A COMPARISON OF RESTART IMPLEMENTATIONS

Marnix J.J. Garvels

Centre for Telematics and
Information Technology
University of Twente
P.O. Box 217, 7500 AE Enschede
The Netherlands

Dirk P. Kroese

Faculty of Mathematical Sciences
University of Twente
P.O. Box 217, 7500 AE Enschede
The Netherlands

ABSTRACT

The RESTART method is a widely applicable simulation technique for the estimation of rare event probabilities. The method is based on the idea to restart the simulation in certain system states, in order to generate more occurrences of the rare event. One of the main questions for any RESTART implementation is *how* and *when* to restart the simulation, in order to achieve the most accurate results for a fixed simulation effort.

In this paper we investigate and compare, both theoretically and empirically, different implementations of the RESTART method. We find that the original RESTART implementation, in which each path is split into a fixed number of copies, may not be the most efficient one. It is generally better to fix the total simulation effort for each stage of the simulation. Furthermore, given this effort, the best strategy is to restart an equal number of times from each state, rather than to restart each time from a randomly chosen state.

1 INTRODUCTION

The RESTART (REpetitive Simulation Trials After Reaching Thresholds) method is a simple simulation method for the estimation of small probabilities. It was introduced in Villen-Altamirano (1991) and enhanced in Villen-Altamirano (1994), but it is similar to an older technique called *splitting* proposed in Kahn et al. (1951). Theoretical aspects of the method were considered in Glasserman et al (1996:1) and Glasserman et al. (1996:2). Other studies include Glasserman et al. (1997), where large deviations aspects of the RESTART method were considered, and Schreiber et al. (1996), where a method to control the variance buildup in the estimator was presented.

The basic idea of the RESTART method is to consider the rare event as the intersection of a nested sequence of events. The probability of the rare event is thus the product of conditional probabilities, each of which can usually be estimated much more accurately than the rare event itself, for a given simulation effort.

Although RESTART has been shown to be an efficient and flexible simulation method in many cases, it is not clear what the best implementation is for a given class of problems. To clarify this issue, we investigate and compare various implementations and discuss their advantages and disadvantages. In the original RESTART implementation, at every restart stage each run is split into a fixed number of copies. We call this the *Fixed Splitting* (FS) method. Analytical and empirical results suggest however that the FS method may not be the best implementation. We propose another implementation, in which the total simulation effort per stage is fixed. We call this the *Fixed Effort* (FE) method. This strategy not only yield more accurate results (for a fixed simulation budget), but is also more robust, in the sense that it is much less sensitive to the choice of the states in which the simulation is restarted. On the negative side, it requires somewhat more computer memory than the FS method.

The rest of the paper is organized as follows. In Section 2 we describe the general setting in which our rare event estimation takes place. We shortly review the RESTART method, and introduce the FE and FS implementations. We discuss some of the properties of the estimator of the rare event probability. In Section 3 we have a closer look at the implementation alternatives, and argue which alternative gives the better performance. In Section 4 a number of simulation experiments are conducted using two different models. Finally, in Section 5 we give our conclusions and some directions for future research.

2 OVERFLOW PROBABILITIES

In this section we describe the class of problems for which we wish to use the RESTART method. The basic setting is the following (for examples see Section 4): Consider a Markov process $X := (X_t,\, t \geq 0)$ with space E, and let f be a real-valued measurable function on E. Define $Z_t := f(X_t)$, for all $t \geq 0$. Assume for definiteness that $Z_0 \geq 0$. For any *threshold* or *level* $L > 0$, let T_L denote the first time that the process $Z := (Z_t,\, t \geq 0)$ hits the set $[L, \infty)$; and let T_0 denote the first time, after 0, that Z hits the set $(-\infty, 0]$. We assume that T_L and T_0 are well-defined (possibly infinite) stopping times with respect to the history of X.

We are interested in the probability, γ say, of the event $D_L := \{T_L < T_0\}$, i.e., the probability that Z up-crosses level L before it down-crosses level 0. Note that γ depends on the initial distribution of X.

An exact analysis of γ is often not possible. A standard way to estimate γ by simulation is the following. Generate independently r realizations (sample paths) of the Markov process X. Each path $x^{(i)} := (x_t^{(i)})$ defines a realization $z^{(i)} := (z_t^{(i)})$ of Z. Let I_i be the indicator that $z^{(i)}$ up-crosses level L before it down-crosses level 0. An unbiased estimate for γ is given by

$$\widehat{\gamma} := \frac{1}{r} \sum_{i=1}^{r} I_i.$$

For small values of γ this method is not very efficient. We can see this by examining the *relative error* (RE) of the corresponding estimator (We use the same notation for estimate and estimator, as is often done in statistical inference), which is defined as

$$\mathrm{RE}(\widehat{\gamma}) := \frac{\sqrt{\mathrm{Var}\,\widehat{\gamma}}}{\mathrm{E}\,\widehat{\gamma}} = \sqrt{\frac{1-\gamma}{r\,\gamma}}.$$

Note that the relative error tends to infinity as γ tends to 0. An alternative way to estimate γ is based on the following observation: If $L > K$ then $D_L \subset D_K$, where, of course, D_K denotes the event that Z up-crosses level K before it down-crosses level 0. Therefore, we have by basic conditional probability,

$$\gamma = p_1\, p_2,$$

with $p_1 := \mathrm{P}(D_K)$ and $p_2 := \mathrm{P}(D_L | D_K)$.

