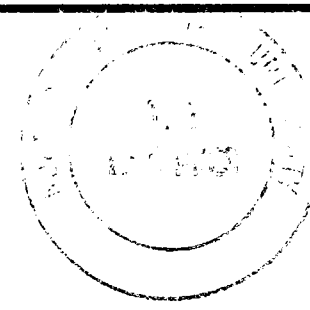
Statistical Optimization of Dynamic Importance Sampling Parameters for Efficient Simulation of Communication Networks



Michael Devetsikiotis

J. Keith Townsend

Center for Communications and Signal Processing
Department of Electrical and Computer Engineering
North Carolina State University

TR-93/9
April 1993

To appear (with revisions) in the *ACM/IEEE Transactions on Networking*

Statistical Optimization of Dynamic Importance Sampling Parameters for Efficient Simulation of Communication Networks

Michael Devetsikiotis, *Student Member IEEE*
J. Keith Townsend, *Member IEEE*
Center for Communications and Signal Processing,
Department of Electrical & Computer Engineering,
North Carolina State University, Raleigh, NC 27695-7914

Abstract

Importance sampling (IS) is recognized as a potentially powerful method for reducing simulation run times when estimating the probabilities of rare events in communication systems using Monte Carlo simulation. Of special interest is the probability of buffer overflow in networks of queues.

When simulating networks of queues, regenerative techniques make the application of IS feasible and efficient. The application of regenerative techniques is also crucial in obtaining correct confidence intervals for the estimates involved. However, using the most favorable IS settings very often makes the length of regeneration cycles infinite or impractically long. We discuss here a methodology that uses IS dynamically within each regeneration cycle, in order to drive the system back to the regeneration state, after an accurate estimate has been obtained.

To obtain large speed-up factors in simulation run time using IS, the modification, or bias of the underlying probability measures must be carefully chosen. Analytically or numerically minimizing the variance of the IS estimator with respect to the biasing parameters or finding the optimal exponential change of measure is only possible under certain conditions. We extend in this paper a technique we developed for finding near-optimal biasing parameters for link simulations to discrete-event simulations of queueing systems, especially in the case of complex systems with *bursty* arrival processes. We also present a methodology for simulating realistic systems which optimizes IS parameter settings using the mean field annealing (MFA) optimization algorithm in conjunction with statistical estimates of the IS estimator variance.

We demonstrate the combination of these techniques by evaluating blocking probabilities for the M/M/1/K, M/D/1/K, GI/D/1/K, Geo/Geo/1/K, and IBP/Geo/1/K queues, a 16×16 synchronous Clos ATM switch, and a 4×4 ATM switch with priority and push-out. Run time speed-up factors of two to eleven orders of magnitude over conventional Monte Carlo are obtained for these examples.

1. This work was supported by the Center for Communications & Signal Processing as a core project.

2. Portions of this paper have been presented at the 30th ACM Annual Southeast Conference, Raleigh, NC, April 1992, and at the IEEE International Conference on Communications, ICC '92, Chicago, June 1992.

1 Introduction

A significant problem when using Monte Carlo (MC) simulation for the performance analysis of communication networks is the large run times required to obtain the desired results with acceptable accuracy. Under proper conditions, Importance Sampling (IS) [1] is a technique that can speed up simulations involving rare events, of both physical layer (link) and network (queueing) systems [2, 3, 4, 5, 6, 7, 8]. Such an event of special interest is the event of a buffer overflow in networks of queues.

Regenerative techniques make the application of IS feasible and efficient [5]. An important issue we address in this paper is that *static* IS parameter settings and regenerative simulation are in conflict — near-optimal but static IS parameter settings typically result in impractically long regenerative cycles. Thus, *dynamic* IS techniques are required [9, 10]. The idea behind *dynamic* IS is to use initially, in each regeneration cycle, the IS settings that will lead to an accurate estimate with maximum efficiency and then *change* IS values during the simulation so that the system will be driven to regeneration as quickly as possible. Thus the benefits of optimal IS and of short regeneration periods are achieved simultaneously.

In contrast, when IS is used in the customary, static way, regeneration cycles will most likely be impractically, or even infinitely long. Using static IS, the only techniques to circumvent this are either to *force* regeneration at chosen instants — which may not always be theoretically justifiable, or choose IS parameter values under the constraint that regeneration cycles be of manageable length — which may decrease the efficiency dramatically.

To obtain large speed-up factors in simulation run time using IS, the modification, or bias of the underlying probability measures must be carefully chosen, otherwise the run times may increase. Most promising IS biasing schemes are parametric. Analytically minimizing the variance of the importance sampling estimator with respect to the biasing parameters [3, 11], or analytically finding the optimal exponential change of measure [4, 6, 8], has typically yielded results for systems which could either be solved analytically or utilized restrictive assumptions (e.g., Poisson arrivals or independent inter-arrival times). These approaches utilize exact or approximate analytical knowledge of the variance expression or the "large deviation" rates which for many realistic systems is not available. As a result, efficient simulation methodologies have not yet been proposed for B-ISDN systems, which are characterized by correlated, bursty arrivals.

To overcome these difficulties, we have previously presented a technique for finding near-optimal biasing parameter values, based on repetitive, short simulation runs and *statistical measures of performance*, which were statistical *estimates* of the estimator variance [12].

For the general problem of optimizing IS parameter values using statistical measures of performance, we are faced with the task of minimizing a *stochastic* cost function over a parameter space of higher dimensionality. Conventional optimization techniques (e.g., gradient descent, Fletcher-Powell, Newton) do not work well when applied to noisy cost functions. MFA [13] is a deterministic approximation to the *simulated annealing* (SA) optimization algorithm [14]. MFA works well for noisy objective functions of moderate or high dimensionality. A brief overview of SA and MFA are given in Section 4.

There are two significant contributions in this paper: First, in Section 3 we present a

method that uses IS *dynamically* in order to allow maximum improvement while still maintaining an efficient regenerative evolution of the system. Three major advantages of using regenerative simulations are: Overcoming the deleterious effects of system memory on the efficiency of IS, no need for a warm-up period, and improved accuracy of confidence interval calculations [15].

Second, we extend our method of estimating optimal IS biasing parameter values presented in [16, 12] to queueing system simulation. In fact, we combine statistical measures of performance and our underestimation theory from [12] with MFA to obtain near-optimal sets of IS parameter values in cases where analytical and numerical methods are intractable, as discussed in Section 4. We use this technique to evaluate very low blocking probabilities for seven finite queueing systems, M/M/1/K, M/D/1/K, Geo/Geo/1/K, GI/D/1/K, IBP/Geo/1/K, a 16×16 synchronous Clos ATM switch, and a 4×4 ATM switch with priority and push-out. The last five systems are characterized by bursty, correlated arrivals, as discussed in Sections 5 and 6.

2 Formulation

2.1 MC Estimates

We will restrict our attention to formulations based on *discrete-time Markov chains.* Even when the actual process under study is a continuous-time Markov chain, simulation of the embedded, discrete-time Markov chain leads to lower estimator variance (see [5, 9] and references within). Furthermore, any discrete-event system simulation can be modeled as a generalized semi-Markov process (GSMP) [5]. Finally, simulations involving simple i.i.d. observations, e.g., BER estimation for communication links, can be thought of special cases of discrete-time Markov chains.

Using the formulation of [9], let $\{\mathbf{X}_i\}_{i\geq 0}$ be the discrete-time Markov chain with finite state space $\mathcal{E}$ and transition matrix $\mathbf{P}$. Assume that $\{\mathbf{X}_i\}_{i\geq 0}$ has a steady-state distribution, and converges in distribution to $\mathbf{X}$. The goal is to estimate the expectation $E[f(\mathbf{X})]$ of some function $f(\mathbf{X}) = h(\mathbf{X})/g(\mathbf{X})$. The expectation of f can be estimated as

$$\widehat{E}[f] = \frac{\sum_{i=0}^{N-1} h(\mathbf{X}_i)}{\sum_{i=0}^{N-1} g(\mathbf{X}_i)} \tag{1}$$

where $h(\mathbf{X}_i)$, $g(\mathbf{X}_i)$, $i = 0, \ldots, N-1$ are observations of h and g obtained during a simulation run. Although this estimator is consistent, it is also, in general, biased because of the strong correlation of h's and g's to the initial state. In order to obtain i.i.d. observations and hence, correct confidence intervals, regenerative techniques [15] exploit the fact that there exists a state that is visited infinitely often, such that the process starts afresh probabilistically each time this state is visited. Let $\mathbf{r}$ be a such a regeneration state, and $\mathbf{s}$ denote a sample path in the evolution of the system under study. Let $H(\mathbf{s}) = \sum_{i=0}^{\tau_1 - 1} h(\mathbf{X}_i)$ and $G(\mathbf{s}) = \sum_{i=0}^{\tau_1 - 1} g(\mathbf{X}_i)$, where $\mathbf{X}_0 = \mathbf{r}$, and τ_1 is the first time i greater than zero that $\mathbf{X}_i = \mathbf{r}$. Let $E_P[G(\mathbf{s})]$ denote the expectation of $G(\mathbf{s})$ with respect to the probability measure $P(\mathbf{s})$. Then, the expectation

of f above can be written as

$$E[f] = \frac{E_P[H(\mathbf{s})]}{E_P[G(\mathbf{s})]} \tag{2}$$

This leads to the MC estimate

$$\widehat{E}[f] = \frac{1/N \sum_{k=1}^{N} H_k(\mathbf{s})}{1/M \sum_{k=1}^{M} G_k(\mathbf{s})} \tag{3}$$

where $H_k(\mathbf{s})$ and $G_k(\mathbf{s})$ are i.i.d. observations of $H(\mathbf{s})$ and $G(\mathbf{s})$, respectively. This estimator is still biased but asymptotically consistent, with variance $\sigma^2_{MC}(P)$, and can be used to derive asymptotically correct confidence intervals [5, 15].

