Proceedings of the 2002 Winter Simulation Conference
E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, eds.

ESTIMATING BUFFER OVERFLOWS IN THREE STAGES USING CROSS-ENTROPY

P.T. de Boer

Department of Computer Science
University of Twente
P.O.Box 217
7500 AE Enschede, THE NETHERLANDS

D.P. Kroese

Department of Mathematics
University of Queensland
Brisbane 4072, AUSTRALIA

R.Y. Rubinstein

Faculty of Industrial Engineering
and Management
Technion, Haifa, ISRAEL.

ABSTRACT

In this paper we propose a fast adaptive Importance Sampling method for the efficient simulation of buffer overflow probabilities in queueing networks. The method comprises three stages. First we estimate the minimum Cross-Entropy tilting parameter for a small buffer level; next, we use this as a starting value for the estimation of the optimal tilting parameter for the actual (large) buffer level; finally, the tilting parameter just found is used to estimate the overflow probability of interest. We recognize three distinct properties of the method which together explain why the method works well; we conjecture that they hold for quite general queueing networks. Numerical results support this conjecture and demonstrate the high efficiency of the proposed algorithm.

1 INTRODUCTION

The performance of computer and communications systems is often characterized by the probability of certain *rare events*. For example, the cell loss probability in asynchronous transfer mode (ATM) switches should typically be less than 10^{-9}, see e.g., L'Ecuyer and Champoux (2001). The performance of such systems is frequently studied through simulation. However, estimation of rare event probabilities with naive Monte Carlo techniques requires a prohibitively large number of trials in most interesting cases. One way to deal with this problem is to use *Importance Sampling* (IS). The main idea of IS, when applied to rare events, is to make their occurrence more frequent, or in other words, to "speed up" the simulation. Technically, IS aims to select a probability distribution (change of measure) that minimizes the *variance* of the IS estimator. Finding the right change of measure is often described by a large deviation result. This type of analysis is feasible only for relatively simple models, see also Asmussen and Rubinstein (1995) and Heidelberger (1995) for surveys.

In Lieber, Rubinstein, and Elmakis (1997) and Rubinstein (1997) an *adaptive* IS algorithm for rare events simulation was proposed in which the change of measure is *estimated* by minimizing the sample variance of the IS estimator. In de Boer (2000) and Lieber and Rubinstein (1998) this IS algorithm was further modified to minimize the Kullback-Leibler distance, or Cross-Entropy, with respect to the tilted parameter, instead of minimizing the variance. In de Boer (2000), several efficient heuristics based on state-dependent exponential changes of measure are presented to overcome the difficulties when the state-independent CE method fails. An attractive feature of the CE method is that it can be readily modified for solving NP-hard combinatorial optimization problems (see Alon, Raviv, and Rubinstein (2001), Rubinstein (1999), Rubinstein (2001b), Rubinstein (2002), Rubinstein (2001a)).

In this paper we investigate an adaptive IS algorithm for the efficient simulation of buffer overflow probabilities in queueing systems. The difference between this algorithm and existing adaptive algorithms (de Boer, Nicola, and Rubinstein 2000, Lieber, Rubinstein, and Elmakis 1997, Rubinstein 1997) is that the latter ones always required many stages, where the present one comprises only *three* stages: First, in the *pilot* stage we estimate the minimum CE tilting parameter for a small buffer level; next, we use this as a starting value for the estimation of the optimal tilting parameter for the actual (large) buffer level; finally, the tilting parameter just found is used to estimate the overflow probability of interest.

The reason why the three-stage approach works well (for arbitrary overflow levels) is that under the initial change of measure the buffer process is unstable, and moreover, that this change of measure is "close" to the change of measure for the second stage. In other words, the initial tilting vector is in some sense a "good" tilting vector. We have investigated these two properties, which we will call the *instability property* and the *robustness property* in more detail for the M/M/1 queue. We conjecture that these properties hold in more general network as well. Numerical results support this conjecture and demonstrate the high efficiency of the proposed algorithm.

The rest of the paper is organized as follows. In Section 2 we summarize the main ideas behind the *adaptive* approach to Importance Sampling. In Section 3 we formulate the simulation model and give the main algorithm for simulating overflows in queueing networks. Results from a closer investigation of the M/M/1 queue are summarized in Section 4. In Section 5 we demonstrate numerically the effectiveness of the algorithm by investigating various queueing models, and in Section 6 concluding remarks are given. Finally, some auxiliary results and proofs are given in the appendix.

2 IMPORTANCE SAMPLING AND THE CROSS-ENTROPY METHOD

In this section we briefly review the ideas behind Importance Sampling (IS) and the Cross-Entropy (CE) method. For details the reader is referred to Rubinstein and Melamed (1998) and Rubinstein (1999).

Let $\boldsymbol{X} = (X_1, \ldots, X_n)$ be a random vector taking values in some space $\mathcal{X}$. Let $\{f(\cdot; \boldsymbol{v})\}$ be a family of probability densities on $\mathcal{X}$, with respect to some (unspecified) base measure. Here $\boldsymbol{v}$ is a real-valued parameter (vector).

Let H be some real function on $\mathcal{X}$. Suppose we wish to estimate, via simulation,

$$\gamma_{\boldsymbol{v}} := \mathbb{E}_{\boldsymbol{v}}\, H(\boldsymbol{X}),$$

where $\mathbb{E}_{\boldsymbol{v}}$ denotes the expectation under $f(\cdot\,; \boldsymbol{v})$. In this paper we will be mostly concerned with functions H that are *indicators* of certain events; for example $H(\boldsymbol{X}) = I_A$, with $A = \{\boldsymbol{X} \in \mathcal{X}_0\}$ for some subset $\mathcal{X}_0 \subset \mathcal{X}$. When the probability of A is very small we say that A is a *rare event*.

