Telecommun Syst (2016) 62:289–302
DOI 10.1007/s11235-015-0055-0

Study on estimating probabilities of buffer overflow in high-speed communication networks

Izabella Lokshina[1]

Published online: 13 May 2015

Abstract The paper recommends new methods to estimate effectively the probabilities of buffer overflow in high-speed communication networks. The probability of buffer overflow in queuing system is very small; therefore the overflow is defined as a rare event and can be estimated using rare event simulation with continuous-time Markov chains. First, a two-node queuing system is considered and the buffer overflow at the second node is studied. Two efficient rare event simulation algorithms, based on the Importance sampling and Cross-entropy methods, are developed and applied to accelerate the buffer overflow simulation with Markov chain modeling. Then, the buffer overflow in self-similar queuing system is studied and simulations with long-range dependent self-similar traffic source models are conducted. A new efficient simulation algorithm, based on the RESTART method with limited relative error technique, is developed and applied to accelerate the buffer overflow simulation with SSM/M/1/B modeling using different parameters of arrival processes and different buffer sizes. Numerical examples and simulation results are provided for all methods to estimate the probabilities of buffer overflow, proposed in this paper.

Keywords High-speed communication networks · Estimating probability of buffer overflow · Two-node queuing system with feedback · Importance sampling method · Cross-entropy method · Self-similar queuing system · RESTART method with limited relative error technique

✉ Izabella Lokshina
lokshiiv@oneonta.edu

[1] Department of Management, Marketing and Information Systems, SUNY, Oneonta, NY 13820, USA

1 Introduction

Since networks must be designed and provisioned sufficiently due to requirements for high quality of service (QoS) levels, overflows in high-speed communication networks are uncommon and defined as rare events. The probability of rare events is very small, e.g. 10^{-6} or less; however, rare event probability can be used to simulate, estimate and analyze many queuing system characteristics. Queuing systems are appropriate reference models being used in different methodologies and techniques to accelerate rare event simulation in high-speed communications networks. Estimation of rare event probability using Monte Carlo simulation requires a very long computing time and cannot easily be implemented [2,4,19]. Lately two basic methods of the rare event simulation were developed based on cross-entropy approach that can be applied to a wide range of optimization tasks [6,9,10]:

- Splitting of the sample path that to reach definite intermediate level between the starting level and rare event [5]; and
- importance sampling (IS) generation [3].

The probability density function (PDF) is used in the IS approach as a rare event evaluation measure, which can be compared and changed based on the likelihood ratio of the less rare event PDF [17].

One of the rare event simulation objectives is estimating total network population overflow. Exact large deviation analysis leading to asymptotically efficient change of measure is rather difficult. Instead, heuristic change of measure is proposed, which interchanges the arrival rate to the first queue and the slowest service rate. A similar change of measure is suggested based on time reversal arguments. However, analysis shows that the IS estimator based on this change

of measure is not necessarily asymptotically efficient. In fact, it has an infinite variance in some parameter regions [10].

Another rare event simulation objective is estimating buffer overflow observed at individual network nodes. This objective is the purpose of our study. If the node of concern is a bottleneck relative to all preceding nodes, then asymptotically efficient exponential change of measure can be obtained by interchanging the arrival rate and the service rate at this particular node; and the service rates at all other nodes are kept unchanged [11,12].

However, this change of measure is not asymptotically efficient when overflow of buffer at the consequent node is considered. Effective bandwidth is used to derive heuristics for an efficient feed-forward discrete-time queuing network simulation. This class of networks essentially resembles feed-forward fluid-flow networks. The analogous approach to continuous-time queuing networks has not yet been introduced even for Markov chains.

Initially, two-node queuing systems are considered in this paper; and the event of buffer overflow at the second node is studied. The discrete-time Markov chain (DTMC) with its regular structure is highly efficient model used for performance evaluation of the queuing system. On one hand, the states are easily arranged as a grid in the DTMC (with as many dimensions as the number of queues). On the other hand, any transition in the DTMC corresponds to a particular elementary event at one of the queues (e.g., arrival or service completion). These events are known as transition events, and they are defined separately from the states; i.e., there is only one transition event for a service completion at a given queue, and this particular transition event corresponds to a transition out of each state in the DTMC while the particular queue is non-empty [13,16].

However, not all transition events are enabled in every state. For example, the service completion event of the particular queue is not possible in a state where the queue is empty.

In this paper we propose a new simulation method based on Markov additive continuous-time process (MAP) modeling. We develop and apply an Importance sampling algorithm with exponential tilting of the unbiased PDF estimation to the appropriate MAP representation, which lets us estimate effectively the probability of buffer overflow at the second node. Unlike several heuristic changes of measure described in the literature, the derived change of measure depends on the content of the first buffer.

When the first buffer is finite, we confirm that the proposed simulation procedure yields the estimation with a relative error that is bound independent of the buffer overflow level. This result is much stronger than the asymptotic efficiency, which cannot be observed with other known methods.

When the first buffer is infinite, we propose a natural extension of change of measure for finite buffer case. Applying the orthogonal polynomial model we obtain two types of simulation behavior. When the second buffer is a bottleneck, we confirm once more that simulation yields the estimation with a relative error that is bound independent of the buffer overflow level. However, when the first buffer is a bottleneck, the simulation results prove that the relative error is asymptotic and linearly bound to the buffer overflow level.

Finally, long-range dependent self-similar queuing systems are considered in this paper; and the event of buffer overflow is studied. We propose a new steady-state simulation method, based on SSM/M/1/B modeling. We develop and apply a RESTART with limited relative error (LRE) algorithm, which lets us estimate effectively the probability of buffer overflow in a long-range dependent self-similar queuing system with different parameters of arrival processes and different buffer sizes.

Overall, estimating the probabilities of rare events using traditional simulation methods is computationally challenging and extremely time consuming. This paper recommends new methods to estimate effectively the probability of buffer overflow in high-speed communication networks and contributes to the literature by developing algorithms for accelerating buffer overflow simulations.

2 Estimating probability of buffer overflow in continuous time queueing systems

A Markov additive continuous-time process is a stochastic process $(J_t,\ Z_t)$, where (J_t) is Markov chain with the denumerable state space, and (Z_t) has stationary and independent increments during the time intervals when (J_t) is in any given state. That is, if the given J_t has not changed in the interval $(t_1,\ t_2)$, then for any $t_1 < s_1 < \cdots < s_n < t_2$, the increments $Z_{s_2} - Z_{s_1}, \ldots, Z_{s_n} - Z_{s_{n-1}}$ are mutually independent, and the total increment during the interval $[t_1,\ t_2]$ depends on t_1 and t_2 only through the difference $t_2 - t_1$.

Moreover, the transition from state i to state j (J_t) has a certain probability (depending only on i and j) of triggering the transition of (Z_t) at the same time. The size of the transition in the process (Z_t) has a fixed distribution, which depends only on i and j.

A Markov additive continuous-time process $(J_t,\ Z_t)$ is characterized by the family of the matrices $(M_t(s), t \geq 0)$, where $(i,\ j)$-th element of $M_t(s)$ is (1),

$$[M_t(s)]_{ij} = \mathrm{E}_i \left[e^{s(Z_t - Z_0)} I_{\{J_t = j\}} \right] \tag{1}$$

where E_i denotes the expectation operator given to the initial MAP state $J_0 = i$. Let us notice that $M_t(\cdot)$ is a generaliza-

tion of the moment generating function for ordinary random variables, as shown in (2).

$$\begin{aligned}
&\mathrm{E}_i\left[e^{s(Z_{t+h}-Z_0)}I_{\{J_{t+h}=j\}}\right] \\
&= \sum_k \mathrm{E}_i\left[e^{s(Z_{t+h}-Z_0)}I_{\{J_t=k\}}I_{\{J_{t+h}=j\}}\right] \\
&= \sum_k \mathrm{E}_i\left[e^{s(Z_t-Z_0)}I_{\{J_t=k\}}\right] \\
&\quad \mathrm{E}_i\left[e^{s(Z_{t+h}-Z_t)}I_{\{J_{t+h}=j\}}|J_t=k\right] \\
&= \sum_k [M_t(\mathrm{s})]_{ik}\mathrm{E}_k\left[e^{s(Z_h-Z_0)}I_{\{J_h=j\}}\right] \qquad (2)
\end{aligned}$$

Consequently, if for all k and j (3) can be defined,

$$[A(s)]_{kj} := \lim_{h\downarrow 0}\frac{1}{h}\mathrm{E}_k\left[e^{s(Z_h-Z_0)}I_{\{J_h=j\}}-\delta_{kj}\right] \qquad (3)$$

where δ is usual notation of Dirac (delta function vs. inner product), then (4) can be easily obtained,

$$\frac{d}{dt}M_t(s) = M_t(s)A(s), \qquad t \geq 0 \qquad (4)$$

with $M_0(s) = I$ (the identity matrix). It is true as soon as (5) is true.

$$M_t(s) = e^{tA(s)}, \qquad t \geq 0 \qquad (5)$$

The matrix $A(s)$ is known as the MAP (infinitesimal) generator.

