conducted simulations using the two rate Markov fluid model described above for the sources to the buffer. The following values were chosen for the parameters: $B = 1800$ cells, $\Lambda_0 = 0$, $\Lambda_1 = 2500$ cells/s, $c = 15000$ cells/s ($\approx$ 6 Mbps) and

$$Q_r = \begin{bmatrix} -10 & 10 \\ 20 & -20 \end{bmatrix}.$$

The number of sources, N, was varied. We took $\epsilon = 1/N$ and $N+1 \in \{13, 14, .., 17\}$. The above parameters were chosen so that we would be simulating on-off sources with burst rates in the Mbps range (1 cell = 53 bytes), and the fraction of overflowing cycles among non-"trivial" busy cycles was small: a trivial busy cycle being one in which a cell arrives to an empty buffer and leaves before the next cell arrives ($\dot{Y} < 0$ and $\sum r_i > 0$). The values of N were chosen in the above range to simulate heavy traffic conditions: $\rho(\frac{N}{c}) = \frac{N}{c}\pi_1\Lambda_1 = \frac{N}{18}$ where $\pi = (\pi_0, \pi_1) = (2/3, 1/3)$.

The first simulation checks the $N - c$ scaling property of section 3 for a finite B when a large amount of time is available to accurately estimate the F_i, $i = 1, 2, 3$. For $N+1 \in 13, ..17$, we measured $F(N+1, B, c)$ using direct sample averaging (we stopped our simulation when the 95% confidence interval estimate of $F(N+1, B, c)$ was less than 0.3 [9]). Using three small buffers, we estimated $F(N+1, B, c)$ from estimates of $F_i = F(N, \frac{B}{k_i}, \frac{c}{1+N^{-1}})$ using the formulas above with $\epsilon = 1/N$ (we stopped our simulation when the 95% confidence interval estimate of $F(N, \frac{B}{k_i}, \frac{c}{1+N^{-1}})$ was less than 10%). We used two different sets of three k_i: (20,15,9) and (15,12,9). Figure 1 contains plots of the logarithm of the fraction of cells lost vs. the number of buffer sources, $N+1$. There are three curves: the measured value of $F(N+1, B, c)$ and the estimate of this value using three small buffers for the two sets of k. The estimates are both well within an order of magnitude of the measured value.

In the second simulation, we fixed $N+1 = 13$ so that $F(N+1, B, c) \approx 5 \times 10^{-6}$ and $\Phi(N+1, B, c) \approx 3 \times 10^{-9}$. We ran the simulation for $n = 10^7$ busy cycles of the "actual" buffer process (size B). Since the probability of even seeing one overflow in this amount of time is 30%, an estimate of F from direct sample averages would probably be zero. Figure 2 shows the performance of the estimate of F using three small buffers for two sets of k_i. In the six trials (abscissa of the graph) of this simulation, no overflows were observed in the actual buffer. The sample standard deviation was less than the estimate in every trial.

These two simulations were repeated using four-rate Markov fluids with the following parameters chosen to have the traffic characteristics described above: $B = 1800$ cells, $\Lambda_i \in \{0, 2000, 3000, 4000\}$ cells/s, $c = 15000$ cells/s,

$$Q_r = \begin{bmatrix} -10 & 10 & 0 & 0 \\ 20 & -30 & 10 & 0 \\ 0 & 30 & -40 & 10 \\ 0 & 0 & -40 & -40 \end{bmatrix},$$

$N+1 \in \{9, 10, 11, 12\}$, and two different sets of three k_i: (30,15,5) and (20,15,5). For the short time simulation, we fixed $N+1 = 9$ and the simulation time $n = 5 \times 10^7$ cycles. Since the measured value of $\Phi(N+1, B, c) = 6 \times 10^{-10}$, we get that the probability of seeing an overflow in one n cycle trial is 30% as above. The results are shown in figures 3 and 4.

We repeated the simulation with two-rate Markov fluid sources to estimate Φ using the method based on minimizing the Kullback-Leibler distance. We fixed $N+1 = 15$ so that $\Phi(N+1, B, c) \approx \times 10^{-6}$ and ran the simulation for $n = 10^5$ busy cycles. In the six trials of this simulation plotted in figure 5, one overflow was observed in the actual buffer. The estimates are optimistic but are within an order of magnitude of the measured value, 10^{-6}. Also, the sample standard deviation was less than the estimate in every trial. Given this amount of time, the three small buffers estimate was very noisy.

When estimating Φ, trivial busy cycles were counted. When these cycles were not counted, the simulations yielded good results anyway; the values of Φ were, of course, much higher in this case. Note that, if we assume the buffer is an M/M/1 queue with traffic intensity $\rho = 15\pi_1\Lambda_1/c = 15/18$ and approximate Φ with the stationary probability that this M/M/1 queue exceeds B, we get $\Phi(N+1, B, c) \approx \rho^B \approx 10^{-140}$–an extremely poor estimate.