A naive way to estimate $\gamma_{\boldsymbol{v}}$ is to use crude Monte-Carlo simulation: Draw a random sample $\boldsymbol{X}^{(1)}, \ldots, \boldsymbol{X}^{(N)}$ from $f(\cdot\,; \boldsymbol{v})$; then $\frac{1}{N}\sum_{i=1}^{N} H(\boldsymbol{X}^{(i)})$ is an unbiased estimator of $\gamma_{\boldsymbol{v}}$. However this poses serious problems when H is the indicator of a rare event. In that case a large simulation effort is required in order to estimate $\gamma_{\boldsymbol{v}}$ accurately.

An alternative is to use Importance Sampling simulation: Draw a random sample $\boldsymbol{X}^{(1)}, \ldots, \boldsymbol{X}^{(N)}$ from $f(\cdot\,; \tilde{\boldsymbol{v}})$; then

$$\frac{1}{N}\sum_{i=1}^{N} H(\boldsymbol{X}^{(i)})\, W(\boldsymbol{X}; \boldsymbol{v}, \tilde{\boldsymbol{v}}), \tag{1}$$

with *likelihood ratio*

$$W(\boldsymbol{X}; \boldsymbol{v}, \tilde{\boldsymbol{v}}) := \frac{f(\boldsymbol{X}^{(i)}\,; \boldsymbol{v})}{f(\boldsymbol{X}^{(i)}\,; \tilde{\boldsymbol{v}})},$$

is an unbiased estimator of $\gamma_{\boldsymbol{v}}$. We say that we perform the simulation under a *change of measure* parameterized by the *tilting* parameter (vector) $\tilde{\boldsymbol{v}}$. The aim is now to find an optimal tilting parameter ${}_*\boldsymbol{v}$ such that the variance, or equivalently, the second moment, of the IS estimator is minimal. In other words we wish to find

$${}_*\boldsymbol{v} = \arg\min_{\tilde{\boldsymbol{v}}} \mathbb{E}_{\tilde{\boldsymbol{v}}} \left[H(\boldsymbol{X})\, W(\boldsymbol{X}; \boldsymbol{v}, \tilde{\boldsymbol{v}})\right]^2 . \tag{2}$$

More generally, using again the principle of IS, this is equivalent to finding

$${}_*\boldsymbol{v} = \arg\min_{\tilde{\boldsymbol{v}}} \mathbb{E}_{\boldsymbol{v}_j}\, H^2(\boldsymbol{X})\, W(\boldsymbol{X}; \boldsymbol{v},\ \tilde{\boldsymbol{v}})\, W(\boldsymbol{X}; \boldsymbol{v}, \boldsymbol{v}_j) \tag{3}$$

for *any* tilting parameter $\boldsymbol{v}_j$.

An analytic expression for the optimal tilting parameter ${}_*\boldsymbol{v}$ is typically not available. However, it can be estimated by minimizing, possibly numerically, the estimator of the expectation in (3), leading to the approximation

$$\boldsymbol{v}_{j+1} = \arg\min_{\tilde{\boldsymbol{v}}} \sum_{i=1}^{N} H^2(\boldsymbol{X}^{(i)})\, W(\boldsymbol{X}^{(i)}; \boldsymbol{v}, \tilde{\boldsymbol{v}})\, W(\boldsymbol{X}^{(i)}; \boldsymbol{v}, \boldsymbol{v}_j)\ , \tag{4}$$

where $\boldsymbol{X}^{(1)}, \ldots, \boldsymbol{X}^{(N)}$ is a random sample from $f(\cdot, \boldsymbol{v}_j)$. This formula forms the basis of an iterative scheme to estimate the true optimal tilting parameter. Note that the evaluation of (4) in general involves *numerical* optimization, which may be quite time-consuming. A much more convenient approach is to replace (2) with its Cross-Entropy equivalent introduced in Lieber and Rubinstein (1998), Rubinstein (1999). This typically leads to much more simple (analytical) updating rules than (4).

2.1 Cross-Entropy Method

It is well known that the best possible change of measure to estimate $\gamma_{\boldsymbol{v}}$ is such that $\boldsymbol{X}$ has a density g given by

$$g(\boldsymbol{x}) = \frac{H(\boldsymbol{x})f(\boldsymbol{x};\boldsymbol{v})}{\gamma_{\boldsymbol{v}}}, \qquad (5)$$

for all $\boldsymbol{x} \in \mathcal{X}$. However, this density may not belong to the family $\{f(\cdot;\boldsymbol{v})\}$. Instead of trying to find a tilting parameter ${}_*\boldsymbol{v}$ which minimizes the variance of the estimator (1) we could try to find a density $f(\cdot;\boldsymbol{v}^*)$ which, in some sense, is closest to the density given in (5). One way of doing this is by minimizing the Kullback-Leibler or *Cross Entropy* (CE) "distance" between g and $f(\cdot;\boldsymbol{v}^*)$ which is given (see e.g. Kapur and Kesavan (1992)) by

$$\mathbb{E}_g \log \frac{g(\boldsymbol{X})}{f(\boldsymbol{X};\boldsymbol{v}^*)}, \qquad (6)$$

where $\mathbb{E}_g$ denotes the expectation under g. It is not difficult to see that this is equivalent to finding

$$\boldsymbol{v}^* = \arg\max_{\tilde{\boldsymbol{v}}} \mathbb{E}_{\boldsymbol{v}}\, H(\boldsymbol{X}) \log f(\boldsymbol{X};\tilde{\boldsymbol{v}}) \;. \qquad (7)$$