Let us consider a simple Markov chain that consists of two queues in tandem. The calls arrive to the first queue (e.g., buffer) according to Poisson process with the rate λ. The departure time is exponentially distributed with the rate μ_1. The calls that leave the first queue enter the second queue. The departure time at the second queue has an exponential distribution with the rate μ_2.

The queuing system stability is assumed, i.e. $\lambda < \min\{\mu_1, \mu_2\}$. The size of the first buffer may be finite or infinite; in fact, let us consider both cases. Let X_t and Y_t denote the number of the calls in the first and the second queues at the time t, respectively. Let P_i denote the probability measure under which (X_t) starts from i at the time 0 (i.e., $X_0 = i$, $i \geq 0$); and let E_i denote the corresponding expectation operator.

Assuming that the second buffer is initially non-empty, say, $Y_0 = 1$, the probability that, starting from $(X_0, Y_0) = (i, 1)$, content of the second buffer hits some high level $\mathrm{L} \in \mathrm{N}$ before hitting 0, can be estimated. This probability is noted by γ_i and referred to it as the second buffer overflow probability, given that the initial number of calls in the first queue is i.

2.1 Rare event simulation with Markov chain modeling

Let us consider the IS approach for the rare event simulation, where the probability density function is used as the measure of rare event evaluation, which is compared and changed with the likelihood ratio of the probability density function of a less rare event. First, let us determine the rare event. Let $\mathbf{X} = (\mathbf{X}_1, \ldots, \mathbf{X}_N)$ be a random vector, which values belong to the certain state space χ. Let $\{f(\cdot, \mathbf{v})\}$ be the family of the probability density functions on χ, with respect to some base measure υ. Here $\mathbf{v}$ is a real-valued parameter (vector). Then, for any measurable function H is obtained as (6).

$$\mathrm{E}\left[H(\mathbf{X})\int H(\mathbf{x})f(\mathbf{x};\mathbf{v})\upsilon(d\mathbf{x})\right] \qquad (6)$$

In many cases f is often called the probability mass function (PMF), but in this paper the generic term density, or the probability density function (PDF), is used. Let S be some real function on χ. The probability that $\mathrm{S}(\mathbf{X})$ is greater than some real number γ, under $\mathrm{f}(\cdot; \mathbf{u})$ can be defined. Therefore, probability can be written as (7),

$$l = \mathrm{P}_{\mathbf{u}}(S(\mathbf{X}) \geq \gamma) = \mathrm{E}_{\mathbf{u}}[I_{\{S(\mathbf{X})\geq\gamma\}}] \qquad (7)$$

where $I_{\{S(\mathbf{X})\geq\gamma\}}$ is the indicator function. If this probability is very small, like 10^{-6} or less, then $\{S(\mathbf{X}) \geq \gamma\}$ is a rare event. The simplest way to estimate l is to use the basic Monte Carlo simulation. Draw a random sample $\mathbf{X}_1, \ldots, \mathbf{X}_\mathrm{N}$ from $f(\cdot; \mathbf{u})$, and use (8) as the unbiased estimator of l.

$$\frac{1}{N}\sum_{i=1}^{N} I_{\{S(\mathbf{X}_i)\geq\gamma\}} \qquad (8)$$

However, it poses serious problems when $\{S(\mathbf{X}) \geq \gamma\}$ is a rare event. In that case a large simulation effort is required in order to estimate l accurately. An alternative is based on the IS. Take a random sample $\mathbf{X}_1, \ldots, \mathbf{X}_\mathrm{N}$ from the IS density g on χ, and evaluate l using the unbiased estimator, called the likelihood ratio estimator (9).

$$\hat{l} = \frac{1}{N}\sum_{i=1}^{N} I_{\{S(\mathbf{X}_i)\geq\gamma\}}\frac{f(\mathbf{X}_i;\mathbf{u})}{g(\mathbf{X}_i)} \qquad (9)$$

It is well known that the optimal way to estimate l is to use the change of the measure with the density (10)

$$g^*(\mathbf{x}) = \frac{I_{\{S(\mathbf{x})\geq\gamma\}}f(\mathbf{x};\mathbf{u})}{l} \qquad (10)$$

Specifically, applying this change of the measure to (9), (11) can be obtained for all i.

$$I_{\{S(\mathbf{X}_i)\geq\gamma\}}\frac{f(\mathbf{X}_i;\mathbf{u})}{g^*(\mathbf{X}_i)} = l \qquad (11)$$

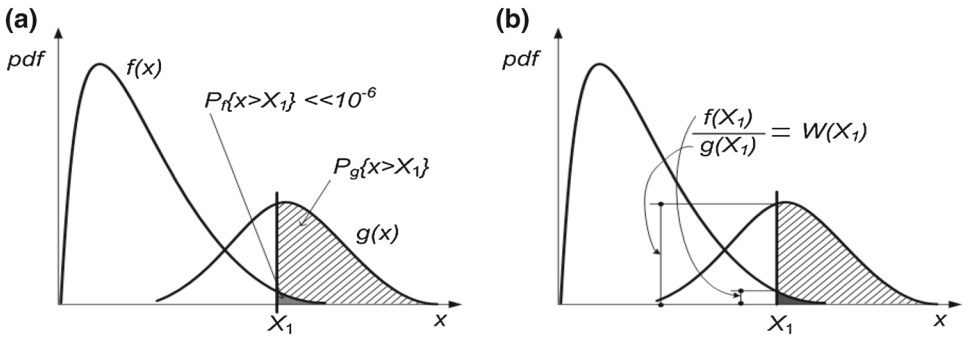

Fig. 1 Conditional probability of rare event $P_f\{x > X_1\}$ (**a**); and its acceleration with likelihood ratio $W(X_1)$ (**b**)

In other words, the estimator (9) has a zero variance, and only $N = 1$ sample needs to be produced. The obvious difficulty is, of course, that the g* depends on the unknown parameter l. Moreover, one often wishes to choose this g from the density family $\{f(\cdot, \mathbf{v})\}$. Now the plan is to choose the tilting parameter **v**, such that the distance between the densities g* and $f(\cdot, \mathbf{v})$ is minimal.

A particular suitable measure of the distance between two densities g and f is the Kullback-Leibler distance, which is defined in (12).

$$D(g, f) = \mathrm{E}_g\left[\ln \frac{g(\mathbf{X})}{f(\mathbf{X})}\right] = \int g(\mathbf{x}) \ln g(\mathbf{x}) \upsilon(d\mathbf{x}) - \int g(\mathbf{x}) \ln f(\mathbf{x}) \upsilon(d\mathbf{x}) \tag{12}$$

Therefore, minimizing the Kullback-Leibler distance between g in (11) and $f(\cdot, \mathbf{v})$ is the same as choosing **v**, such that $-\int g(\mathbf{x}) \ln f(\mathbf{x}; \mathbf{v}) \upsilon(d\mathbf{x})$ is minimized, or equivalently, such that $\int g(\mathbf{x}) \ln f(\mathbf{x}; \mathbf{v}) \upsilon(d\mathbf{x})$ is maximized. Formally, it can be written as (13).

$$\max_{\mathbf{v}} D(\upsilon) = \max_{\mathbf{v}} \int g(\mathbf{x}) \ln f(\mathbf{x}; \mathbf{v}) \upsilon(d\mathbf{x}) \tag{13}$$

Applying g from (10) to (13) as the substitution, the following optimization program can be obtained as (14).

$$\max_{\mathbf{v}} D(\mathbf{v}) = \max_{\mathbf{v}} \int \frac{I_{\{S(\mathbf{x}) \geq \gamma\}} f(\mathbf{x}; \mathbf{u})}{l} \ln f(\mathbf{x}; \mathbf{v}) \upsilon(d\mathbf{x}) = \max_{\mathbf{v}} \mathrm{E}_{\mathbf{u}}\left[I_{\{S(\mathbf{X}) \geq \gamma\}} \ln f(\mathbf{X}; \mathbf{v})\right] \tag{14}$$

Using the IS again, with the change of measure $f(\cdot; \mathbf{w})$, (14) can be re-written into (15),

$$\max_{\mathbf{v}} D(\mathbf{v}) = \max_{\mathbf{v}} \mathrm{E}_{\mathbf{w}}\left[I_{\{S(\mathbf{X}) \geq \gamma\}} W(\mathbf{X}; \mathbf{u}, \mathbf{w}) \ln f(\mathbf{X}; \mathbf{v})\right] \tag{15}$$

for any tilting parameter **w**, where the likelihood ratio at **x** between $f(\cdot; \mathbf{u})$ and $f(\cdot; \mathbf{w})$ is W. This can be presented according to (16).

$$W(\mathbf{x}; \mathbf{u}, \mathbf{w}) = \frac{f(\mathbf{x}; \mathbf{u})}{f(\mathbf{x}; \mathbf{w})} \tag{16}$$

The basic idea of the IS method regarding rare events is to accelerate its frequency with iterative tilting of the unbiased estimation of the probability density function to appropriate MAP representation.