2.2 Efficient Simulation Using IS

Conventional MC simulation techniques require extremely long simulation runs when used to estimate the steady-state probability of rare events. Importance Sampling (IS) has been proposed [1] as a variance reduction technique. Let $\mathbf{P}^*$ be an alternative, *sampling* transition matrix, with $P^*(\mathbf{s})$ the induced probability measure. IS is based on the observation that $E_P[G(\mathbf{s})] = E_{P^*}[G(\mathbf{s})\, L^*(\mathbf{s})]$, where $L^*(\mathbf{s}) = P(\mathbf{s})/P^*(\mathbf{s})$, and provided that $P^*(\mathbf{s}) \neq 0$ whenever $G(\mathbf{s})\, P(\mathbf{s}) \neq 0$. L^* is a *likelihood ratio* and, in the language of IS, a *weight* function. Clearly, $E[f]$ can then be estimated as

$$\widehat{E_*}[f] = \frac{1/N \sum_{k=1}^{N} H_k(\mathbf{s}) L_k^*(\mathbf{s})}{1/M \sum_{k=1}^{M} G_k(\mathbf{s}) L_k^*(\mathbf{s})} \tag{4}$$

where $\widehat{E_*}[f]$ denotes an estimate of $E[f]$ using IS. Write $H^*(\mathbf{s}) = \sum_{i=0}^{\tau_1 - 1} h(\mathbf{X}_i) L_i^*$ and $G^*(\mathbf{s}) = \sum_{i=0}^{\tau_1 - 1} g(\mathbf{X}_i) L_i^*$. Then, an equivalent [5] IS estimator is

$$\widehat{E_*}[f] = \frac{1/N \sum_{k=1}^{N} H_k^*(\mathbf{s})}{1/M \sum_{k=1}^{M} G_k^*(\mathbf{s})} \tag{5}$$

where $L_{ik}^* = P(\mathbf{X}_{0k}, \ldots, \mathbf{X}_{ik}) / P^*(\mathbf{X}_{0k}, \ldots, \mathbf{X}_{ik})$. Due to the Markov chain assumption, $P(\mathbf{X}_{0k}, \ldots, \mathbf{X}_{ik}) / P^*(\mathbf{X}_{0k}, \ldots, \mathbf{X}_{ik}) = \prod_{j=0}^{i-1} p(\mathbf{X}_{jk}, \mathbf{X}_{j+1,k}) / \prod_{j=0}^{i-1} p^*(\mathbf{X}_{jk}, \mathbf{X}_{j+1,k})$, where $p(\mathbf{X}_j, \mathbf{X}_{j+1})$ are the transition probabilities of the Markov chain. Call the variance of this estimator $\sigma^2_{IS}(P, P^*)$.

In (5) above, the likelihood ratio (or weight) at time i during the simulation depends on all random transitions (e.g., arrivals or service completions) which previously occurred in the same regeneration cycle (RC). Thus, from the IS standpoint, the "memory" of the system is increasing within each RC. An additional motivation to use regeneration techniques is in order to avoid the deleterious effects of large system memory on the efficiency of IS. In fact, as was shown in [5], at least one version of non-regenerative IS breaks down as the length of the simulation approaches infinity.

In (5) it is implied that IS is implemented in a *static* way, where the modified or *biased* transition probabilities p^* do not depend on the state $\mathbf{X}_i$ at time i. As was shown in [9], under

certain conditions for the simulation of Markov chains, the optimal IS is *dynamic*. When using IS dynamically, the modified transition probabilities $p^*(\mathbf{X}_j, \mathbf{X}_{j+1})$ become $p^*_{\mathbf{X}_j \mathbf{X}_{j+1}}(\mathbf{X}_j, \mathbf{X}_{j+1})$, to denote the dependence of the modified probabilities on the specific state transition.

We focus on the problem of estimating the probability that an arriving customer (i.e., cell or packet) will be blocked (lost) because the queue capacity is exceeded. In this case, $g(\mathbf{X}_i) = 1$ if an arrival occurs at time i, and 0 otherwise, and $h(\mathbf{X}_i) = 1$ if a cell arrives and is blocked at time i, and 0 otherwise. Then, $G(\mathrm{s})$ is the number of arrivals in a RC, $H(\mathrm{s})$ is the number of blocked cells in a RC, and $E[f] = \Pr[blocking] = E_P[H(\mathrm{s})]/E_P[G(\mathrm{s})]$. This blocking probability can be estimated by (3). Clearly, for very low blocking probabilities, conventional MC estimation is very inefficient and IS (as in (5)) can be used to improve the statistical accuracy and speed up run times. Note that the denominator in (5) should be estimated conventionally, since it does not involve a rare event [7].

3 A Dynamic IS Methodology

3.1 Motivation

Consider a single-queue, single-server system. Denote the length of the queue by $K > 0$. Let the number of cells in the system, at instant k be denoted by X_k, and assume that a regenerative state $\mathbf{r}$ is chosen such that $X_k = 0$. We distinguish here between two types of regeneration: Type I regeneration occurs when the system revisits $\mathbf{r}$ without ever reaching $\{X_k = 0\}$. Type II regeneration occurs when $\{X_k = 0\}$ is encountered at least once before visiting $\mathbf{r}$ again.

Denote the utilization factor by $\rho = \lambda/\mu$, where $1/\lambda$ is the average inter-arrival time and $1/\mu$ the average service time, and let ρ^* be the utilization factor when IS is used. For the original system (no IS), under light traffic conditions ($\rho \ll 1.0$) and with K large, it is clear that the system will be relatively empty most of the time, regeneration will occur frequently but blocked arrivals will be rare events. That is, type-I regenerations will be frequent but type-II regenerations will be rare, as illustrated in Fig. 1-(a).

A necessary condition for speed-up when using IS is an increased frequency of "important events" (i.e., blocked arrivals). This implies *increasing* the effective arrival rate and *decreasing* the effective service rate of the system. As illustrated in Fig. 1-(b), when IS settings are chosen so that the traffic load is light (say, $\rho < \rho^* < 1.0$), the system will still visit $\{X_k = 0\}$ although with reduced frequency. In this case, the proportion of type-II regenerations will be increased. On the other hand, when IS statically modifies the probability measures so that the system traffic load becomes excessively large (say, $\rho^* > 1.0$) the average length of RC's, which is at least as long as the mean recurrence time T_0 of $\{X_k = 0\}$, grows to an impractical size (Fig. 1-(c)). In this case, nearly all regenerations will be of type II but the duration of the regeneration cycle will become very long. As an example, T_0 would be *infinite* for practically any GI/GI/1 queue with $\rho^* > 1.0$ (unstable system). Furthermore, for the M/M/1/K system $T_0 = O(\rho^{*K})$, which shows the exponential increase of T_0 when $\rho^* > 1.0$. Other queueing systems behave similarly, demonstrating the requirement for low ρ^*'s.

Clearly, unless restrictive assumptions on the traffic type allow regeneration to be *forced*

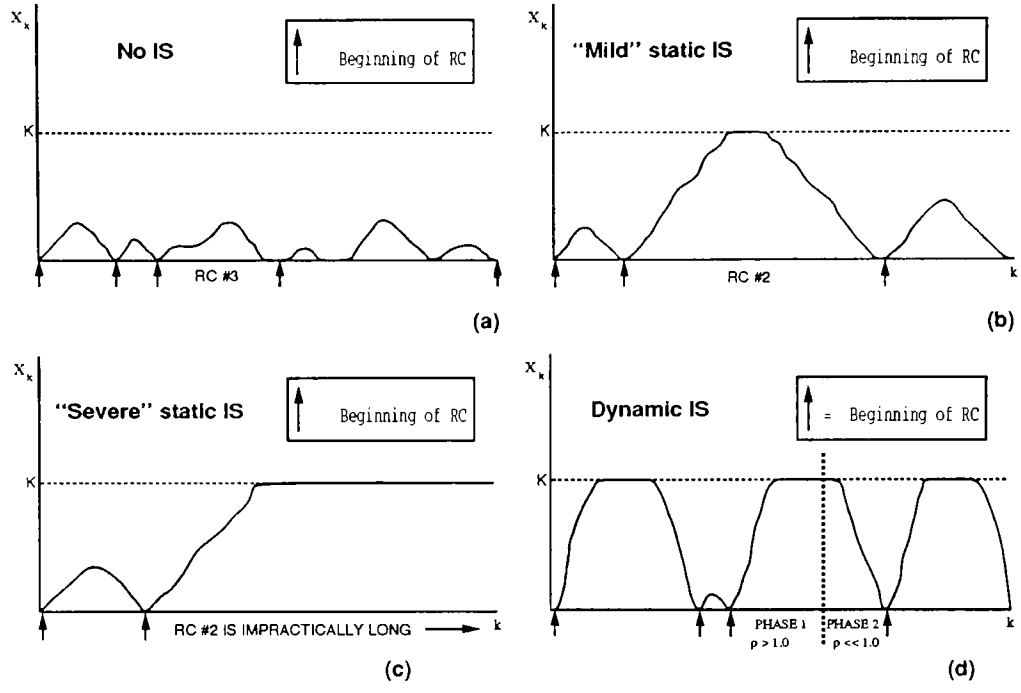


Figure 1: Example system trajectories with no IS, mild static IS, severe static IS, and dynamic IS.

after the first blocked arrival (as in [4, 8]), under static IS we are required to maintain at least moderate load conditions. This can limit dramatically the potential improvement that can be realized with IS; analytical results for simple systems have shown that the optimal biasing typically corresponds to $\rho^* > 1.0$ [4], a fact that is supported by our empirical findings.

3.2 Dynamic Application of IS

To circumvent these difficulties, and efficiently combine IS with regenerative simulation we propose a technique in which IS is implemented *dynamically*. IS parameter settings are varied during each RC initially set to allow important events (i.e., blocked arrivals) to occur frequently, and then changed to facilitate driving the system back to regeneration.