6 Discussion and Conclusions

The above method can be used to handle multiple types of calls sharing a buffer. Say there are six voice calls (same type) and two video calls currently using the buffer, and we wish to estimate the effect of adding a video call. Define a new type of call that is the sum of three voice calls and one video call. Therefore there are two calls of that type currently using the buffer. Instead of estimating the fraction of cells lost when another video call is added, we estimate F when another call of that type is added. This, of course, may be a very conservative estimate of the affect of another video call on the buffer.

In order to estimate the number of Mips required by one small buffer to estimate

F_i, we let the peak arrival rate into the buffer be $p \times c$ cells/s. The worst case occurs during cell loss when we have to handle the buffer occupancy and perform a compare every $(pc + c)^{-1}$ seconds, and update the cells lost and cells arrived counters every $(pc)^{-1}$ seconds. Thus we require $2(pc + c) + 2pc = (4p + 2)c$ Mips. For $c = 3.5 \times 10^5$ cells/s (150 Mbps) and $p = 5$ we get 7.7 Mips required by one small buffer.

In summary, we have described a call set-up algorithm which monitors the traffic in a switch buffer and makes quick and direct estimates of the effect of routing more calls through that buffer on the fraction of cells lost in that buffer. The method is robust: it has been shown, in principle, to work for batch Poisson and Markov fluid sources. Finally, simulations were conducted which demonstrated the predictive property of the algorithm as well as the significant variance reduction. with finite buffer size.

Since many idealizations were made above, experiments on actual networks are clearly required to determine how useful this method will be.

References

[1] J. Hui, "Network, transport, and switching integration for broadband communications," *IEEE Network*, vol. March, pp. 40–51, 1988.

[2] A. S. Tanenbaum, *Computer Networks, 2nd Ed.* Englewood Cliffs, NJ: Prentice Hall, 1988.

[3] M.Decina, T.Toniatti, P.Vaccari, and L.Verri, "Bandwidth assignment and virtual call blocking in ATM networks," *IEEE Infocom'90*, vol. Vol.3, pp. 881–888, 1990.

[4] G. Kesidis and J. Walrand, "Large deviations of traffic in high speed digital networks with a view towards network control," *in preparation.*

[5] D. Anick, D. Mitra, and M. M. Sondhi, "Stochastic theory of a data-handling system with multiple sources," *Bell Sys. Tech. J.*, vol. 61 No.8, pp. 1871–1894, 1982.

[6] J. Lynch and J. Sethuraman, "Large deviations for processes with independent increments," *Annals of Probability*, vol. 15 No. 2, pp. 610–627, 1987.

[7] M. Cottrell, J.-C. Fort, and G. Malgouyres, "Large deviations and rare events in the study of stochastic algorithms," *IEEE Trans. on Auto. Control*, vol. 28 No. 9, pp. 907–920, 1983.

[8] S. Parekh and J. Walrand, "A quick simulation method for excessive backlogs in networks of queues," *IEEE Trans. on Auto. Control*, vol. 34 No. 1, pp. 54–66, 1989.

[9] S. M. Ross, *A Course in Simulation.* New York, NY: Macmillan, 1990.

[10] J. Bucklew, *Large Deviation Techniques in Decision, Simulation and Estimation.* New York, NY: John Wiley and Sons, Inc., 1990.

[11] M. Freidlin and A. Wentsell, *Random Perturbations of Dynamical Systems.* New York, NY: Springer-Verlag, 1984.

[12] R. Dupuis, R. Ellis, and A. Weiss, "Large deviations for Markov processes with discontinuous statistics 1: General upper bounds," *to appear.*

7 Appendix A: Scaling Property of Φ

We can express this rate matrix of dY/dt in terms of z_i, instead of i, to get,

$$Q(z_i, z) = \begin{cases} \frac{q_0}{\Delta}(N\Lambda_1 - c - z_i) =: \lambda(z_i) & \text{if} \quad z = z_i + \Delta = z_{i+1} \\ \frac{q_1}{\Delta}(-N\Lambda_0 + c + z_i) =: \mu(z_i) & \text{if} \quad z = z_i - \Delta = z_{i-1} \\ 0 & \text{else} \end{cases}$$

where $\Delta = \Lambda_1 - \Lambda_0$.

Define $\lambda(z) = q_0(N\Lambda_1 - c - z)/\Delta$ for all real z in the interval $[z_0, z_N]$, $\lambda(z) = Nq_0$ for $z < z_0$, and $\lambda(z) = 0$ for $z > z_N$. Similarly extend the definition of μ to a continuous function on the whole real line. We will use the Markov process $R(t)$, with generator

$$(\mathcal{L}f)(z) = \lambda(z)(f(z+\Delta) - f(z)) + \mu(z)(f(z-\Delta) - f(z)),$$

to model the dY/dt process. Note that R and dY/dt will have the same distribution if the initial values are the same (in the set $\{z_i\}$).

