Sojourn times and Rare Events

Norbert Th. Müller, Freimut Brenner

Abteilung Informatik — Universität Trier

D-54286 Trier, Germany

E-mail: mueller@uni-trier.de, brenner@info04.uni-trier.de

Abstract: In this paper, we present and analyze three different algorithms for the evaluation of rare event simulation results. They are loosely based on the AA^C method, which in turn is an enhancement of the LRE-algorithm, derived from the sojourn time analysis of that algorithm.

The AA^C method is based on weaker assumptions than the LRE and thus usually creates confidence intervals that are more reliable than those delivered by the LRE-algorithm.

Keywords: Rare events, Markov renewal process, discrete event simulation, LRE-algorithm, confidence interval, sojourn times

1 Introduction

Goal of the algorithms proposed in this paper is the evaluation of rare event simulations using Markov renewal theory (e.g.[Ci75]). Background is an analysis of the LRE algorithm (e.g. [GöSc96]) in [Mu00] that led to a first enhanced algorithm AA^C in [Mu02].

At least three different kinds of values can be of interest when trying to simulate rare events:

- The probability of a system to return to a given initial state avoiding a (rare) subset of the state space [GHSZ96a, GHSZ96b], or equivalently, the probability to enter the rare states before returning to the initial state. (The latter will be called 'ruin probability' in the following).

- The mean time to failure (MTTF) [HMT96], i.e. the mean time necessary to reach a rare condition from an initial state.

- The fraction of time spent in a rare event condition (like Cell Loss Ratios, [BBK99]), determined by the ratio MTTR/(MTBF+MTTR) of the mean time to recovery, MTTR, and the mean time between failures, MTBF.

Obviously, the three values are quite independent, as we can see from the following simple 4-state Markov chain in continuous time with corresponding transition rates λ_{ij}:
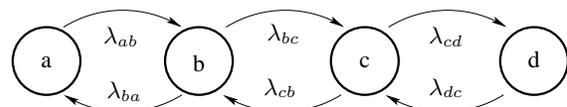
If a is the initial state and if we define $\{c, d\}$ to be the 'rare' states, then the ruin probability is solely determined by λ_{ba} and λ_{bc}. For the MTTF, we additionally need λ_{ab}, and the MTBF also needs λ_{cb}. The ratio MTTR/(MTBF+MTTR) finally depends on all the six rates.

Our model under consideration will be much more general: Markov renewal processes with a finite state space E, i.e. continuous time jump processes where the sojourn times in states $s \in E$ need not have exponential distribution, they are even allowed to depend on the next state into which the process changes. We only require that the process viewed at state changes is a Markov process. This essentially is the requirement for the processes considered e.g. in [GHSZ96b].

Our interest is in the evaluation of simulations of multilevel RESTART type, with an emphasis on the *reliability* of the resulting estimators, similar to the the analysis in [GHSZ96b] revealing the standard deviation of the given estimator, but only for the ruin probability. The LRE suite of algorithms is aiming at MTTR/(MTBF+MTTR), but the reliability of the resulting estimators is questionable; an attempt [BBK99] to improve the reliability of the LRE is not mathematically stringent, however. Their results even had to be calibrated to fit the given examples.

2 Markov renewal processes

In this section we will briefly recall needed notions from [Ci75]. We will consider jump processes $S(t)$ with state space E. X_n will be the corresponding sequence of visited states.

$S(t)$ is called a Markov renewal process, if X_n is a Markov chain and if the sojourn time in X_n only depends on the states X_n and X_{n+1} themselves. For convenience, we assume that X_n is irreducible and positive recurrent. (In our case of a finite E, irreducibility implies positive recurrence.)

Let $P = (p_{ss'})_{s,s' \in E}$ be the matrix of transition probabilities of the Markov chain X_n.

Let $\nu = (\nu_s)_{s \in E}$ be the steady state distribution for the chain X_n, i.e. the unique probability vector with

$$\nu = \nu \cdot P \qquad (1)$$

Let μ_s be the mean sojourn time in state $s \in E$. Please note that this mean sojourn time is taken independently from the next state.

The following figure shows a state diagram of the general situation that we are dealing with.
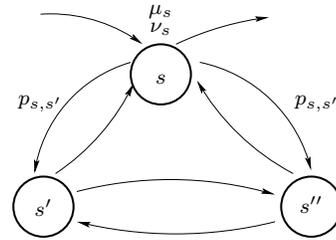
The most important result from [Ci75] concerning our point of interest is the following: The probability to find the process in state s is given as

$$P(S(t) = s) = \frac{\nu_s \cdot \mu_s}{\sum_{s'} \nu_{s'} \cdot \mu_{s'}} \qquad (2)$$

3 The AA^C method and the sojourn times analysis

Our basic evaluation method for fractions like (2) will be the AA^C-method introduced in [Mu02], being a significant enhancement of the LRE method [Schr84]. We will briefly recall the method and its analysis in the following.

