Estimating Small Cell-Loss Ratios in ATM Switches via Importance Sampling

PIERRE L'ECUYER
Université de Montréal
and
YANICK CHAMPOUX
Alcatel

The cell-loss ratio at a given node in an ATM switch, defined as the steady-state fraction of packets of information that are lost at that node due to buffer overflow, is typically a very small quantity which is hard to estimate by simulation. Cell losses are rare events and importance sampling is sometimes the appropriate tool in this situation. However, finding the right change of measure is generally hard. In this paper, importance sampling is applied to estimate the cell-loss ratio in an ATM switch modeled as a queueing network fed by several sources emitting cells according to a Markov-modulated ON/OFF process, and where all the cells from the same source have the same destination. The change of measure is obtained via an adaptation of an heuristic proposed by Chang et al. [1994] for intree networks. The numerical experiments confirm important efficiency improvements even for large non-intree networks and a large number of sources. Experiments with different variants of the importance sampling methodology are also reported, and a number of practical issues are illustrated and discussed.

Categories and Subject Descriptors: G.3 [**Probability and Statistics**]

General Terms: Algorithms

Additional Key Words and Phrases: Importance sampling, variance reduction, rare events, ATM;

This work has been supported by NSERC-Canada grants no. ODGP0110050 and SMF0169893, and FCAR-Québec grant no. 93ER1654, to the first author, as well as a joint scholarship from NSERC-Canada and *Newbridge Networks Corporation* to the second author. We thank the Guest Editor James R. Wilson and the three referees for their very careful reading and for their several constructive and useful comments.
Name: P. L'Ecuyer
Address: Département d'Informatique et de Recherche Opérationnelle, Université de Montréal, C.P. 6128, Succ. Centre-Ville, Montréal, H3C 3J7, Canada;
e-mail: `lecuyer@iro.umontreal.ca`
Name: Y. Champoux
Address: Alcatel, 600 March Road, Kanata, K2K 2E6, Canada;
e-mail: `yanick.champoux@cid.alcatel.ca`

1. INTRODUCTION

An Asynchronous Transfer Mode (ATM) communication switch can be modeled as a network of queues with finite buffer sizes. Packets of information (called *cells*) join the network from several sources according to stochastic processes, and some cells may be lost due to buffer overflow. The long-term (or steady-state) fraction of cells that are lost at a given node is called the cell-loss ratio (CLR) at that node. Typical CLRs are small (e.g., less than 10^{-8}) and the cell losses tend to occur in bunches. Cell losses are thus so rare that estimating the CLR with good precision by straightforward simulation is very time-consuming, and in some cases practically impossible.

Importance sampling (IS) (e.g., [Heidelberger 1995]) and the *splitting* method (e.g., [Glasserman et al. 1999]) are two important candidate methods for improving the quality of the estimator in such a situation. IS changes the probability laws governing the system so that the rare events of interest occur more frequently, eventually to the point of being no longer rare events. The estimator is modified accordingly so that it remains unbiased: It is multiplied by a quantity called the *likelihood ratio*. The hope is that the IS estimator is *more efficient* in the sense that the product of its variance and its computing cost is smaller than for the regular estimator. The most difficult problem in applying IS is (in general) to figure out how to change the probability laws so that the variance gets reduced to an acceptable level. Theoretically, there always exists a change of measure that reduces the variance to zero, but it is usually much too complicated and impractical, as it frequently leads to a non-Markovian process [Glynn and Iglehart 1989; Heidelberger 1995].

For general background on efficiency improvement (or variance reduction), we refer the reader to Bratley et al. [1987], Fishman [1996], Glynn [1994], and L'Ecuyer [1994]. IS is well explained in Glynn and Iglehart [1989], Shahabuddin [1994], Heidelberger [1995], Sadowsky [1996], and the several other references given there. Application of IS to the simulation of communication systems is studied, e.g., by Parekh and Walrand [1989], Chang et al. [1994], Chang et al. [1995], Bonneau [1996], Falkner et al. [1999], and Heegaard [1998].

Chang et al. [1994] derived an *asymptotically optimal* change of measure, based on the theories of *effective bandwidth* and *large deviations*, for estimating the probability p that a queue length exceeds a given level x before returning to empty, given that the queue is started from empty, for a *single queue* with multiple independent arrival sources that satisfy a large deviation principle. Roughly, *asymptotically optimal* means that the standard error of the IS estimator converges to zero exponentially fast with the same decay rate (exponent) as the quantity to be estimated, as a function of the level x. A precise definition can be found in Chang et al. [1994] and Heidelberger [1995]. An asymptotically optimal change of measure does not minimize the variance, but it can reduce it by several orders of magnitude. Chang et al. [1994] extended their method to *intree* networks of queues, which are acyclic tree networks where customers flow only towards the root of the tree. For intree networks, assuming infinite buffers at all nodes, they obtained a lower bound on the exponent in the (exponential) convergence rate of the variance of the IS estimator, but they could not prove that this estimator is asymptotically optimal. The missing step for proving the latter is to show that the square mean converges exponentially to 0 at the same rate (i.e., with the same exponent, and not faster) than the variance. In a related paper, Chang [1995] obtained the exact exponent in the convergence rate of $P[q_\infty > x]$ towards 0 as $x \to \infty$, where the random variable q_∞ is the steady-state queue length in an intree queueing network with infinite buffers. This exponent is linked with exactly the same change of measure as in Chang et al. [1994]. The probabilities p and $P[q_\infty > x]$ just described are closely related to the CLR when x equals the buffer size (it measures similar events), so it seems reasonable to use the change of measure proposed by Chang et al. [1994] to estimate the CLR as well. In fact, these authors applied their IS strategy to estimate the steady-state average number of cell losses per unit of time, another quantity closely related to the CLR, and observed large empirical variance reductions for examples of queueing models with a *single node* and *two nodes in series*.

In this paper, we pursue these experiments by considering much larger networks, some of them being non intree. We consider queueing networks having a large number of nodes, fed by a large number of Markov-modulated ON/OFF sources (up to 300 in our numerical examples). The nodes are organized in successive layers. Each cell (or customer) goes through exactly one node of each layer, following a path uniquely determined by its source. This type of queueing network models the traffic in an ATM switch [Dabrowski et al. 1999; Falkner et al. 1999; Giroux and Ganti 1999]. We apply IS to estimate the CLR at any prespecified node of the network, using a direct adaptation of the IS methodology of Chang et al. [1994]. The idea is to increase the traffic to the target node by increasing the average ON/OFF ratio for all the sources (and only those) feeding that node. The exact change of measure is determined by a heuristic which turns out to be equivalent to Algorithm 3.2 of Chang et al. [1994] if we neglect all the traffic not directed towards the target node and the corresponding sources, and if we assume that the average arrival rate after IS does not exceed the service rate at the nodes upstream of the target node. The latter condition is always satisfied in our examples, but if it was not, we could simply enforce it by choosing a change of measure that gives an arrival rate equal to the service rate at those nodes where the condition is violated, exactly as in Algorithm 3.2 of Chang et al. [1994]. Falkner et al. [1999] have proposed an additional heuristic, based on the concept of *decoupling bandwidth* introduced in de Veciana et al. [1994], to reduce the change of measure when the effect of the additional traffic not directed towards the target node may be non-negligible.

For both small and large networks, we obtain large efficiency improvements (empirically), especially when the CLR is small. When the size of the buffer at the target node increases, IS seems to improve the efficiency by a factor that increases exponentially with the buffer size. This is consistent with the *conjecture* that the change of measure is asymptotically optimal.

It must be emphasized that this IS methodology is *heuristic* because (i) CLR is not the same as p; (ii) it is still unproven if the IS estimator proposed in Chang et al. [1994] is asymptotically optimal for p, and proving it for CLR in the presence of finite buffers at all nodes appears quite messy; (iii) even if the IS estimator was asymptotically optimal for the CLR, this would only be an asymptotic result. It would not necessarily imply that the variance is reduced sufficiently and that the estimator is practically acceptable for the concrete models that one is interested in.

Actual experimentation with this IS methodology, with examples similar to real-life problems of interest, is therefore needed to see how practical the method is, and whether or not additional heuristics can improve on it. The aim of the present paper is to report on such experimentation.

Our examples also illustrate the fact, typical of rare-event simulation contexts, that the variance estimators tend to be more noisy (in terms of relative error) than the mean estimators, even after IS is applied. One must therefore be careful when comparing empirical variances, or with the interpretation of confidence intervals. (This important fact is typically left unmentioned in papers that report empirical results on rare-event simulation.) The noise in the variance estimator is not *caused* by the IS methodology, but is due to the fact that the cell losses are rare events. In fact, in our examples, IS improves the variance estimator by a much larger factor than the mean estimator. This is along the lines of the results of Sadowsky [1993], who has shown, in the context of estimating large deviations probabilities for a sum of independent and identically distributed random variables (a much simpler situation than our model), that an optimal exponentially twisted change of measure stabilizes all the moments of the estimator, in addition to being asymptotically optimal. These results have been generalized to a more abstract setup in Sadowsky [1996].

Beck et al. [1999] and Dabrowski et al. [1998] also study the application of IS to a discrete-time queueing network model of an ATM switch. Their model is very general. They obtain the asymptotics of the tail of the queue size distribution in steady-state, and they use that to propose a change of measure for estimating the CLR at a given node. Their IS methodology is related to (but different from) that proposed by Chang et al. [1994] (see also Section 6.3).

We have limited the generality of the model in order to avoid excessive notation and unnecessary complexity. The model is flexible enough to illustrate how the IS strategy behaves in several important situations. In addition to our basic model, we also made several experiments with IS for generalizations and other variants of this model. The proposed IS scheme works fine for some variants and fails for others. This is summarized in Section 5.5. In particular, the fact that each source is assigned a fixed destination (in contrast to having random routing for each cell or group of cells) is an important factor for the success of the method.

The geometric distribution of the ON and OFF periods for the sources (implied by the Markov-modulated model) is certainly not always realistic. Statistical analyses of broadband traffic traces (for Ethernet, video, etc.) indicate that the traffic is often long-range dependent, and that a more representative model in this case is obtained when the sojourn-time distributions in the ON/OFF states are heavy-tailed (e.g., Pareto-type, with infinite variance) [Beran et al. 1995; Leland et al. 1994; Willinger et al. 1995; Neame et al. 1999]. Unfortunately, the IS methodology used in this paper relies heavily on the exponential tails of the ON/OFF distributions, and it is not obvious how (and if) it can be adapted to heavy-tailed distributions. On the other hand, despite the empirical evidence of the limited representativity of the Markov-modulated model in certain contexts, there remain supporters of this model who argue, based on experiments, that in networks with finite buffers the correlations of very long range are usually not important to model anyway [Heyman and Lakshman 1996; Krunz and Makowski 1998]. In any case, we think that the Markov-modulated model is still an important model, and that a better understanding of the efficiency improvement methods proposed for it remains worthwhile.