The acceleration of conditional probability of rare event X_1 with one-parameter function f(x) is shown in Fig. 1. The conditional probability of rare event $P_f\{x > X_1\}$ is changing with conditional probability of $g(x) - P_g\{x > X_1\}$.

At each iteration of the IS simulation, N independent samples are generated, which distribution g(x) can be evaluated with the likelihood ratio $W(X_1)$. The optimal solution for (15) can be written as (17).

$$\mathbf{v}^* = \underset{\mathbf{v}}{\mathrm{argmax}} \mathrm{E}_{\mathbf{w}}\left[I_{\{S(\mathbf{X}) \geq \gamma\}} W(\mathbf{X}; \mathbf{u}, \mathbf{w}) \ln f(\mathbf{X}; \mathbf{v})\right] \tag{17}$$

It can be obtained by solving the following stochastic program, which can be considered as a stochastic counterpart of (15) and written according to (18),

$$\max_{\mathbf{v}} \hat{D}(\mathbf{v}) = \max_{\mathbf{v}} \frac{1}{N} \sum_{i=1}^{N} I_{\{S(\mathbf{X}) \geq \gamma\}} W(\mathbf{X}_i; \mathbf{u}, \mathbf{w}) \ln f(\mathbf{X}_i; \mathbf{v}) \tag{18}$$

where $\mathbf{X}_1, \ldots, \mathbf{X}_N$ is a random sample from $f(\cdot; \mathbf{w})$.

The solution for (18) can be obtained by solving the following system of equations with respect to **v** in (19),

$$\frac{1}{N} \sum_{i=1}^{N} I_{\{S(\mathbf{X}_i) \geq \gamma\}} W(\mathbf{X}_i; \mathbf{u}, \mathbf{w}) \nabla \ln f(\mathbf{X}_i; \mathbf{v}) = \mathbf{0} \tag{19}$$

where the gradient is defined regarding **v**.

This confirms the expectation that the differentiation operators and the function $\hat{D}$ can be interchanged in (18) with respect to **v**. The advantage of this approach is in the fact that often the solution of (19) can be calculated analytically. In particular, this happens when the random variable distributions belong to the natural exponential family. The cross-entropy program (19) is useful only when the probability of the target event $\{S(\mathbf{X}) \geq \gamma\}$ is not too small, say $l \geq 10^{-5}$.

Then, the above program is useful in terms of potentially more accurate estimator determination. However, in a rare

event context, (say, $l \leq 10^{-6}$), the program (19) is useless to rarity of events $\{S(\mathbf{X}_i) \geq \gamma\}$, because the random variables $I_{\{S(\mathbf{X}_i)\geq\gamma\}}$, i = 1, . . . , N and the associated derivatives of $\hat{D}(\mathbf{v})$ vanish with high probability, as given in the right-hand side of (19), for reasonable sizes of N.

2.2 Exponential change of measure

Let us initially think that the first buffer has a finite capacity N. In this case the state space of the driving process (X_t) is finite in {0, . . . , N}. Let us consider Markov additive continuous-time process (X_t, Z_t). To create a corresponding MAP generator (i.e., matrix A(s) in (5)), the infinitesimal expectations $E_i[e^{s(Z_h-Z_0)}I_{\{X_h=j\}} - \delta_{ij}]$ as h ↓ 0, for all i, j in {0, . . . , N} have to be determined, where $Z_0 = 1$ and $\delta_{ij} = 0$ for j ≠ i.

For instance, since the downward transition of (X_t) leads to the upward transition of (Z_t), (20) is used for i = 1, . . . , N, as h ↓ 0.

$$\begin{aligned} &E_i\left[e^{s(Z_h-Z_0)}I_{\{X_h=i-1\}} - \delta_{i,i-1}\right] \\ &= E_i\left[e^{s(Z_h-Z_0)}|X_h = i-1\right]P_i(X_h = i-1) \\ &= e^s(\mu_1 h + o(h)) = \mu_1 h e^s + o(h) \end{aligned} \tag{20}$$

Therefore, the (i,i-1)-th element of the matrix A(s) exists and is equal to $\mu_1 e^s$. Other elements of the matrix A(s) can be defined similarly. Consequently, (5) holds with A(s) for Markov additive continuous-time process (X_t, Z_t) with the given (N + 1, N + 1)-tri-diagonal matrix (21).

$$\mathbf{G}_N(s) = \begin{pmatrix} -\lambda-\mu_2+\mu_2 e^{-s} & \lambda & & \\ \mu_1 e^s & -\lambda-\mu_1-\mu_2+\mu_2 e^{-s} & \lambda & \\ & \ddots & \ddots & \ddots \\ & & \mu_1 e^s & -\mu_1-\mu_2+\mu_2 e^{-s} \end{pmatrix} \tag{21}$$

Let us note that the MAP generator $G_N(s)$ is now equal to the matrix (22).

$$\hat{\mathbf{Q}}^{(n)}(u) = \begin{pmatrix} -\lambda-\mu_2+\mu_2 u & \lambda & & \\ \mu_1/u & -\lambda-\mu_1-\mu_2+\mu_2 u & \lambda & \\ & \ddots & \ddots & \ddots \\ & & \mu_1/u & -\mu_1-\mu_2+\mu_2 e^{-s} \end{pmatrix} \tag{22}$$

Next, the change of the measure based on the family of the matrices $\mathbf{G}_N(s)$ is defined. For any s ≥ 0, $k_N(s) := \log(sp(M_t(s)))/t$ has to be defined, where $sp(M_t(s))$ denotes the spectral radius (or, the maximum Eigen value) of $M_t(s)$. Using (5) $k_N(s)$ can be identified as the largest positive Eigen value of $\mathbf{G}_N(s)$. Let $\mathbf{w}(s) = \{w_k(s), 0 \leq k \leq N\}$ represent the correspondent right-eigenvector.

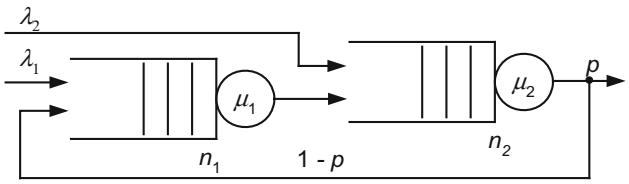

Fig. 2 Two-node queuing network with feedback

When the first buffer has the infinite capacity, the process (X_t, Z_t) is still Markov additive continuous-time process, but the state space for Markov process (X_t) is now infinite. Equation (5) is still used, but A(s) is now given by the infinite-dimensional tri-diagonal matrix (23).

$$\mathbf{G}(s) = \begin{pmatrix} -\lambda-\mu_2+\mu_2 e^{-s} & \lambda & & \\ \mu_1 e^s & -\lambda-\mu_1-\mu_2+\mu_2 e^{-s} & \lambda & \\ & \ddots & \ddots & \ddots \end{pmatrix} \tag{23}$$

Let us note that the MAP generator G(s) is now equal to the infinite tri-diagonal matrix (24).

$$\mathbf{Q}(u) = \begin{pmatrix} -\lambda-\mu_2+\mu_2 u & \lambda & & \\ \mu_1/u & -\lambda-\mu_1-\mu_2+\mu_2 u & \lambda & \\ & \ddots & \ddots & \ddots \end{pmatrix} \tag{24}$$

2.3 Importance sampling algorithm and simulation results

The Markovian network that consists of tandem queues with feedback is shown in Fig. 2 and used as a simulation example,

Table 1 Simulation results

Overflow level		λ_1	λ_2	λ_1	λ_2	p	Overflow probability
n_1	n_2						
25	25	0.13	0.11	0.29	0.18	0.42	2.22×10^{-8}
50	50	0.23	0.18	2.15	2.26	0.31	1.15×10^{-15}
60	40	1.18	1.42	4.91	3.58	0.44	6.29×10^{-25}
100	40	2.02	1.67	5.82	2.94	0.37	1.86×10^{-50}

with the entry parameters follow: $\lambda_1 = \lambda_2 = 1,\ \mu_1 = \mu_2 = 6,\ p = 0.5$.

The Importance Sampling (IS) algorithm to accelerate rare event simulation with Markov chain modeling in high-speed communication networks is suggested in this paper. It consists of the following six steps provided below.