Suppose we are given a jump process $\overline{S}(t)$ with state space $\overline{E}$. Given a subset $A \subseteq \overline{E}$, the problem is to determine

$$P(A) := P\{\overline{S}(t) \in A\} \qquad (3)$$

The basic idea of the LRE method was to aggregate the state set A into just one state 1 and to aggregate the complementary state set $A^c = \overline{E} \setminus A$ into a state 0. Then the resulting indicator process $\mathbb{1}_A(t)$ is analyzed. Of course, the state diagram now takes a very simple form:
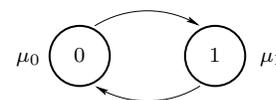
Here again, μ_0 and μ_1 are the mean sojourn times, i.e. the mean time from entering A until A^c is reached (or vice versa). Obviously, the process $\mathbb{1}_A(t)$ will *not* be a Markov renewal process in

general. However, in order to evaluate $P(A)$ we will *assume* that $\mathbb{1}_A(t)$ is such a Markov renewal process.

In this case we have $p_{01} = p_{10} = 1$ and $\nu_0 = \nu_1 = 1/2$ leading to

$$P(A) = \frac{\mu_1}{\mu_1 + \mu_0} \tag{4}$$

So in a simulation environment, we get point estimators for $P(A)$ by simply evaluating the means m_1 and m_0 of the observed sojourn times in A and A^c:

$$P(A) \approx \frac{m_1}{m_1 + m_0} \tag{5}$$

To get interval estimators, we need additional information on the variances σ_1^2 and σ_0^2 of the sojourn times.

The basic assumption behind the LRE method (see e.g. [Gö97, Schr84, Schr99]) was that $\mathbb{1}_A(t)$ should be treated even like a Markov chain, implying not only independentness of subsequent sojourn times but also implying that the variances of the sojourn times are known to be

$$\sigma_1^2 = \mu_1^2 \quad , \quad \sigma_0^2 = \mu_0^2 \tag{6}$$

In the following we will not use this very strong assumption. Instead we will consider the AA^c method [Mu02], that does not use any further assumption on $\mathbb{1}_A(t)$ than the Markov renewal property. Compared to the LRE method, the simulation environment must additionally be able to deliver the sample variances s_1^2 and s_0^2 of the observed sojourn times.

Then, if the number n of observed sojourn times in A and A^c is large enough, we get $\sigma_1^2 \approx s_1^2$ and $\sigma_0^2 \approx s_0^2$. The central limit theorem furthermore implies that the sample means m_1 and m_2 are approximately normal distributed:

$$m_1 \sim N(\mu_1, \sigma_1^2/n) \tag{7}$$

$$m_0 \sim N(\mu_0, \sigma_0^2/n) \tag{8}$$

However, if n is so large that $\sqrt{\sigma_1^2/n} \ll m_1$ and $\sqrt{\sigma_0^2/n} \ll m_0$, then $(m_1 + m_0)/(\mu_1 + \mu_0) \approx 1$, so

$$
\begin{aligned}
&\frac{m_1}{m_1 + m_0} - \frac{\mu_1}{\mu_1 + \mu_0} \\
&= \frac{(m_1 - \mu_1) \cdot \mu_0 - \mu_1 \cdot (m_0 - \mu_0)}{(m_1 + m_0) \cdot (\mu_1 + \mu_0)} \\
&\approx \frac{(m_1 - \mu_1) \cdot \mu_0 - \mu_1 \cdot (m_0 - \mu_0)}{(\mu_1 + \mu_0) \cdot (\mu_1 + \mu_0)} \\
&= m_1 \cdot \frac{\mu_0}{(\mu_1 + \mu_0)^2} - m_0 \cdot \frac{\mu_1}{(\mu_1 + \mu_0)^2}
\end{aligned}
\tag{9}
$$

As m_1 and m_2 are independent and normal distributed, (9) is obviously normal distributed. Its mean is 0, and its variance σ^2 turns out to be

$$\sigma^2 = \frac{\frac{\sigma_1^2}{n}\mu_0^2 + \frac{\sigma_0^2}{n}\mu_1^2}{(\mu_1 + \mu_0)^4} \approx \frac{\frac{s_1^2}{n}m_0^2 + \frac{s_0^2}{n}m_1^2}{(m_1 + m_0)^4} \tag{10}$$

if $\sigma_1^2 \approx s_1^2$ and $\sigma_0^2 \approx s_0^2$.