Analogously to (3) this is equivalent to

$$\boldsymbol{v}^* = \arg\max_{\tilde{\boldsymbol{v}}}\ \mathbb{E}_{\boldsymbol{v}_j}\, H(\boldsymbol{X})\, W(\boldsymbol{X};\boldsymbol{v},\boldsymbol{v}_j) \log f(\boldsymbol{X};\tilde{\boldsymbol{v}}), \qquad (8)$$

for any tilting parameter $\boldsymbol{v}_j$. Similarly to (4) we may estimate $\boldsymbol{v}^*$ by

$$\boldsymbol{v}_{j+1} = \arg\max_{\tilde{\boldsymbol{v}}} \sum_{i=1}^{N} H(\boldsymbol{X}^{(i)})\, W(\boldsymbol{X}^{(i)};\boldsymbol{v},\boldsymbol{v}_j)\, \log f(\boldsymbol{X}^{(i)};\tilde{\boldsymbol{v}}) \;, \qquad (9)$$

where $\boldsymbol{X}^{(1)},\ldots,\boldsymbol{X}^{(N)}$ is a random sample from $f(\cdot,\boldsymbol{v}_j)$. Since under quite mild conditions (Rubinstein and Shapiro 1993) the sum in (9) is convex and differentiable with respect to $\tilde{\boldsymbol{v}}$, the tilting vector $\boldsymbol{v}_{j+1}$ in (9) may be readily obtained by solving (with respect to $\tilde{\boldsymbol{v}}$) the following system of nonlinear equations:

$$\sum_{i=1}^{N} H(\boldsymbol{X}^{(i)})\, W(\boldsymbol{X}^{(i)};\boldsymbol{v},\boldsymbol{v}_j)\, \nabla \log f(\boldsymbol{X}^{(i)};\tilde{\boldsymbol{v}}) = \boldsymbol{0}, \qquad (10)$$

where the gradient is with respect to $\tilde{\boldsymbol{v}}$. This, of course, provided that the expectation and differentiation operators can be interchanged (Rubinstein and Shapiro 1993) and the function (8) is convex and differentiable with respect to $\tilde{\boldsymbol{v}}$. The advantage of this approach is that $\boldsymbol{v}_{j+1}$ can often be calculated *analytically*. In particular, this happens if the distributions of the random variables belong to a *Natural Exponential Family* (NEF); this is demonstrated in the Appendix for a simple case, and in the next section for a general queueing model.

3 ESTIMATING BUFFER OVERFLOW PROBABILITIES

In this section we present the main algorithm for estimating buffer overflow probabilities in queueing networks.

Consider an open network of GI/G/1 queues with Markovian routing. We are interested in the probability $\gamma(\ell)$ of the event A that the content of a certain queue, or the combined contents of several queues, exceeds a certain level ℓ during an interval $[0, T]$, where T is some stopping time for the process $\boldsymbol{X}$ of interarrival times (from outside the system) and service times and routing decisions. Typically, T is the length of a busy cycle, or the first time until either the content of a queue exceeds level ℓ or the system becomes empty.

We wish to estimate $\gamma(\ell)$ by using an IS procedure, in which we can change the service and interarrival time distribution at each queue. We assume that for each queue the interarrival and service time distributions belong to a NEF family that is reparametrized by the mean (vector of means) $\boldsymbol{v}$, as discussed in the Appendix. Note that such an IS procedure is *state independent*: the change of the distributions is made globally and does not vary with the state variables of the system (e.g., the content of the queues).

The idea is to first estimate the optimal tilting parameter via the iterative schemes (4) or (9) and then to use this to estimate $\gamma(\ell)$ via ordinary IS.

In most cases of interest $\gamma(\ell)$ is a rare event probability. This means that the choice of a "good" *initial* tilting parameter $\boldsymbol{v}_0$ for the scheme (4) or (9) is crucial. For general queueing networks it is unclear what comprises a good initial guess. Obviously, the system should be instable, but it is far from trivial to determine which instable regimes are good and which are not good.

We now make three conjectures. All conjectures have been observed numerically and some can be proved in certain simple situations, (see below).

1. **Instability property.** The optimal tilting parameter corresponding to overflow of a *low* level ℓ_0 (e.g. $\ell_0 = 3$ or $\ell_0 = 4$) renders the system instable.
2. **Robustness property.** An optimal parameter corresponding to overflow of a *low* level ℓ_0 is a "good" initial tilting vector for finding the optimal tilting parameter for the high level ℓ. I.e., the estimation of the tilting parameter for the high level ℓ is robust (insensitive) to the choice of ℓ_0.
3. **CE optimality property.** The minimum variance tilting parameter *asymptotically* coincides with the minimum CE tilting parameter (see Lieber and

Rubinstein (1998) for the proof for certain simple situations).

The third property means that we can use a very simple updating formula for the tilting vectors. In particular, let $\boldsymbol{v} = (v_1, \dots, v_K)$ be the (nominal) vector of means corresponding to the pdfs $(f_1, \dots, f_K)$ of interarrival times (customers arriving to the queue from outside the system) and service times at the queues, assuming that the interarrival and service times are random. For simplicity we assume for the moment that the routing probabilities remain fixed; see however Remark 3.2. Let $H(\boldsymbol{X})$ be the indicator of the event A. Note that each parameter v_k corresponds to a service time or an (external) interarrival time at a certain queue. For each such service or interarrival time (indexed by k) there will be τ_k service completions/inter-arrivals. Denote these by $Y_{k1}, \dots, Y_{k\tau_k}$. It follows that the density $f(\boldsymbol{X}; \boldsymbol{v})$, corresponding to the history of the process $\boldsymbol{X}$ during $[0, T]$, is the *product*

$$f(\boldsymbol{X}; \boldsymbol{v}) = \prod_{k=1}^{K} \prod_{j=1}^{\tau_k} f_k(Y_{kj}; v_k) \, . \tag{11}$$