We assume that the buffer size, B, is large so that, even in heavy traffic, loss probabilities are still small. For $t \in [0,T]$, define the scaled process $R^B(t) = B^{-1}R(Bt)$. In order to get and expression for the probability of buffer overflow in a busy cycle, we will use a large deviations result for $\{R^B(t) : t \in [0,T]\}$ as $B \to \infty$. To this end define the cumulant of R ([11], p.145):

$$H(z,\alpha) = \lambda(z)(e^{\alpha\Delta} - 1) + \mu(z)(e^{-\alpha\Delta} - 1).$$

Let L be the Legendre transform of H with respect to the second variable:

$$L(z,\zeta) = \sup_{\alpha\in\mathbf{R}}\{\alpha\zeta - H(z,\alpha)\}.$$

In the following, we will indicate the dependence of L and H on N and c: $L(N,c,z,\zeta)$ and $H(N,c,z,\alpha)$ respectively.

Define the set of trajectories of R^B that result in an overflow at time BT in the first busy cycle:

$$\Gamma^*(T) \;=\; \{z \in C^1[0,T] \;:\; \int_0^T z(t)dt = B^{-1} > \int_0^u z(t)dt > -B^{-2} \;\; \forall u \in (0,T)\}.$$

By the large deviations result in [12] we get,

$$\Phi(N,B,c) \;\leq\; \exp(-BI(N,c) + o(B))$$

where

$$I(N,c) \;=\; \inf_{T > \frac{1}{N\Lambda_1 - c}} \; \inf_{z \in \Gamma^*(T)} \int_0^T L(N,c,z(t),\dot{z}(t))dt.$$

Note that $\frac{1}{N\Lambda_1 - c}$ is the shortest possible time to overflow. Since $L \geq 0$, we can replace $\Gamma^*(T)$ in the expression for $I(N,c)$ by

$$\Gamma(T) \;=\; \{z \in C^1[0,T] \;:\; \int_0^T z(t)dt = B^{-1}, z(0) = 0, z(t) > 0 \;\text{ a.e. } t \in (0,T)\}$$

where a.e. abbreviates "for almost every".

Because L is the supremum of linear functions of $(z,\dot{z})$ (Legendre transform above), L is a convex function of $(z,\dot{z})$. Thus, by Jensen's inequality,

$$\int_0^T L(N,c,z(t),\dot{z}(t))dt \;\geq\; TL(N,c,\frac{1}{T}\int_0^T z(t)dt, \frac{1}{T}\int_0^T \dot{z}(t)dt) = TL(N,c,\frac{1}{BT},\frac{z(T)}{T})$$

where, in the last two equalities, we have used $z \in \Gamma(T)$. Now we assumed that the traffic intensity $\rho(\frac{N}{c}) < 1$, which is equivalent to $\lambda(0) < \mu(0)$. This implies that $\mu(z) > \lambda(z)$

for all $z \geq 0$.

Claim: For all $z \in \Gamma(T)$, $\int_0^T L(N,c,z(t),\dot{z}(t))dt \geq TL(N,c,\frac{1}{BT},0)$.

Proof: $z(T) > 0$ and Jensen's inequality above imply that it suffices to show that $L(N,c,(BT)^{-1},\epsilon) \geq L(N,c,(BT)^{-1},0)$ for all $\epsilon > 0$. Since $\mu((BT)^{-1}) > \lambda((BT)^{-1})$, $H(N,c,(BT)^{-1},-a) > H(N,c,(BT)^{-1},a)$ for all $a > 0$. This implies that

$$a^* \equiv \arg\sup_{a \in \mathbf{R}}\{-H(N,c,(BT)^{-1},a)\} \geq 0.$$

Thus, for all $\epsilon > 0$,

$$\begin{aligned} L(N,c,(BT)^{-1},\epsilon) &\equiv \sup_{a \in \mathbf{R}}\{a\epsilon - H(N,c,(BT)^{-1},a)\} \\ &\geq a^*\epsilon - H(N,c,(BT)^{-1},a^*) \\ &\geq -H(N,c,(BT)^{-1},a^*) \\ &= L(N,c,(BT)^{-1},0) \end{aligned}$$

as desired.