Putting all things together, we get the following summary for $\mathbb{1}_A$, valid under the Markov renewal assumption and if suitable large sample conditions are met:

$$\frac{m_1}{m_1 + m_0} \sim N\left(\frac{\mu_1}{\mu_1 + \mu_0} \quad , \quad \frac{\frac{s_1^2}{n}m_0^2 + \frac{s_0^2}{n}m_1^2}{(m_1 + m_0)^4}\right) \tag{11}$$

It is easy to see which large sample conditions are necessary: n should allow application of the the central limit theorem:

$$n > 30 \tag{12}$$

Furthermore the approximation (9) should be sufficiently precise, so we need

$$\sqrt{s_1^2/n} \ll m_1 \quad , \quad \sqrt{s_0^2/n} \ll m_0 \tag{13}$$

and

$$\sigma_1^2 \approx s_1^2 \quad , \quad \sigma_0^2 \approx s_0^2 \tag{14}$$

The estimator (5) is surely biased, especially if the variances s_1^2/n and s_0^2/n are still large. But given the large sample conditions in (13), the approximation in (9) (leading to (11)) shows that it is it

is (approximately) unbiased. We will use this in the following sections to construct estimators for processes that are of more complex structure.

In the following we present the results of a few tests to illustrate the behavior of the AA^C method. Using three different types of discrete time Markov processes $\overline{S}(t)$ and for a range of samples from 10^5 to 10^7 taken from the original processes $\overline{S}(t)$, we measured 99% confidence intervals for the following three methods of evaluation:

1. An implementation of the Law-Carson algorithm for the batch means method (see e.g. [BFS87]).

2. The AA^C-method as described above with confidence intervals based on (11).

3. An implementation of the LRE-algorithm using confidence intervals instead of the limited relative error, so essentially using (11) with simply replacing s_1^2 and s_0^2 with m_1^2 and m_0^2 according to (6). In the following diagrams, this slightly modified version of the LRE will be called LRE/sojourn. See [Mu02] for a precise comparison of the original LRE, the LRE/sojourn and the AA^C method.

The first example was a simulation of a discrete Markov chain with just two nodes, being the ideal setting for the LRE method. The transition probabilities were chosen such that the exact result was known to be 0.1. As expected, the confidence intervals turned out to be almost identical:


Confidence intervals, 2-Node chain, p_1=0.1
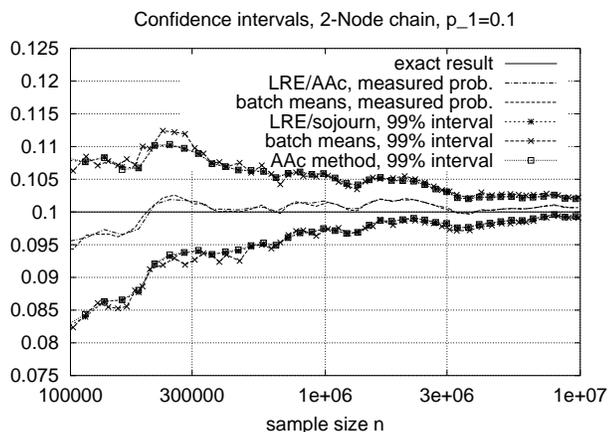
As a second example we considered a 2k-Node Markov chain, where the left nodes on the left build set A while the others correspond to set A^c. This leads to sojourn times of a discrete hypo-exponential type distribution with small variance.
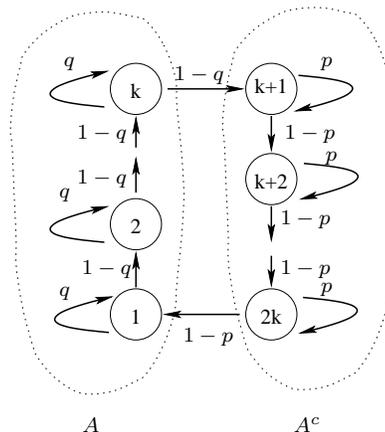

We tried this example with $k = 10$ and $p = 0.99$. q was chosen such that the exact solution was $P(S(t) \in A) = 0.1$. As expected, the confidence intervals of LRE/sojourn are much larger than for the other two methods that are of equal quality.