Thus the likelihood ratio $W(\boldsymbol{X}; \boldsymbol{v}, \boldsymbol{v}_j)$, corresponding to history of the process $\boldsymbol{X}$ during $[0, T]$, is the quotient the products of the form above. Now, combining (11), (9) and the Appendix, it is not difficult to see that for NEFs the components of the tilting vector should be updated as

$$v_{j+1,k} = \frac{\displaystyle\sum_{i=1}^{N} \left(H^{(i)}(\boldsymbol{X}) W^{(i)}(\boldsymbol{X}; \boldsymbol{v}, \boldsymbol{v}_j) \sum_{j=1}^{\tau_k^{(i)}} Y_{kj}^{(i)} \right)}{\displaystyle\sum_{i=1}^{N} H^{(i)}(\boldsymbol{X}) W^{(i)}(\boldsymbol{X}; \boldsymbol{v}, \boldsymbol{v}_j) \, \tau_k^{(i)}} \, , \tag{12}$$

where the simulation is performed under tilting vector $\boldsymbol{v}_j$.

Based on the three properties above we now have the algorithm shown in Figure 1.

Remark 3.1. To assess if an initial tilting vector $\boldsymbol{v}_0$ is "good" we have to consider how effective the second stage of the Main Algorithm is. Numerical evidence shows that vectors $\boldsymbol{v}_1, \boldsymbol{v}_2, \dots$ converge *accurately* and *fast* to the optimal tilting vector $\boldsymbol{v}^*$.

Remark 3.2. In the above, each random variable (and thus each element of $\boldsymbol{v}$) was assumed to correspond to a service or interarrival time. However, the same formalism also applies to random routing among two destinations: this involves a Bernoulli random variable, with outcomes 0 and 1 corresponding to the two destinations. The mean of this random variable is just the routing probability, so the routing probability can be directly incorporated into $\boldsymbol{v}$, thus allowing our algorithm to also find the optimal routing probability.

Main Algorithm

Pilot stage:

1. Choose an initial buffer level ℓ_0. Choose the initial tilting vector $\boldsymbol{v}_0 = \boldsymbol{v}$.
2. Simulate N_1 paths, using the tilting vector $\boldsymbol{v}_0$, for overflow level ℓ_0.
3. Find the tilting vector $\boldsymbol{v}_1$ from (12), for overflow level ℓ_0.

Second stage:

1. Initialize as follows: $j := 0$ (iteration counter); Choose as initial tilting vector $\boldsymbol{v}_0$ the resulting tilting vector $(\boldsymbol{v}_1)$ of the pilot stage.
2. Simulate N_2 replications with tilting vector $\boldsymbol{v}_j$.
3. Find the tilting vector $\boldsymbol{v}_{j+1}$ from (12), for overflow level ℓ.
4. Increment j and repeat steps 2–4, until the tilting vector has converged.

Third stage:

Estimate the probability $\gamma_{\boldsymbol{v}}$ via IS simulation, as in (1), with the final tilting vector obtained in the second stage.

Figure 1: The Main Algorithm

4 IS AND THE CE-METHOD APPLIED TO THE M/M/1 QUEUE

For a single $M/M/1$ queue the behaviour of the proposed method, and in particular the three properties conjectured earlier, can be studied analytically. A detailed analysis of the CE method applied to the $M/M/1$ queue can be found in de Boer, Kroese, and Rubinstein (2002); we only summarize the results here.

A few preliminaries: the $M/M/1$ queue is simulated as a discrete-time Markov chain (as opposed to the continuous-time repesentation used in the rest of this paper). The probability of arrival in the DTMC is $p = \lambda/(\lambda + \mu)$, where λ is the arrival rate and μ is the service rate. The probability of service completion is $q = 1 - p$. The tilting is described by the exponential tilting parameter θ, from which the tilted arrival and service probabilities follow:

$$\tilde{p} = \frac{p e^{\theta}}{p e^{\theta} + q e^{-\theta}} \quad \text{and} \quad \tilde{q} = 1 - \tilde{p}.$$

4.1 Instability Property

It can be shown that the optimal tilted traffic intensity $\tilde{\rho}(\ell)$ for the buffer overflow probability in a M/M/1 queue is *greater than unity* regardless of the buffer size ℓ, $(\ell \geq 2)$. This obviously is the instability property.

In addition, $\tilde{\rho}(\ell)$ decreases in ℓ and $\lim_{\ell \to \infty} \tilde{\rho}(\ell) = \rho^{-1}$; the latter is the well-known asymptotically optimal tilt for a single queue (Sadowsky 1991).

4.2 Robustness Property

In each iteration, the tilting parameter for the next iteration is estimated using an equation like (12), which is a *ratio estimator*: the new tilting parameter is given as the ratio of two sample averages. A sufficient condition for such an estimator to have finite variance is that the variances of both the numerator and the denominator, and the expectation of their product, are all finite, and that the denominator is nonzero. For the case of IS-simulation of an $M/M/1$ queue, it can be shown that this condition is only satisfied if the simulation is run at a not-too-large tilting θ.

As noted in Section 4.1, for a lower overflow level the optimal tilting is larger. So the first step, using a low ℓ_0, may produce a rather high tilting parameter θ for use in the second iteration. Consequently, in the second iteration the sufficient condition may not be satisfied; then it is not guaranteed that the estimate for θ found there (for use as tilting parameter in the third iteration) has finite variance (the theory does not tell). As an example, if $\rho = 0.3/0.7$, ℓ_0 should be chosen at least 7 in order for the sufficient condition to be satisfied.

Clearly, the above results do not fully support the conjectured robustness property: ℓ_0 must not be too small for robustness to be proved. Still, as will be shown in the experiments section, even with a small ℓ_0 the method converges, although a few more iterations are needed. This needs further study.