At the beginning of a RC, during the phase we call "Efficient Estimation" or EE phase, a high utilization factor ρ^*_{EE} is used (the search for optimal IS values during this phase is further discussed in Section 4) causing the queue to fill-up quickly, and blocked arrivals to occur frequently. It will be shown below that $\sum_i h(\mathbf{X}_{ik})L^*_{ik}$ converges a.s. within each cycle k (given enough time). Thus, after such convergence has been detected (usually after a finite number of blocked arrivals, less than 50, have been observed), the "Accelerated Regeneration" or AR phase can be entered, where IS settings can be changed to $\rho^*_{AR} < \rho$ to favor the reoccurrence of the regenerative state (Fig. 1-(d)). This second phase regards the achievement of the regenerative state as the important event and modifies the probability measures in order to accelerate the return to such a state, e.g. $\rho^*_{AR} \ll \rho$ and/or $\rho^*_{AR} \ll 1.0$.

3.3 Justification and Discussion

Recall that, under the IS scheme described previously, the empirical estimate of $E_P[H(\mathbf{s})]$ becomes

$$\widehat{E_{P^*}}[H^*(\mathbf{s})] = 1/N \sum_{k=1}^{N} \sum_{i=0}^{\tau_1 - 1} h(\mathbf{X}_{ik}) L_{ik}^* \tag{6}$$

During the EE phase of the RC, the likelihood of the observed system trajectory becomes increasingly smaller with respect to the trajectory under the unmodified measures. Therefore, the weight function decreases within each RC since weights smaller than 1.0 dominate, and the effect of successive important events on the cumulative estimate decreases as well. Eventually, the weight function becomes so small that blocked arrivals contribute insignificant amounts to the summation in (6).

As Glynn and Iglehart show in [5], in both the cases of a discrete-time Markov chain and the previously mentioned GSMP, the cumulative weight (likelihood ratio)

$$L_i^* = \prod_{j=0}^{i-1} p(\mathbf{X}_j, \mathbf{X}_{j+1}) / p^*_{\mathbf{X}_j \mathbf{X}_{j+1}}(\mathbf{X}_j, \mathbf{X}_{j+1})$$

goes to zero (a.s.), as $i \to \infty$. It is straightforward to extend their approach to show that not only L_{ik}^* but also $i^2 L_{ik}^*$ goes to zero (a.s.):

Theorem: Let $\{\mathbf{X}_i\}_{i \geq 0}$ be an irreducible Markov chain, on a finite state space $\mathcal{E}$ under transition matrix $\mathbf{P}$. Let $\mathbf{P}^*$ be the IS transition matrix. Let $L_i^* = P(\mathbf{X}_0, \ldots, \mathbf{X}_i)/P^*(\mathbf{X}_0, \ldots, \mathbf{X}_i) = \prod_{j=0}^{i-1} p(\mathbf{X}_j, \mathbf{X}_{j+1}) / \prod_{j=0}^{i-1} p^*(\mathbf{X}_j, \mathbf{X}_{j+1})$. Then, unless $p(\cdot,\cdot) = p^*(\cdot,\cdot)$, $\lim_{i \to \infty} i^2 \, L_i^* = 0$.

Proof: If $p(\mathbf{u}, \mathbf{v})$ vanishes when $p^*(\mathbf{u}, \mathbf{v})$ is positive, for some state $(\mathbf{u}, \mathbf{v}) \in \mathcal{E}$, then the result is immediate, since the finite state space and the irreducibility of the Markov chain guarantee that such a state $(\mathbf{u}, \mathbf{v})$ will eventually be visited. Otherwise, observe that

$$\lim_{i \to \infty} i^2 \, L_i^* = lim_{i \to \infty} \exp \left[\sum_{j=0}^{i-1} \phi(\mathbf{X}_j, \mathbf{X}_{j+1}) + 2 \log i \right] \tag{7}$$

where $\phi(\mathbf{X}_j, \mathbf{X}_{j+1}) = \log \left(p(\mathbf{X}_j, \mathbf{X}_{j+1}) / p^*(\mathbf{X}_j, \mathbf{X}_{j+1}) \right)$. But, since $2 \log i / i \to 0$,

$$\frac{1}{i} \sum_{j=0}^{i-1} \phi(\mathbf{X}_j, \mathbf{X}_{j+1}) + \frac{2 \log i}{i} \to \sum_{u,v} \pi(\mathbf{u}) p^*(\mathbf{u}, \mathbf{v}) \phi(\mathbf{u}, \mathbf{v}) \qquad P^*\text{-a.s.} \tag{8}$$

where $(\pi(\mathbf{u}) : \mathbf{u} \in \mathcal{E})$ are the stationary probabilities of $p^*(\cdot,\cdot)$. By the strict concavity of $\log(\cdot)$,

$$\sum_{u,v} \pi(\mathbf{u}) p^*(\mathbf{u}, \mathbf{v}) \log \left(\frac{p(\mathbf{u}, \mathbf{v})}{p^*(\mathbf{u}, \mathbf{v})} \right) < \log \left(\sum_{u,v} \pi(\mathbf{u}) p(\mathbf{u}, \mathbf{v}) \right) = 0$$

since $p(\mathbf{u}, \mathbf{v}) \neq p^*(\mathbf{u}, \mathbf{v})$. By (7), $\sum_{j=0}^{i-1} \phi(\mathbf{X}_j, \mathbf{X}_{j+1}) + 2 \log i \to -\infty$ and thus $\lim_{i \to \infty} i^2 \, L_i^* = 0$ by (8). □

It follows from the theorem above, that $\sum_{i=0}^{\infty} L_i^*$ *converges* (P^*-a.s.), and since $0 \leq h(\mathbf{X}_i) \leq 1$, $\sum_{i=0}^{M} h(\mathbf{X}_i)\, L_i^*$ also converges P^*-a.s. as $M \to \infty$. We can therefore choose to switch to the AR phase after the difference between the summation values at two successive blocked arrival instants, M_1, M_2 within the kth RC becomes smaller than a prespecified tolerance ϵ:

$$0 < \sum_{i=0}^{M_2} h(\mathbf{X}_{ik}) L_{ik}^* - \sum_{i=0}^{M_1} h(\mathbf{X}_{ik}) L_{ik}^* = L_{M_2 k}^* < \epsilon$$

In practice, this usually occurred only after 10 or 20 blocked arrivals had been collected. This behavior has been consistently verified in our experimental observations.

4 Optimal IS

4.1 Near-Optimal IS Based on Statistical Estimates

The general, non-parametric, globally optimal IS measure can easily be found but it represents essentially a tautology, since it requires knowledge of the quantity to be estimated, $E[f]$ [17, 5]. Most useful and practical IS schemes are parametric [2, 3, 18].

In the parametric case, the optimal IS problem can be posed as a multidimensional, non-linear optimization problem, where the values of the IS parameters must be set to optimize some measure of performance, usually the estimator variance $\sigma_{IS}^2(P, P^*)$. When an exact closed-form representation of the variance is available, the calculus of variations can be used to minimize the estimator variance [2, 3, 11]. Similarly, analytical methods can be used when optimizing the exponential change of measure [4, 6, 8]. Still, these approaches utilize exact or approximate analytical knowledge of the variance expression or the "large deviation" rates which for most realistic systems is not available.

To overcome this fundamental difficulty we have presented in [12] *statistical measures of performance*, which are statistical estimates of the variability and/or scatter of the MC observations involved. In other words, instead of minimizing the true estimator variance $\sigma_{IS}^2(P, P^*)$ which is usually unknown in closed form, our approach minimizes statistical estimates, of the variance, $\hat{\sigma}_{IS}^2(P, P^*)$, with respect to the IS parameter values:

$$min_{P^*} \widehat{\sigma_{IS}^2}(P, P^*)$$

The estimates we developed can be obtained during each simulation run with minimal computational overhead.

Under certain conditions, the dimensionality of the optimization problem can be reduced to unity. This happens when the search in the parameter space for optimal settings can be confined on a direction (line) or trajectory [11] in the search space. In such cases, our one-dimensional algorithm from [12] has been shown to provide excellent performance in finding near-optimal IS parameter settings. Furthermore, for one- and two-dimensional problems, optimal solutions can also be obtained visually [16].

4.2 Multidimensional Optimization Methods for the IS Problem

For the general problem of optimizing IS using *statistical* measures of performance, we are faced with the task of minimizing a *stochastic* cost function over a parameter space of higher dimensionality. Specifically, let the non-negative cost function $C(\mathbf{a}) = \widehat{\sigma^2_{IS}}(P, P^*)$ be defined as the estimated IS variance given as a function of d IS parameters, $a_1, \ldots, a_d$. In this setting, C is a *random variable* with distribution parameterized by the vector $\mathbf{a} = [a_1, \ldots, a_d]$.

In order to optimize the IS speed-up factor we wish to find $\mathbf{a}_{opt} \in \Re^d$ that minimizes $C(\mathbf{a}) = \widehat{\sigma^2_{IS}}(P, P^*)$. Conventional optimization techniques (e.g., gradient descent, Fletcher-Powell, Newton) do not work well when applied to noisy (random) cost functions, since "uphill" moves are not allowed.

Simulated annealing (SA) [14] is a widely used optimization method, which employs stochastic techniques to avoid becoming trapped in local optima. While changes to $\mathbf{a}$ which decrease the cost function are always accepted, a move which causes an increase of ΔC will be taken with probability $\Pr[\text{uphill move} = \Delta C] = \exp(-\Delta C/T)$ that depends on a parameter T called the *temperature*, thus providing a mechanism for escaping from local minima. Over time, the temperature is lowered from T_{max} to T_{min}, thus lowering the probability of accepting uphill moves and forcing the system into a global optimum.

The *mean field annealing* (MFA) algorithm [13] is a variation of simulated annealing that retains the ability of SA to avoid local minima and arrive at optimal or near-optimal solutions while demonstrating more rapid convergence. In applying MFA, a randomly selected parameter i is stepped through the entire set of M quantized values in the range $(A_{min,i}, A_{max,i})$ and the cost function C_j, $0 \leq j \leq M-1$, is determined at each value. The selected parameter is then set to a weighted average of the quantized values

$$a_i = \frac{\sum_{j=0}^{M-1}(A_{min,i} + jq)\exp(-C_j/T)}{\sum_{j=0}^{M-1}\exp(-C_j/T)} \tag{9}$$

where bins with a larger cost function contribute less to the average. This procedure is performed for every parameter and, as the temperature decreases, each parameter increasingly avoids values with a high cost function.