Confidence intervals, Erlang-type chain, p_1=0.1
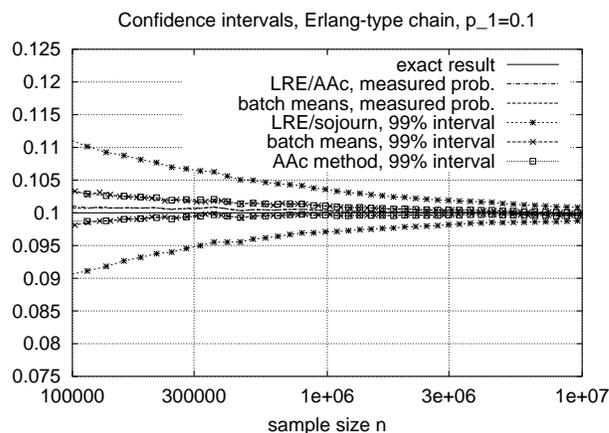
The third test used a 4-Node chain with highly variant sojourn times:
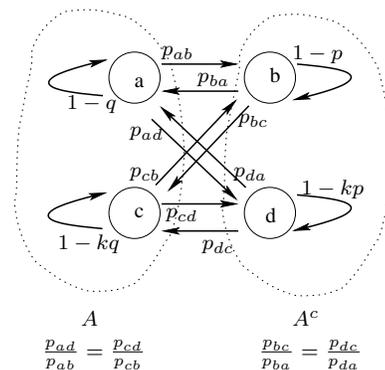
Again the nodes on the left build set A while the others correspond to set A^c. We tried an example with $k = 100$, so the sojourn times are build by mixtures of two geometric distributions differing by a factor of 100. These sojourn times have a kind of discrete hyper-exponential type distribution. p and q were chosen such that again $P(S(t) \in A) = 0.1$. As expected, the LRE underestimates the true variance and the resulting confidence intervals are much too small in this case, while the other two methods again give similar results:

Confidence intervals, high variance chain, p_1=0.1
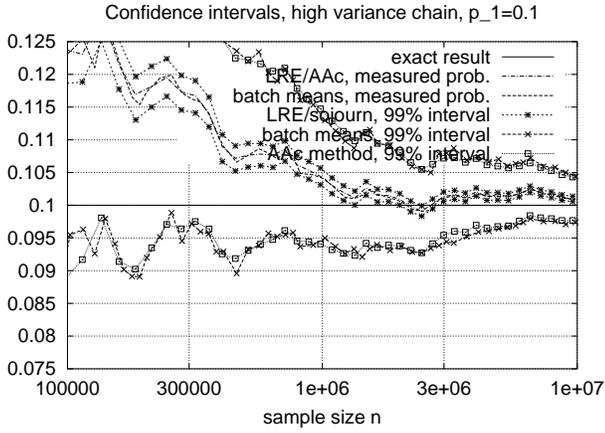
4 Skip-free chains

In this section, we consider the most simple case of renewal processes corresponding to queueing systems that are skip-free in both directions. For simplicity, we use the state set $E = \{0, 1, \ldots, k\}$, i.e. here we only consider Markov renewal processes $S(t)$ of the following kind:


$$(15)$$
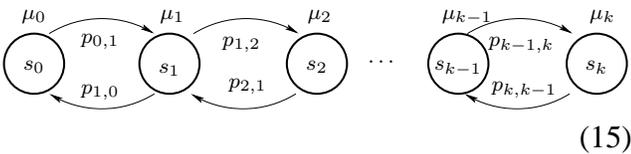
In this special case we have $p_{01} = 1 = p_{k\,k-1}$ and $p_{i\,i-1} = 1 - p_{i\,i+1}$.

Our goal will be to determine the relative amount of time $P(S(t) = k)$ spent in state k. We use the following identity:

$$P(S(t) = k) = p_{k|k-1} \cdot p_{k-1|k-2} \cdot \ldots \cdot p_{2|1} \cdot p_{1|0}$$

$$(16)$$

where $p_{i+1|i} := P(S(t) \geq i + 1 \mid S(t) \geq i)$.

To determine the single values $p_{i+1|i}$, we reduce the sojourn times of all transitions of the process starting s_j for $j < i$ to zero. This is equivalent to replace them all by a single instantaneous transition from $i-1$ to i. Then we aggregate the states as in the following figure:

$$(17)$$
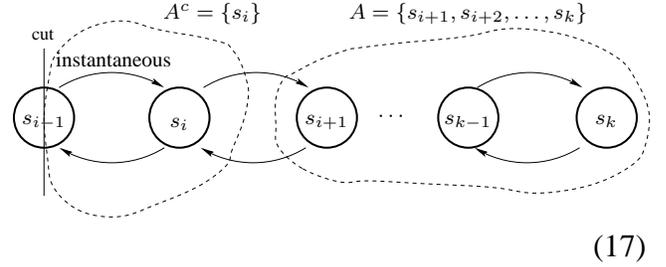
The result is still a Markov renewal process and we are able to apply the AA^c method for each i to get independent estimates h_i for the values $p_{i+1|i}$. The AA^c method will also give us variances s_i such that $h_i \sim N(p_{i+1|i}, s_i^2)$ approximately.