4.3 CE Optimality Property

For the $M/M/1$ queue, it can be shown that the minimization of variance and cross-entropy are asymptotically equivalent, in the sense that the optimal tilting factors for $\ell \to \infty$ both converge to the same limit value (which corresponds to exchanging the arrival and service rates). This is illustrated in Figure 2, which gives the tilting parameter value $_*\theta$ that minimizes the variance, and θ^* that minimizes the cross-entropy, as a function of ℓ.

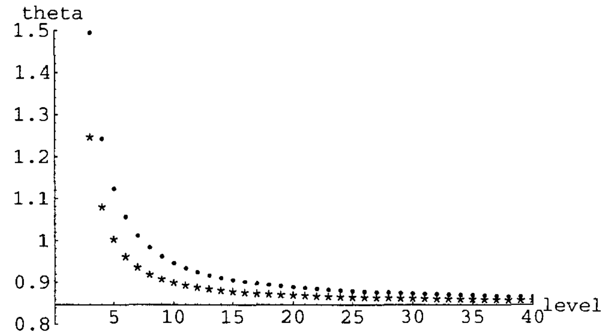

Figure 2: Optimal Tilting Parameter $_*\theta$ for Minimizing Variance (Stars) and θ^* for Minimizing CE (Dots) for Various Values of ℓ; with $p = 3/10$ and $i = 1$. Note: $\theta^*(\ell) > {}_*\theta(\ell)$. Also, $\theta^*(2) = \theta^*(2) = \infty$.

5 SIMULATION RESULTS

In sections 5.1 - 5.4 we give some numerical examples of the application of our Main Algorithm with the view to illustrate the three properties we have discussed above.

5.1 Single $M/M/1$ Queue

As a first example, we consider the $M/M/1$ queue, with arrival rate $\lambda = 0.3$, service rate $\mu = 0.7$, and overflow level (buffer size) $\ell = 20$.

The results are presented in Table 1. The table has one row for every simulation run (iteration), listing the number of replications (busy cycles) simulated, the values of (in principle) the tilting parameters $\boldsymbol{v}_k$, and the estimate for the overflow probability found in that simulation run along with its relative error (RE). In the present model all distributions are exponential, and tilting them exponentially gives again an exponential distribution. Therefore, instead of listing the tilting parameters $\boldsymbol{v}_k$ explicitly, we prefer to show the resulting rates, since these are more intuitive. The same applies to routing probabilities in later examples.

Table 1 shows results for two different values of the overflow level ℓ_0 in the pilot run, namely 2 and 8. The former is the minimum that can work; for $\ell_0 = 1$, the system would already have reached the "rare" target event in its initial state. In the case with $\ell_0 = 8$, the overflow in the pilot run is rather rare, so a large number of replications are needed to observe it a reasonable number of times (16 in this experiment).

The results for the case $\ell_0 = 8$ show that a total of three iterations can indeed be enough. The first (pilot run) makes the system unstable; i.e., the λ and μ that the pilot run calculates as optimal for the second iteration, are such that $\lambda > \mu$. The second run does not yet yield an optimal

Table 1: Simulation Results for the $M/M/1$ Queue, for $\ell = 20$. For Comparison: Direct Calculation Yields an Overflow Probability of $5.826 \cdot 10^{-8}$.

$\ell_0 = 2$					
iter.	repl.	λ	μ	estimate	rel. error
1	100	0.3	0.7	–	–
2	1000	1.41	0.45	$8.31 \cdot 10^{-9}$	0.3031
3	1000	1.00	0.32	$4.64 \cdot 10^{-8}$	0.1332
4	1000	0.79	0.28	$5.29 \cdot 10^{-8}$	0.0514
5	1000	0.74	0.30	$5.60 \cdot 10^{-8}$	0.0419
6	1000	0.73	0.30	$5.96 \cdot 10^{-8}$	0.0406
$\ell_0 = 8$					
iter.	repl.	λ	μ	estimate	rel. error
1	10^4	0.3	0.7	–	–
2	1000	0.81	0.29	$5.61 \cdot 10^{-8}$	0.0573
3	1000	0.73	0.29	$6.15 \cdot 10^{-8}$	0.0398
4	1000	0.72	0.30	$5.94 \cdot 10^{-8}$	0.0406
5	1000	0.72	0.30	$6.21 \cdot 10^{-8}$	0.0385

(i.e., low RE) estimate of the overflow probability, since it uses a tilting found in the first iteration and thus optimal for an overflow level of 8 rather than 20. However, the second run does find optimal values for λ and μ to be used in the third iteration: the third iteration achieves a relative error of 0.0398, and further iterations do not significantly improve this.

In the case of $\ell_0 = 2$, things look a bit different. Clearly, five iterations are needed here before λ and μ are sufficiently close to their final values to achieve a low relative error. This is not surprising: in Section 4.2 it was noted that if ℓ_0 is chosen too low, the estimator for the tilting parameter becomes the ratio of two infinite-variance estimates, and thus has unknown behaviour. The present simulation results suggest that the estimator for the tilting vector is biased in this situation, causing more iterations to be needed; with every iteration we move closer to the correct tilting and thus away from the "problematic" region.

5.2 Two Non-Markovian Queues with Random Feedback

As a second example, we consider the network depicted in Figure 3. It consists of two queues in tandem, where customers departing from the second queue either leave the network (with probability p), or go back to the first queue (with probability $1-p$). We are interested in the probability that the total number of customers in the network exceeds some high level, 50 in this example, during one busy cycle.