The model is defined is Section 2. Section 3 recalls the A-cycles method and the batch-means method, which are used jointly to compute confidence intervals. In Section 4 we explain how IS is applied to estimate the CLR at a given target node. The change of measure is the same as in Chang et al. [1994] for intree networks. However, our explanation of the heuristic differs from that of these authors. It is given directly in terms of the CLR and is along the lines of the arguments given in Heidelberger [1995], pages 62–63. Numerical results are reported in Section 5. We also summarize our numerical experiments with several other variants of the model. In Section 6, we consider various refinements of the basic IS scheme, and test them empirically to see how much additional variance reduction they can bring. We report both positive and negative results. Section 7 explains how the CLR can be estimated in functional form, as a function of certain parameters of the model. Section 8 gives a conclusion. A preliminary report of this work was presented in L'Ecuyer and Champoux [1996] and additional numerical results and details can be found in Champoux [1998].

2. THE MODEL

We consider an acyclic queueing network with 4 layers of *nodes*, as illustrated in Figure 1. Each node is a single-server FIFO queue with finite buffer size. The ℓ-th layer is called *level* ℓ and the nodes at level 4 transmit cells to *destinations*. Levels 2 and 3 have m_2 nodes each, while levels 1 and 4 have $m_1 m_2$ nodes each. Each level-2 node is fed by m_1 level-1 nodes, while each level-3 node feeds m_1 nodes at level 4. *Cells* (i.e., packets of information) arrive at level 1, visit one node of each level, in succession, then leave the network. Each node at level 1 is fed by m_0 arrival *sources*. These $m_0 m_1 m_2$ sources are assigned to specific destinations; i.e., all the cells produced by a given source follow exactly the same path. The trajectory of a cell at the switching stage between levels 2 and 3 is thus determined by its fixed destination. The arrival sources are time-synchronized, but otherwise independent, stochastically identical, discrete-time ON/OFF Markov modulated processes. A source is OFF for a while, then ON for a while, then OFF again, and so on. The source produces one cell per unit of time during an ON period, and none during an OFF period. The durations of OFF and ON periods are independent geometric random variables with means κ_0 and κ_1, respectively, so the arrival rate is $\rho = \kappa_1/(\kappa_1 + \kappa_0)$. The parameter κ_1 is called the *average burst size*. If we denote OFF and ON by 0 and 1, respectively, these assumptions imply that the state of a source evolves as a discrete-time Markov chain with two states, 0 and 1, with transition probability matrix

$$R = \begin{pmatrix} r_{00} & r_{01} \\ r_{10} & r_{11} \end{pmatrix} = \begin{pmatrix} 1 - 1/\kappa_0 & 1/\kappa_0 \\ 1/\kappa_1 & 1 - 1/\kappa_1 \end{pmatrix}. \tag{1}$$

These Markov chains comprise all the stochasticity of the model; everything else is deterministic. The arrival sources are numbered from 1 to $m_0 m_1 m_2$ and the nodes are numbered from 1 to $2m_2(1+m_1)$, level by level. When two or more cells reach a given node simultaneously, they are placed in the queue (the buffer) by order of the number of the node or source where they come from. This deterministic ordering rule is for simplification and tends to favor the cells coming from certain sources and nodes. One could order the cells randomly instead, but that would have no major qualitative impact on our results.

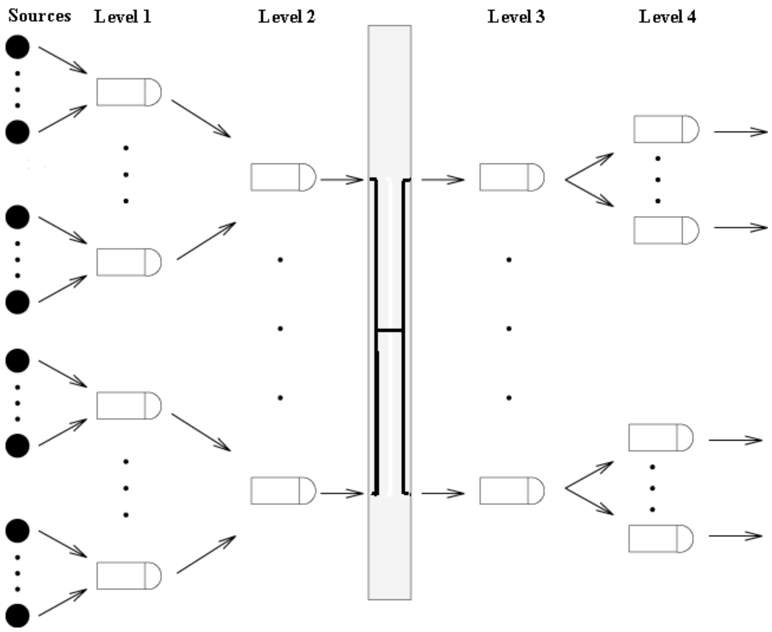

Fig. 1. An ATM Switch Modeled as 4 Layers of Queues with Finite Buffer Sizes

All the nodes at level ℓ have the same buffer size B_ℓ and the same constant service time $1/c_\ell$ (so c_ℓ is the service rate). Whenever a cell arrives at a node where the buffer is full, it is *lost* and disappears from the network. Our aim is to estimate the CLR at a given node of the network, say node q^* at level ℓ^*, where the CLR is defined as

$$\mu = \lim_{\tau\to\infty} \mathrm{E}[N_\mathrm{L}(\tau)]/E[N_\mathrm{T}(\tau)], \tag{2}$$

where $N_\mathrm{T}(\tau)$ is the total number of cells reaching node q^* during the time interval $(0, \tau]$ and $N_\mathrm{L}(\tau)$ is the number of those cells that are lost due to buffer overflow at node q^*. We assume that the total arrival rate is less than the service rate at each node. That is, if the cells from m sources pass through a given node at level ℓ, then $m\rho < c_\ell$, and this holds for all nodes.

To simplify the discussion, we assume that each c_ℓ is an integer. Since the buffers are finite, the entire model is then a discrete-time Markov chain with finite state space. It is also aperiodic, and the *zero* state (the state where all sources are OFF and all the nodes are empty) is positive recurent and is accessible from every other state. As a consequence, there exists a limiting distribution ν over the state space of that chain, defined as

$$\nu(\cdot) = \lim_{\tau\to\infty} P\{\mathcal{S}(\tau) \in \cdot\},$$

where τ is an integer and $\mathcal{S}(\tau)$ is the state of the chain at time τ.

This model could of course be generalized in several directions and our approach would be easy to adapt for certain types of generalizations (see also Section 5.5). For example, the buffer sizes and constant service times can differ between nodes at a given level, different sources can have different transition probability matrices R, a source could produce a cell only with some probability when it is ON, and additional sources may feed the nodes at levels higher than 1. IS still works in these situations. We keep our simpler model to avoid burying the key ideas under a complicated notation. On the other hand, if the destinations were determined randomly and independently for each cell, or for each ON period at each source, finding an efficient way of applying IS would be more difficult. The fixed source-destination assignment model is reasonable if we assume that a typical connection between a source and a destination lasts for several orders of magnitude longer than the service times $1/c_\ell$, and much longer than the time required for buffer overflow in the most likely path to overflow. This assumption is commonly made in the literature [Dabrowski et al. 1999; Falkner et al. 1999; Giroux and Ganti 1999] and by the switch manufacturers.

In the case where $\ell^* = 3$ or 4 and all the nodes at level ℓ^* are statistically identical, an alternative estimator would take the global (empirical) average CLR for all nodes at level ℓ^*. With a straightforward simulation approach (no IS), this would yield a better estimator than concentrating on a single node q^*. But with IS, according to our empirical investigations, it is much better to concentrate on a single node and increase only the traffic to that node. With the latter, the likelihood ratio is less noisy and there is less (undesired) overflow at the nodes of level $\ell < \ell^*$.

3. A REGENERATION APPROACH FOR CONFIDENCE INTERVALS

IS is generally easier to apply to a model defined over a short time horizon or when the model's evolution can be decomposed into short regenerative cycles. Here, the model is over an infinite horizon, and to decompose its trajectory into cycles, we apply a generalization of the classical regenerative method introduced in Nicola et al. [1993] and Chang et al. [1994], and called the *A-cycle* method. Let A be a subset of the state space of the system. Here we take A as the set of states for which the buffer at q^* is empty. Let $\tau_0 = 0$ and for $i \geq 1$, let $\tau_i = \inf\{\tau > \tau_{i-1}$: the buffer at q^* is empty at time τ but not at time $\tau - 1\}$. These τ_i are the successive hitting times of the set A by the Markov chain $\{\mathcal{S}(\tau),\, \tau \in \mathbb{N}\}$. The pair $\{(\mathcal{S}(\tau-1), \mathcal{S}(\tau)),\, \tau \in \mathbb{N}\}$ is also a positive recurrent aperiodic Markov chain over a finite state space, so it has a pointwise limit distribution $\tilde{\nu}$, say, as $\tau \to \infty$. Then the state of the chain at the hitting times τ_i has the pointwise limit distribution π, over A, defined by:

$$\pi(\cdot) = \lim_{i\to\infty} P\{\mathcal{S}(\tau_i) \in \ \cdot\} = \frac{\tilde{\nu}(\overline{A},\, \cdot \cap A)}{\tilde{\nu}(\overline{A},\, A)},$$

where $\overline{A}$ is the complement of the set A.

The process over the time interval $(\tau_{i-1}, \tau_i]$ is called the ith A-cycle. Let X_i be the number of cells reaching node q^* during the ith A-cycle, and Y_i be the number of those X_i cells that are lost due to buffer overflow at q^*. Let E_π denote the mathematical expectation over an A-cycle when the initial state (at the beginning of the A-cycle) has distribution π. One has:

$$\mu = \frac{\mathrm{E}_\pi[Y_1]}{\mathrm{E}_\pi[X_1]}. \tag{3}$$

In the limit, as the number of A-cycles increases, the *average* distribution of the system states at the times τ_i approaches π. By taking the average of the Y_i and X_i over the first n A-cycles, one obtains the consistent estimator of μ:

$$\hat{\mu} = \frac{\sum_{i=1}^n Y_i}{\sum_{i=1}^n X_i}. \tag{4}$$

This estimator is biased because the initial state at time 0 is normally *not* generated from π (this would be too difficult) and because it is a ratio estimator. However, the bias can be reduced by *warming up* the system, e.g., by running n_0+n A-cycles and discarding the first n_0 from the statistics.