A point estimate for $P(S(t) = k)$ is given by

$$P(S(t) = k) \approx \overline{h} := h_0 \cdot \ldots \cdot h_{k-1} \qquad (18)$$

As the estimates h_i are independent, an estimator $\overline{s}^2$ for the variance $\overline{\sigma}^2$ of the product $\overline{h}$ is given by

$$\overline{s}^2 = \prod_{1 \leq i \leq k} (s_i^2 + h_i^2) - \prod_{1 \leq i \leq k} h_i^2 \qquad (19)$$

For large simulation runs we will have $s_i \ll h_i$ for any $1 \leq i \leq k$. As mentioned in the previous section, this implies that the estimators h_i are (almost) unbiased and of normal distribution. In consequence, the product $\overline{h}$ will also be nearly unbiased.

So in order to get an interval estimate for $P(S(t) = k)$ via $\overline{h}$ and $\overline{s}^2$ we could use Tchebychev's inequality valid for arbitrary distributions. E.q. for a 99%-level, we have

$$P\left(|P(S(t) = k) - \overline{h}| < 10 \cdot \overline{s}\right) \geq 0.99 \qquad (20)$$

But $\overline{h}$ will be approximately normal distributed, too, as we have:

$$p_{3|2} \cdot p_{2|1} - h_2 \cdot h_1$$
$$= (p_{3|2} - h_2) \cdot p_{2|1} + (p_{2|1} - h_1) \cdot h_2$$
$$\approx (p_{3|2} - h_2) \cdot p_{2|1} + (p_{2|1} - h_2) \cdot p_{3|2} \qquad (21)$$

So from the normal distribution of h_1 and h_2 we may follow that also $h_1 \cdot h_2$ has (almost) normal distribution. Inductively, this holds for all the products, so also $\overline{h}$ should be normal distributed. In consequence, we also may use the following, much better confidence intervals:

$$P\left(|P(S(t) = k) - \overline{h}| < 2.58 \cdot \overline{s}\right) \geq 0.99 \quad (22)$$

As a modified way of evaluation without the use of (19), we could compute a fixed number m of samples $\overline{h}_i$ for (18) and determine sample mean $\tilde{h}$ and sample variance $\tilde{s}^2$:

$$\tilde{h} = \sum \overline{h}_i/m \quad , \quad \tilde{s}^2 = \sum(\overline{h}_i - \tilde{h})^2/(m-1) \quad (23)$$

Using the Student-t-distribution we get confidence intervals, e.g. for $m = 30$ using the factor of 2.75 we get the confidence level of 99%:

$$P\left(|P(S(t) = k) - \tilde{h}| < 2.75 \cdot \frac{\tilde{s}}{\sqrt{m}}\right) \geq 0.99 \quad (24)$$

5 A general method

As an alternative and more general method, we may use that the solution to ν to equation (1) is continuous in P, so the solutions v of $v = v \cdot H$ for good approximations H to P must be good approximations to ν. In consequence, we may as well compute a fixed number m of matrices $H^{(i)} = (h_{ss'}^{(i)})_{s,s' \in E}$ with $h_{ss'}^{(i)} \approx p_{ss'}$ and vectors $(m_s^{(i)})_{s \in E}$ of mean sojourn times $m_s^{(i)} \approx \mu_s$, solve the systems

$$v^{(i)} = v^{(i)} \cdot H^{(i)} \quad (25)$$

for those sets of means and determine the m ratios

$$x_i := \frac{v_s^{(i)} \cdot m_s^{(i)}}{\sum_{s'} v_{s'}^{(i)} \cdot m_{s'}^{(i)}} \quad (26)$$

Unfortunately, the estimator $\overline{x}$ will be biased in general, as e.g. the formula (31) below involves the division of random variables. Nevertheless,

the continuity of the mapping from $H^{(i)}$ and $(m_s^{(i)})$ to the solution x_i implies that asymptotically, i.e. for small variances of $h_{ss'}^{(i)}$ and of $m_s^{(i)}$, the bias will vanish, similar to equations (9,21). So under sufficient large sample conditions, the mean

$$\overline{x} := \frac{1}{m} \cdot \sum x_i \quad (27)$$

will have expectation $P(S(t) = k)$. It is almost impossible to derive the variance of this estimator from the single variances, so we take the sample variance $\overline{s}^2$ of the values x_i:

$$\overline{s}^2 := \frac{1}{m-1} \cdot \sum(x_i - \overline{x})^2 \quad (28)$$

The choice of $m \geq 30$ additionally allows the application of the central limit theorem, so we use confidence intervals built on normal distributions like the following, again using the student-t distribution:

$$P\left(|P(S(t) = k) - \overline{x}| < 2.75 \cdot \frac{\overline{s}}{\sqrt{m}}\right) \geq 0.99 \quad (29)$$