Algorithm 1 Importance Sampling Simulation Method

Step 1 Set $t := 1$ *(initialization of iteration counter). Define the likelihood ratio* $v_1 := 0$ *(in this case Monte Carlo simulation is appropriate).*
Step 2 Generate a sample $\boldsymbol{X}_1, \ldots, \boldsymbol{X}_k, \ldots, \boldsymbol{X}_N$*, from the density* $f(\boldsymbol{X}_k; \boldsymbol{v}_{t-1})$ *in such a way that for the* ρ*-th part of samples* ($\rho = 0.01$) *the condition of the rare event* $S(\boldsymbol{X}_k) > M$ *is executed.*
Step 3 Determine the full paths and sort ascending in following way $S_{(1)} \leq S_{(2)} \leq \cdots \leq S_{(N)}$.
Step 4 Calculate the conditional probability $\gamma_t = S_{[(1-\rho)N]}$.
Step 5 For each $S(\boldsymbol{X}_k) > \gamma$ *define the rare event indicator* $I_{\{S(\mathbf{X}_i) \geq M\}} = 1$ *and then determine the likelihood ratio for the next iteration* $\boldsymbol{v}_{t+1}$ *according to* (15), (16) *and* (19).
Step 6 If $\gamma_t < M$ *then* $t := t + 1$ *and repeat steps 2, 3 and 4.*

The simulation results for the Poisson distribution and fixed numbers of calls n_1 and n_2, for four different overflow cases are provided in Table 1. As can be seen, the overflow probability exponentially decreases with an increase of the fixed number of calls in queues n_1 and n_2. The exponential behavior depends more on the number n_1.

2.4 Cross-entropy algorithm and simulation results

The Cross-entropy algorithm to accelerate rare event simulation with Markov chain modeling in high-speed communication networks is suggested in this paper. The idea is to introduce a sequence of reference parameters $\{\mathbf{v}_t,\ t \geq 0\}$ and a sequence of the levels $\{\gamma_t,\ t \geq 1\}$, and iterate in both γ_t and $\mathbf{v}_t$.

The initialization is done with choosing a not very small ρ, say $\rho = 10^{-2}$, and defining $\mathbf{v}_0 = \mathbf{u}$. Next, we let γ_1 ($\gamma_1 < \gamma$) be such that under the original density $f(\mathbf{x}; \mathbf{u})$, the probability $l_1 = E_{\mathbf{u}}[I_{\{S(\mathbf{X}) \geq \gamma_1\}}]$ is at least ρ.

After, let $\mathbf{v}_1$ be the optimal cross-entropy reference parameter to estimate l_1, and repeat the last two steps iteratively with the purpose to estimate the pair $\{l,\ \mathbf{v}^*\}$. In other words, the iteration of the algorithm consists of two main phases. In the first phase γ_t is updated, in the second phase $\mathbf{v}_t$ is updated. Particularly, starting with $\mathbf{v}_0 = \mathbf{u}$, the subsequent γ_t and $\mathbf{v}_t$ are obtained as described below.

Phase 1 includes adaptive update of γ_t. For a fixed $\mathbf{v}_{t-1}$, let γ_t be a $(1 - \rho)$-quintile of $S(\mathbf{X})$ under $\mathbf{v}_{t-1}$. That is, γ_t satisfies (25) and (26).

$$P_{v_{t-1}}(S(\mathbf{X}) \geq \gamma_t) = E_{\mathbf{u}}\left[I_{\{S(\mathbf{X}) \geq \gamma\}}\right] \tag{25}$$

$$P_{v_{t-1}}(S(\mathbf{X}) \leq \gamma_t) \geq 1 - \rho \tag{26}$$

where $X \sim f(\cdot; \mathbf{v}_{t-1})$. The simple estimator $\hat{\gamma}_t$ of γ_t can be obtained by drawing a random sample $\mathbf{X}_1, \ldots, \mathbf{X}_N$ from $f(\cdot;\ v_{t-1})$, calculating the performances $S(\mathbf{X}_i)$ for all i, ordering them from the smallest to the biggest: $S_{(1)} \leq \cdots \leq S_{(N)}$ and finally, evaluating the $(1 - \rho)$ sample quintile as (27).

$$\hat{\gamma}_t = S_{[(1-\rho)N]} \tag{27}$$

Let us note that $S_{(j)}$ is called jth order-statistic of the sequence $S(\mathbf{X}_1), \ldots, S(\mathbf{X}_N)$. Let us note also that $\hat{\gamma}_t$ is chosen such that the event $\left\{S(\mathbf{X}) \geq \hat{\gamma}_t\right\}$ is not too rare (it has a probability of around ρ), and therefore, the reference parameter updated with a procedure such as (27) is not void of the meaning.

Phase 2 contains adaptive update of v_t. For fixed γ_t and $\mathbf{v}_{t-1}$, let us derive $\mathbf{v}_t$ from the solution of the following cross-entropy program according to (28).

$$\max_{\mathbf{v}} D(\mathbf{v}) = \max_{\mathbf{v}} E_{\mathbf{v}_{t-1}}\left[I_{\{S(\mathbf{X}) \geq \gamma\}} W(\mathbf{X}; \mathbf{u}, \mathbf{v}_{t-1}) \ln f(\mathbf{X}; \mathbf{v})\right] \tag{28}$$

The stochastic counterpart of (28) is shown as follows: for fixed $\hat{\gamma}_t$ and $\hat{\mathbf{v}}_{t-1}$, derive $\hat{\mathbf{v}}_t$ from the solution of the program according to (29).

Table 2 Simulation results

Iteration	Repetitive trails	λ	μ_1	μ_2	p	Estimation
1	10^4	0.2	0.8	0.2	0.5	–
2	10^4	0.216	0.643	0.258	0.364	$1.426e^{-15}$
3	10^4	0.198	0.621	0.287	0.322	$1.462e^{-15}$
4	10^4	0.196	0.614	0.279	0.318	$1.436e^{-15}$
5	10^4	0.195	0.614	0.282	0.320	$1.372e^{-15}$
6	10^4	0.196	0.612	0.284	0.322	$1.318e^{-15}$
7	10^6	0.196	0.612	0.284	0.322	$1.342e^{-15}$

$$\max_{\mathbf{v}} \hat{D}(\mathbf{v}) = \max_{\mathbf{v}} \frac{1}{N} \sum_{i=1}^{N} I_{\{S(\mathbf{X}_i) \geq \hat{\gamma}_t\}} W(\mathbf{X}_i; \mathbf{u}, \hat{\mathbf{v}}_{t-1}) \ln f(\mathbf{X}_i; \mathbf{v}) \tag{29}$$

Therefore, at the first iteration starting with $\hat{\mathbf{v}}_0 = \mathbf{u}$, the target event is artificially made less rare with temporary use of the level $\hat{\gamma}_1$, which is chosen to be smaller than γ that for to get a good estimate for $\hat{\mathbf{v}}_1$.

The value $\hat{\mathbf{v}}_1$ obtained in this way will make the event $\{S(\mathbf{X}) \leq \gamma\}$ less rare in the next iteration, so the value $\hat{\gamma}_2$ can be used in the next iteration, which is closer to γ itself.

The algorithm terminates when the level is reached at some iteration t, which is at least γ and after the original value of γ can be used without getting too few samples. As mentioned before, the optimal solution of (28) and (29) can be often obtained analytically, in particular when f($\mathbf{x}$; $\mathbf{v}$) belongs to the natural exponential family.

The above rationale results are placed in the multi-level Cross-entropy algorithm for accelerated rare event simulation with Markov chain modeling in high-speed communication networks. This efficient algorithm consists of the following five steps provided below.

Algorithm 2 Multi-Level Cross-Entropy Simulation Method

Step 1 Define $\hat{\mathbf{v}}_0 = \mathbf{u}$. *Set* $t = 1$. *(Iteration = level counter).*
Step 2 Generate a sample $X_1, \ldots, X_N$ *with the density* $f(\cdot; v_{t-1})$ *and compute the sample* $(1-\rho)$ *- quintile* $\hat{\gamma}_t$ *performance according to* (28) *with* $\hat{\gamma}_t < \gamma$. *Otherwise, set* $\hat{\gamma}_t = \gamma$.
Step 3 Use the same sample $X_1, \ldots, X_N$ *to solve the stochastic program* (29). *Denote the solution by* $\hat{\mathbf{v}}_t$.
Step 4 If $\hat{\gamma}_t < \gamma$, *then set* $t = t + 1$ *and reiterate from Step 2. Else, proceed with step 5.*
Step 5 Estimate the rare event probability l using (30),

$$\hat{l} = \frac{1}{N} \sum_{i=1}^{N_1} I_{\{S(\mathbf{X}_i) \geq \gamma\}} W\left(\mathbf{X}_i; \mathbf{u}, \hat{\mathbf{v}}_T\right) \tag{30}$$

where T denotes the final number of iterations, or number of the levels used.