The A-cycles are asymptotically identically distributed (with probability law π for their initial state) but they are *dependent*. To reduce the dependence, and also improve the normality, one can *batch* the cycles, as in the usual batch means method. One then applies the standard methodology for computing a confidence interval for a ratio of expectations, using the batch means as observations, and obtain a confidence interval for μ [Law and Kelton 2000]. In our experiments, we fix the number of cycles per batch (and not the simulation time per batch), as explained in Section 5.1.

4. APPLYING IMPORTANCE SAMPLING

When μ is very small, the vast majority of the Y_i's in (4) are 0 and the *relative error* of $\hat{\mu}$ (i.e., its standard deviation divided by μ) blows up. In (3), the denominator $\mathrm{E}_\pi[X_1]$ is easy to estimate, but the numerator is hard to estimate because it depends on rare events. In fact, denoting $\mu_Y = \mathrm{E}_\pi[Y_1]$ and observing that Y_1 is a non-negative integer, one has $\mathrm{Var}_\pi[Y_1] = \mathrm{E}_\pi[Y_1^2] - \mu_Y^2 \geq \mathrm{E}_\pi[Y_1] - \mu_Y^2 = \mu_Y(1-\mu_Y)$, so the squared relative error satisfies

$$\mathrm{RE}^2[Y_1] = \frac{\mathrm{Var}_\pi[Y_1]}{\mu_Y^2} \geq \frac{1}{\mu_Y} - 1 \to \infty \tag{5}$$

as $\mu_Y \to 0$. Following Chang et al. [1994], we will use IS for the numerator of (3) and not for the denominator.

Let S^* denote the set of sources feeding q^*. The IS strategy for increasing the traffic towards q^* is to increase r_{01} and r_{11} in the matrix R, for all the sources that belong to S^* and only those, so that the total long run arrival rate at q^* becomes larger than the service rate. The system starts with an empty buffer at q^* (a state in A) and the change remains in effect until the buffer at q^* empties again or overflows. When the buffer overflows, R is set back to its original for all the sources until the buffer at q^* empties again, which marks the end of the A-cycle. We call this an *A-cycle with IS*. Under this strategy, if the traffic to q^* can be increased sufficiently, cell losses are no longer rare events. This can certainly be achieved if $m^* > c^*$, where m^* is the cardinality of S^* and $c^* = c_{\ell^*}$ is the service rate at the target node.

It remains to decide how to change R. For this, we proceed in a standard way [Heidelberger 1995]. For a real-valued parameter θ, define

$$\Gamma(\theta) = \begin{pmatrix} r_{00} & r_{01}e^{\theta} \\ r_{10} & r_{11}e^{\theta} \end{pmatrix},$$

let $\lambda(\theta)$ be the spectral radius (largest eigenvalue) of $\Gamma(\theta)$, and let $(f_0(\theta), f_1(\theta))'$ be the corresponding right eigenvector (the prime means transposed), so that

$$\begin{pmatrix} r_{00} & r_{01}e^{\theta} \\ r_{10} & r_{11}e^{\theta} \end{pmatrix} \begin{pmatrix} f_0(\theta) \\ f_1(\theta) \end{pmatrix} = \lambda(\theta) \begin{pmatrix} f_0(\theta) \\ f_1(\theta) \end{pmatrix}.$$

The eigenvalue $\lambda(\theta)$ can be written explicitly as

$$\lambda(\theta) = \frac{1}{2}\left(r_{00} + r_{11}e^{\theta} + \sqrt{(r_{00} - r_{11}e^{\theta})^2 + 4e^{\theta}r_{01}r_{10}}\right).$$

For IS, we will change R to the stochastic matrix

$$\tilde{R} = \begin{pmatrix} \tilde{r}_{00} & \tilde{r}_{01} \\ \tilde{r}_{10} & \tilde{r}_{11} \end{pmatrix} = \frac{1}{\lambda(\theta)} \begin{pmatrix} r_{00} & r_{01}e^{\theta} f_1(\theta)/f_0(\theta) \\ r_{10} f_0(\theta)/f_1(\theta) & r_{11}e^{\theta} \end{pmatrix}.$$

This formulation is quite flexible, because the mean arrival rate from a source can be set to an arbitrary value between 0 and 1 by choosing an appropriate θ, and it leads to important simplifications in the likelihood ratio over an A-cycle, as we will see.

During a given A-cycle, let N_{ij} be the number of times a source in S^* goes from state i to state j *while using the probabilities* $\tilde{r}_{ij}$, for $i = 0, 1$ and $j = 0, 1$. The total number of transitions generated from $\tilde{R}$ is then $N_{00} + N_{01} + N_{10} + N_{11} = m^*t$, where t is the number of time steps where IS is on. The state transitions of the sources are assumed to occur right before the (discrete) times of cell production. The number of cells generated for q^* during the time interval $(0, t]$ is thus $N_{01} + N_{11}$. By a simple counting argument, assuming that the buffer overflows at time t, one can decompose

$$N_{01} + N_{11} + Q_0 = B^* + Q_t + (c^*t - F_t) + H_t, \qquad \text{if } Y_1 > 0, \tag{6}$$

where B^* is the buffer size at q^*, Q_t is the number of cells already generated and on their way to node q^* at time t, Q_0 is the corresponding number at time 0, F_t is the difference between the total capacity of service c^*t of the server at q^* during $(0, t]$ and the actual number of cells served at q^* during that interval of time, and H_t is the number of cells headed to q^* but lost due to buffer overflow either at q^* or upstream during $(0, t]$. In (6), B^* is the number of cells filling up the buffer and $c^*t - F_t$ is the total number of cells served at q^* during $(0, t]$. We denote $\Delta = N_{01} + N_{11} - B^* - c^*t$, which equals $Q_t - Q_0 + H_t - F_t$ when $Y_1 > 0$.

The transition probability of a source from state i to state j is r_{ij} without IS, and $\tilde{r}_{ij}$ under IS, so each such transition under IS contributes a factor $r_{ij}/\tilde{r}_{ij}$ to the *likelihood ratio* associated with this change of probabilities. Since there are N_{ij} transitions from i to j, the likelihood ratio becomes

$$L = \left(\frac{r_{00}}{\tilde{r}_{00}}\right)^{N_{00}} \left(\frac{r_{01}}{\tilde{r}_{01}}\right)^{N_{01}} \left(\frac{r_{10}}{\tilde{r}_{10}}\right)^{N_{10}} \left(\frac{r_{11}}{\tilde{r}_{11}}\right)^{N_{11}}$$

$$
\begin{aligned}
&= (\lambda(\theta))^{N_{00}} \left(\frac{\lambda(\theta) f_0(\theta)}{f_1(\theta) e^{\theta}}\right)^{N_{01}} \left(\frac{\lambda(\theta) f_1(\theta)}{f_0(\theta)}\right)^{N_{10}} \left(\frac{\lambda(\theta)}{e^{\theta}}\right)^{N_{11}} \\
&= W(\theta)\lambda(\theta)^{m^* t} \exp[-\theta(N_{01} + N_{11})] \\
&= W(\theta) \exp[-\theta B^*] \exp[m^* t \ln \lambda(\theta) - \theta(c^* t + \Delta)] \qquad (7)
\end{aligned}
$$

where

$$W(\theta) \;=\; (f_0(\theta)/f_1(\theta))^{N_{01} - N_{10}}.$$

It is well known that if V is a random variable defined over an A-cycle with initial state that has distribution π, then $\mathrm{E}_\pi[V] = \tilde{\mathrm{E}}_\pi[LV]$ where $\tilde{\mathrm{E}}_\pi$ denotes the expectation under the probabilities $\tilde{R}$, over an A-cycle with IS, with initial state drawn from π. Thus, computing LY_1 over the A-cycle with IS yields an unbiased estimator of $\mathrm{E}_\pi[Y_1]$. One always has $\tilde{\mathrm{E}}_\pi[L] = 1$. IS is effective (roughly) if L tends to have a small variance conditional on $Y_1 > 0$. In other words, we would like to match the small values of L with the positive values of Y_1 (i.e., the buffer overflows) and the large values of L with the no-overflow situations.

We still have the choice of θ at our disposal. Since t is an unbounded random variable likely to have significant variance, a standard strategy in this situation (e.g., [Heidelberger 1995]) is to choose θ so that t disappears from L, i.e, take $\theta = \theta^* > 0$ such that

$$m^* \ln \lambda(\theta^*) = \theta^* c^*. \qquad (8)$$

This θ^* gives the same IS strategy as that provided in Chang et al. [1994] if we consider only the sources that feed q^* and if we neglect the possibility that the (mean) arrival rate exceeds the service rate at upstream nodes. The latter condition was satisfied in all the examples that we have considered, but when it is not, one can reduce θ to meet the condition exactly as in Chang et al. [1994]. Note that $\ln \lambda(0) = 0$, that $\theta^{-1} \ln \lambda(\theta)$ is a strictly increasing and differentiable function of θ (see, e.g., Chang et al. [1994], Example 2.6), and that $\theta^{-1} \ln \lambda(\theta) \to 1$ as $\theta \to \infty$. Therefore, this θ^* exists if and only if $m^* > c^*$, which we assume (otherwise, one cannot overload the node q^* and overflows should be negligible anyway unless B^* is near 0). With $\theta = \theta^*$, the likelihood ratio becomes

$$L \;=\; e^{-\theta^*(B^* + \Delta)} W(\theta^*).$$

The variance of the estimator of μ_Y is $\tilde{\mathrm{Var}}_\pi[LY_1] = \tilde{\mathrm{E}}_\pi[L^2 Y_1^2] - \mu_Y^2$ where

$$\tilde{\mathrm{E}}_\pi[L^2 Y_1^2] \;=\; e^{-\theta^* B^*} \tilde{\mathrm{E}}_\pi[L Y_1 e^{-\theta^* \Delta} Y_1 W(\theta^*)]. \qquad (9)$$

The squared relative error of the IS estimator satisfies

$$\tilde{\mathrm{RE}}^2[LY_1] \;=\; \frac{\tilde{\mathrm{Var}}_\pi[LY_1]}{\mu_Y^2} \;\le\; \frac{\tilde{\mathrm{E}}_\pi[L^2 Y_1^2]}{\mu_Y^2} \;=\; \frac{\tilde{\mathrm{E}}_\pi[(e^{-\theta^* \Delta} Y_1 W(\theta^*))^2]}{\tilde{\mathrm{E}}_\pi^2[e^{-\theta^* \Delta} Y_1 W(\theta^*)]}. \qquad (10)$$

The ratio of expectations in (10) is ≥ 1, by the Cauchy-Schwarz inequality. Bounding this ratio by a constant independent of B^* would prove that the relative error under IS is bounded, but we do not have a proof. Sadowsky [1991] was able to prove that IS based on an optimal exponential twisting found by solving an equation similar to (8) is asymptotically optimal for estimating p in a single $GI/GI/m$ queue with finite buffer. The results of Chang et al. [1994] generalize this to *dependent* interarrival times, still for a single queue.