We should mention that in the special case of skip free processes like in graph (15), there is a simpler direct formula for x_i: The solution of (1) gives

$$\nu_j = \nu_k \cdot \frac{p_{k\ k-1}}{p_{k-1\ k}} \cdot \ldots \cdot \frac{p_{j+1\ j}}{p_{j\ j+1}} \quad (30)$$

leading to

$$x_i = \frac{m_k^{(i)}}{\sum_{j=0}^{k} m_j^{(i)} \prod_{l=j+1}^{k} \frac{h_{l\ l-1}^{(i)}}{h_{l-1\ l}^{(i)}}} \quad (31)$$

6 Comparison of the methods

In the following, we present the results of extensive tests of the methods, where we used the $M/M/1/85$ reference model from [BBK99, Gö97]: Given an $M/M/1/85$-system with a load of 0.8, the question is to determine the probability of finding 85 waiting customers (i.e. a total of

86 customers) in the system. The exact solution for this simple Markovian system is $\frac{0.8^{-87}-1}{0.8^{-1}-1} \approx 9.263 \times 10^{-10}$.

We used the following three methods of evaluation, named by the main characterizing properties:

- "Skip free, Inner variance, Normal distribution" (SIN), based on equations (18,19,22)

- "Skip free, Outer variance, Student-t" (SOS), based on equations (18,23, 24) with $m = 30$

- "General method, Outer variance, Student-t" (GOS), based on equations (25,26, 27,28,29) with $m = 30$

The results soon showed that it was not necessary to use (20), as the approximation to a normal distribution in (22) was justified in all tests.

From the 87 states $\overline{E} = \{0, 1, 2, 3, \ldots, 86\}$ of the $M/M/1/85$ queue process $\overline{S}(t)$ we selected subsets $E := \{0, d, 2d, 3d, \ldots, 85, 86\}$ to construct Markov renewal processes $S(t)$ with about $86/d$ states. Including both the states 85 and 86 in E implied that the time spent in state 86 in $S(t)$ is equal to the time spent in state 86, but for the original process $\overline{S}(t)$.

This state 86 corresponds to the queue being full, so the value we want to estimate is the time spent in this state.

The step width d corresponds to the choice of thresholds in the usual RESTART setting. We tried the following values: $d \in \{2, 3, 5, 7, 10, 15, 20\}$.

Our simulation environment was based on a very simple C++ class implementing state changes in Markov chain using a member function Step, cf. the following excerpt from the program:

```
class  State { ...
   double Step(){
// this member function performs
// state changes in the original chain
// and returns the sojourn time
   ...
   }
...
}
```

Using this basic routine, steps and sojourn times in the embedded Markov renewal process $S(t)$ could easily be implemented as follows:

```
double MRP_Step(){
// this member function performs
// state changes in the embedded chain
// and returns the sojourn time
  double sojourn_time=0.0;
  do {
    sojourn_time = sojourn_time + s.Step();
    } while ( ! s._is_MRP_state() );
  return sojourn_time;
  }
```

Using these sojourn times, the evaluation of the simulation runs was straightforward following the respective formulas. The number m for (GOS) and (SOS) was fixed to 30.

In (GOS), for each state $s \in E$, the same number n of samples from the Markov renewal process were drawn, so each evaluation took $30 \times |E| \times n$ samples from $S(t)$.

Similarly, for (SOS) and (SIN) the same number of samples of the reduced chain from graph (17) were drawn for each state. So here we had $30 \times 2 \times |E| \times n$ or $2 \times |E| \times n$ samples from the reduced chains per evaluation.

However, the underlying number of samples from $\overline{S}(t)$ could differ from case to case. We took exponentially increasing values for n, but each simulation run was allowed to draw at most 2.3×10^8 samples from $\overline{S}(t)$. We chose to evaluate this number of samples in each run, as this was the largest number of samples tried in [BBK99].

For the evaluation of (25) we used the GNU Scientific Library (GSL). However, as the Markow renewal processes all had less than 50 states, solving the linear system was not an important aspect of the implementation.