Thus we have found a lower bound for $B \times I(N,c)$:

$$B \times I(N,c) \;\geq\; J(N,B,c) := \inf_{T > \frac{1}{N\Lambda_1 - c}} BTL(N,c,(BT)^{-1},0) = \inf_{S > \frac{B}{N\Lambda_1 - c}} SL(N,c,S^{-1},0).$$

Given the expressions for λ and μ above, it is easy to see that $H(N,c,z,\alpha) = N \cdot H(1,c/N,z/N,\alpha)$. This in turn implies that $L(N,c,z,\dot{z}) = N \cdot L(1,c/N,z/N,\dot{z}/N)$. Therefore,

$$J(N,B,c) = \inf_{NS > \frac{B}{\Lambda_1 - c/N}} NSL(1, \frac{c}{N}, \frac{1}{NS}, 0) = J(1, B, \frac{c}{N})$$

and

$$\Phi(N,B,c) \leq \exp(-J(1,B,\frac{c}{N}) + o(B))$$

which is the desired scaling property for an upper bound of Φ. One can show this (N,c) scaling property for $m > 2$ rate Markov fluid sources as well using a similar approach [4].

We can show directly that

$$J(N,B,c) = B\frac{\left(\sqrt{\mu(0)} - \sqrt{\lambda(0)}\right)^2}{N\Lambda_1 - c} + o(B).$$

Under heavy traffic ($\rho \approx 1 \Leftrightarrow \lambda(0) \approx \mu(0)$) we can approximate

$$J(N,B,c) \approx B\frac{\mu(0) - \lambda(0)}{N\Lambda_1 - c} + o(B) = B\frac{(q_0 + q_1)\frac{c}{N}(1 - \rho(\frac{N}{c}))}{(\Lambda_1 - \Lambda_0)(\Lambda_1 - \frac{c}{N})} + o(B).$$

Thus, under heavy traffic,

$$\Phi(N,B,c) \leq \exp\left(-B\frac{(q_0 + q_1)\frac{c}{N}(1 - \rho(\frac{N}{c}))}{(\Lambda_1 - \Lambda_0)(\Lambda_1 - \frac{c}{N})} + o(B)\right).$$

Having an upper bound for Φ we now show a lower bound. Define the buffer process, X, with $dX/dt(t) = \sum_{k=1}^{N} r_k(t) - c1\{X(t) > 0\}$. Anick et al [5] show that the stationary probability of $\{X(t) > B\}$ is

$$\exp\left(-B\frac{(q_0 + q_1)\frac{c}{N}(1 - \rho(\frac{N}{c}))}{(\frac{c}{N} - \Lambda_0)(\Lambda_1 - \frac{c}{N})} + o(B)\right).$$

Define the "actual" buffer process, β, defined with $d\beta/dt(t) = \sum_{k=1}^{N} r_k(t)1\{\beta(t) \leq B\} - c1\{\beta(t) > 0\}$. We take $X(0) = Y(0) = \beta(0)$. Consider a *typical* busy cycle of the buffer, starting at the time zero, followed by an idle cycle. That is, define $S > T > 0$ such that $\beta(0-) = 0, \beta(0) = 1, \beta(t) > 0 \ \forall t \in (0, T)$, and $\beta(t) = 0 \ \forall t \in [T, S)$. Thus

$$\begin{aligned}
\exp\left(-B\frac{(q_0+q_1)\frac{c}{N}(1-\rho(\frac{N}{c}))}{(\frac{c}{N}-\Lambda_0)(\Lambda_1-\frac{c}{N})} + o(B)\right) &= \mathrm{P}\{X(t) > B\} \ \text{ typical } t \in [0, S) \\
&\leq \mathrm{P}\{X(t) > B\} \ \text{ typical } t \in [0, T] \\
&= \mathrm{P}\{Y(t) > B\} \ \text{ typical } t \in [0, T] \\
&\leq \mathrm{P}\{\max_{0 \leq t < T} Y(t) > B\} = \Phi.
\end{aligned}$$

Note that X, Y and β all agree up to the end of the busy cycle (T) or an overflow, whichever comes first. X and Y agree up to the end of the busy cycle.