As a simulation example, let us apply this efficient algorithm to a similar tandem queue with feedback as was given in Fig. 2, but at this time with the following entry parameters: $\lambda = 0.2$; $\mu_1 = 0.8$; $\mu_2 = 0.2$; $p = 0.5$.

The simulation results for generating $N = 10,000$ samples are shown in Table 2. As could be seen, overflow is obtained when there are $j = 6$ iterations. The accuracy increases up to $N = 1,000,000$ in the seventh iteration, and the overflow probability is obtained equal to $\hat{\ell}_{IS} = 1.342e^{-15}$.

3 Estimating probability of buffer overflow in self-similar queuing systems

Recent studies of high-speed communication network traffic have clearly shown that teletraffic (technical term, identifying all phenomena of transport and control of information within the high-speed communication networks) exhibits long-range dependent self-similar properties over a wide range of time scales. Therefore, self-similar queuing systems are appropriate reference models being also used in different methodologies and techniques to accelerate rare event simulation in high-speed communication networks [13].

Long-range dependent self-similar teletraffic is usually observed in LAN and WAN, where superposition of strictly independent alternating ON/OFF traffic models, whose ON- or OFF-periods have heavy-tailed distributions with infinite variance, can be used to model aggregate queuing network traffic that exhibits long-range dependent self-similar behavior, typical for measured LAN traffic over a wide range of time scales [14,18].

Long-range dependent self-similar teletraffic is also observed in ATM networks: when arriving at an ATM buffer, it results in a heavy-tailed buffer occupancy distribution, and a buffer cell loss probability decreases with the buffer size not exponentially, like in traditional Markovian models, but hyperbolically [16,18].

Furthermore, long-range dependent self-similar teletraffic is observed in the Internet as many characteristics can be modeled using heavy-tailed distributions, including the distributions of traffic times, user requests for documents, and document sizes. In IP with TCP self-similar queuing networks the transfer of files or messages shows that the reliable transmission and flow control mechanisms serve to maintain long range dependent structure included by heavy-tailed file size distributions [1].

Long-range dependent self-similar video traffic provides the possibility for developing models for Variable Bit Rate (VBR) video traffic using heavy-tailed distributions [16,18]. Therefore; we can clearly see that impact of self-similar models on the queuing and network performance is very significant.

The properties of long-range dependent self-similar teletraffic are very different from the properties of traditional models based on Poisson, Markov-modulated Poisson, and related processes. More specifically, while tails of the queue length distributions in traditional teletraffic models decrease exponentially, those of long-range dependent self-similar teletraffic models decrease much slower.

Therefore, the use of traditional models in high-speed communication networks characterized by long-range dependent self-similar processes can lead to incorrect conclusions about the queuing and network performance. Traditional models can lead to over-estimation of the queuing and network performance, insufficient allocation of communication and data processing resources, and consequently difficulties in ensuring the QoS.

Self-similarity can be classified into two types: deterministic and stochastic. In the first type, deterministic self-similarity, a mathematical object is assumed to be self-similar (or fractal) if it can be decomposed into smaller copies of itself. That is, deterministic self-similarity is a property, in which the structure of the whole is contained in its parts [14,18].

This work is focused on stochastic self-similarity. In that case, probabilistic properties of self-similar processes remain unchanged or invariant when the process is viewed at different time scales. This is in contrast to Poisson processes that lose their burstiness and flatten out when time scales are changed [18].

However, the time series of self-similar processes exhibit burstiness over a wide range of time scales. Self-similarity can statistically describe teletraffic that is bursty on many time scales [14].

One can distinguish two types of stochastic self-similarity. A continuous-time stochastic process $\mathbf{Y}_t$ is strictly self-similar with a self-similarity parameter H $(1/2 < H < 1)$, if $\mathbf{Y}_{ct}$ and $c^H\mathbf{Y}_t$ (the rescaled process with time scale ct) have identical finite-dimensional probability for any positive time stretching factor c. This definition, in a sense of probability distribution, is quite different from that of the second-order self-similar process, observed at the mean, variance and autocorrelation levels [14].

The process $\mathbf{X}$ is asymptotically second-order self-similar with $0.5 < H < 1$, if for each k large enough $\rho_k^{(m)} \to \rho_k$, as $m \to \infty$, where $\rho_k = \mathrm{E}[(X_i - \mu)(X_{i+k} - \mu)]/\sigma^2$.

In this work the exact or asymptotic self-similar processes are used in an interchangeable manner, which refers to the tail behavior of the autocorrelations [14,18].

3.1 Long-range dependent self-similar processes

We have to say that the most striking feature of some second-order self-similar processes is that the accumulative functions of the aggregated processes do not degenerate with the non-overlapping batch size m increasing to infinity. Such processes are known as Long-Range Dependent (LRD) processes [1,14,18].

This is in contrast to traditional processes used in modeling high-speed communication network traffic, all of which include the property that the accumulative functions of their aggregated processes degenerate as the non-overlapping batch size m increasing to infinity, i.e., $\rho_k^{(m)} \to 0$ or $\rho_k^{(m)} = 0(|k| > 0)$, for $m > 1$. The equivalent definition of long-range dependence is given as (31).

$$\sum_{k=-\infty}^{\infty} \rho_k = \infty \tag{31}$$

Another definition of LRD is presented as (32),

$$\rho_k \sim L(t)k^{-(2-2H)}, \quad \text{as} \quad k \to \infty \tag{32}$$

where $1/2 < H < 1$ and $L(\cdot)$ slowly varies at infinity, i.e. for all $x > 0$ it could be determined as (33).

$$\lim_{t\to\infty} \frac{L(xt)}{L(t)} = 1 \tag{33}$$

The Hurst parameter H characterizes the relation in (32), which specifies the form of the tail of the accumulative function. One can show that is true for $1/2 < H < 1$, as given in (34).

$$\rho_k = \frac{1}{2}\left[(k+1)^{2H} - 2k^{2H} + (k-1)^{2H}\right] \tag{34}$$

For $0 < H < 1/2$ the process is Short-Range Dependent (SRD) and could be presented as (35).

$$\sum_{k=-\infty}^{\infty} \rho_k = 0 \tag{35}$$

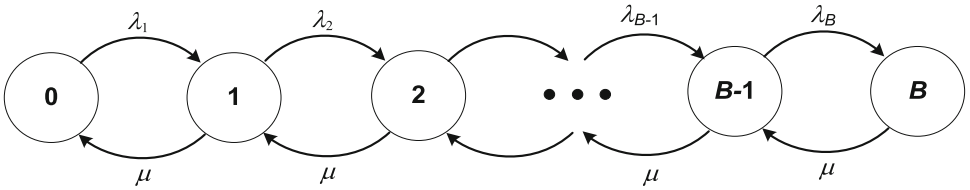

Fig. 3 State transition diagram for a SSM/M/1/B self-similar queuing system

For $H = 1$ all autocorrelation coefficients are equal to one, no matter how far apart in time the sequences are. This case has no practical importance in real high-speed communication network traffic modeling. If $H > 1$, then (36) is true.

$$\rho_k = \begin{cases} 1 & \text{for} \quad k = 0 \\ \frac{1}{2}k^{2H}g(k^{-1}) & \text{for} \quad k > 0 \end{cases} \tag{36}$$

where

$$g(x) = (1+x)^{2H} - 2 + (1-x)^{2H} \tag{37}$$

One can see that $g(x) \to \infty$ as $H > 1$. If $0 < H < 1$ and $H \neq 1/2$, then the first non-zero term in the Taylor expansion of g(x) is equal to $2H(2H-1)x^2$. Therefore, (38) is true.

$$\rho_k / \left(H(2H-1)k^{2H-2}\right) \to 1, \quad \text{as} \quad k \to \infty \tag{38}$$

In the frequency domain, an essentially equivalent definition of LRD for a process X with given spectral density (39),

$$f(\lambda) = \frac{\sigma^2}{2\pi} \sum_{k=-\infty}^{\infty} \rho_k e^{ik\lambda} \tag{39}$$

is that in the case of LRD processes, this function is required to satisfy the following property (40),

$$f(\lambda) \sim c_{f_1} \lambda^{-\gamma}, \quad \text{as} \quad \lambda \to 0 \tag{40}$$

where c_{f_1} is a positive constant and $0 < \gamma < 1$, $\gamma = 2H - 1 < 1$. As a result, LRD manifests itself in the spectral density that obeys a power-law in the vicinity of the origin. This implies that $f(0) = \Sigma_k \rho_k = \infty$. Consequently, it requires a spectral density, which tends to $+\infty$ as the frequency λ approaches 0.