The following graphs shows the typical evolution of confidence intervals during simulation runs. They show these intervals versus the size of the underlying sample set (from the original $M/M/1/85$ queue). Using this setting, we may compare the precision of the results for the different algorithms based on the size of the underlying sample sets.

M/M/1/85, Method SIN, d=3

M/M/1/85, Method SOS, d=3

M/M/1/85, Method GOS, d=3
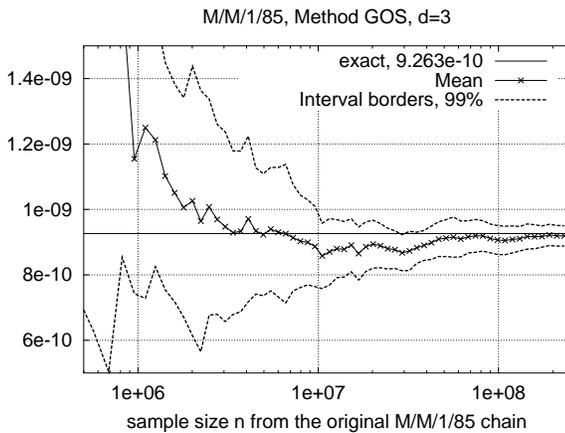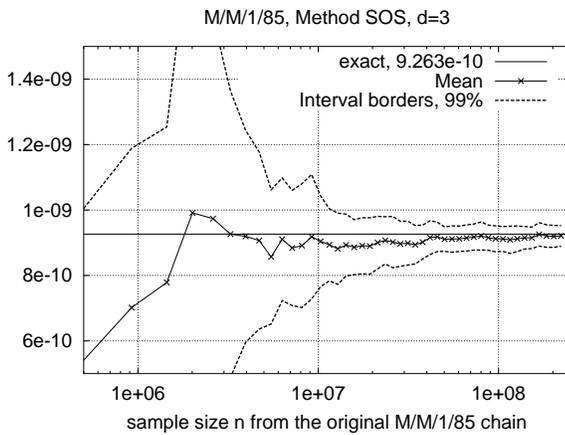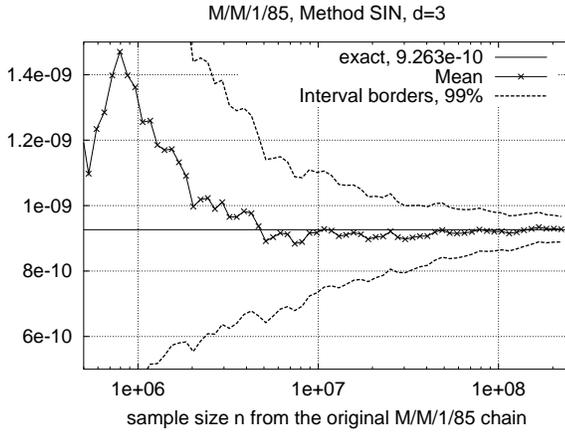
For each of the three methods, we tried 100 of the simulations and computed the mean relative error of the final estimators as well as the mean of the interval radius at the end of the simulation runs.

mean relative error after $\approx 2.3 \times 10^8$ samples							
d	2	3	5	7	10	15	20
SIN	1.17	1.19	1.10	1.55	1.51	2.46	3.69
SOS	1.01	1.19	1.15	1.49	1.52	2.31	4.60
GOS	1.32	0.96	0.81	1.70	1.49	2.64	5.90

mean radius of intervals from $\approx 2.3 \times 10^8$ samples (all values $\times 10^{-11}$)							
d	2	3	5	7	10	15	20
SIN	3.60	3.91	4.74	5.45	6.56	9.73	15.01
SOS	2.99	3.23	3.78	4.26	5.19	7.67	11.92
GOS	3.17	2.98	3.10	4.26	4.61	8.01	17.32

The resulting confidence intervals of all the experiments were quite sharp, as the following example shows. It contains the upper and lower bounds of the final 99% intervals for method GOS and step width $d = 3$. For 3 of the 100 tests, the final interval did not contain the exact value, but misses the value by a very small amount.