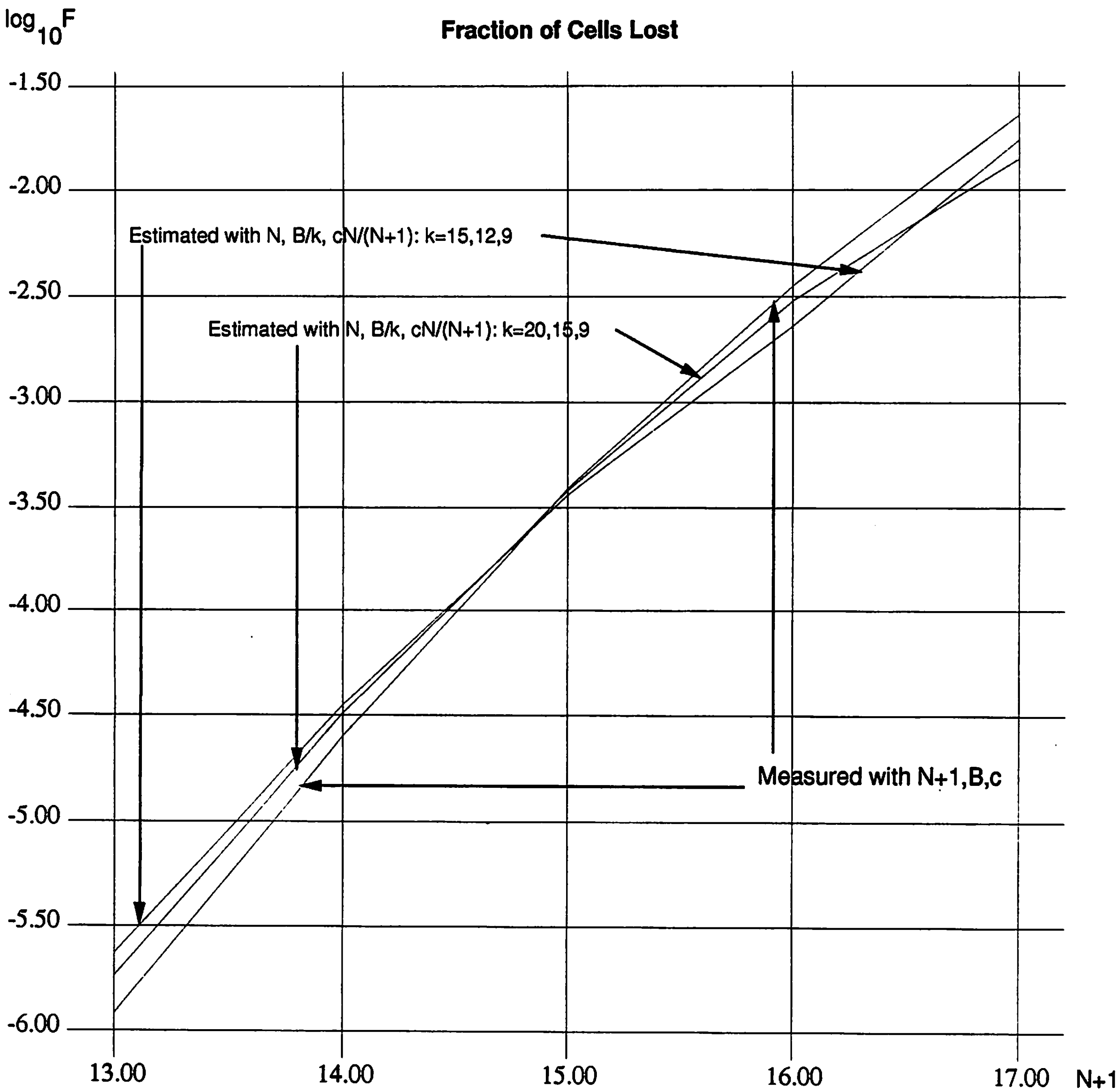

Figure 1: Large Time, 2 Rate Sources