4.3 Optimization of IS Parameters Using MFA

MFA combines the effectiveness of SA with reduced run times, therefore we select it over SA to minimize the above noisy cost function, $C(\mathbf{a}) = \hat{\sigma}^2_{IS}(P, P^*)$ with respect to $\mathbf{a}$. Our MFA-based algorithm that estimates near-optimal IS parameter settings $a_{1,opt}, \ldots, a_{d,opt}$ is given in Fig. 2.

In the algorithm we also exploit a theoretically justifiable relationship, for small sample sizes, between the IS estimate $\widehat{E_*}[f]$ and the amount of IS bias. As we have proven in [12], for a wide range of biasing schemes, small sample sizes and increasing biasing amounts ("over-biasing"), $\widehat{E_*}[f]$ increasingly *underestimates* the unknown expectation $E[f]$ in a given simulation run with probability asymptotically approaching unity. From an algorithmic standpoint, this allows us to set a threshold for the estimates (based on *a priori* knowledge), such

```
/* Initialize parameters to random values */
for (i ⇐ 1; i ≤ d; i ⇐ i + 1)
    a_i ⇐ random(A_min,i, A_max,i)

/* Anneal (i.e., reduce) T from T_max to T_min using γ < 1 */
for (T ⇐ T_max; T > T_min; T ⇐ γT)

    N_EQ ⇐ E      /* Initialize equilibrium counter */
    /* Repeat until equilibrium is established */
    do until(N_EQ = 0)
        i ⇐ random(1, 2, ..., d)      /* Randomly pick one of the d parameters */
        a_sum ⇐ 0      /* Reset weighted amplitude accumulator */
        p_sum ⇐ 0      /* Reset exponential weight accumulator */

        /* Compute estimated variance C and estimate Ê_P*[f] for each quantized parameter level */
        for (j ⇐ 0; j < M; j ⇐ j + 1)
            a_i ⇐ A_min,i + jq      /* Compute next parameter level */
            (C_j, Ê_P*[f]) ⇐ simulate(a, N)      /* Simulate using parameter vector a and sample
                                                    size N to find new cost function C_j and estimate Ê_P*[f] */
            if(Ê_P*[f] < Th) then C_j = MAX /* If estimate less than threshold,
                                               set cost function to maximum value */
            p_sum ⇐ p_sum + exp(−C_j/T)      /* Accumulate exponentials */
            a_sum ⇐ a_sum + a_i · exp(−C_j/T)      /* Accumulate weighted level */

        a_i ⇐ a_sum/p_sum      /* Update parameter with its new value */
        N_EQ ⇐ N_EQ − 1      /* Decrement equilibrium counter */
```

Figure 2: Pseudo-code for MFA-based algorithm used to minimize $C(\mathbf{a}) = \hat{\sigma}^2_{IS}(P, P^*)$ with respect to the IS biasing parameter vector, $\mathbf{a}$.

that when the estimate is lower than this threshold, the cost function is set to a very large value (MAX in Fig. 2). This practically eliminates the corresponding biasing parameters as possibilities for the optimal settings, thus reducing the search space drastically. *A priori* knowledge in the form of (at least loose) lower bounds on the probability to be estimated is usually available. Note that this type of *a priori* knowledge is very different from the "analytical" type of knowledge required in other IS optimization methods.

For each simulation run (i.e., cost function evaluation) in the algorithm, the sample size N should be made as large as practically possible, since larger sample sizes result in less noisy cost function observations.

5 Calculation of Speed-Up Factors

To estimate the speed-up factor over conventional MC simulation provided by importance sampling we use the following method which is more complete but also more conservative than the one used in [10, 19]: At the chosen (near-optimal) IS parameter settings we perform

N_R independent runs of N_{RC} RC's each. Denote the probability estimate corresponding to the ith run be $\hat{P}_i$, $i = 1, \ldots, N_R$. Our overall probability estimate is then the sample mean

$$\hat{P} = \frac{1}{N_R} \sum_{i=1}^{N_R} \hat{P}_i \tag{10}$$

We estimate the variance of $\hat{P}_i$ by the sample variance

$$\hat{\sigma}^2 = \frac{1}{N_R - 1} \sum_{i=1}^{N_R} (\hat{P}_i - \hat{P})^2 \tag{11}$$

Based on the normal assumption we obtain the 95% confidence interval for $\hat{P}_i$:

$$(\hat{P}_i - 1.96\sqrt{\hat{\sigma}^2}, \hat{P}_i + 1.96\sqrt{\hat{\sigma}^2}) \tag{12}$$

as in [20], where 1.96 is the upper $1 - \alpha/2$ critical point for the standard normal random variable, $\alpha = 0.05$.

Let $N_{RC,MC}$ be the number of conventional Monte Carlo RC's required to achieve the same confidence level as given by (12). Then the speed-up factor R_{IS} in terms of *number of RC's* is the ratio $R_{IS} = N_{RC,MC}/N_{RC}$.

To estimate $N_{RC,MC}$ we first assume that in each conventional Monte Carlo RC there is a constant number of arrivals, equal to the expected number $\bar{A}$ of arrivals per RC. This is an approximation we make to simplify the calculations. Furthermore, we *conservatively* assume that successive customer losses are independent. This means that, in our calculation, $N_{RC,MC}$ conventional Monte Carlo RC's are assumed equivalent to $N_{RC,MC} \times \bar{A}$ i.i.d. observations. We then calculate the $N_{RC,MC}$ required to make the confidence interval in equation (8) of [21] equal to the confidence interval in (12) above.

In reality, successive losses are far from independent, especially when traffic is bursty. Our assumption is conservative because, in general, a greater number of observations is required when observations are dependent. In [21], page 158, an example is given that illustrates how the variance of the estimator increases when observations are correlated. The implication for our calculation is that, in actuality, a greater number of conventional Monte Carlo RC's would have to be run in order to achieve the same accuracy.

Note that this measure of comparison is based on the *number* of RC's, not the actual simulation run time that would have to include the *length* of RC's as well. Assuming the computational effort required to complete the simulation of the i-th RC to be equal to its length D_i in number of arrivals, the total run time required to obtain an estimate based on N RC's is $D = \sum_{i=1}^{N} D_i$. Then, $\bar{D} = \mathrm{E}\{D\} = N\,\mathrm{E}\{D_i\}$. A fair comparison of simulation efficiency can then be based on the *time-reliability product* $\bar{D}\,\sigma_*^2$, where σ_*^2 is the variance of $H_i^*(\mathrm{s})$, the number of (weighted) blocked cells during the i-th RC. In the case of conventional MC, since the queue rarely fills up, RC's are extremely short at the cost of a very large σ_*^2. Under favorable IS settings, the (expected) length of RC's increases but σ_*^2 decreases so that the resulting time-reliability product can be orders of magnitude smaller than that of conventional MC. The choice of ρ_{AR}^* should make $\mathrm{E}\{D_i\}$ as short as possible, without

increasing σ_*^2. Hence, while the purpose of the EE phase is mainly to make σ_*^2 low, the AR phase ensures that regeneration will occur frequently, keeping D_i's short.

When IS is applied in order to increase the frequency of losses, the length of RC's inevitably increases over its value under unbiased conditions. In the following, we denote the factor of increase (in time units) in the expected length of a RC under IS by R_t. Denoting the speed-up in number of RC's by R_{IS}, the net run time improvement over conventional MC simulation is then $R_{net} = R_{IS}/R_t$.

6 Application to Continuous-Time Queueing Systems

We focus here on continuous-time queueing models. In this section we use the dynamic and regenerative IS techniques discussed earlier in order to estimate the average probability of blocked arrival for M/M/1/K, M/D/1/K, and GI/D/1/K systems. IS performance is optimized using the one-dimensional methods mentioned in Section 4.1 [12, 16].

6.1 The M/M/1/K Queue

The M/M/1/K system had an average arrival rate $\lambda = 1.0$, an average service rate $\mu = 1.333$, and a system capacity $K = 101$. The blocking probability could be calculated analytically and was found to be 6.01×10^{-14}. For this system RC's coincided with *busy cycles*. The average number of arrivals per RC was calculated analytically to be $E_P[G(\mathrm{s})] = 4.0$. As discussed earlier, we only needed to use IS to estimate the average number blocked per RC, $E_P[H(\mathrm{s})]$. For this example, $E_P[H(\mathrm{s})] = 2.41 \times 10^{-13}$, and this is the number that we estimated by (6) using our technique.

Under IS, the inter-arrival and service times were still exponential, but the rates λ and μ were independently multiplied by biasing parameters. Using our algorithm in [12], optimal parameter values were estimated to be 0.73 and 1.36 for the interarrival time multiplier and service time multipliers respectively. Shown in Figs. 3 and 4 are cross-sections of a 3-D plot of $\widehat{E_{P^*}}[H^*(\mathrm{s})]$ vs. the IS parameter values, through this optimal point. Each point on these plots represents one simulation run of $N = 100$ RC's. Also shown in these figures are plots of the corresponding variance estimators. As discussed in [12], the algorithm minimizes a cost function which combines the local scatter of the $\widehat{E_{P^*}}[H^*(\mathrm{s})]$ curve and the sample variance for each estimate. The effectiveness of the algorithm is clearly evident in the figures. Our estimated near-optimal IS settings were similar to those analytically calculated in [4] for the M/M/1/∞ queue.

Based on $N_R = 100$ runs with $N_{RC} = 1000$ RC's per run, using the above estimated optimal IS values, we obtained $\widehat{E_{P^*}}[H^*(\mathrm{s})] = 2.41 \times 10^{-13}$, corresponding to a blocking probability of 6.025×10^{-14}. The 95% confidence coefficient for this set of runs was calculated to be 7.44×10^{-16}.