We continue the discussion with *intuitive heuristic arguments* suggesting that L is likely to be small when $Y_1 > 0$, and that the variance of LY_1 is also likely to be small. Firstly, since q^* is stable without IS and since IS is stopped as soon as the buffer overflows, Y_1 is unlikely to take very large values (it should be possible to prove that its distribution has an exponentially bounded tail under IS). Concerning $W(\theta^*)$, for the matrices R and the values of m^*/c^* that correspond to typical examples (see Section 5), one usually has $f_0(\theta^*)/f_1(\theta^*) < 1$. Moreover, when $Y_1 > 0$, it is unlikely that less sources are ON under IS than without IS, because the ON state has a higher steady state probability under IS, so $N_{01} \ll N_{10}$ is extremely unlikely. Therefore, it seems reasonable to expect that $W(\theta^*)$ is less than 1 with large probability and has a small variance, bounded independently of B^*. (In fact, $W(\theta^*)$ is bounded independently of B^* because $|N_{01} - N_{10}|$ is bounded by m^*, but this bound can be very large unless m^* is small.) Now it suffices to hope that the large negative values of Δ are rare enough to be negligible when $Y_1 > 0$. This hope appears reasonable for the following reasons. Since the sources are overproducing cells headed towards q^* when IS is ON, and that the initial state at time 0 follows approximately the limiting distribution π *without* IS, one should normally expect $Q_t \geq Q_0$, and it should be (intuitively) extremely unlikely that $Q_t \ll Q_0$. (Remember that Q_0 and Q_t are the total number of cells in the upstream nodes, in steady-state (roughly), without IS and under IS, respectively.) Moreover, since the production rate of the sources exceeds the capacity at q^*, the server at q^* is not expected to remain idle very long, so a large positive value of F_t is very unlikely. Since $\exp(-\theta^* B^*)$ is a constant, the above arguments suggest that L should tend to be very small when $Y_1 > 0$. From another viewpoint, when $Y_1 = 0$ (no overflow), the value of Δ is likely to be slightly smaller than $-B^*$, i.e., the total number of cells produced during $(0, t]$ should be slightly less than the total service capacity $c^* t$, because the buffer at q^* is empty when IS is turned OFF. This suggests that L should tend to be larger than 1 when $Y_1 = 0$, which is good news in view of the fact that $\tilde{E}_\pi[L] = 1$.

As a *very crude heuristic*, suppose we assume that $e^{-\theta^* \Delta} Y_1 W(\theta^*)$ in (9) is bounded by a constant K_1 independent of B^*. This would give

$$\tilde{\mathrm{E}}_\pi[L^2 Y_1^2] \;\leq\; e^{-\theta^* B^*} K_1 \tilde{\mathrm{E}}_\pi[L Y_1] \;=\; e^{-\theta^* B^*} K_1 \mu_Y, \tag{11}$$

in which case IS would provide the approximate variance reduction factor

$$\frac{\tilde{\mathrm{Var}}_\pi[LY_1]}{\mathrm{Var}_\pi[Y_1]} \;\leq\; \frac{\tilde{\mathrm{E}}_\pi[L^2 Y_1^2] - \mu_Y^2}{\mu_Y - \mu_Y^2} \;\leq\; \frac{e^{-\theta^* B^*} K_1 - \mu_Y}{1 - \mu_Y} \;\approx\; e^{-\theta^* B^*} K_1.$$

Again, these arguments are only hand-waving heuristics and the real test is to see how the variance is actually reduced on concrete examples. One may be tempted to modify the IS scheme adaptively (e.g., by stopping IS earlier or later) in order to reduce the variability of the quantity $e^{-\theta^* \Delta} Y_1 W(\theta^*)$. We will return to this in Section 6.

This discussion assumed implicitly that m^* is not too large (e.g., for bounding $W(\theta^*)$). One may expect the method to eventually break down as $m^* \to \infty$ for fixed B^*, in view of the fact that the asymptotic overflow frequency for fixed B^*

and $m^* \to \infty$ generally differs from that when $B^* \to \infty$ (e.g., [Shwartz and Weiss 1995]). Courcoubetis and Weber [1996] compare these asymptotics and give numerical examples, for fluid source models, where the asymptotic based on large m^* gives a better approximation than the one based on large B^* to the steady-state probability that the queue exceeds B^*. (The asymptotic based on large B^* overestimated the probability.) This suggests that the change of measure that we use might perform poorly when m^* is large and B^* is small. On the other hand, the large m^* asymptotic for fixed B^* converges to the large B^* asymptotic as $B^* \to \infty$ for their model. Empirically, we tried examples with m^* up to 50 and the method still worked fine. We also observed that taking θ slightly less than θ^* in our experiments tended to improve the efficiency when m^* is large (see Section 6.1).

What about the variance of the variance estimator, with and without IS? They can be compared by comparing $\tilde{\mathrm{E}}_\pi[L^4 Y_1^4]$ with $\mathrm{E}_\pi[Y_1^4]$. Using the same crude argument as in (11) above, one can conjecture the approximation

$$\frac{\tilde{\mathrm{E}}_\pi[L^4 Y_1^4]}{\mathrm{E}_\pi[Y_1^4]} = \frac{\tilde{\mathrm{E}}_\pi[L^4 Y_1^4]}{\tilde{\mathrm{E}}_\pi[L Y_1^4]} \le e^{-3\theta^* B^*} K_2$$

for some constant K_2 independent of B^*. We expect not only the estimator itself to be less noisy with IS than without, but also its sample variance to be less noisy, and by a larger factor (at least if B^* is large).

We now explain how the A-cycles are simulated to estimate both the numerator and the denominator in (3), in the IS case. One simulates two *versions* of each A-cycle, one with IS and the other without, both starting from the same initial state. Thus, the A-cycles come in pairs. For the ith A-cycle pair, one first simulates an A-cycle with IS, which provides an estimation $L_i Y_i$ of the numerator, where L_i and Y_i are the value of the L and the number of cell losses for this cycle. Then, the state of the system is reset to what it was at the beginning of this A-cycle with IS, and a second A-cycle is simulated to obtain an estimator X_i of the denominator. The final state of the no-IS A-cycle, which obeys approximately the distribution π, is then saved and is taken as the initial state for the next pair of A-cycles. After a warmup of n_0 cycles without IS, n pairs of A-cycles are thus simulated and the IS estimator of μ is

$$\hat{\mu} = \frac{\sum_{i=1}^n L_i Y_i}{\sum_{i=1}^n X_i}.$$

A confidence interval is computed using batch means as explained in Section 3.

Starting the two A-cycles of each pair from the same state means that we must save or reset the entire state of the system after each cycle. This means copying how many cells are at each node of the network, the destinations of these cells, and the state (ON or OFF) of each source. We also memorize/reset the state of each random number generator, so that the two A-cycles of a pair use common random numbers. This tends to increase the correlation between $L_i Y_i$ and X_i, and to decrease the variance of $\hat{\mu}$ as a result.

5. SIMULATION EXPERIMENTS

5.1 The Setup

For several examples and parameter sets, we ran the simulation first using the standard approach without IS, for C A-cycles, and then with IS for C' pairs of A-cycles. In each case, the values of C and C' were chosen so that the total CPU time was about the same for both IS and no-IS, and these A-cycles were regrouped into $b = 200$ batches. (For sensitivity analysis with respect to b, we tried different values of b ranging from 50 to 3200, for several examples, and found that the variance estimates were practically independent of b, in that range, for the values of C and C' that we use). For $1 \leq j \leq b$, let $\bar{X}_j$ and $\bar{Y}_j$ denote the samples means of the X_i and Y_i (or of the X_i and $L_i Y_i$, for IS), respectively, within batch j. The tables that follow report the value of the CLR estimator $\hat{\mu}$ and of its variance estimator

$$\hat{\sigma}^2 = (S_Y^2 + \hat{\mu}^2 S_X^2 - 2\hat{\mu} S_{XY})/(b\bar{X}^2), \tag{12}$$

where $\hat{\mu} = \bar{Y}/\bar{X}$, and $\bar{Y}$, $\bar{X}$, S_Y^2, S_X^2, and S_{XY} are the sample means, sample variances, and sample covariance of the $\bar{Y}_j$ and $\bar{X}_j$, respectively. The tables also report the relative half-width $\hat{\delta} = 2.57\hat{\sigma}/\hat{\mu}$ of a 99% confidence interval on μ (under the normality assumption), the CPU time T_{CPU} (in seconds) required to perform the simulation, and the *relative efficiency* (eff.), defined as $\hat{\mu}^2/(\hat{\sigma}^2 T_{\mathrm{CPU}})$. These values are all noisy estimates but give a good indication of what happens.

For the cases where no cell loss was observed in all the A-cycles simulated, we put $\hat{\mu} = 0$ and the entries for the variance and efficiency are left blank. The simulation with IS takes more CPU time than no-IS for the same total number of simulated cells, but the relative efficiency takes both the variance reduction and the overhead into account. Beware: Efficiencies and CPU times can be compared within a given table, but not across the tables, because the models are different and the experiments were run on different machines (SUN SparcStations 4, 5, and 20). Within each table, common random numbers were used for the corresponding A-cycles across the different lines of the table.

5.2 CLR Estimation at Level 2

EXAMPLE 1. Let $\ell^* = 2$, $B_1 = 512$, $m_0 = 2$, $m_1 = 25$, $c_1 = 1$, $c_2 = 3$, $\kappa_1 = 50$, $\rho = 1/101$ (i.e., $r_{11} = 49/50 = 0.9800$ and $r_{00} = 0.9998$) and vary the buffer size $B^* = B_2$. There are 50 sources feeding the target node q^* (i.e., $m^* = 50$), so the average arrival rate at q^* is $50/101 \approx 0.495$, while the service rate is 3. With these numbers, we compute $\theta^* = 0.018127$, $f_0(\theta^*) = 0.0581$, $f_1(\theta^*) = 0.3676$, $\tilde{r}_{11} = 0.99684$, $\tilde{r}_{00} = 0.99871$, and IS increases the total arrival rate at q^* from 0.495 to 14.48.