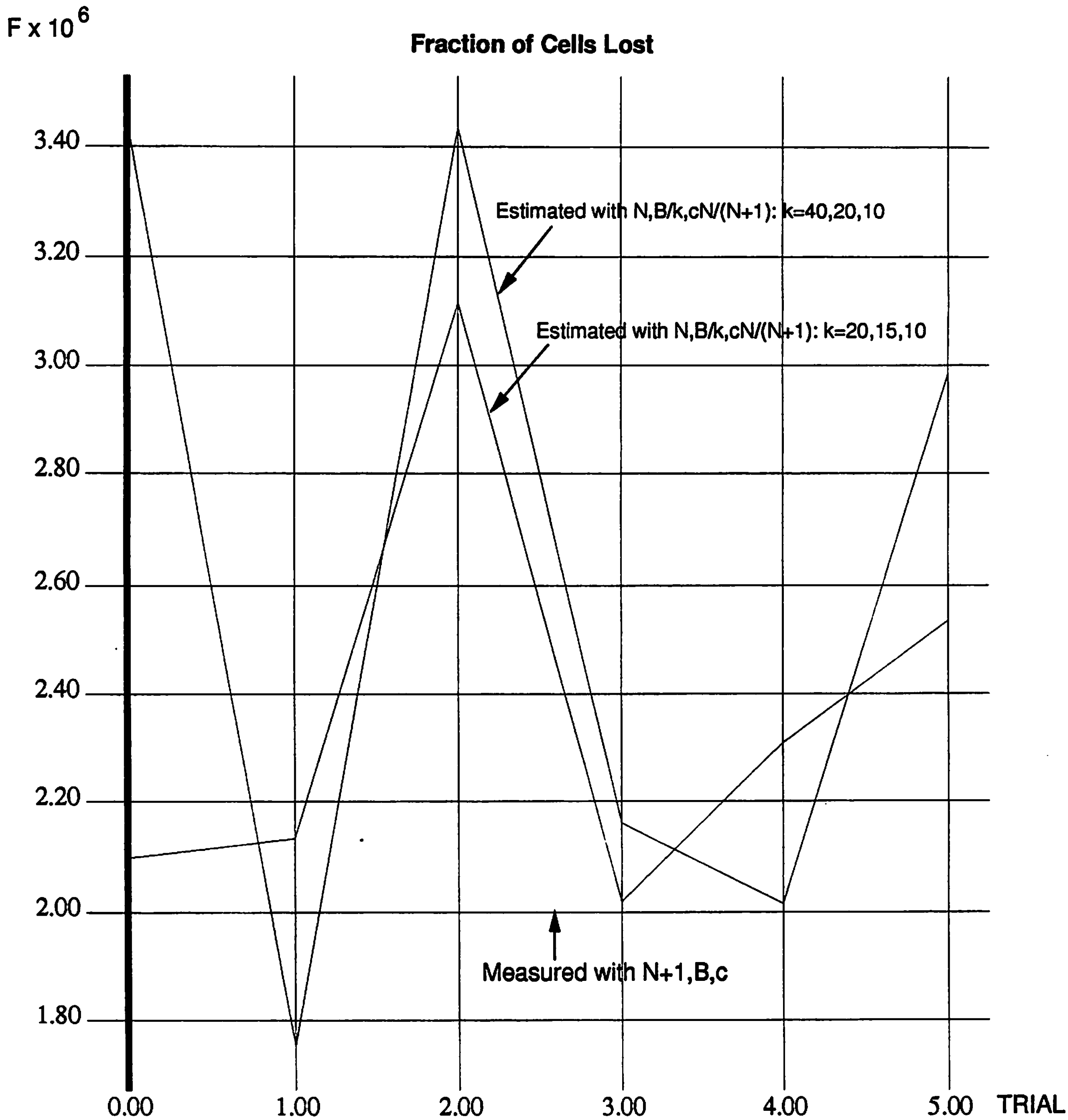

Figure 2: Small Time, 2 Rate Sources

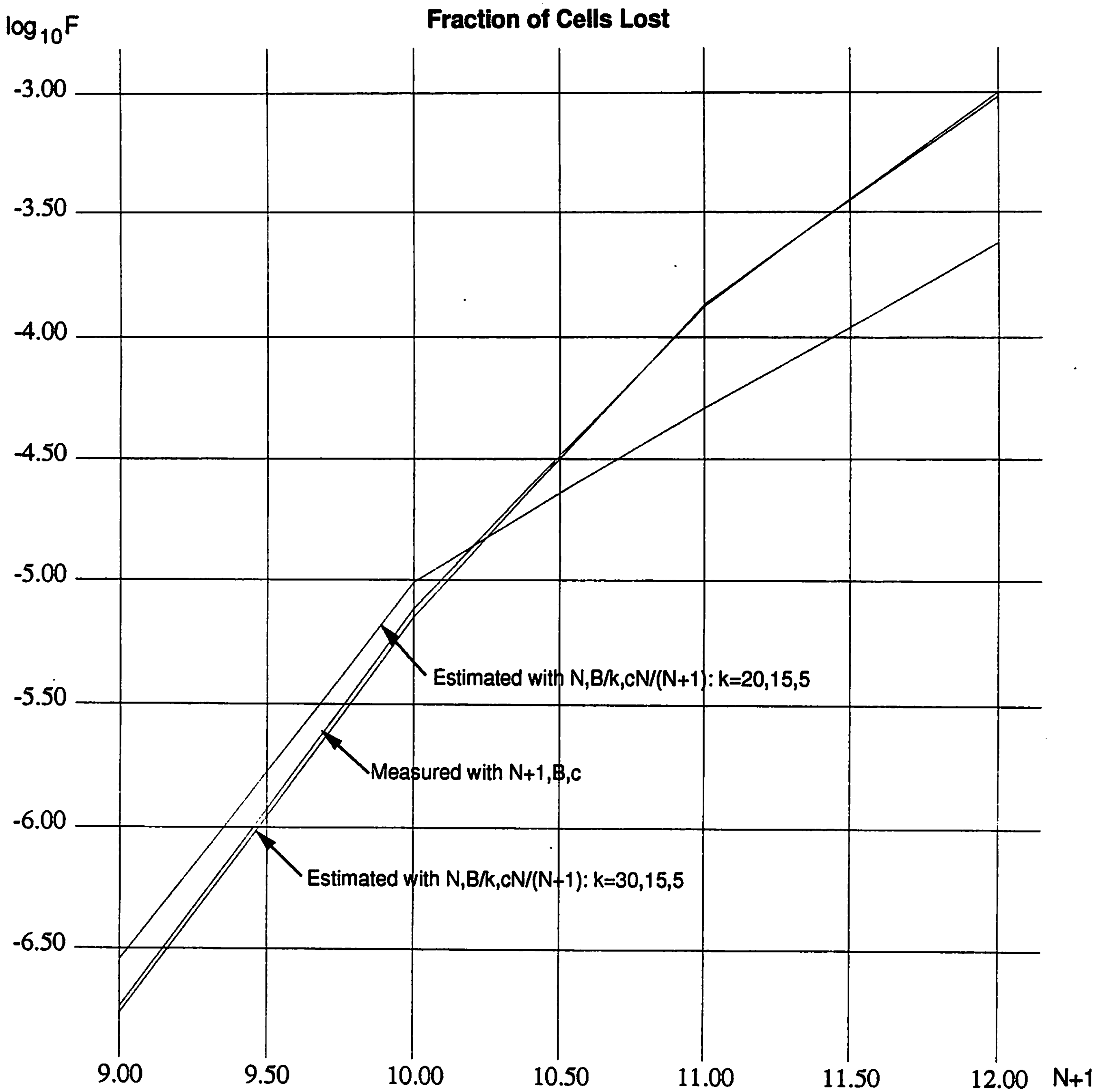