Using the procedure in Section 4 we calculated a speed-up factor of 1.057×10^{12}, i.e., our simulation estimated the average number blocked in a RC, $\widehat{E_{P^*}}[H^*(\mathrm{s})]$, with a factor of 1.057×10^{12} fewer RC's than would have been required by conventional Monte Carlo simulation.

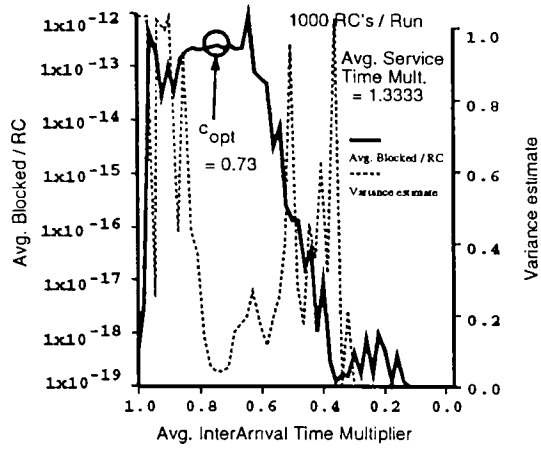


Figure 3: Cross-section of the 3-D plot of $\widehat{E_{P^*}}[H^*(\mathbf{s})]$ overlaid with estimated sample variance, as a function of the interarrival time multiplier at the near-optimal value for the service time multiplier.

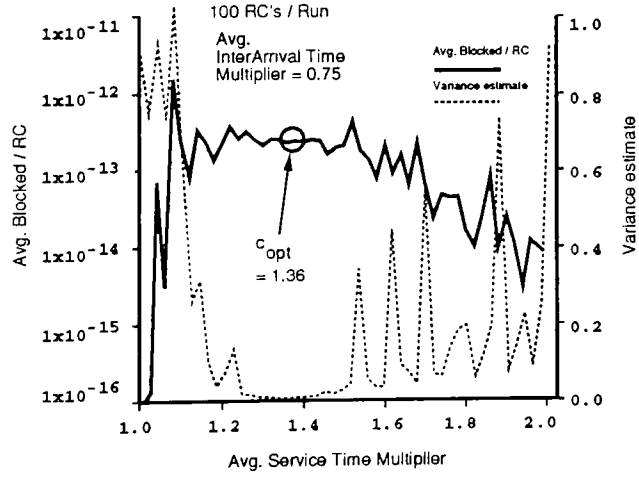


Figure 4: Cross-section of the 3-D plot of $\widehat{E_{P^*}}[H^*(\mathbf{s})]$ overlaid with estimated sample variance, as a function of the service time multiplier at the near-optimal value for the interarrival time multiplier.

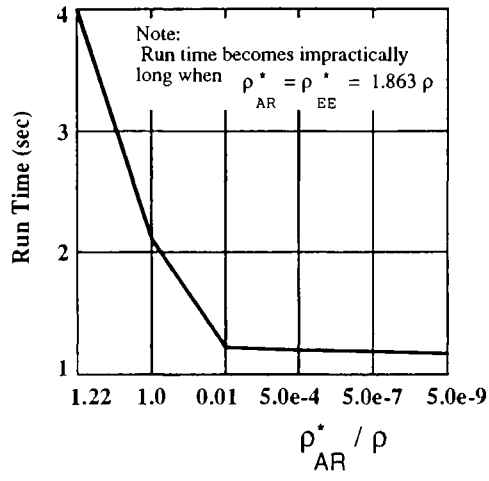


Figure 5: Plot of the average simulation run time in seconds versus ρ^*_{AR} expressed as a fraction of the original utilization ρ. Run times are based on starting with the optimal ρ^*_{EE} and switching to ρ^*_{AR} after 20 lost cells, $N = 100$ RC's per run.

In our example M/M/1/K system, R_t was estimated to be $R_t = 38.97$, leading to $R_{net} = 2.712 \times 10^{10}$.

To see the effect of AR settings on the simulation length, refer to Fig. 5, which corresponds to the same M/M/1/K system above. As ρ^*_{AR} decreases, the RC average length and hence, the total run time, becomes shorter, as would be expected. The value of ρ^*_{AR} shown on the rightmost point is the value used in all simulation results reported in this section. The advantage of dynamic IS is most evident in the case $\rho^*_{AR} = \rho^*_{EE} = 1.863\rho$. In this case, RC's would be impractically long, as discussed earlier.

One other characteristic alluded to earlier was that, as the system trajectory in the kth RC evolves under IS, $\sum_{i=0}^{M} h(\mathrm{X}_{ik})\, L^*_{ik}$ converges a.s. as $M \to \infty$. This can be seen in Fig. 6, which shows $\widehat{E_{P^*}}[H^*(\mathrm{s})]$ for the same M/M/1/K system above as a function of the number of blocked arrivals observed in the RC before switching to the AR phase. After 10 to 20 blocked arrivals have been observed in an RC, the estimate has converged to a desired level of precision.

6.2 The M/D/1/K Queue

The next example is an M/D/1/K queue where $\lambda = 1.0$, the deterministic service rate, was fixed at 1.333, and the system capacity was $K = 59$. Again for this system, RC's coincided with busy cycles. The average number of arrivals per RC was estimated to be $E_P[G(\mathrm{s})] = 4.0$.

Under IS, the inter-arrival times were still exponential, but the rate λ was multiplied by a biasing parameter. A plot of $\widehat{E_{P^*}}[H^*(\mathrm{s})]$ and the sample variance estimate as a function of the interarrival time multiplier is shown in Fig. 7. As before, the optimal IS biasing parameter value was estimated using our algorithm. It is clear from Fig. 7 that the chosen value $c_{opt} = 0.55$ corresponds to the minimum scatter and minimum variance point. Repeating

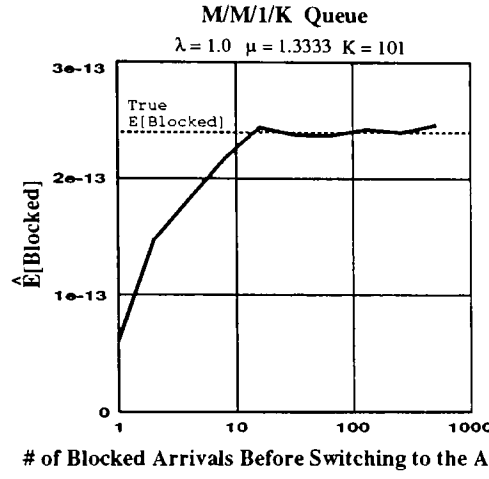


Figure 6: Plot of $\widehat{E_{P^*}}[H^*(\mathbf{s})]$ as a function of the number of blocked arrivals observed before switching to the AR phase.

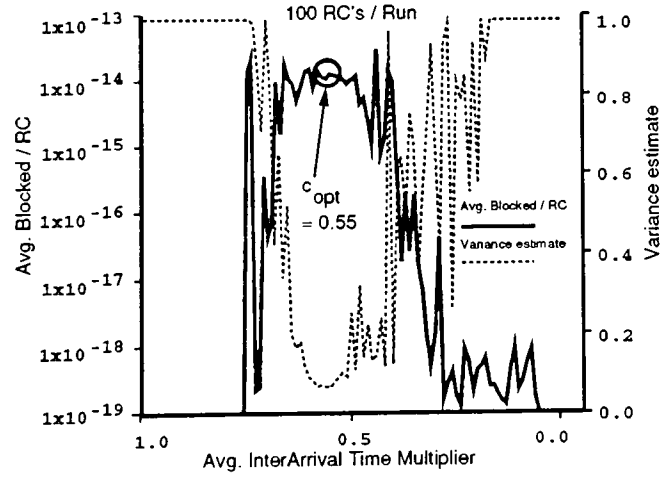


Figure 7: Plot of $\widehat{E_{P^*}}[H^*(\mathbf{s})]$ overlaid with estimated sample variance, as a function of the interarrival time multiplier for an M/D/1/K system.

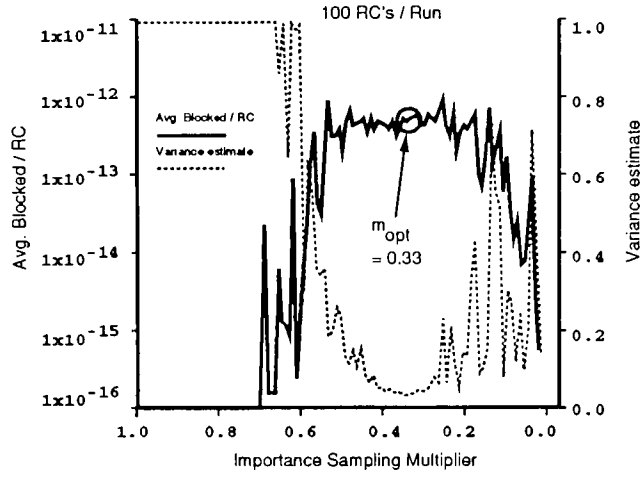


Figure 8: Plot of $\widehat{E_{P^*}}[H^*(\mathbf{s})]$ overlaid with estimated sample variance, as a function of the multiplier m for an GI/D/1/K system.

the same procedure as above in the M/M/1/K case, we obtained an estimate of $\widehat{E_{P^*}}[H^*(\mathbf{s})] = 1.16 \times 10^{-14}$, corresponding to a blocking probability of 2.9×10^{-15}. The 95% confidence coefficient for $N_R = 100$ runs at $N_{RC} = 1000$ RC's per run was 2.88×10^{-17}. From the same set of 100 runs we obtained $R_{IS} = 3.38 \times 10^{13}$, $R_t = 47.34$, and $R_{net} = 7.15 \times 10^{11}$.