We took $C = 7\,200\,000$ for no-IS and $C' = 300\,000$ for IS (note that the IS cycles are much longer than the no-IS on the average, and their average length increases with B^*, because most of them fill up the buffer before emptying it again, whereas for most of the no-IS cycles the buffer empties after just a few cell arrivals). Table 1 gives the results. For $B_2 \geq 512$, without IS, not a single cell loss was observed, so the estimates are useless. On the other hand, the relative error of the IS estimators does not increase significantly as a function of B_2, and these estimators can estimate very small CLRs. The efficiency decreases slowly with B_2. (The outlier at $B_2 = 768$ will be discussed later on.)

EXAMPLE 2. Same as the preceding example, except that B_2 is now fixed at 512 and we vary the average burst size κ_1. For large κ_1, μ is large and easy to estimate, but not for small κ_1 (the other parameters remaining the same). The results are in Table 2. Without IS, cell losses were observed only for $\kappa_1 \geq 100$, and even in that case IS is more efficient. The total arrival rate with IS decreases with κ_1: It goes from 22.5 for $\kappa_1 = 25$ to 5.95 with $\kappa_1 = 150$. The squared relative error with IS (not show in the table) is approximately constant as a function of κ_1.

Table 1. CLR estimation at level 2 for different buffer sizes

B_2	$\hat{\mu}$	$\hat{\sigma}^2$	$\hat{\delta}$	CPU	eff.
			no-IS		
128	2.8E-5	2.5E-11	45%	2828	0.0113
256	6.8E-7	4.6E-13	257%	2828	0.0003
512	0			2828	
768	0			2829	
1024	0			2827	
			IS		
128	3.0E-5	6.3E-13	7%	1675	0.838
256	9.8E-7	1.5E-15	10%	1993	0.315
512	2.5E-9	5.4E-20	24%	2593	0.043
768	3.7E-11	5.9E-22	170%	3108	0.001
1024	5.6E-14	3.6E-29	28%	3634	0.023

Table 2. CLR estimation at level 2 for different average burst sizes

κ_1	$\hat{\mu}$	$\hat{\sigma}^2$	$\hat{\delta}$	CPU	eff.
			no-IS		
10	0			1969	
25	0			2245	
50	0			2828	
100	8.1E-5	1.0E-10	32%	4591	0.014
150	3.0E-3	2.9E-9	5%	7641	0.414
			IS		
10	1.1E-39	1.5E-79	87%	2897	0.003
25	2.1E-17	8.6E-36	36%	2884	0.017
50	2.5E-9	5.4E-20	24%	2593	0.043
100	7.2E-5	3.2E-12	6%	2445	0.659
150	3.0E-3	3.9E-9	5%	2585	0.905

An important question now arises: How noisy are the variance and efficiency estimates given in the tables? One way of estimating the distribution of the variance and efficiency estimators is to bootstrap from the b batch means, as follows. Put the b pairs $(\bar{X}_1, \bar{Y}_1), \ldots, (\bar{X}_b, \bar{Y}_b)$ in a table. Draw b random pairs from that table, with replacement, and compute the quantities $\hat{\sigma}^2$ and eff. that correspond to this sample of size b. Repeat this N times and compute the empirical *distributions* of the N values of $\hat{\sigma}^2$ and of eff. thus obtained. These empirical distributions are bootstrap estimators of the distributions of $\hat{\sigma}^2$ and eff., and the interval between the 2.5th and 97.5th percentiles of the empirical distribution is a 95% bootstrap confidence interval for the variance of $\hat{\mu}$ or for the efficiency. Table 3 gives the xth percentiles Q_x of the bootstrap distributions obtained from the results of Example 1, for $x = 2.5$, 50, and 97.5, with $N = 10\,000$.

Table 3. Bootstrap quantile estimates for Example 1

	$\hat{\sigma}^2$			eff.		
B_2	$Q_{2.5}$	Q_{50}	$Q_{97.5}$	$Q_{2.5}$	Q_{50}	$Q_{97.5}$
128	4.3E-13	6.2E-13	8.8E-13	0.23	0.31	0.42
256	8.1E-16	1.5E-15	2.4E-15	0.08	0.12	0.19
512	6.2E-21	5.4E-20	1.5E-19	7.5E-3	1.6E-2	1.0E-1
768	3.6E-25	5.9E-22	1.8E-21	2.4E-4	3.7E-4	3.7E-3
1 024	9.9E-30	3.4E-29	8.1E-29	5.1E-3	8.7E-3	2.2E-2

We already pointed out the very low empirical efficiency of the IS estimator with $B_2 = 768$ in Table 1. A closer look at the 200 batch means $\bar{Y}_j$ reveals that one of the $\bar{Y}_j$ in that case is 4.19×10^{-8}, whereas all others are less than 10^{-9}, except one which is 1.92×10^{-9}. It seems that a rare event has happened within that particular batch. We did not observe such outliers for the other values of B_2, but we found some in other examples, although rarely as excessive. The presence of these outliers indicates that there remains an important tail in the distribution of $\bar{Y}_j$ with IS, despite the large reduction in the variance of $\bar{Y}_j$. This outlier has an important effect not only on the variance and efficiency estimators, but also on the bootstrap distributions, as can be seen from Table 3 (compare the behavior of the quantiles for $B_2 = 768$ with those for the other values of B_2). To assess the effect of the outlier, we repeated the bootstrap after removing it from the sample (i.e., with the 199 remaining pairs), and obtained the following quantiles for $\hat{\sigma}^2$: $(Q_{2.5}, Q_{50}, Q_{97.5})$ =(2.7E-25, 1.7E-24, 4.5E-24). The effect is significant. The numbers suggest that for $B_2 = 768$, the variance is highly overestimated, that the efficiency is underestimated, and that the bootstrap distribution is more widely spread than the true distribution. To confirm these suspicions, we made 5 additional replications of the entire experiment, independently, with $B_2 = 768$ and IS. The results, in Table 4, give an idea of the variability. Table 5 provides similar results for $B_2 = 512$. One can see that the efficiency estimator is (unfortunately) noisy. On the other hand, $\hat{\mu}$ is (fortunately) much less noisy, and this is reassuring. Another reassuring empirical observation is that the distribution of the variance estimator is skewed towards conservatism, in the sense that we observed large overestimation of the variance (like here) from time to time, but did not observe large underestimation of the variance. Moreover, the variance estimators tended to be more noisy when m^* was large. Often, in practice, only crude estimates of μ are sought (e.g., up to a factor of 10 or so), so these variance estimators may suffice. Otherwise, one must increase the sample size. We warn the reader that in a situation where the change of measure is not very effective, $\hat{\mu}$ might be noisy as well. This is in fact what happens without IS. However, this never happened in our examples.

Table 4. Five additional independent replications for $B_2 = 768$ with IS

$\hat{\mu}$	$\hat{\sigma}^2$	$\hat{\delta}$	CPU	eff.
1.2E-11	1.1E-24	22%	3132	0.044
1.2E-11	2.1E-24	30%	3100	0.023
1.1E-11	8.0E-25	21%	3108	0.048
1.1E-11	1.8E-24	31%	3119	0.022
9.5E-12	2.9E-25	14%	3100	0.101

Table 5. Five additional independent replications for $B_2 = 512$ with IS

$\hat{\mu}$	$\hat{\sigma}^2$	$\hat{\delta}$	CPU	eff.
3.1E-9	1.8E-19	35%	2592	0.020
2.5E-9	2.0E-20	15%	2587	0.115
3.8E-9	2.0E-18	94%	2591	0.003
3.3E-9	3.5E-19	46%	2600	0.012
2.4E-9	1.0E-20	11%	2588	0.223

5.3 CLR Estimation at Level 3

EXAMPLE 3. Let $\ell^* = 3$, $B_1 = B_2 = 512$, $c_1 = 1$, $c_2 = c_3 = 2$, $m_0 = 2$, $m_1 = 3$, $m_2 = 10$, $\kappa_1 = 50$, $\rho = 1/21$, and we vary the buffer size $B^* = B_3$. We assign 6 of the 60 sources to q^* (so $m^* = 6$). One node at level 2 is fed by 2 of these 6 *hot* sources, while no other node at levels 1 and 2 is fed by more than 1 of them. Here, $\theta^* = 0.027287$, $f_0(\theta^*) = 0.0851186$, $f_1(\theta^*) = 0.839641$, and the total arrival rate at q^* is 6/21 without IS and 5.0 with IS. We take $C = 1\,800\,000$ and $C' = 100\,000$. The results appear in Table 6. IS works nicely while the no-IS observes no cell loss except at the smallest buffer size. With IS, the relative error and the relative efficiency are almost constant with respect to B^*.

EXAMPLE 4. Same as the preceding example, except that B_3 is fixed at 256 and we vary the average burst size κ_1. Table 7 gives the results. While no-IS has difficulty to observe cell losses, IS gives reasonable estimations.

Table 6. CLR estimation at level 3 for different buffer sizes

B_3	$\hat{\mu}$	$\hat{\sigma}^2$	$\hat{\delta}$	CPU	eff.
			no-IS		
128	2.4E-5	1.1E-10	112%	7036	0.002
256	0			7024	
512	0			7059	
768	0			7037	
1024	0			7027	
			IS		
128	4.1E-5	5.3E-12	14%	5779	0.056
256	6.0E-7	7.2E-16	11%	7316	0.069
512	3.4E-10	2.9E-22	13%	10246	0.040
768	2.5E-13	1.7E-28	13%	12930	0.029
1024	2.1E-16	1.4E-34	14%	15649	0.020

Table 7. CLR estimation at level 3 for different average burst sizes

κ_1	$\hat{\mu}$	$\hat{\sigma}^2$	$\hat{\delta}$	CPU	eff.
			no-IS		
10	0			1088	
25	0			1050	
50	0			1125	
100	2.1E-5	2.1E-10	178%	1134	0.007
150	1.1E-4	1.6E-9	94%	1136	0.015
			IS		
10	5.4E-20	8.6E-42	14%	1454	0.24
25	1.3E-10	2.8E-23	11%	1213	0.49
50	6.0E-7	7.2E-16	11%	1042	0.48
100	4.1E-5	6.7E-12	16%	881	0.28
150	1.7E-4	9.1E-11	15%	813	0.38

5.4 CLR Estimation at Level 4

Example 5. Let $\ell^* = 4$, $B_1 = B_2 = B_3 = 512$, $c_1 = c_4 = 1$, $c_2 = c_3 = 4$, $m_0 = 5$, $m_1 = 10$, $m_2 = 6$, $\kappa_1 = 50$, $\rho = 1/41$, and we vary the buffer size $B^* = B_4$. We assign 6 of the 300 sources to q^*. They are distributed as in Example 3. Here, $\theta^* = 0.021218$, $f_0(\theta^*) = 0.0813754$, $f_1(\theta^*) = 0.644124$, and the total arrival rate at q^* is 6/41 without IS and 3.692 with IS. We take $C = 800\,000$ and $C' = 50\,000$. The results are in Table 8 and they resemble what was observed at level 3 (the efficiencies are smaller because we have a larger network and more sources to simulate). For this example, we also varied κ_1 with B_2 fixed at 512, and the results were qualitatively similar to those of Table 7.