Figure 3: Large Time, 4 Rate Sources

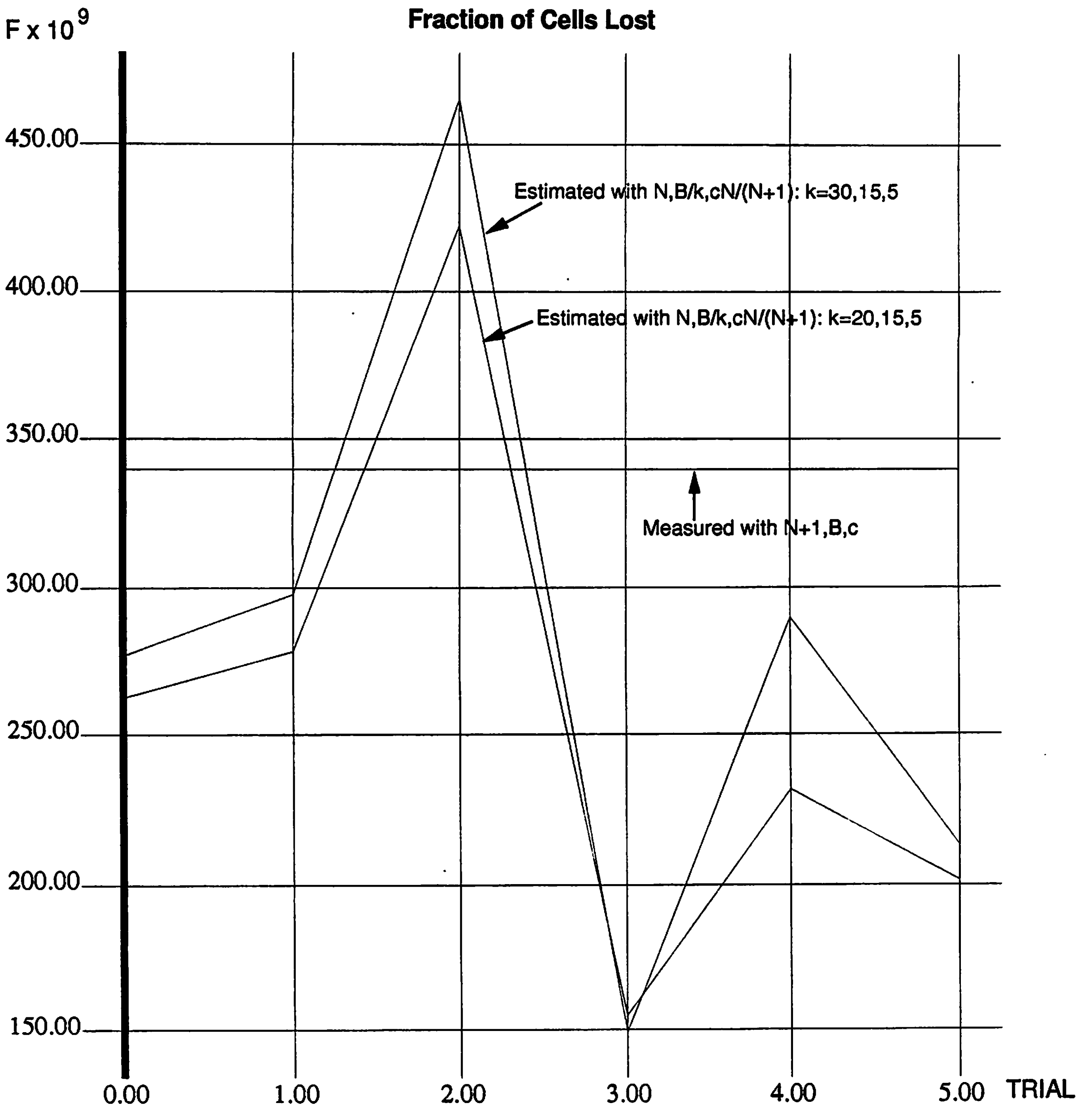

Figure 4: Small