Interestingly, for this model (and in general, any model with random feedback) we cannot work with $\ell_0 = 2$, as we could in the single $M/M/1$ queue. The reason for this is the following. Consider using $\ell_0 = 2$. This means that after starting the busy-cycle with 1 customer in the network, we are interested in the probability of reaching a state where 2 customers are in the network, before the network becomes empty. So, until the overflow there will be always exactly 1 customer in the network: if less than 1, the busy-cycle would already end, and if more than 1 the overflow would already happen. Therefore, no departures from the system can occur on a sample path to the overflow. Consequently, if ever a service completion happens at the second queue on the sample path, the customer leaving that queue *must* be routed back to the first queue, otherwise the busy-cycle would end. Therefore, we will observe customers being routed back to the first queue with probability 1, which then becomes the value of the routing probability for the next iteration due to the CE algorithm. And once a routing probability has become 1, later iterations will never observe the alternative routing decision, so the probability will remain 1. So using a pilot run with $\ell_0 = 2$ forces the routing probability to be 1 in all later iterations, which is incorrect if $\ell > 2$ in those iterations.

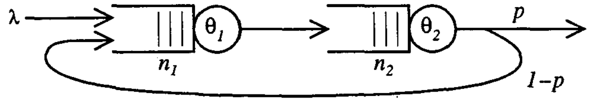

Figure 3: Two Queues in Tandem with Feedback

In this example, the interarrival time distribution is a two-stage Erlang distribution, with exponential parameter $\lambda = 0.2$. The service time distributions are uniform on $[0, 3.333]$ and $[0, 5]$, for the first and second server, respectively. The results are shown in Table 2. In this table, θ_1 and θ_2 are the exponential tilting factors applied to the non-Markovian service time distributions; basically, these are the θ from (13).

Table 2: Simulation Results for the Non-Markovian Network for $\ell = 50$.

$\ell_0 = 3$							
iter.	repl.	λ	θ_1	θ_2	p	estimate	RE
1	10^2	0.2	0	0	0.5	–	–
2	10^4	0.34	0.12	0.09	0.21	$3.48 \cdot 10^{-25}$	0.155
3	10^4	0.36	-0.00	0.17	0.23	$3.37 \cdot 10^{-25}$	0.015
4	10^4	0.36	0.00	0.15	0.24	$3.34 \cdot 10^{-25}$	0.014
5	10^4	0.36	0.00	0.15	0.24	$3.29 \cdot 10^{-25}$	0.012
6	10^6	0.36	0.00	0.15	0.24	$3.29 \cdot 10^{-25}$	0.001
$\ell_0 = 7$							
iter.	repl.	λ	θ_1	θ_2	p	estimate	RE
1	10^4	0.2	0	0	0.5	–	–
2	10^4	0.34	0.05	0.14	0.20	$3.35 \cdot 10^{-25}$	0.041
3	10^4	0.36	-0.00	0.15	0.24	$3.24 \cdot 10^{-25}$	0.011
4	10^4	0.36	0.00	0.16	0.24	$3.27 \cdot 10^{-25}$	0.012
5	10^4	0.36	-0.00	0.15	0.24	$3.29 \cdot 10^{-25}$	0.011
6	10^4	0.36	-0.00	0.16	0.24	$3.22 \cdot 10^{-25}$	0.011
7	10^6	0.36	0.00	0.16	0.24	$3.28 \cdot 10^{-25}$	0.001

The algorithm converges quickly, already reaching the final accuracy in the third iteration. No numerical results are available for validation; therefore, we did the last iteration with 100 times more replications, to see whether relative error decreases appropriately (i.e., by a factor of $\sqrt{100} = 10$). The fact that this is indeed the case, gives confidence.

5.3 Five-node Jackson Network

As a final example, consider the estimation of the overflow probability of the total population of the five-node Jackson network with random routing depicted in Figure 4.

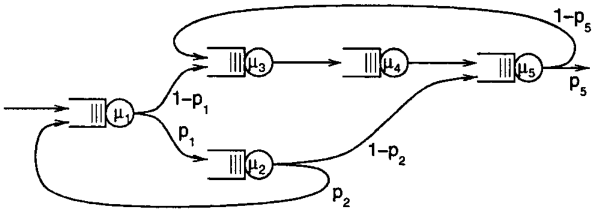

Figure 4: A Five-node Jackson Network.

We first simulate this network at a parameter setting where server 3 is the bottleneck queue: it has a load of 0.2, while the other servers have a load of 0.1. These parameters are as follows: $\lambda = 3, \mu_1 = 40, \mu_2 = 20, \mu_3 = 25, \mu_4 = 50, \mu_5 = 60$, with all routing probabilities equal to 0.5. The overflow level during the pilot run, ℓ_0, was set to 5: this level is reached by about 1% of all sample paths under the original measure.

The results are shown in Table 3. For an overflow level of 80 the method still converges fine; and although the relative error tends to vary notably among further iterations, the estimates do appear to be consistent. We have repeated the simulation for various overflow levels and have observed that the relative error does not increase much between $\ell = 20$ and $\ell = 80$, suggesting that the method is asymptotically efficient.

It is noteworthy that the parameters found by the CE procedure are close to those calculated by the method of Frater, Lennon, and Anderson (1991).

Table 3: Simulation Results for the Five-node Network with One Bottleneck; $\ell_0 = 5, \ell = 80$. Note That Only Four of the Nine Tilting Parameters are Shown Here.

iter.	repl.	λ	p_2	μ_3	μ_4	estimate	RE
1	10^5	3.0	0.500	25.0	50.0	–	–
2	10^5	10.5	0.641	20.0	45.0	$2.51 \cdot 10^{-55}$	0.363
3	10^5	13.3	0.564	16.7	47.6	$8.03 \cdot 10^{-55}$	0.060
4	10^5	13.0	0.589	15.3	49.8	$7.82 \cdot 10^{-55}$	0.024
5	10^5	12.9	0.595	15.3	49.4	$7.50 \cdot 10^{-55}$	0.014
6	10^5	13.0	0.594	15.4	49.6	$7.67 \cdot 10^{-55}$	0.048
7	10^5	13.0	0.594	15.4	49.7	$7.60 \cdot 10^{-55}$	0.017

The above experiment has been repeated with a different set of service rates, chosen such that all servers had an equal load. For an overflow level $\ell = 20$, the method still converged fine, but a much larger number of replications was needed (10^7). For a higher overflow level, no convergence was obtained: presumably, this is a case where a state-independent change of measure does not work well enough. A state-dependent change of measure does help here, at the expense of complexity; see de Boer (2000) or de Boer and Nicola (2002).