Table 8. CLR estimation at level 4 for different buffer sizes

B_4	$\hat{\mu}$	$\hat{\sigma}^2$	$\hat{\delta}$	CPU	eff.
			no-IS		
128	1.6E-3	7.5E-8	44%	3580	0.004
256	8.3E-6	6.8E-11	255%	3593	0.013
512	0			3586	
768	0			3592	
1024	0			3595	
			IS		
128	1.1E-3	4.0E-9	15%	1881	0.15
256	5.5E-5	1.4E-10	55%	2440	0.008
512	1.4E-7	4.4E-16	39%	3580	0.012
768	3.6E-10	2.7E-21	36%	4488	0.011
1024	1.0E-12	1.5E-26	31%	5550	0.012

5.5 Other Variants of the Model

We made several experiments with variants of the model to explore the effectiveness of the proposed IS strategy in other (sometimes more realistic) situations.

The original model is called *variant A*. For *variant B*, the sources are no longer affected to fixed destinations, but the destination of each cell is chosen randomly, independently of other cells, uniformly over all destinations. *Variant C* is similar except that each *burst* (i.e., all the cells from a source during a given ON period) has a random destination. The IS approach of Section 4 did very badly for variant B, and gave improvement for variant C only when μ was very small. An appropriate IS strategy for these models should also change the probabilities over the destinations to increase the traffic towards q^*. Variants B and C are not very realistic for ATM switches, but the next 2 variants have higher practical interest.

In variant D, each node at level 3 has k buffers, the first one receiving the cells originating from the sources 1 to $m_0m_1m_2/k$, the second one taking those from the sources $1 + m_0m_1m_2/k$ to $2m_0m_1m_2/k$, and so on. A server at level 3 takes cells from those buffers according to either a *round-robin* or *longest-queue-first* policy.

In variant E, the sources produce two classes of cells: High priority *constant bit rate* (CBR) cells and low priority *variable bit rate* (VBR) cells. The VBR sources are Markov modulated as before, whereas the CBR sources have constant ON and OFF periods (they are completely deterministic). Each node has two buffers, one for the CBR cells and one for the VBR cells, and the CBR cells are always served before the VBR ones.

The IS strategy of Section 4 works fine for the variants D and E: It provides reasonable estimates for values of μ that standard simulation cannot handle. We also observed in our empirical results that the longest-queue-first policy gives a CLR generally smaller than round robin.

6. REFINING THE IMPORTANCE SAMPLING SCHEME

6.1 Optimizing θ

The IS approach of Section 4 provides a good change of measure, but based only on a heuristic and asymptotic argument, not necessarily the optimal value of θ for a given buffer size. Moreover, when choosing θ, the approach does not take into account the computational costs which may depend on θ. To evaluate the sensitivity with respect to θ, we performed additional experiments where θ was varied around θ^*, and the variance and efficiency were estimated. As a general rule, for the class of examples examined, we found that the optimal θ was around 20% to 25% less than θ^*, and increased the efficiency by a factor between 2 to 60 compared with θ^*, at level 2 or 3 where m^* is typically large. At level 1 or 4, where m^* is usually small, the optimal θ tends to be much closer to (and no significantly better than) θ^*. We emphasize that there is noise in these estimated factors, due to the variance of the efficiency estimators. However, the tendency persisted when we replicated the experiments. Moreover, the θ that gives the best efficiency also tends to reduce significantly the variance of the variance estimator. These improvements are important, so it seems reasonable to use, e.g., $\theta = (4/5)\theta^*$ instead of θ^* at levels 2 and 3, for networks similar to those that we have considered here, and perhaps try to optimize θ adaptively in a small neighborhood around that value, during the simulation. Taking a smaller θ is more conservative, in the sense that it produces a smaller change of measure. Using $\theta > \theta^*$, which is more aggressive, is very dangerous because the variance increases very fast with θ in that area, and may even become infinite for finite θ. The next examples illustrate typical behavior at levels 3 and 4.

Example 6. Let $\ell^* = 3$, $B_1 = B_2 = B_3 = 256$, $m_0 = 2$, $m_1 = 3$, $m_2 = 10$,

$c_1 = 1$, $c_2 = c_3 = 2$, $\kappa_1 = 50$, and $\rho = 1/21$. The node q^* is fed by 6 sources, whose traffic passes through as in example 3. We take $C = 1\,920\,000$. Here, $\theta^* = 0.0272$, and the results for different values of θ around θ^* are in Table 9. Taking $\theta = 0.0185$ improves the empirical efficiency by a factor of approximately 25 compared with θ^*. By examining the data more closely, we found that the efficiency improves because the smaller θ gives a smaller value of $S_Y^2/(b\bar{X}^2)$, which is the dominant term in $\hat{\sigma}^2$. Further replications showed similar results, with $\theta = 0.0185$ registering efficiencies 15 to 60 times higher than θ^*. The variance estimator was also much less noisy at $\theta = 0.0185$ (where the value of $\hat{\delta}$ was always between 3.6% and 4.2%) than at θ^*.

Table 9. Comparing different values of θ, for $\ell^* = 3$

θ	$\hat{\mu}$	$\hat{\sigma}^2$	$\hat{\delta}$	CPU	eff.
0.0170	6.3E-7	9.3E-17	3.9 %	16672	0.258
$\Rightarrow$ 0.0185	6.4E-7	8.0E-17	3.6 %	16744	0.304
0.0200	6.4E-7	1.4E-16	4.8 %	16321	0.176
0.0215	6.2E-7	1.3E-16	4.7 %	15088	0.201
0.0230	6.2E-7	3.4E-16	7.7 %	14307	0.079
0.0245	6.2E-7	1.4E-15	15.5 %	13701	0.020
0.0260	5.7E-7	2.5E-15	22.4 %	15478	0.008
$\rightarrow$ 0.0272	5.2E-7	1.7E-15	20.4 %	12945	0.012
0.0290	5.2E-7	5.1E-15	35.2 %	12667	0.004

EXAMPLE 7. Let $\ell^* = 4$, $B_1 = B_2 = B_3 = B_4 = 512$, $m_0 = 2$, $m_1 = m_2 = 5$, $c_1 = c_4 = 1$, $c_2 = c_3 = 5$, $\rho = 1/41$, and $\kappa_1 = 50$. Only 2 sources feed the node q^*. Both sources feed the same node at level 3, but different nodes at levels 1 and 2. We take $C = 120\,000$. In this case, $\theta^* = 0.0394$, and we found empirically that taking $\theta < \theta^*$ brings no significant efficiency improvement. We made similar experiments with exactly the same data as in Example 5, with $B_4 = 256$, and observed an efficiency improvement by a factor between 1.5 and 2.

6.2 Defining the A-Cycles Differently

Instead of starting the A-cycles when the buffer at q^* becomes empty, one can start them when the number of cells in the buffer crosses β upward, where β is a fixed integer. There is essentially nothing to gain in that direction, however, because when increasing β the no-IS A-cycles tend to become excessively long (typically, the buffer at q^* remains nearly empty most of the time).

Another idea is to impose a lower bound, say t_0, on the length of the A-cycles, to get rid of the extremely short (and wasteful) A-cycles which tend to occur frequently under both the IS and no-IS setup. The A-cycle ends at the maximum time between t_0 and the first time when node q^* becomes empty. How to choose t_0? We want to choose it large enough to make sure that most A-cycles under IS see some overflow, but not too large, so that the A-cycles end at the first return to the empty state after overflow. According to our arguments in Section 4, if overflow occurs at time t_1, then the total production by the twisted sources up to time t_1 should be approximately equal to the number of cells required to keep the server busy until time t_1 and fill up the buffer at node q^*, that is, $m^*\tilde{\rho}t_1 \approx B^* + c^*t_1$, where $\tilde{\rho}$ is the average production rate of a twisted source. The additional time t_2 to empty the buffer (with IS turned off) should satisfy $(c^* - m^*\rho)t_2 \approx B^*$. We want (roughly) $t_0 \le t_1 + t_2$, i.e.,

$$t_0 \le \frac{B^*}{m^*\tilde{\rho} - c^*} + \frac{B^*}{c^* - m^*\rho}.$$

We suggest taking t_0 somewhere between 20% and 50% of the value of that upper bound. In our experiments, this always gave efficiency improvement. Since the variance associated with the IS cycles is the dominant term in the variance of $\hat{\mu}$, a good strategy is to choose t_0 just large enough so that most of the IS cycles fill up the buffer. Taking t_0 too large (close to $t_1 + t_2$) is not a good idea because it makes us spend too much time on the no-IS cycles without bringing much additional variance reduction. Beyond a certain point, increasing t_0 eventually *decreases* the efficiency.

Example 8. We used the same data as in Example 5 (for $\ell^* = 4$), with $B_4 = 256$ and $C' = 160\,000$, with IS. For $\theta = \theta^*$, we have $t_1 \approx 95$ and $t_2 \approx 300$. For $\theta = 0.80\,\theta^*$, we have a total arrival rate of 1.82 with IS, which give $t_1 \approx 312$ and $t_2 \approx 300$. Table 10 give the results. With $\theta = \theta^*$, raising t_0 from 0 to 75 increases the (empirical) efficiency approximately by a factor of 4. With $\theta = 0.8\,\theta^*$, raising t_0 from 0 to 150 improves the (empirical) efficiency by a factor of more than 10. This gain is related to the rapid increase of $\bar{X}$, which decreases $\hat{\sigma}^2$ (see Eq. (12)), when t_0 is small. We made 2 additional replications of this experiment and the results were similar, although the empirical efficiency for $\theta = \theta^*$ and $t_0 = 0$ was 0.02 and 0.04, which suggests that the factor of efficiency improvement from this setup to $\theta = 0.8\,\theta^*$ and $t_0 = 150$ is more around 20 to 30 instead of 10. The variance estimator is also (empirically) much less noisy with $(\theta/\theta^*,\, t_0) = (0.8, 150)$.