Final Confidence Intervals, Method GOS, d=3
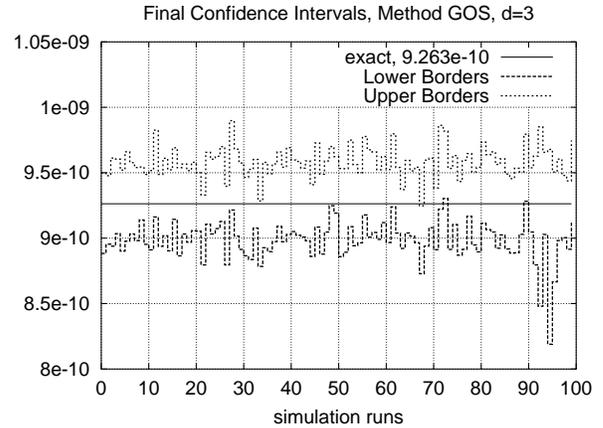
Similar experiments have been done in [BBK99]. Unfortunately, a comparison with those results is not easy, as that paper uses a different simulation system and completely different algorithms. The authors are giving two different error measures: a relative error (from the underlying LRE method) and a percentage of conformity with the exact result. In the following we will nevertheless try some comparisons:

Using a Pentium III with 500 MHz, our computation of 2.3×10^8 samples of the original $M/M/1/85$ queue together with the evaluations took 5m59s in the case of GOS with $d = 2$, i.e. we got about 6.4×10^5 samples per second. This is an enormous speedup of about 45 compared to the 1.4×10^4 trials/s mentioned in [BBK99] for a Pentium II 400 MHz. The faster CPU and (pre-

sumably) better memory can only account for a small part of this speedup, the main speedup must be due to the use of our very simple simulation environment instead of the Ptolemy simulator used in [BBK99].

Our final results due to method GOS with width $d = 3$ had a mean interval width of $\overline{s} = 2.98 \times 10^{-11}$ for a confidence level of 99%. Reduced to a confidence level of 95%, the mean interval radius would be around $2.98 \times 10^{-11} \cdot \frac{2.042}{2.75} = 2.22 \times 10^{-11}$, i.e. they would have a relative error of about 2.4%. In [BBK99], the same number of trials lead to results that had a relative error of 2.5% in about 94% of the tried simulations.

So with exception of the different simulation speed, the quality of the results seems to be roughly comparable to those in [BBK99]. The advantage of our method is that is delivers the variance of the results and so gives immediate access to this common error measure. In contrast, the method in [BBK99] even had to be calibrated with an otherwise unmotivated constant c in order to fit the required accuracy!

References

[BBK99] Below, Kai & Battaglia, Lorenzo& Killat, Ulrich, *RESTART/LRE Simulation: The Reliability Issue*, Second International Workshop on Rare Event Simulation, RESIM99, Enschede, The Netherlands

[BFS87] Bratley, Paul & Fox, Bennett L.& Schrage, Linus E., *A Guide to Simulation* (Springer, New York, 1987)

[Ci75] Cinlar, Erhan, Introduction to Stochastic Processes, Prentice-Hall, 1975

[GHSZ96a] Glasserman, P. & Heidelberger, Ph. & Shahabuddin, P.& Zajic, T., *Multilevel Splitting for Estimating Rare Event Probabilities* IBM Research Report RC 20478 (06/17/96)

[GHSZ96b] Glasserman, P. & Heidelberger, Ph. & Shahabuddin, P.& Zajic, T., *Splitting for rare event simulation: Analysis of simple cases* (1996 Winter Simulation Conference, pp302-308, Coronado, California USA)

[GöSc96] Görg, C., Schreiber, F., *The RESTART/LRE Method for Rare Event Simulation* (1996 Winter Simulation Conference, pp390-397, Coronado, California USA)

[Gö97] Görg, C., *Verkehrstheoretische Modelle und stochastische Simulationstechniken zur Leistungsanalyse von Kommunikationsnetzen* (Habilschrift, Aachen, 1997)

[HMT96] Heidelberger, Ph. & Muppala, J. & Trivedi, K., *Accelerating mean time to failure computations*

[Mu00] Müller, N.Th., *An analysis of the LRE-Algorithm using Sojourn Times* (14th European Simulation Multiconference, 2000, Ghent (ESM 2000)) preliminary version: www.informatik.uni-trier.de/ mueller/Forschung/sojourn.pdf

[Mu02] Müller, N.Th., *The AAc-Method for the Evaluation of Simulation Results* (16th European Simulation Multiconference, 2002, Darmstadt (ESM 2002)) preliminary version: www.informatik.uni-trier.de/ mueller/Forschung/aac.pdf

[Schr84] Schreiber, F., *Time Efficient Simulation: The LRE-algorithm for producing empirical distribution functions with limited relative error* (AEÜ, 38 93-98, 1984)

[Schr99] Schreiber, F., *A simplified Formula Basis for the LRE-Algorithm Time Efficient Simulation* (MMB '99, Trier)

[VA94] Villén-Altamirano, M. & Villén-Altamirano, J., *Restart: a straightforward method for fast simulation of rare events* (1994 Winter Simulation Conference 1994, pp282-289)