Hence, if we estimate both p_1 and p_2 and multiply the results, we obtain an estimate for γ. When p_1 and p_2 are considerably larger than γ, this estimation procedure is likely to be more efficient than the standard method in (1). Moreover, the same arguments may be used when we divide the interval $[0, L]$ into *multiple* subintervals, instead of just two. We will investigate this next.

2.1 Fixed Effort RESTART

We describe in this section a simple implementation of the RESTART method for estimating the probability γ defined previously. First, we partition the interval $[0, L)$ into m subintervals $[L_0, L_1), [L_1, L_2), \ldots, [L_{m-1},\, L_m)$, with $0 =: L_0 < L_1 < \ldots < L_m := L$. Let D_i denote the event that process Z reaches level L_i before returning to 0. It is assumed that Z actually *hits* all thresholds $L_1, \ldots, L_m$ if event D_L occurs. Then $D_1, D_2, \ldots, D_m$ is a nested sequence of events, decreasing to D_m. And, with $p_1 := \mathrm{P}(D_1), p_2 := \mathrm{P}(D_2|D_1), \ldots$, we have

$$\gamma = p_1\, p_2 \cdots p_m.$$

We wish to estimate at the kth *stage* $(k = 1, \ldots, m)$ the conditional probability p_k. We do this by generating a *fixed* number of samples $I_1^{(k)}, \ldots, I_{r_k}^{(k)}$ of the indicator that process Z reaches level L_k before returning to 0, starting from level L_{k-1}. We call r_k the simulation *effort* at stage k, and refer to this RESTART implementation as the *Fixed Effort* (FE) method.

How these indicators are generated from samples of X still remains to be specified, and we will postpone this issue till Section 3. Nevertheless, we may investigate some properties of the FE method for the simplest case in which all the random variables $I_1^{(1)}, \ldots, I_{r_m}^{(m)}$ are independent and identically distributed, with $\mathrm{E}\, I_j^{(k)} = p_k,\ j = 1, \ldots, r_k,\ k = 1, \ldots, m$. For this case natural estimators of $p_1, \ldots, p_m$ are given by

$$\hat{p}_k := \frac{R_k}{r_k},\ k = 1, \ldots, m, \tag{1}$$

where $R_k = \sum_{i=1}^{r_k} I_i^{(k)}$ is the total number of successes at the kth stage. Moreover, the natural estimator of γ is

$$\hat{\gamma} := \prod_{k=1}^{m} \hat{p}_k. \tag{2}$$

Obviously these are unbiased estimators. Moreover, the variance of $\hat{\gamma}$ is given by

$$\begin{aligned}
\mathrm{Var}\,\hat{\gamma} &= \mathrm{E}\,\hat{\gamma}^2 - (\mathrm{E}\,\hat{\gamma})^2 = \mathrm{E} \prod_{i=1}^{m} \hat{p}_i^2 - \gamma^2 \\
&= \prod_{i=1}^{m} \{\, \mathrm{Var}\,\hat{p}_i + (\mathrm{E}\,\hat{p}_i)^2 \} - \gamma^2 \\
&= \prod_{i=1}^{m} \{ \frac{p_i(1-p_i)}{r_i} + p_i^2 \,\} - \gamma^2 \\
&= \gamma^2 \left(\prod_{i=1}^{m} \{ \frac{1-p_i}{p_i\, r_i} + 1 \} - 1 \right).
\end{aligned}$$

2.1.1 Parameter optimization

In the estimator $\hat{\gamma}$ we still have considerable freedom in choosing the simulation parameters. Let us assume that the simulation time of sampling from each Bernoulli $I_j^{(i)}$ is about the same, and that we are given a fixed total simulation effort $r := r_1 + \cdots + r_m$. How we should choose the parameters m, $L_1, \ldots, L_{m-1}$ and $r_1, \ldots, r_m$ in order to minimize the variance of $\hat{\gamma}$? To clarify this issue, first notice that

$$\operatorname{Var} \hat{\gamma} = \gamma^2 \sum_{i=1}^{m} \frac{1 - p_i}{p_i\, r_i} + o\,(1/r), \tag{3}$$

as $r \to \infty$. Suppose thus that r is large enough such that $\operatorname{Var}\hat{\gamma}$ is approximated well by the sum above. For any given choice of m and $p_1, \ldots, p_m$, the optimal choice for $r_1, \ldots, r_m$ is determined by minimizing $\sum_{i=1}^m b_i/r_i$, with $b_i := (1-p_i)/p_i$, under the condition $r_1 + \cdots + r_k = r$. To get an idea what the solution of this *discrete* optimization program is for large r, we consider the corresponding *continuous* version (where the r_i are elements of R_+), adding a Lagrange multiplier. We arrive at the following minimization problem:

$$\text{minimize} \quad \sum_{i=1}^{m} \frac{b_i}{r_i} + K(r_1 + \cdots + r_m - r),$$