Table 10. Imposing Lower Bounds on the A-cycle lengths, for $\ell^* = 4$

t_0	$\hat{\mu}$	$\hat{\sigma}^2$	$\hat{\delta}$	$\bar{X}$	Var($\bar{X}$)	CPU	eff.
			$\theta = \theta^*$				
0	4.2E-5	4.2E-11	39.2 %	0.2	0.03	404	0.10
50	4.8E-5	8.8E-13	5.0 %	8.1	1.25	7089	0.37
100	4.8E-5	5.0E-13	3.8 %	15.8	2.43	10523	0.43
150	5.0E-5	6.7E-13	4.2 %	23.1	2.90	12501	0.29
200	4.9E-5	7.6E-13	4.5 %	30.7	3.25	13890	0.23
250	5.0E-5	3.0E-12	8.9 %	38.0	5.45	14933	0.06
			$\theta = 0.8\,\theta^*$				
0	5.1E-5	8.2E-11	45.3 %	0.2	0.03	399	0.08
50	4.8E-5	6.6E-13	4.4 %	8.1	1.24	4646	0.75
100	4.8E-5	3.0E-13	2.9 %	15.7	2.42	7886	0.98
150	4.8E-5	1.9E-13	2.4 %	23.0	2.90	10468	1.13
200	4.8E-5	2.5E-13	2.6 %	30.7	3.24	12645	0.75
250	4.8E-5	2.2E-13	2.5 %	37.9	5.45	14439	0.73

Example 9. Let $\ell^* = 3$, $B_1 = B_2 = B_3 = 256$, $m_0 = 2$, $m_1 = 3$, $m_2 = 10$, $c_1 = 1$, $c_2 = c_3 = 2$, $\rho = 1/21$ and $\kappa_1 = 50$. Six sources feed the target node q^*, as in Example 3, which gives an average arrival rate of $6/21 \approx 0.286$ to that

node. We run simulations for different values of t_0 both with the $\tilde{r}_{ij}$ associated to $\theta^* = 2.73 \times 10^{-2}$ (with $\tilde{\rho} = 5/6$, a total arrival rate of 5.00, $t_1 \approx 85$, and $t_2 \approx 150$) and $0.80\,\theta^* = 2.18 \times 10^{-2}$ (with $\tilde{\rho} \approx 0.54$, a total arrival rate of 3.26, $t_1 \approx 200$, and $t_2 \approx 150$). Using $t_0 = 100$ together with $\theta = 0.8\theta^*$ gave the best empirical efficiency in this case, about 20 times the empirical efficiency observed with $t_0 = 0$ and $\theta = \theta^*$.

6.3 Stopping IS Earlier

Suppose that $\ell^* = 4$ and that we use IS. When the target buffer at q^* overflows and IS is turned off, there may be several cells already in the network at previous levels, and this may produce more overflow than necessary. Because of that, it could make sense to turn off IS earlier, e.g., when the total number of cells in buffer q^* or at previous nodes but on their way to q^*, reaches some threshold N_0. Beck et al. [1999] and Dabrowski et al. [1998] use this criterion for turning off IS, with $N_0 = B^*$. Our experiments with this idea showed no significant improvement compared with the method which turns off IS when q^* overflows. With $N_0 \leq B^*$, this idea seems to *reduce* the efficiency instead. Here is a typical illustration.

EXAMPLE 10. Let $\ell^* = 4$, $B_1 = B_2 = B_3 = B_4 = 512$, $m_0 = 2$, $m_1 = 5$, $m_2 = 5$, $c_1 = c_4 = 1$, $c_2 = c_3 = 5$, $\rho = 1/5$, and $\kappa_1 = 50$. Two sources feed the node q^*, which gives an arrival rate at q^* of $2/5 = 0.4$. When IS is applied the arrival rate increases to 1.5887. These 2 hot sources feed different nodes at level 2. In Table 11, $\bar{Y}$ is the average number of cell losses per cycle with IS and $N_0 = \infty$ corresponds to turning off IS when q^* overflows. Taking N_0 between 520 and 600 appears to be about as good as our usual method, but $N_0 \leq 510$ is definitely worse.

Table 11. Different stopping criteria for IS

N_0	$\hat{\mu}$	$\hat{\sigma}^2$	$\hat{\delta}$	$\bar{Y}$	Var($\bar{Y}$)	CPU	eff. ($\times 10^{-3}$)
500	4.72E-9	6.7E-18	19.8%	1.51	0.188	3308	1.1
510	4.86E-9	1.5E-17	29.2%	1.97	0.148	3354	0.5
520	4.36E-9	1.8E-18	11.3%	2.77	0.322	3397	4.0
540	4.60E-9	2.1E-18	11.3%	4.37	0.515	3508	3.5
560	4.68E-9	1.4E-18	9.3%	6.24	0.610	3659	4.8
580	4.76E-9	1.4E-18	8.9%	8.36	0.935	3603	5.1
∞	4.71E-9	1.4E-18	9.1%	29.5	3.947	3959	4.5

6.4 Retroactive Manipulations to Control the Overflow

The criterion for turning off IS earlier, considered in the previous subsection, is rather blind. Remember that all the randomness in our model is in the state transitions of the sources. It is therefore possible, in principle, to compute at any given point τ in time whether or not there will be overflow at q^* caused only by the cells generated so far, and turn off IS as soon as this happens (this is a stopping time with respect to the filtration generated by the trajectory of $\mathcal{S}(\cdot)$). In this way, IS is turned off *before* the target buffer fills up, but only when overflow is *guaranteed* to occur. In practice, this can be implemented by actually running the simulation until there is overflow, and then turning off IS *retroactively* right after the time τ

when all the cells having reached q^* when the first cell overflows (at time $\tau + \epsilon$, say) were already produced by a source. (This τ remains a stopping time in our model; retroaction would be used only to avoid too much computation at each time step.) This is complicated to implement and implies significant overhead. Despite spending a lot of time on experimenting with this idea, we were unsuccessful in improving the efficiency with it.

6.5 Combining IS with Indirect Estimation

Srikant and Whitt [1999] proposed the following indirect estimator of the CLR. (This approach was presented by Ward Whitt during the keynote address of the 1997 Winter Simulation Conference.) The CLR at node q^* satisfies

$$\mu = 1 - \lambda/\lambda_0 = 1 - \gamma c^*/\lambda_0 \tag{13}$$

where $\lambda_0 = m_0\kappa_1/(\kappa_0 + \kappa_1)$ is the total (average) production rate of the m_0 sources feeding node q^*, λ is the (average) output rate from node q^*, $1/c^*$ is the service time at node q^*, and γ is the steady-state fraction of time where the server is busy at node q^*. The second equality follows from the conservation equation $\gamma = \lambda/c^*$. Using (13), μ can be estimated indirectly by estimating γ. Srikant and Whitt [1999] showed that the indirect estimator brings substantial variance reduction in heavy traffic situations, especially for queues with several servers and random service times, but not in light traffic. In our context, the traffic at q^* is light, but becomes heavy when IS is applied, so it was not clear to us *a priori* if the indirect estimator combined with IS could help.

The results of our extensive numerical experiments can be summarized as follows. For a single queue with several servers, without IS, the indirect estimator reduces the variance by large factors when the total arrival rate exceeds the service capacity, and increases the variance by large factors when the total arrival rate is much less than the service capacity. This is true even for constant service times and single-server queues, but less servers or less variability in the service times favors the direct estimator. A larger buffer at q^* tends to accentuate the factor of variance reduction or variance increase. When the indirect estimator was combined with IS, we observed a variance increase instead of a variance reduction, even if the total arrival rate after IS was larger than the service rate. An intuitive explanation seems to be that because IS is turned off as soon as the buffer overflows, the condition favoring the indirect estimator (sustained overloading at q^*) does not hold for a large enough fraction of the time.

7. FUNCTIONAL ESTIMATION

So far we have considered the problem of estimating the CLR for fixed values of the model parameters. But in real life one is often interested in a wide range of values of the r_{ij}'s and of the buffer sizes. We now examine how the CLR can be estimated in functional form, as a function of the matrix R, from a single simulation, and also as a function of B^* by re-using certain portions of the simulation.

Let R and $\tilde{R}$ be as before, where $\tilde{R}$ is the twisted version of R determined as in Section 4, but suppose that we now want to estimate the CLR μ for R replaced by $\check{R}$, for several $\check{R}$ in some neighborhood of R, by simulating pairs of A-cycles with $\tilde{R}$ and R only. This can be achieved as follows. One simulates pairs of A-

cycles and computes X_i, Y_i, and the likelihood ratio L_i for each pair just as before. Afterwards, the estimators L_iY_i and X_i of the numerator and the denominator are multiplied by the likelihood ratios

$$L_i'(\check{R}) = \left(\frac{\check{r}_{00}}{r_{00}}\right)^{N_{00}'} \left(\frac{\check{r}_{01}}{r_{01}}\right)^{N_{01}'} \left(\frac{\check{r}_{10}}{r_{10}}\right)^{N_{10}'} \left(\frac{\check{r}_{11}}{r_{11}}\right)^{N_{11}'}$$

and

$$L_i''(\check{R}) = \left(\frac{\check{r}_{00}}{r_{00}}\right)^{N_{00}''} \left(\frac{\check{r}_{01}}{r_{01}}\right)^{N_{01}''} \left(\frac{\check{r}_{10}}{r_{10}}\right)^{N_{10}''} \left(\frac{\check{r}_{11}}{r_{11}}\right)^{N_{11}''},$$

respectively, where N_{kl}' and N_{kl}'' are the total number of transitions of the sources from state k to state l during the A-cycle with IS and without IS, respectively. The functional estimator of μ is then

$$\hat{\mu}(\check{R}) = \frac{\sum_{i=1}^{n} L_i'(\check{R}) L_i Y_i}{\sum_{i=1}^{n} L_i''(\check{R}) X_i}$$

and it can be evaluated *a posteriori* for as many different matrices $\check{R}$ as desired, as long as $\check{R}$ is not too far away from R. The additional overhead during the simulation amounts only to storing the values of N_{kl}' and N_{kl}'', together with those of X_i and L_iY_i, for all the pairs of A-cycles. This type of functional estimator based on a likelihood ratio is discussed in a more general context by Rubinstein and Shapiro [1993] and L'Ecuyer [1993], for example.

EXAMPLE 11. We give an example of functional estimation at level 4. Let $B_1 = B_2 = B_3 = B_4 = 500$, $m_0 = 4$, $m_1 = 6$, $m_2 = 8$, $c_1 = c_4 = 1$, $c_2 = c_3 = 3$, $\kappa_1 = 50$, and $\rho = 1/21$. We assign 4 sources to the node q^* and take $C = 150\,000$. We find $\tilde{R}$ and run the simulation as usual, and then compute two functional estimators. For the first one, κ_1 is fixed and μ is estimated as a function of κ_0, whereas for the second one, $\rho = \kappa_1/(\kappa_1 + \kappa_0)$ is fixed and μ is estimated as a function of κ_1. Tables 12 gives a partial view of the results for the first estimator.

The relative half-widths of pointwise 99% confidence interval, $\hat{\delta}$, remain reasonable for a good range of values of κ_0 and κ_1. If one is interested in a wider region, that region can be partitioned into a few subintervals and a different $\tilde{R}$ can be used for each subinterval.