For a Fractional Gaussian Noise (FGN) process, the spectral density $f(\lambda, H)$ is given by (41),

$$f(\lambda, H) = 2c_f(1 - \cos(\lambda))B(\lambda, H) \tag{41}$$

with $0 < H < 1$ and $-\pi \leq \lambda \leq \pi$, where (42) is true,

$$c_f = \sigma^2(2\pi)^{-1} \sin(\pi H)\Gamma(2H+1)$$
$$B(\lambda, H) = \sum_{k=-\infty}^{\infty} |2\pi k + \lambda|^{-2H-1} \tag{42}$$

and $\sigma^2 = \text{Var}[X_k]$ and $\Gamma(\cdot)$ is the gamma function. The spectral density $f(\lambda, H)$ in (41) complies with a power-law at the origin, as shown in (43),

$$f(\lambda, H) \to c_f \lambda^{1-2H}, \quad \text{as} \quad \lambda \to 0 \tag{43}$$

where $1/2 < H < 1$.

3.2 Steady-state simulation with SSM/M/1/B modeling

As we have previously accepted, there is a significant difference in the queuing and network performance between traditional models of teletraffic, such as Poisson processes and Markovian processes, and those exhibiting long-range dependent self-similar behavior. More specifically, while tails of the queue length distributions in traditional models of high-speed communication network traffic decrease exponentially, those of self-similar traffic models decrease much slower ([14,18]).

Let us consider the potential impacts of traffic characteristics, including the effects of long-range dependent self-similar behavior on queuing and network performance, protocol analysis, and network congestion controls. Steady-state simulation of long-range dependent self-similar queuing system includes:

- Generation of long-range dependent self-similar traffic [14,18];
- Simulation of long-range dependent self-similar queuing process [14]; and
- Simulation of the overflow probability [14,15].

This can be demonstrated with the buffer overflow simulation in SSM/M/1/B queuing systems ($B < \infty$, i.e. queuing systems with the finite buffer capacity) with long-range dependent self-similar queuing processes. In this case, the difference with M/M/1/B queuing system is that the arrival rate λ_j into SSM/M/1/B queuing system is not a constant value. It depends on the sequential number of time-series i, the total number of observations n and the Hurst parameter H, which determine the rate of self-similarity. The analyzed SSM/M/1/B queuing system has exponential service times with constant rates $1/\mu$ as is shown in Fig. 3. The flow balance equations are given below [8,14]:

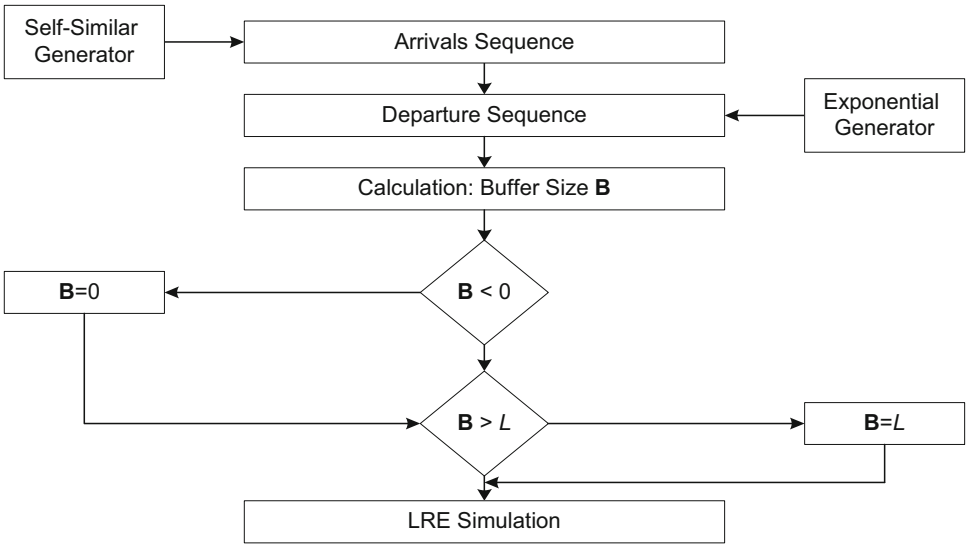

Fig. 4 Steady-state simulation of self-similar queuing system

$$\begin{aligned} &\lambda_j = \lambda\,(i, n, H)\,; \quad j = 1, 2, \ldots, B \\ &\lambda_j = 0; \qquad\qquad j \geq B + 1 \\ &\mu_j = \mu; \qquad\qquad j = 1, 2, \ldots, B + 1 \end{aligned} \tag{44}$$

This system is stable with a throughput $\rho = \frac{\lambda(i,n,H)}{\mu} < 1$. Let us consider two separated cases: $\rho = 1$, and $\rho \neq 1$. For $j = 0, 1, 2, \ldots, B$ the distribution of the number of flows in the system is $P_j = \rho^j P_0$, which is determined according to

$$\begin{aligned} &P_j = \frac{\rho^j\,(1-\rho)}{1-\rho^{B+1}}; \quad \rho \neq 1 \\ &P_j = \frac{1}{B+1}; \quad \rho = 1 \end{aligned} \tag{45}$$

Therefore, the rate at which the flows are blocked and lost is λP_B. The self-similar queuing process is described with the steady-state simulation procedure [16], presented in Fig. 4.

The self-similar traffic can be generated and the sequence of arrivals is obtained. The fixed length of self-similar traffic is extracted by fixing the number of observations. As the service process is Markovian, the sequence of departures has exponential distribution, generated with an inverse transform generator [14].

The next step is the calculation of the buffer size. If the service size is greater than the size of arrivals, then the buffer size $B = 0$, as it is impossible to have a negative buffer size. In cases when the buffer size is greater than the overflow L, i.e. $B > L$, the traffic is lost, therefore we have made an assumption that $B = L$.

The simulation is performed with splitting of the sample path [15], using a variant based on the RESTART method [7], where any chain is split by a fixed factor when it hits a level upward, and one of the copies is tagged as the original for that simulation level. When any of those copies hits that same level downward, if the copy is the original it just continues its path, otherwise it is killed immediately. This rule applies recursively, and the method is implemented in a depth-first fashion, as follows: whenever there is a split, all the non-original copies are simulated completely, one after the other; then the simulation continues for the original chain [20].

The reason for eliminating most of the paths that go downward is to reduce the work. Therefore, the buffer size calculations being made for all sequences provide the opportunity to estimate the overflow probability using the steady-state simulation based on the RESTART method with the limited relative error (LRE) algorithm.

3.3 RESTART method with limited relative error algorithm and simulation results

The limited relative error algorithm helps to determine the complementary cumulative function of arrivals at single server buffer queues with Markov processes. In order to describe the LRE principles for steady-state simulation in Discrete-Time Markov Chains (DTMC), let us consider a homogeneous two-node Markov chain, which is extended to regular DTMC, consisting of $(k + 1)$ nodes with states, respectively $S_0, S_1, \ldots, S_k$, as shown in Fig. 5.

We obtain the random generated sequence $x_1, x_2, \ldots, x_t$, $x_{t+1}, \ldots$ for $x = 0, 1, \ldots, k$, for which a transition for state S_j at the time t exists, e.g. $x_t = j$ and there are no constraints to the parameters of the transition probabilities:

$$p_{ij} = \mathrm{P}\,(j|i)\,; \quad (i, j) = 0, 1, \ldots, k; \quad \sum_{j=1}^{k} p_{ij} = 1 \tag{46}$$

There are no absorbing states S_i at $p_{ii} = 1$ for all stationary probabilities $\mathrm{P}_j\, j = 0, 1, \ldots, k$, which satisfy the constraint condition:

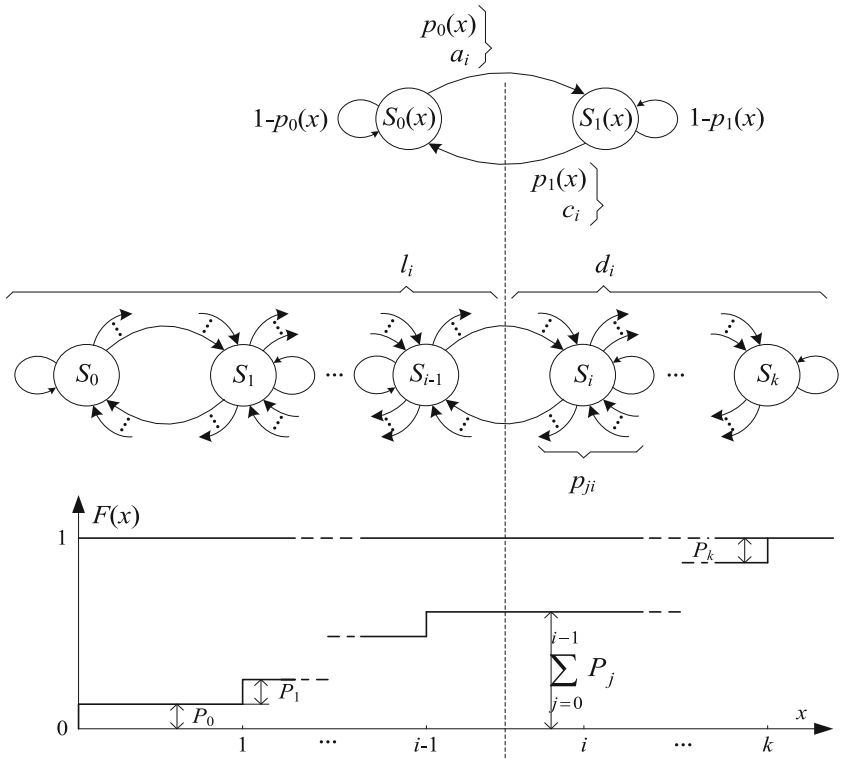

Fig. 5 Cumulative function F(x) for $(k+1)$-node Markov chain

$$0 \le \mathrm{P}_j < 1; \qquad \sum_{j=0}^{k} \mathrm{P}_j = 1 \tag{47}$$

The cumulative distribution $F(x)$ can be presented as:

$$\left.\begin{aligned} &F(x) = F_i; \ \ (i-1) \le x < i; \quad i = 1, 2, \ldots, k+1; \\ &F_i = \sum_{j=0}^{i-1} \mathrm{P}_j; \ F_0 = 0; \quad F_{k+1} = 1; \end{aligned}\right\} \tag{48}$$

In order to simulate the $(k+1)$ nodes of Markov chain, the complementary cumulative distribution $G(x) = 1 - F(x)$ that is more significant, can be determined along with the local correlation coefficient $\rho(x)$ through the limited relative error approach.

After having the homogeneous two-node Markov chain defined as shown in Fig. 3, with changing the states n times, an estimation of the local correlation coefficient $\hat{\rho}(x)$ can be obtained, which connects the number of transitions through a dividing line $a_i \approx c_i$, with the total number of observed events $l_i = n - d_i$ $(\beta = 0, 1, \ldots, i-1)$ at left side, and d_i at right side $(\beta = i, i+2, \ldots, k)$.

The value of simulated complementary cumulative distribution $\hat{G}_i$ can be defined directly by using relative frequency d_i/n, if there is enough number of samples:

$$n \ge 10^3; \qquad \left(l_i, d_i \ge 10^2\right); (a_i, c_i, l_i - a_i, d_i - c_i) \ge 10 \tag{49}$$

The posterior equations can be used for the complementary function $\hat{G}(x)$, the average number of generated values of $\hat{\beta}$, the local correlation coefficient $\hat{\rho}(x)$, the correlation coefficient Cor$[x]$ and the relative error RE$[x]$:

$$\begin{aligned} &\hat{G}(x) = \hat{G}_i = d_i/n \quad \hat{\beta} = \frac{1}{n}\sum_{i=1}^{k} d_i \\ &\hat{\rho}(x) = \hat{\rho}_i = 1 - \frac{c_i/d_i}{1 - d_i/n} \qquad \begin{matrix} i-1 \le x < i \\ i = 1, \ldots, k \end{matrix} \\ &\mathrm{Cor}[x] = \mathrm{Cor}_i = (1+\hat{\rho}_i)(1-\hat{\rho}_i) \qquad \mathrm{RE}[x]^2 = \mathrm{RE}_i \\ &\qquad\qquad = \frac{1 - d_i/n}{d_i} \cdot \mathrm{Cor}_i \end{aligned} \tag{50}$$

The main advantage of this approach is that the relationships between transitions c_i are obtained with routine statistical calculations. The necessary total number of simulation trails n is determined with the maximal relative error $\mathrm{RE}_{\max}[x]^2$ and with the less value of the function $G(x)$, presented as $G_{\min} = \hat{G}_k$ in approximation equation:

$$\begin{aligned} &n = \frac{(1 - G_{\min})}{G_{\min}.\mathrm{RE}_{\max}[x]^2} \approx \frac{\mathrm{Cor}_k}{\hat{G}_k.\mathrm{RE}_{\max}[x]^2}; \\ &\mathrm{Cor}_k = \frac{1+\hat{\rho}_k}{1-\hat{\rho}_k} \end{aligned} \tag{51}$$

This method can be described with a standard version of limited relative error algorithm for random discrete

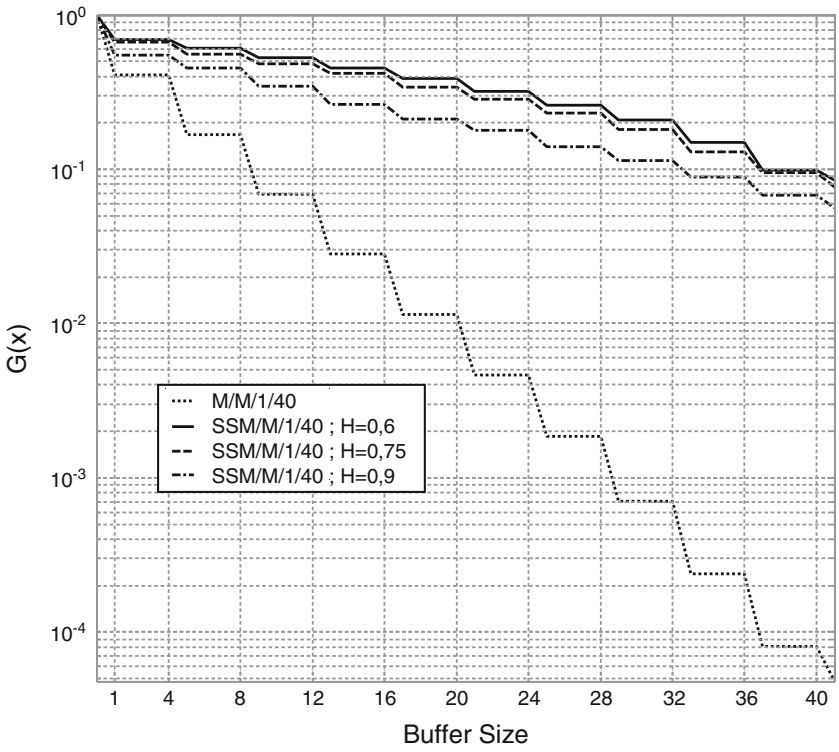

Fig. 6 Buffer overflow probability (L = 41) in SSM/M/1/40 self-similar queuing system

sequences of buffer arrivals. It consists of the following three steps provided below.

Algorithm 3 RESTART with Limited Relative Error Simulation Method

Step 1 Initialization of minimal and maximal values of the simulation parameter.
Step 2 Estimation and management of the simulation time.
Cycle L_1 *Determine the current variable for calculating the Markov chain, e.g.* $\omega{:=}\beta$*;generate a new value for* β *with given distribution.*
Increase the number of state h_β.
If the condition $\beta < \omega$ *is true, then increase the number of transitions* $c_{\beta+1}$ *while it reaches the value of* c_ω.
Cycle L_2 *Determine the total number of events at the left part* l_s *and at the right part* d_s *of the Markov chain and number of transitions* $a_s := c_s$*; check on the constraint condition* (49) *for the index* $i = s$.
If the constraint condition (49) *is true, then calculate the posterior values of the local correlation coefficient* $\hat{\rho}_s$ *and relative error* RE $[x]$ *with use of* (50)*. Calculate whether the relative error* RE $[x] \le$ RE$_{\max}$ $[x]$.
If $s < k$*, than leave the cycle* L_2.
If the index $s = k$ *is reached, than leave the cycle* L_1 *and increase the index of the simulation time* $s := s + 1$*;*

Step 3 Printing out the experimental results for $i = 1, 2, \ldots, k$*. The results for the total frequency* d_i *are determined according to* (52)*:*

$$d_i = \sum_{j=1}^{k} h_j \quad \text{for } i = 0, 1, \ldots, k \quad \text{where} \quad d_0 = n \tag{52}$$

The values of the complementary function $\hat{G}_i$, the local correlation coefficient $\hat{\rho}_s$ and the relative error RE $[x]$ are calculated as given in (50).

As an example, the overflow probability of an SSM/M/1/B self-similar queuing system has been simulated with different characteristics of long-range dependent self-similar arrival processes. In order to demonstrate the effects of self-similarity on the buffer overflow probability, the obtained experimental results were compared with the complementary cumulative distribution in the traditional single server finite buffer queue M/M/1/B. The obtained results in a logarithmic scale are given in Fig. 6.

For example, for the value of Hurst parameter $H = 0.6$ the overflow probability was $G(L) = 1.045 \times 10^{-1}$, and for $H = 0.9$ it was G$(L) = 5.6 \times 10^{-2}$. On the other hand, the overflow probability of self-similar queuing system has increased significantly in comparison with the theoretical M/M/1/B self-similar queuing system, for which $G(L) = 4.79 \times 10^{-5}$.