5.4 Root Finding

In practical problems, one often needs to do *root finding*: finding a buffer size for which the overflow probability is less than a given value. The present simulation technique can easily be used for that, because for high overflow levels the optimal tilting turns out to be almost independent of the overflow level (cf. Section 4.1). Thus, after finding a good tilting for some high overflow level, one can estimate the overflow probability for a large range of levels in one run. (Note that the CE algorithm for static models, as in Rubinstein and Melamed (1998), does not have this property, making root finding more involved.)

As an example, the two-node network from Section 5.2 is used. A simulation run with the tilting found by the CE method for overflow level $\ell = 50$, is used to obtain all the overflow probabilities given in Table 4. Clearly, these estimates have almost equal relative error, and one easily concludes that a buffer size of 22 is the minimum that will make the overflow probability less than 10^{-10}.

6 CONCLUDING REMARKS

In this paper, we have presented an efficient Cross-Entropy method for estimation of buffer overflow probabilities in queueing networks via simulation. We have recognised

Table 4: Numerical Results for the Root Finding Example.

level	overflow probability	relative error
2	0.422	0.0058
3	0.170	0.0082
⋮	⋮	⋮
21	$1.62 \cdot 10^{-10}$	0.0111
22	$5.05 \cdot 10^{-11}$	0.0109
⋮	⋮	⋮
48	$3.40 \cdot 10^{-24}$	0.0113
49	$1.05 \cdot 10^{-24}$	0.0111
50	$3.27 \cdot 10^{-25}$	0.0109

three properties (CE optimality, instability and robustness) which explain why the method works well. Numerical results support the conjectured properties and demonstrate the high efficiency of the proposed algorithm for queueing networks up to five queues.

Some issues for further research are the following.

- Extension of the proofs of the three properties to more general queueing models.
- Further investigation of the behaviour of the ratio estimators of type (12) for the $M/M/1$ queue and more general queueing models.
- Finding conditions under which a state-independent change of measure, as used in this method, can or cannot lead to an (asymptotically) efficient simulation.

APPENDIX

A.1 Natural Exponential Families (NEFs)

Consider a univariate family of distributions with densities (pmf's, pdf's) $\{f_\theta, \theta \in \Theta\}$, for some subset $\Theta \subset \mathbb{R}$. The family is said to be a NEF if

$$f_\theta(x) = \mathrm{e}^{x\theta - \kappa(\theta)}\, h(x), \tag{13}$$

where h is a positive (normalization) function, cf. Morris (1982) and Jorgensen (1997). For example, if we take $\theta = \lambda/\sigma^2$ and $\kappa(\theta) = \sigma^2\theta^2/2$, then f_θ is the density of the $\mathsf{N}(\lambda, \sigma^2)$ distribution, where σ^2 is fixed.

There are many NEFs. In fact, every distribution with pdf f_0 for which the moment generating function exists in a neighbourhood of 0 generates its own NEF by letting κ be the cumulant function

$$\kappa(\theta) = \log \int \mathrm{e}^{\theta x} f_0(x)\, dx$$

and by substituting $h = f_0$ into (13). We say that f_θ is obtained from f_0 by an *exponential twist/tilt* with *twisting/tilting parameter* θ.

Now let X have a distribution in some NEF $\{f_\theta\}$. It is not difficult to see that

$$v := \mathbb{E}_\theta X = \kappa'(\theta) \quad \text{and} \quad \mathrm{Var}_\theta X = \kappa''(\theta).$$

Since κ' is increasing we may reparametrize the family using the mean v. In particular, to the NEF above corresponds a family $\{g_v\}$ such that for each pair (θ, v) satisfying $\kappa'(\theta) = v$ we have $g_v = f_\theta$.

Now consider (8) for the case where X is a random variable from a NEF $\{f(\cdot; v)\}$, reparametrized by the mean v. Hence,

$$f(x; v) = \exp\left(\theta(v)x - \kappa(\theta(v))\right)\, h(x),$$

where θ is some differentiable function of v. We wish to maximize, with respect to $\tilde{v}$ the function D defined as

$$D(\tilde{v}) = \mathbb{E}_{v_j}\, H(X)\, W(X; v, v_j) \log f(X; \tilde{v})\ .$$

Solving $D'(\tilde{v}) = 0$ for $\tilde{v}$ gives

$$\begin{aligned} &\mathbb{E}_{v_j}\, H(X)\, W(X; v, v_j) \left\{\theta'(\tilde{v})X - \kappa'(\theta(\tilde{v}))\,\theta'(\tilde{v})\right\} \\ &= \ \mathbb{E}_{v_j}\, H(X)\, W(X; v, v_j)\, \theta'(\tilde{v})(X - \tilde{v}) = 0, \end{aligned}$$

which is solved for $\tilde{v} = v^*$, with

$$v^* = \frac{\mathbb{E}_{v_j}\, H(X)\, W(X; v, v_j) X}{\mathbb{E}_{v_j}\, H(X)\, W(X; v, v_j)}\ . \tag{14}$$

That v^* is a global maximum follows from the convexity of D and the fact that $D''(v^*) = -\theta'(v^*)\mathbb{E}_v H(X) < 0$, because $\theta'(v^*) = 1/\mathrm{Var}_{v^*}(X) > 0$.

ACKNOWLEDGMENTS

We would like to thank Søren Asmussen for his helpful comments, and Zinovy Landsman for introducing us to the natural exponential family. This research was partly supported by the Australian Research Council (ARC), grant number DP0209921.