For the estimation of μ as a function of B^*, one cannot use the likelihood ratio approach, because B^* is not a parameter of a probability distribution in the model. However, observe that when an A-cycle is simulated, the sample path of the system is independent of B^* as long as there is no overflow at q^*. Therefore, when estimating μ for several large values of B^*, the initial part of the simulation (until overflow occurs) does not have to be repeated for each value. One can start with a *single* simulation (or sample path) and create a new subpath (or branch) each time the number of cells at q^* exceeds one of the buffer sizes of interest. If one is interested in N distinct values of B^*, one eventually ends up with N parallel simulations, but a lot of work is saved by starting these parallel simulations only when needed. This type of approach is studied in more generality in L'Ecuyer and Vázquez-Abad [1997]. In our experiments with this method, the savings in CPU time were typically around 50%.

Table 12. Functional estimation at level 4, for fixed κ_1

κ_0	$\hat{\mu}$	$\hat{\sigma}^2$	$\hat{\delta}$
500	2.4E-6	6.5E-14	27%
541	1.6E-7	1.7E-14	22%
588	9.8E-7	4.0E-15	17%
645	6.0E-7	8.8E-16	13%
714	3.5E-7	1.9E-16	10%
800	2.0E-7	4.3E-17	7%
909	1.1E-7	1.0E-17	7%
1050	5.8E-8	2.7E-18	7%
1250	2.8E-8	7.2E-19	8%
1540	1.2E-8	2.1E-19	10%

The development of Section 4 suggests an approximately linear relationship between $\ln \mu$ and B^*, at least asymptotically. Our empirical experiments confirm that the linear model $\ln \mu = \beta_0 + \beta_1 B^*$ fits very well for large enough B^*. We can therefore recommend, for estimating μ as a function of B^*, to perform simulations at 4 or 5 large values of B^* only, and fit a linear model to the observations $(B^*, \ln \hat{\mu})$ by least squares regression.

8. CONCLUSION

We have discussed how to implement IS for estimating the CLR in an ATM switch model, by a direct adaptation of the approach proposed in Chang et al. [1994]. Our extensive empirical experiments with large networks indicate that the method is viable, at least for the type of model and examples that we have considered. IS improves the CLR estimator and also improves, by a larger factor, the corresponding variance estimator (although this variance estimator often remains noisy even with IS, in which case the natural way of improving the reliability of the confidence intervals would be to increase the sample size). We have also explored heuristic refinements of the IS scheme. Some gave improvements with respect to the basic scheme, others did not. Those that improved the efficiency of the CLR estimator, namely taking θ slightly less than θ^* and imposing a lower bound on the A-cycle length, also (empirically) reduced the noise of the variance estimator.

This IS methodology can be adapted to other variants of the model, in addition to those considered in Section 5.5, e.g., by replacing the geometric sojourn time distributions in the ON and OFF states by other distributions with an exponential tail. Statistical studies of traffic traces suggest that heavy-tailed distributions, with infinite variance, are more appropriate, although some people still argue that the exponential-tailed distributions do the job well-enough in many cases. The IS scheme considered in this paper does not appear easily adaptable to heavy-tailed distributions. Finding an effective way of applying IS for such distributions remains a challenging problem.

REFERENCES

Beck, B., Dabrowski, A. R., and McDonald, D. R. 1999. A unified approach to fast teller queues and ATM. *Journal of Applied Probability 31*, 758–787.

BERAN, J., SHERMAN, R., TAQQU, M. S., AND WILLINGER, W. 1995. Long-range dependence in variable-bit-rate video traffic. *IEEE Transactions on Communications 43*, 1566–1579.

BONNEAU, M.-C. 1996. Accelerated simulation of a leaky bucket controller. Master's thesis, Department of Mathematics and Statistics, University of Ottawa.

BRATLEY, P., FOX, B. L., AND SCHRAGE, L. E. 1987. *A Guide to Simulation* (Second ed.). Springer-Verlag, New York.

CHAMPOUX, Y. 1998. Estimation du taux de perte de réseaux ATM via la simulation et le changement de mesure. Master's thesis, Département d'IRO, Université de Montréal.

CHANG, C.-S. 1995. Sample path large deviations and intree networks. *Queueing Systems 20*, 7–36.

CHANG, C. S., HEIDELBERGER, P., JUNEJA, S., AND SHAHABUDDIN, P. 1994. Effective bandwidth and fast simulation of ATM intree networks. *Performance Evaluation 20*, 45–65.

CHANG, C. S., HEIDELBERGER, P., AND SHAHABUDDIN, P. 1995. Fast simulation of packet loss rates in a shared buffer communications switch. *ACM Transactions on Modeling and Computer Simulation 5*, 4, 306–325.

COURCOUBETIS, C. AND WEBER, R. 1996. Buffer overflow asymptotics for a buffer handling many traffic sources. *Journal of Applied Probability 33*, 3, 886–903.

DABROWSKI, A. R., LAMOTHE, G., AND MCDONALD, D. R. 1999. Accelerated simulation of ATM switching fabrics. In *Conference Proceedings of the Fields Institute Workshop on Monte Carlo Methods* (1999), pp. 193–206. AMS.

DE VECIANA, G., COURCOUBETIS, C., AND WALRAND, J. 1994. Decoupling bandwidths for networks: A decomposition approach to resource management. In *IEEE Infocom'94 proceedings*, Volume 2 (Los Alamitos, CA, 1994), pp. 466–473. IEEE Computer Society Press.

FALKNER, M., DEVETSIKIOTIS, M., AND LAMBADARIS, I. 1999. Fast simulation of networks of queues with effective and decoupling bandwidths. *ACM Transactions on Modeling and Computer Simulation 9*, 1, 45–58.

FISHMAN, G. S. 1996. *Monte Carlo: Concepts, Algorithms, and Applications*. Springer Series in Operations Research. Springer-Verlag, New York.

GIROUX, N. AND GANTI, S. 1999. *Quality Service in ATM Networks*. Prentice Hall PTR, Upper Saddle River, NJ.

GLASSERMAN, P., HEIDELBERGER, P., SHAHABUDDIN, P., AND ZAJIC, T. 1999. Multilevel splitting for estimating rare event probabilities. *Operations Research 47*, 4, 585–600.

GLYNN, P. W. 1994. Efficiency improvement techniques. *Annals of Operations Research 53*, 175–197.

GLYNN, P. W. AND IGLEHART, D. L. 1989. Importance sampling for stochastic simulations. *Management Science 35*, 1367–1392.

HEEGAARD, P. E. 1998. *Efficient Simulation of Network Performance by Importance Sampling*. Ph. D. thesis, Norwegian University of Science and Technology.

HEIDELBERGER, P. 1995. Fast simulation of rare events in queueing and reliability models. *ACM Transactions on Modeling and Computer Simulation 5*, 1, 43–85.

HEYMAN, D. AND LAKSHMAN, T. 1996. What are the implications of long-range dependence for VBR video traffic engineering. *IEEE/ACM Transactions on Networking 4*, 301–317.

KRUNZ, M. M. AND MAKOWSKI, A. M. 1998. Modeling video traffic using $M/G/\infty$ input processes: A compromise between Markovian and LRD models. *IEEE Journal on Selected Areas in Communications 16*, 5, 733–748.

LAW, A. M. AND KELTON, W. D. 2000. *Simulation Modeling and Analysis* (Third ed.). McGraw-Hill, New York.

L'ECUYER, P. 1993. Two approaches for estimating the gradient in a functional form. In G. W. EVANS, M. MOLLAGHASEMI, E. C. RUSSELL, AND W. E. BILES Eds., *Proceedings of the 1993 Winter Simulation Conference* (Piscataway, NJ, 1993), pp. 338–346. IEEE Press.

L'ECUYER, P. 1994. Efficiency improvement via variance reduction. In *Proceedings of the 1994 Winter Simulation Conference* (1994), pp. 122–132. IEEE Press.

L'ECUYER, P. AND CHAMPOUX, Y. 1996. Importance sampling for large ATM-type queueing networks. In *Proceedings of the 1996 Winter Simulation Conference* (1996), pp. 309–316.

IEEE Press.

L'Ecuyer, P. and Vázquez-Abad, F. 1997. Functional estimation with respect to a threshold parameter via dynamic split-and-merge. *Discrete Event Dynamic Systems: Theory and Applications 7*, 1, 69–92.

Leland, W. E., Taqqu, M. S., Willinger, W., and Wilson, D. V. 1994. On the self-similar nature of ethernet traffic (extended version). *IEEE/ACM Transactions on Networking 2*, 1, 1–15.

Neame, T. D., Zukerman, M., and Addie, R. G. 1999. Application of the M/Pareto process to modeling broadband traffic streams. In *Proceedings of the IEEE ICON'99* (September 1999), pp. 53–58.

Nicola, V. F., Shahabuddin, P., Heidelberger, P., and Glynn, P. W. 1993. Fast simulation of steady-state availability in non-markovian highly dependable systems. In *Proceedings of the 23rd International Symposium on Fault-Tolerant Computing* (1993), pp. 38–47. IEEE Computer Society Press.

Parekh, S. and Walrand, J. 1989. A quick simulation method for excessive backlogs in networks of queues. *IEEE Transactions on Automatic Control AC-34*, 54–56.

Rubinstein, R. Y. and Shapiro, A. 1993. *Discrete Event Systems: Sensitivity Analysis and Stochastic Optimization by the Score Function Method*. Wiley, New York.

Sadowsky, J. S. 1991. Large deviations and efficient simulation of excessive backlogs in a $GI/G/m$ queue. *IEEE Transactions on Automatic Control AC-36*, 1383–1394.

Sadowsky, J. S. 1993. On the optimality and stability of exponential twisting in Monte Carlo estimation. *IEEE Transactions on Information Theory IT-39*, 119–128.

Sadowsky, J. S. 1996. On Monte Carlo estimation of large deviation probabilities. *The Annals of Applied Probability 6*, 2, 399–422.

Shahabuddin, P. 1994. Importance sampling for the simulation of highly reliable markovian systems. *Management Science 40*, 3, 333–352.

Shwartz, A. and Weiss, A. 1995. *Large Deviations for Performance Analysis*. Chapman and Hall, London.

Srikant, R. and Whitt, W. 1999. Variance reduction in simulations of loss models. *Operations Research 47*, 4, 509–523.

Willinger, W., Taqqu, M. S., Sherman, R., and Wilson, D. V. 1995. Self-similarity through high-variability: Statistical analysis of ethernet LAN traffic at the source level. In *Proceedings of SIGCOMM'95* (1995), pp. 100–113. Association for Computing Machinery.