6.3 A GI/D/1/K Queue

The last example we present is an GI/D/1/K queue, where Pr[interarrival time $= \alpha_1$] $= p$, and Pr[interarrival time $= \alpha_2$] $= 1 - p$, $0 < p < 1$. In this example, $\alpha_1 = 2.1$, $\alpha_2 = 0.7$, $p = 0.6$, the deterministic service rate is fixed at 1.25, and the system capacity is $K = 19$. RC's coincided with busy cycles. The average number of arrivals per RC was estimated to be $E_P[G(\mathbf{s})] = 2.213$.

Under IS, p was multiplied by m to obtain the biased distribution Pr*[interarrival time $= \alpha_1$] $= p^* = pm$, and Pr*[interarrival time $= \alpha_2$] $= 1 - pm$, $0 < m < 1/p$. A plot of $\widehat{E_{P^*}}[H^*(\mathbf{s})]$ and the sample variance estimate as a function of the multiplier m is shown in Fig. 8. The optimal IS biasing parameter setting was found to be $m_{opt} = 0.33$. At the optimal IS setting we estimated $R_t = 38.13$. Using $N_R = 100$ runs at $N_{RC} = 1000$ RC's per run, the estimate of the average blocked per RC, blocking probability, 95% confidence coefficient, and speed-up factors over conventional MC simulation were estimated to be $\widehat{E_{P^*}}[H^*(\mathbf{s})] = 3.87 \times 10^{-13}$, 1.75×10^{-13}, 1.29×10^{-15}, $R_{IS} = 1.85 \times 10^{12}$, and $R_{net} = 4.85 \times 10^{10}$, respectively.

7 Application to Slotted-Time Queueing Systems

We focus here on slotted-time queueing models, and use (5) to estimate blocking probabilities. We present in this section applications of using MFA, combined with our dynamic, regenerative

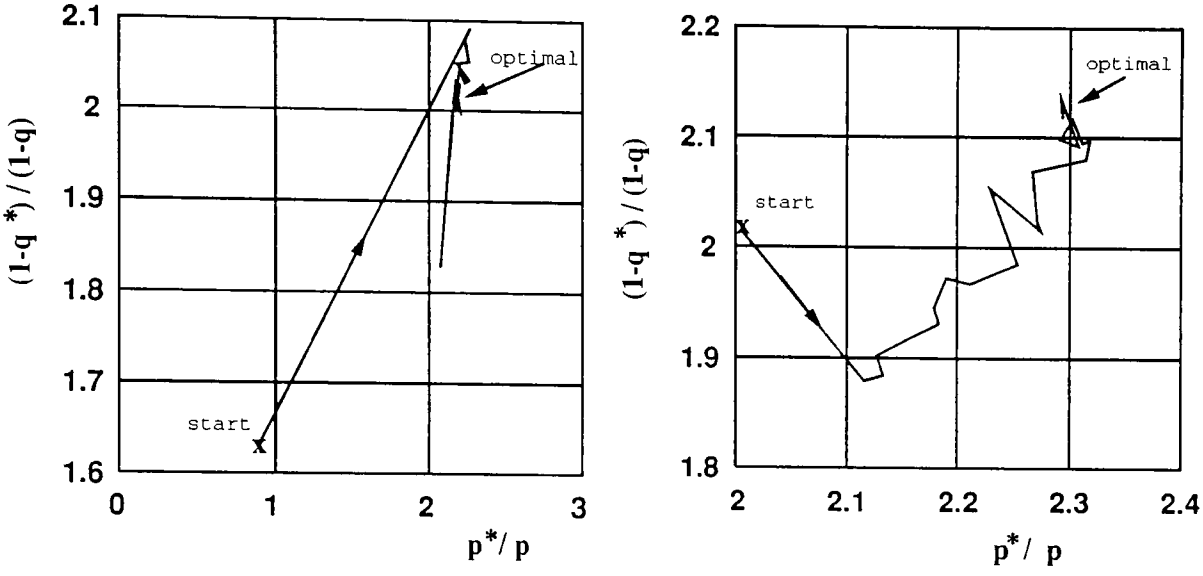


Figure 9: Trajectories of the IS parameters moving towards the near-optimal solution for the Geo/Geo/1/K queue examples.

IS technique, to optimize the IS parameter settings. Four systems are studied, in order of increasing complexity and practical significance:

7.1 Non-bursty System, Geo/Geo/1/K

The Geo/Geo/1/K queue [22] is the simplest finite capacity slotted queueing system, the equivalent of the M/M/1/K queue. Both the arrival process and the service process are *memoryless.* There is a probability p in each slot that a cell will arrive, and a probability q in each slot, when the server is busy, that a cell will depart. Arrivals and service completions are independent. There is a finite capacity of K cells in the system.

Under regenerative IS, we choose the empty state as the regeneration state and the times that a cell arrives to an empty system as the regeneration points. In each RC, we bias initially p and q to p_1^* and q_1^*, until a cell has been blocked, then change IS parameters to p_2^*, q_2^* in order to allow fast regeneration (dynamic IS, [10, 19]).

In our experiments, we set $p_2^* = p$, $q_2^* = q$, and optimized with respect to the settings of p_1^*, q_1^* using MFA. In applying our MFA-based algorithm (Fig. 2) we set $a_1 = p_1^*$, $a_2 = 1 - q_1^*$, $A_{max,1} = 1.0$, $A_{min,1} = 0.8p$, $A_{max,2} = 1.0$, $A_{min,2} = 0.8(1-q)$, $\gamma = 0.8$, $M = 100$, $E = 5$, $N = 100$, $T_{max} = 3.0$, and $T_{min} = 3.0 \times 10^{-2}$. Results were obtained for two queue configurations: $(p = 0.3, q = 0.64853, K = 10)$ and $(p = 0.3, q = 0.64853, K = 20)$. The threshold Th in Fig. 2 was set to 10^{-9} and 10^{-15}, respectively.

Fig. 9 shows an example of the trajectory of the IS parameters moving towards the near-optimal solution for the Geo/Geo/1/K queue. Table 1 summarizes the results, including the near-optimal IS parameters found by MFA, the corresponding estimated blocking probabilities, the estimated confidence intervals and the speed-up factors with respect to conventional MC

Buffer	Pr[loss]	IS values	$\widehat{\Pr}$[loss]	95% Confidence	R_{IS}	R_{net}
$K = 20$	1.121×10^{-13}	2.1684 2.0042	1.161×10^{-13}	$(1.136 \times 10^{-13}$, $1.186 \times 10^{-13})$	5.62×10^{12}	3.88×10^{11}
$K = 10$	2.455×10^{-7}	2.2930 2.1173	2.494×10^{-7}	$(2.427 \times 10^{-7}$, $2.560 \times 10^{-7})$	1.70×10^{6}	2.21×10^{5}

Table 1: Blocking probabilities and speed-up factors using the proposed algorithm for the Geo/Geo/1/K queue, with $p = 0.3$, $q = 0.64853$. For these estimates: $N_R = 100$, $N_{RC} = 100$.

simulation. For these estimates we used $N_R = 100$, $N_{RC} = 100$. We estimated $R_t = 7.69$ when $K = 10$, and $R_t = 14.5$ when $K = 20$. Estimated blocking probabilities are in agreement with the known, analytically calculated probabilities.

7.2 Bursty System, IBP/Geo/1/K

The IBP/Geo/1/K queue is the slotted equivalent of the IPP/M/1/K queue and a special case of the MMBP/Geo/1/K queue. For this queue, although the service process is memoryless, the arrival process is *bursty*. This makes this queue a useful and widely used model for the bursty processes involved in B-ISDN and ATM analyses.

There are two states of the arrival process: active and inactive. In the active state, an arrival can occur with probability α while in the inactive state no arrivals can occur. While the arrival process is in the active state, there is a probability p at each slot that the state will *remain* active and a probability $1 - p$ that it will change to inactive. While the arrival process is in the inactive state, there is a probability q at each slot that the state will remain inactive and a probability $1 - q$ that it will change to active. When the server is busy, there is a probability $1 - \beta$ in each slot that a cell will depart. Arrivals and service completions are independent. There is a finite capacity of K cells in the system. In our experiments, α was assumed to be equal to 1. Let $\tilde{t}$ denote the random interarrival time. The squared coefficient of variation $c^2 = Var(\tilde{t})/[E(\tilde{t})]^2$ of the interarrival times is used to measure the burstiness of the arrival process. Typical values are $c^2 = 1$ corresponding to Poisson arrivals, $c^2 \approx 20$ for voice and c^2 ranging from 10 to 10,000 for video.

Under regenerative IS, we choose the times that a cell arrives to an empty system *and* the arrival process has just changed to active, as the regeneration points. In each RC, we bias initially p, q and β to p_1^*, q_1^* and β_1^*, until a cell has been blocked, then change IS parameters to p_2^*, q_2^* and β_2^* in order to empty the queue and, finally, change to p_3^*, q_3^* and β_3^* in order to allow fast regeneration.

In our experiments, we set $p_2^* = p_3^* = p$, $q_2^* = q_3^* = q$, $\beta_2^* = \beta_3^* = \beta$ and optimized with respect to the settings of p_1^*, q_1^* and β_1^* using MFA. In applying MFA we set $a_1 = p_1^*$, $a_2 = q_1^*$, $a_3 = \beta_1^*$, $A_{max,1} = 1.0$, $A_{min,1} = p$, $A_{max,2} = q$, $A_{min,2} = 0.5q$, $A_{max,3} = 2.0\beta$, $A_{min,3} = \beta$, $\gamma = 0.8$, $M = 100$, $E = 6$, $N = 100$, $T_{max} = 8.0 \times 10^{-2}$, and $T_{min} = 1.0 \times 10^{-3}$. Results were obtained for three queue configurations that corresponded to three different values of c^2, namely 10.0, 20.0, and 30.0: $(p = 0.932075471, q = 0.954716981, \beta = 0.35147, K = 200)$,

c^2	Pr[loss]	IS values	$\widehat{\Pr}$[loss]	95% Confidence	R_{IS}	R_{net}
10.0	7.530×10^{-12}	1.0439 0.9349 1.0926	7.289×10^{-12}	$(7.195 \times 10^{-12}$, $7.384 \times 10^{-12})$	2.66×10^{9}	1.37×10^{8}
20.0	8.301×10^{-7}	1.0237 0.9448 1.0300	8.424×10^{-7}	$(8.251 \times 10^{-7}$, $8.596 \times 10^{-7})$	4.99×10^{3}	4.47×10^{2}

Table 2: Blocking probabilities and speed-up factors using the proposed algorithm for the IBP/Geo/1/K queue, with $\beta = 0.35147$, $K = 200$. For these estimates: $N_R = 100$, $N_{RC} = 500$.