REFERENCES

Alon, G., T. Raviv, and R. Y. Rubinstein. 2001. Application of the cross entropy method for buffer allocation problem in simulation based environment. Manuscript, Technion, Haifa, Israel.

Asmussen, S., and R. Y. Rubinstein. 1995. Complexity properties of steady-state rare events simulation in queueing models. In *Advances in Queueing: Theory, Methods and Open Problems*, ed. J. Dshalalow, Volume I, 429–462.

de Boer, P. T. 2000. *Analysis and efficient simulation of queueing models of telecommunication systems*. Ph. D. thesis, University of Twente.

de Boer, P. T., D. P. Kroese, and R. Y. Rubinstein. 2002. A fast cross-entropy method for estimating buffer overflows in queueing networks. Submitted.

de Boer, P. T., and V. F. Nicola. 2002. Adaptive state-dependent importance sampling simulation of markovian queueing networks. *European Transactions on Telecommunications*. To appear.

de Boer, P. T., V. F. Nicola, and R. Y. Rubinstein. 2000. Adaptive importance sampling simulation of queueing networks. In *Proceedings of the 2000 Winter Simulation Conference, Orlando, Florida*, 646–655.

Frater, M. R., T. M. Lennon, and B. D. O. Anderson. 1991. Optimally efficient estimation of the statistics of rare events in queueing networks. *IEEE Transactions on Automatic Control* 36:1395–1405.

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

Jorgensen, B. 1997. *The theory of dispersion models*. Chapman and Hall, London.

Kapur, J. N., and H. K. Kesavan. 1992. *Entropy optimization principles with applications*. Academic Press.

L'Ecuyer, P., and Y. Champoux. 2001. Estimating Small Cell-Loss Ratios in ATM Switches via Importance Sampling. *ACM Transactions on Modeling and Computer Simulation* 11 (1): 76 – 105.

Lieber, D., and R. Rubinstein. 1998. Rare-event estimation via cross-entropy and importance sampling. Technion, Manuscript.

Lieber, D., R. Y. Rubinstein, and D. Elmakis. 1997. Quick estimation of rare events in stochastic networks. *IEEE Transaction on Reliability* 46:254–265.

Morris, C. N. 1982. Natural exponential families with quadratic variance functions. *The Annals of Statistics* 10:65–80.

Rubinstein, R. Y. 1997. Optimization of computer simulation models with rare events. *European Journal of Operations Research* 99:89–112.

Rubinstein, R. Y. 1999. The simulated entropy method for combinatorial and continuous optimization. *Methodology and Computing in Applied Probability* 2:127–190.

Rubinstein, R. Y. 2001a. Combinatorial optimization, cross-entropy, ants and rare events. In *Stochastic Optimization: Algorithms and Applications*, ed. S. Uryasev and P. M. Pardalos, 304–358: Kluwer.

Rubinstein, R. Y. 2001b. Combinatorial optimization via cross- entropy. In *Encyclopedia of Operations Research and Management Sciences*, ed. S. Gass and C. Harris, 102–106: Kluwer.

Rubinstein, R. Y. 2002. The cross-entropy method and rare-events for maximal cut and bipartition problems. Manuscript, Technion, Haifa, Israel; to be published in ACM Transactions on Modelling and Computer Simulation.

Rubinstein, R. Y., and B. Melamed. 1998. *Modern simulation and modeling*. New York: Wiley.

Rubinstein, R. Y., and A. Shapiro. 1993. *Discrete event systems: Sensitivity analysis and stochastic optimization via the score function method*. Wiley.

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

AUTHOR BIOGRAPHIES

PIETER-TJERK DE BOER holds a Ph.D. degree in Computer Science (working on performance analysis) and an M.S. degree in Applied Physics (working on theoretical physics) from the University of Twente, The Netherlands. Since 2001, he has been an assistant professor in the Department of Computer Science, University of Twente. His research interests include rare event simulation, importance sampling, queueing theory, and large deviations theory, with applications to performance analysis of telecommunication networks. His e-mail address is <ptdeboer@cs.utwente.nl> and his web page is at <http://www.cs.utwente.nl/~ptdeboer/>.

DIRK P. KROESE holds a PhD degree in Mathematical Sciences from the University of Twente, The Netherlands. He has held faculty and research staff positions at the University of Twente, Princeton University, U.S.A. and the University of Melbourne, Australia. From 1998 to 2000 he was a Senior Research Fellow at the Teletraffic Research Centre in Adelaide, Australia. Currently he is a Lecturer in Statistics at the University of Queensland, Australia. His interests include randomised algorithms, efficient simulation, the Cross-Entropy method, queueing theory and performance analysis and computational biology. His e-mail address is <kroese@maths.uq.edu.au> and his web address is <http://www.maths.uq.edu.au/~kroese/>.

REUVEN Y. RUBINSTEIN Prof. Reuven Rubinstein is with the Faculty of Industrial Engineering and Management of the Technion since 1973. His fields of interest are stochasic models, stochastic optimization and simulation. He published over 80 papers and 4 books on simulation and stochastic optimization, all with Wiley. He was the head of operations research division at the Technion for 4 years. He has visited many universities and research centers around the world, among them University of Illinois, Urbana (1978–79 academic year), Harvard University (1985–86 academic year), George Washington University (1986–87 academic year), IBM Research Center (Summer 1980), Bell Laboratories, Holmdel, NJ (Summers 1989 and 1990), NEC (February 1992). Motorola US (1997, 6 months), The Institute of Statistical Mathematics (1997-98, 4 months, Tokyo). He is a Technion Management Chair Professor since 1998. His e-mail address is <ierrr01@ie.technion.ac.il> and his web page is <http://iew3.technion.ac.il:8080/ierrr01.phtml>.