After that, the simulation was repeated for SSM/M/1/B self-similar queuing system by using long-range dependent

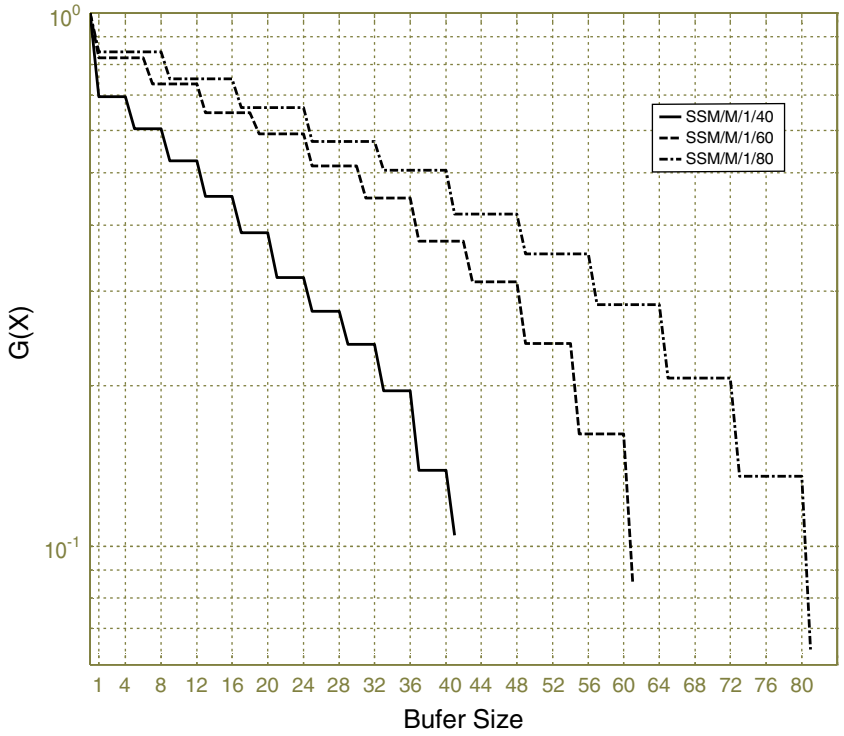

Fig. 7 Buffer Overflow Probability in SSM/M/1/B Self-similar Queuing System with Different Buffer Sizes

self-similar arrival process with $H = 0.6$ and different buffer sizes. The obtained results for buffer size $B = 40$, $B = 60$ and $B = 80$ are shown in Fig. 7. One can see that since the buffer size was increased twice, the overflow probability has been changed simply by about two orders of magnitude—from 1.045×10^{-1} to 6.4×10^{-3}.

Finally, it was confirmed that in order to design a single server finite buffer model with long-range dependent self-similar arrival processes, the buffer size has to be increased many times in order to decrease the overflow probability.

4 Conclusions

The paper recommends new methods to estimate effectively the probability of buffer overflow in high-speed communication networks. The probability of buffer overflow in queuing systems is very small; therefore the overflow can be defined as a rare event and estimated using rare event simulation with continuous-time Markov chains. Estimating the probabilities of rare events using traditional simulation techniques is computationally challenging and extremely time consuming. The paper contributes to the literature by developing algorithms for accelerating buffer overflow simulations.

Initially, two-node queuing systems have been considered in this paper; and an event of buffer overflow at the second node was studied. Two efficient rare event simulation algorithms, based on the Importance sampling and Cross-entropy methods, have been developed and applied to accelerate the buffer overflow simulation with Markov chain modeling. Simulation results were shown and analyzed.

Then, steady-state simulations of self-similar queuing systems have been conducted using the RESTART method with Limited Relative Error algorithm to estimate effectively the probability of buffer overflow.The models of SSM/M/1/40 self-similar queuing system have been applied with different parameters of arrival processes and different buffer sizes. Simulations results were shown and analyzed.

The resulting recommended methods to estimate effectively the probability of buffer overflow are appropriate and particularly efficient being used for performance evaluation in high-speed communication networks, while higher performance networks must be described by lesser buffer overflow probabilities.

Acknowledgments The author would like to thank her colleagues Michael R. Bartolacci from Penn State University—Berks, USA and Cees J. M. Lanting from CSEM, Switzerland for their time, thoughtful insights and review during the preparation of this paper.

References

1. Bobbio, A., Horváth, A., Scarpa, M., & Telek, M. (2003). Acyclic discrete phase type distributions: Properties and a parameter estimation algorithm. *Performance Evaluation*, *54*(1), 1–32.
2. Bolch, G., Greiner, S., Meer, H., & Trivedi, K. (1998). *Queueing networks and Markov Chains: Modeling and performance evaluation with computer science applications*. New York, NY: Wiley.

3. Bueno, D. R., Srinivasan, R., Nicola, V., van Etten, W., & Tattje, H. (2000). Adaptive importance sampling for performance evaluation and parameter optimization of communication systems. *IEEE Transactions on Communications*, *48*(4), 557–565.
4. Bucklew, J. (2004). *An introduction to rare event simulation*., Springer Series in Statistics, XI Berlin: Springer.
5. C'erou, F., LeGland, F., Del Moral, P., & Lezaud P. (2005). Limit theorems for the multilevel splitting algorithm in the simulation of rare events. In *Proceedings of the 2005 winter simulation conference* (pp. 682–691). San Diego, USA.
6. De Boer, P., Kroese, D., & Rubinstein, R. (2002). Estimating buffer overflows in three stages using cross-entropy. In *Proceedings of the 2002 winter simulation conference* (pp. 301–309), San Diego, USA.
7. Georg, C., & Schreiber, F. (1996). The RESTART/LRE method for rare event simulation. In *Proceedings of the winter simulation conference* (pp. 390–397), Coronado, CA, USA.
8. Giambene, G. (2005). *Queueing theory and telecommunications: Networks and applications*. New York: Springer.
9. Heidelberger, P. (1995). Fast simulation for rare event in queueing and reliability models. *ACM Transactions of Modeling and Computer Simulation*, *5*(1), 43–85.
10. Kalashnikov, V. (1997). *Geometric sums: Bounds for rare events with applications: Risk analysis, reliability, queueing*. Berlin: Kluwer Academic Publishers.
11. Keith, J., & Kroese, D. P. (2002). SABRES: Sequence alignment by rare event simulation. In *Proceedings of the 2002 winter simulation conference* (pp. 320–327). San Diego, USA.
12. Kroese, D., & Nicola, V. F. (1999). Efficient simulation of a tandem Jackson Network. In *Proceedings of the second international workshop on rare event simulation RESIM'99* (pp. 197–211).
13. Lokshina, I. (2014). Study on estimating probability of buffer overflow in high-speed communication networks. In *Proceedings of the 2014 networking and electronic commerce conference (NAEC 2014)* (pp. 306–321), Trieste, Italy.
14. Lokshina, I. (2012). Study about effects of self-similar IP network traffic on queuing and network performance. *International Journal of Mobile Network Design and Innovation*, *4*(2), 76–90.
15. Lokshina, I., & Bartolacci, M. (2012). Accelerated rare event simulation with Markov chain modelling in wireless communication networks. *International Journal of Mobile Network Design and Innovation*, *4*(4), 185–191.
16. Radev, D., & Lokshina, I. (2006a). Performance analysis of mobile communication networks with clustering and neural modelling. *International Journal of Mobile Network Design and Innovation*, *1*(3/4), 188–196.
17. Radev, D., & Lokshina, I. (2006b). Rare event simulation with Tandem Jackson networks. In *Proceedings of the fourteen international conference on telecommunication systems: Modeling and analysis—ICTSM 2006* (pp. 262–270). Penn State Berks, Reading, PA, USA.
18. Radev, D., & Lokshina, I. (2010). Advanced models and algorithms for self-similar network traffic simulation and performance analysis. *Journal of Electrical Engineering*, *61*(6), 341–349.
19. Rubino, G., & Tuffin, B. (2009). *Rare event simulation using Monte Carlo methods*. Chichester, UK: Wiley.
20. Villen-Altamirano, M., & Villen-Altamirano, J. (2006). On the efficiency of RESTART for multidimensional systems. *ACM Transactions on Modeling and Computer Simulation*, *16*(3), 251–279.

Izabella Lokshina Ph.D. is Professor of Management Information Systems and chair of Management, Marketing and Information Systems Department at SUNY Oneonta. Her positions included Senior Scientific Researcher at the Moscow Central Research Institute of Complex Automation and Associate Professor of Automated Control Systems at Moscow State Mining University. Her main research interests are complex system modeling (communications networks and queuing systems) and artificial intelligence (fuzzy systems and neural networks).