$(p = 0.965048543, q = 0.976699029, \beta = 0.35147, K = 200)$, and $(p = 0.976470588, q = 0.984313725, \beta = 0.35147, K = 200)$. The threshold Th of Fig. 2 was set to 10^{-13}, 10^{-8} and 10^{-6}, respectively.

Table 2 summarizes the results, including the near-optimal IS parameters found by MFA, the corresponding estimated blocking probabilities, the estimated confidence intervals and the speed-up factors with respect to conventional MC simulation. For these estimates we used $N_R = 100$, $N_{RC} = 500$. We estimated $R_t = 19.38$ when $c^2 = 10.0$, and $R_t = 11.16$ when $c^2 = 20.0$. Estimated blocking probabilities are in agreement with the known, analytically calculated probabilities, as illustrated in Fig. 10.

7.3 Realistic System: ATM Switch

The Asynchronous Transfer Mode (ATM) appears to be the evolving standard for broadband ISDN. We consider an ATM switch with buffers at the input ports. The switch fabric is a Clos three-stage interconnection network. For a detailed description of the switch and an approximate model for its operation see [23] and references within. Such an $N_L \times N_L$ Clos cell switch is shown in Fig. 11.

The bursty arrival process to each input port is modeled as a discrete time Interrupted Bernoulli Process (IBP), described in the previous segment. For this type of switch an approximation algorithm was constructed in [23].

Under regenerative/dynamic IS, we choose the times that a cell arrives to an empty buffer *and* the arrival process has just changed to active, as the regeneration points. Note that this is an approximation, required here due to the impractical length of the "true" regeneration cycles based on *all* the buffers being empty, and *all* the arrival processes just changing to active and producing an arriving cell. Our approximation is supported by the fact that these approximate RC's (ARC's) are long enough to allow us to consider events across cycles "practically independent". It is also supported by the extensive experimental correlation analysis we conducted, i.e., estimates of the correlation existing ARC's were consistent with our assumption of independence.

IS biasing was done in a way similar to the IBP/Geo/1/K case above, except that, for this

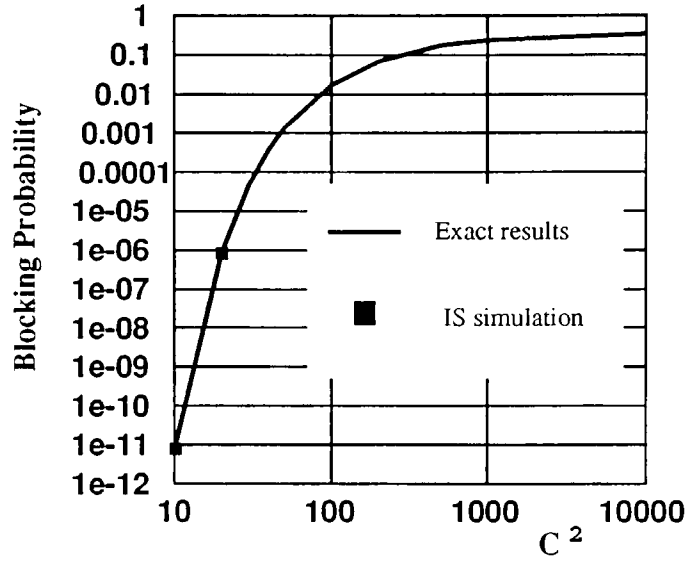


Figure 10: Estimated blocking probabilities and analytically calculated probabilities vs. the squared coefficient of variation, c^2, for the IBP/Geo/1/K queue.

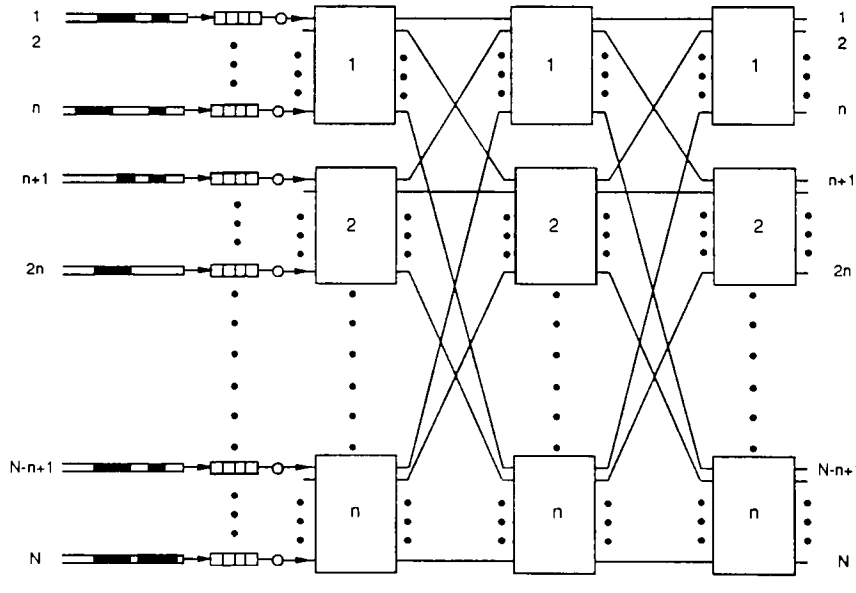


Figure 11: A three-stage, synchronous $N_L \times N_L$ Clos ATM switch [23, 24].

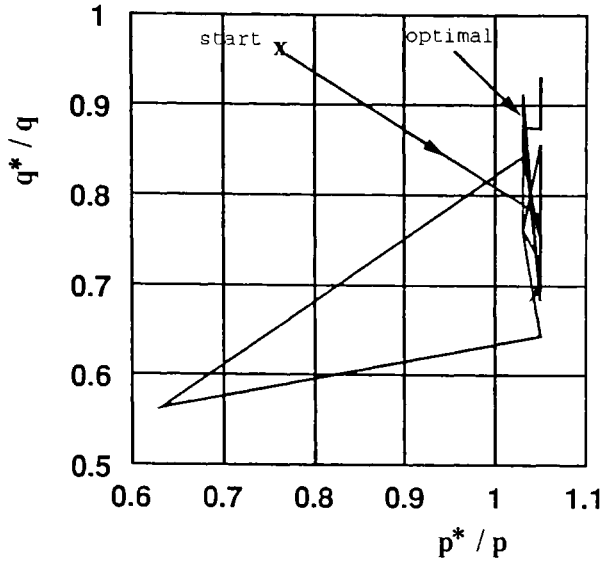


Figure 12: Trajectory of the IS parameters moving towards the near-optimal solution for the ATM switch model.

model, there is no explicit service probability $1-\beta$ available to be modified. In each ARC, we bias initially p and q to p_1^* and q_1^*, until the weight function (likelihood L^*) decreases to a prespecified minimum, then change IS parameters to p_2^* and q_2^* in order to empty the queue and, finally, change to p_3^* and q_3^* in order to allow fast regeneration.

In our experiments, we set $p_2^* = 0.3p$, $p_3^* = p$, $q_2^* = 1.045q$, $q_3^* = q$, and optimized with respect to the settings of p_1^* and q_1^* using MFA. The number of input lines was $N_L = 16$, with input buffers of length $K = 200$ each. In applying MFA we set $a_1 = p_1^*$, $a_2 = q_1^*$, $A_{max,1} = 1.07p$, $A_{min,1} = 0.1p$, $A_{max,2} = 1.045q$, $A_{min,2} = 0.1q$, $\gamma = 0.8$, $M = 50$, $E = 5$, $N = 100$, $T_{max} = 8.0 \times 10^{-2}$, and $T_{min} = 8.0 \times 10^{-5}$. Results were obtained for a configuration that corresponded to $c^2 = 10.0$: $(p = 0.932075471, q = 0.954716981, N_L = 16, K = 200)$. *Symmetric* traffic conditions over the 16 input lines were assumed. The threshold Th of Fig. 2 was set to 10^{-13}.

Fig. 12 shows an example of the trajectory of the IS parameters moving towards the near-optimal solution for the ATM switch model. Table 3 summarizes the results, including the near-optimal IS parameters found by MFA, the corresponding estimated blocking probability, the estimated confidence interval and the speed-up factor with respect to conventional MC simulation. For these estimates we used $N_R = 50$, $N_{RC} = 100$. We estimated $R_t = 25.0$. Fig. 13 favorably compares the approximation results from [23] with conventional MC results (where it is possible) and the simulation result obtained using IS.

Pr[loss]	IS values	$\widehat{\Pr}$[loss]	95% Confidence	R_{IS}	R_{net}
unknown	1.0504 0.8938	1.93×10^{-10}	$(1.501 \times 10^{-10}$, $2.356 \times 10^{-10})$	4.15×10^{6}	1.66×10^{5}

Table 3: Blocking probabilities and speed-up factors using the proposed algorithm for the 16×16 ATM switch, with $p = 0.932075471$, $q = 0.954716981$, $c^2 = 10.0$, and $K = 200$. For these estimates: $N_R = 50$, $N_{RC} = 100$.