Time, 4 Rate Sources

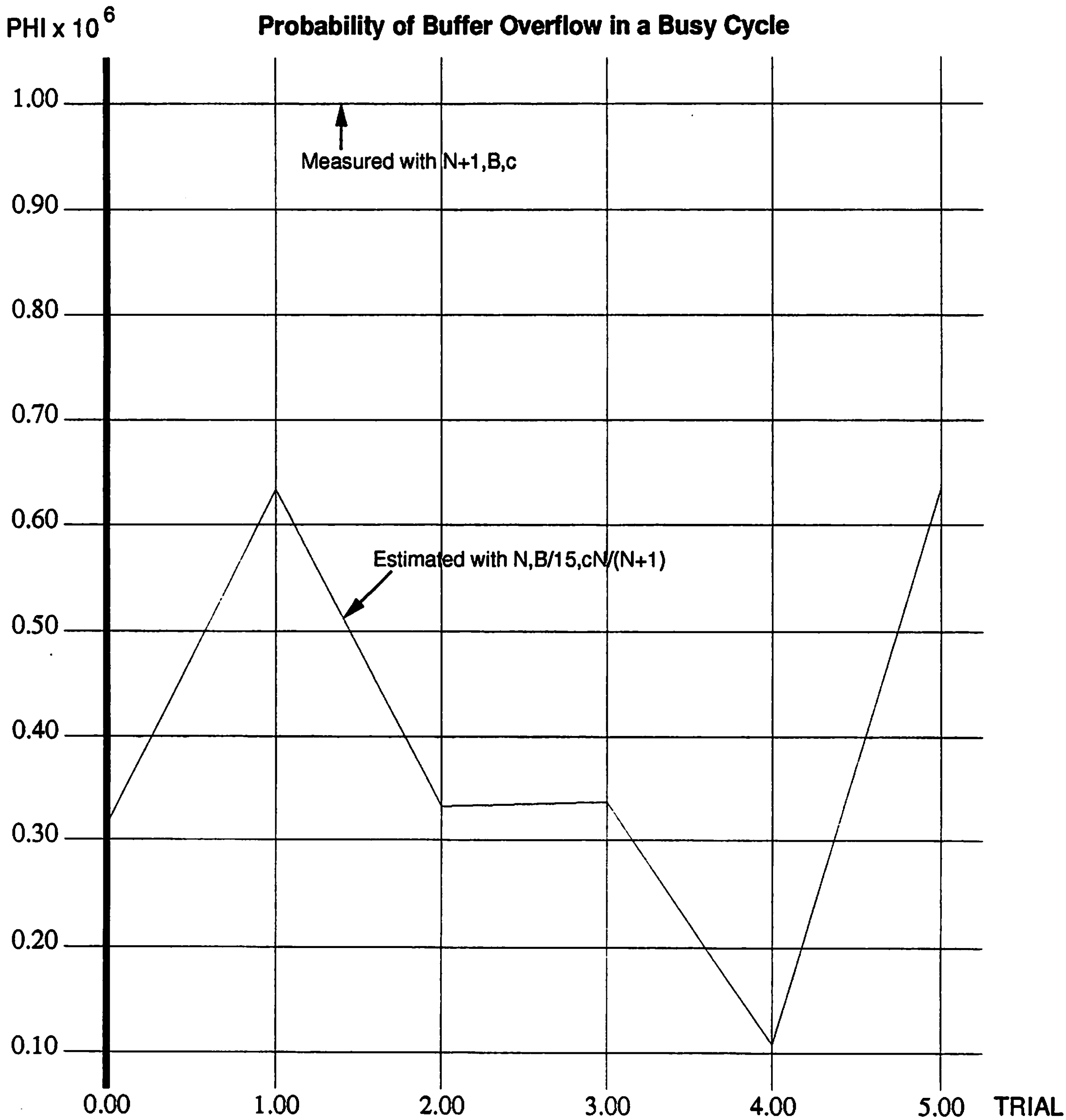

Figure 5: Small Time, 2 Rate Sources