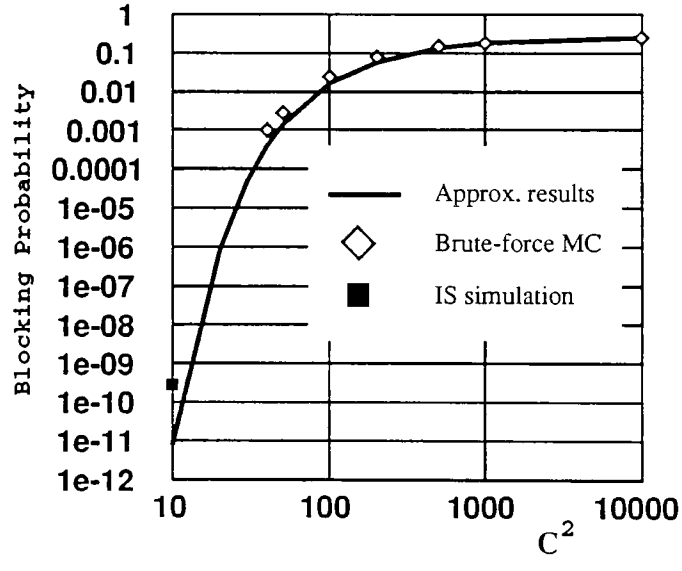


Figure 13: Plot comparing the approximation results with conventional MC results (where it was possible) and the simulation result obtained using IS. Blocking probabilities are plotted vs. the squared coefficient of variation c^2. IS results are consistent with conventional MC results.

Buffer	Pr[loss]	IS values	$\widehat{\Pr}$[loss]	95% Confidence	R_{IS}	R_{net}
$K = 70$	unknown	1.0905 0.8683	1.042×10^{-9}	$(6.978 \times 10^{-10}, 1.381 \times 10^{-9})$	2.63×10^4	2.17×10^3
$K = 85$	unknown	1.0905 0.8683	2.19×10^{-11}	$(1.307 \times 10^{-11}, 3.073 \times 10^{-11})$	8.89×10^5	7.06×10^4

Table 4: Low priority cell blocking probabilities and speed-up factors for the 4×4 ATM switch, with $p = 0.908411$, $q = 0.960747$, $c^2 = 10.0$, and $P_H = 0.3$. For these estimates: $N_R = 20$, $N_{RC} = 10,000$.

7.4 $N_L \times N_L$ ATM Clos Switch with Head-of-Line Priority and Push-Out

Again, we consider an ATM switch with buffers at the input ports, modeled as a *slotted*-time queueing system. Such an $N_L \times N_L$ Clos cell switch is shown in Fig. 11. Furthermore, we assume that there exist two classes of cells, high priority and low priority cells, and that the switch operates with *head-of-line* priority and *push-out*. For a detailed description of the switch and an approximate model for its operation see [24] and references within.

The ATM switch we study here has $N_L = 4$ input lines, symmetric traffic over all input lines, two classes of cell priority (high and low), average rate of arrival in each line $\lambda = 0.3$, probability that a cell has high priority $P_H = 0.3$, and buffers of length $K = 70$.

Approximate regeneration cycles (ARC's) were again used, as described in the previous section.

IS biasing was done dynamically. In each ARC, we biased initially p and q to p_1^* and q_1^*, until the weight function (likelihood L^*) decreased to a prespecified minimum, then changed IS parameters to p_2^* and q_2^* in order to empty the queue and, finally, changed to p_3^* and q_3^* in order to allow fast (approximate) regeneration. In our experiments, we set $p_2^* = p_3^* = p$, $q_2^* = q_3^* = q$, $P_H^* = P_H$.

We optimized IS performance with respect to the settings of p_1^* and q_1^* using MFA, in a way similar to [19]. The blocking probability for low priority cells were estimated that corresponded to $c^2 = 10.0$: $(p = 0.908411, q = 0.960747, N_L = 4, P_H = 0.3, K = 70)$. We used the same IS values to estimate the blocking probability for $c^2 = 10.0$: $(p = 0.908411, q = 0.960747, N_L = 4, P_H = 0.3, K = 85)$. This demonstrates a certain robustness of the optimal IS setting with respect to the queueing capacity, when all other coefficients remain fixed. Table 4 summarizes the results. For these estimates we used $N_R = 20$, $N_{RC} = 10,000$. We estimated $R_t = 12.1$ when $K = 70$, and $R_t = 12.59$ when $K = 85$.

8 Conclusions

We have presented a methodology that uses IS dynamically, within each regeneration cycle, in order to drive the system back to the regeneration state, after an accurate estimate has been obtained. Using this methodology, the benefits of optimal IS and of short regeneration

periods can be achieved simultaneously.

In most realistic systems, the IS estimator variance is not known in closed form. For these cases, minimizing statistical estimates of the variance with respect to the IS parameters can be a useful alternative. The SA and MFA optimization algorithms are appealing because of their increased resistance to the noisiness of the cost function and their ability to escape local minima. We have presented a methodology that uses the MFA algorithm in conjunction with statistical estimates of the IS estimator variance, to obtain near-optimal IS parameter settings.

Run time speed-up factors of two to eleven orders of magnitude over conventional MC simulation are obtained using our methodologies for a wide variety of queueing systems, including systems with correlated arrivals and multiple queues.

References

[1] H. Kahn and A. W. Marshall. Methods of Reducing Sample Size in Monte Carlo Computations. *Oper. Res. Soc. of Amer.*, 1:263–278, Nov. 1953.

[2] K. S. Shanmugan and P. Balaban. A Modified Monte-Carlo Simulation Technique for the Evaluation of Error Rate in Digital Communication Systems. *IEEE Trans. Commun.*, COM-28(11):1916–1924, Nov. 1980.

[3] D. Lu and K. Yao. Improved Importance Sampling Technique for Efficient Simulation of Digital Communication Systems. *IEEE J. Select. Areas Commun.*, 6(1), Jan. 1988.

[4] S. Parekh and J. Walrand. A Quick Simulation Method for Excessive Backlogs in Networks of Queues. *IEEE Trans. Automat. Contr.*, AC-34(1):54–66, Jan. 1989.

[5] P. W. Glynn and D. L. Iglehart. Importance Sampling for Stochastic Simulations. *Management Science*, 35(11):1367–1392, Nov. 1989.

[6] J. S. Sadowsky and J. A. Bucklew. On Large Deviation Theory and Asymptotically Efficient Monte Carlo Estimation. *IEEE Trans. Inform. Theory*, IT-36(3):579–588, May 1990.

[7] V. S. Frost and Q. Wang. Efficient Estimation of Cell Blocking Probability for ATM Systems. In *Proc. of IEEE Int. Conf. Commun.*, Denver, CO, 1991.

[8] M. R. Frater, T. M. Lennon, and B. D. O. Anderson. Optimally Efficient Estimation of the Statistics of Rare Events in Queueing Networks. *IEEE Trans. Automat. Contr.*, AC-36(12):1395–1405, Dec. 1991.

[9] A. Goyal, P. Heidelberger, and P. Shahabuddin. Measure Specific Dynamic Importance Sampling for Availability Simulations. In *Proc. 1987 Wint. Simul. Conf.*, pages 351–357, 1987.

[10] M. Devetsikiotis and J. K. Townsend. A Dynamic Importance Sampling Methodology for the Efficient Estimation of Rare Event Probabilities in Regenerative Simulations of Queueing Systems. In *Proc. IEEE Int. Conf. Commun., ICC '92*, Chicago, June 1992.

[11] R. J. Wolfe, M. C. Jeruchim, and P. M. Hahn. On Optimum and Suboptimum Biasing Procedures for Importance Sampling in Communication Simulation. *IEEE Trans. Commun.*, COM-38(5):639–647, May 1990.

[12] M. Devetsikiotis and J. K. Townsend. An Algorithmic Approach to the Optimization of Importance Sampling Parameters in Digital Communication System Simulation. To appear in the *IEEE Trans. Commun.*

[13] G. Bilbro, R. Mann, T. Miller III, W. Snyder, D. E. Van den Bout, and M. W. White. Mean Field Annealing and Neural Networks. In D. S. Touretzky, editor, *Advances in Neural Information Processing Systems*. Morgan-Kaufmann, San Mateo, CA, 1989.

[14] S. Kirkpatrick, C. Gelatt, and M. Vecchi. Optimization by Simulated Annealing. *Science*, 220(4598):671–680, May 1983.

[15] M. A. Crane and A. J. Lemoine. *An Introduction to the Regenerative Method for Simulation Analysis*. Berlin: Springer-Verlag, 1977.

[16] M. Devetsikiotis and J. K. Townsend. A Useful and General Technique for Improving the Efficiency of Monte Carlo Simulation of Digital Communication Systems. In *Proc. of IEEE GLOBECOM '90*, San Diego, CA, 1990.

[17] R. Y. Rubinstein. *Simulation and the Monte Carlo Method*. New York: John Wiley & Sons, 1981.

[18] H. J. Schlebusch. Nonlinear Importance Sampling Techniques for Efficient Simulation of Communication Systems. In *Proc. IEEE Int. Conf. Commun., ICC '90*, Atlanta, GA, 1990.

[19] M. Devetsikiotis and J. K. Townsend. Optimization of Importance Sampling Parameters for the Efficient Simulation of Communication Networks and ATM Switches Using Mean Field Annealing. In *Proc. 30th Annual ACM Southeast Conf.*, Raleigh, North Carolina, Apr. 1992.

[20] A. M. Law and W. D. Kelton. *Simulation Modeling & Analysis*. New York: McGraw-Hill, 1991.

[21] M. C. Jeruchim. Techniques for Estimating the Bit Error Rate in the Simulation of Digital Communication Systems. *IEEE J. Select. Areas Commun.*, SAC-2(1):153–170, Jan. 1984.

[22] J. J. Hunter. *Mathematical Techniques of Applied Probability*, volume 2, Discrete Time Models: Techniques and Applications. Academic Press, 1983.

[23] A. A. Nilsson, F. Lai, and H. G. Perros. An Approximate Analysis of a Bufferless N x N Synchronous Clos ATM Switch. In *Proc. 13th Int. Teletraffic Congress, ITC 13*, Copenhagen, Denmark, June 1991.

[24] A. A. Nilsson and F. Lai. A Queueing Model of a Bufferless N x N Synchronous Clos ATM Switch with Head-of-Line Priority and Push-Out. Technical Report TR-90/21, CCSP, North Carolina State University, Dec. 1990.