where the minimum is taken over all $r_i > 0$ and $K > 0$. It is easy check that the values of $r_1, \ldots, r_m$ for which all partial derivatives in the expression above are 0, must satisfy

$$\frac{b_i}{b_j} = \frac{r_i^2}{r_j^2},\ i, j \in \{1, \ldots, m\}.$$

In particular, $r_i = r_1 \sqrt{\frac{b_i}{b_1}}$, and therefore $r = r_1 \sum_{i=1}^m \sqrt{\frac{b_i}{b_1}}$, so that

$$r_i = r\, \frac{\sqrt{b_i}}{\sum_{j=1}^m \sqrt{b_j}},\ i = 1, \ldots, m. \tag{4}$$

Hence, for this choice of the r_i we have

$$\operatorname{Var}\hat{\gamma} \approx \gamma^2\, \frac{\left(\sum_{i=1}^m \sqrt{b_i}\right)^2}{r} = \gamma^2\, \frac{\left(\sum_{i=1}^m \sqrt{\frac{1-p_i}{p_i}}\right)^2}{r}. \tag{5}$$

Next, we wish to examine for a *fixed* m the optimal choice for the *partition* $p_1, \ldots, p_m$, under the condition $p_1 \cdots p_m = \gamma > 0$. (From these probabilities we can infer directly the optimal levels $L_1, \ldots, L_{m-1}$.) By (5) this is equivalent to minimizing $\sum_{i=1}^m \sqrt{(1-p_i)/p_i}$, under the same condition. Again, by introducing a Lagrange multiplier, we easily find that the p_i should all be equal. Thus, for a fixed choice of m the variance of $\hat{\gamma}$ is minimal if we choose $p_i = \gamma^{1/m}$ and, from (4), $r_i = r/m$ for all i. The variance is then

$$\operatorname{Var}\hat{\gamma} \approx \frac{\gamma^2\, m^2 (1 - \gamma^{1/m})}{\gamma^{1/m}\ r}.$$

It remains to minimize this last expression with respect to m. For small γ, this is equivalent to minimizing $m^2/\gamma^{1/m}$. This is again a discrete optimization problem. For real positive m the minimum is attained in $m = -\log(\gamma)/2$. This suggests that for small γ we should take approximately $-\log(\gamma)/2$ thresholds. Or equivalently, the number of thresholds should be such that the probability of crossing a threshold when starting from the previous threshold, i.e., p_i, is roughly equal to $\mathrm{e}^{-2} \approx 0.135$. The same probability for the original RESTART method has been found in Villen-Altamirano (1994).

Thus, if we use these optimal choices for the parameters, we have (for small γ and large r)

$$\operatorname{Var}\hat{\gamma} \approx \frac{(\mathrm{e}\,\gamma\, \log\gamma)^2}{4\, r}.$$

This should be compared with the variance of the standard estimator, with the same simulation effort r. For small γ this variance is approximately γ/r.

Therefore, for the RESTART method with fixed effort, the relative error of $\hat{\gamma}$ is

$$\mathrm{RE}(\hat{\gamma}) \approx -\frac{\mathrm{e}\, \log\gamma}{2\sqrt{r}},$$

compared with a relative error of approximately $1/\sqrt{r\gamma}$ for standard simulation, see (1).

2.2 Fixed Splitting RESTART

The more "standard" implementation of the RESTART method is slightly different from the FE method described before. We divide the estimation procedure into m stages and at each stage $k \in \{1, \ldots, m\}$ generate samples $I_1^{(k)}, \ldots, I_{r_k}^{(k)}$ from the indicator that process Z, starting from level L_{k-1}, reaches level L_k before returning to 0. However, now the effort per stage is *random*. Specifically, r_1 is constant, but $r_k = n_k\, R_{k-1}$, where $R_{k-1} = I_1^{(k-1)} + \cdots + I_{r_{k-1}}^{(k-1)}$ is the number of successes at stage $k-1$, and n_k are some *fixed* numbers, possibly depending on k, $k = 2, \ldots, m$. Since each successful path is "split" into a fixed number of copies, we call this the *Fixed Splitting* (FS) method.

As before, we are at this stage not interested in *how* these indicators are generated from samples of X. We will recall some properties of this implementation from Glasserman et al (1996:2) , for the simplest case in which all the random variables $I_1^{(1)}, \ldots, I_{r_m}^{(m)}$ are i.i.d. with $\mathrm{E}\, I_j^{(k)} = p_k,\ j = 1, \ldots, r_k,\ k = 1, \ldots, m$.

The natural estimators of $p_1, \ldots, p_m$ and γ are again given by (1) and (2). By conditioning on the R_k's it is not difficult to see that these estimators are unbiased. Moreover, since $r_k = n_k R_{k-1}$, for $k = 2, \ldots, m$, (2) reduces to the simpler formula

$$\hat{\gamma} := \frac{R_m}{\prod_{k=1}^m n_k}, \tag{6}$$

where we have put $n_1 := r_1$.

The variance of $\hat{\gamma}$ follows through recurrence, and is given (see Glasserman et al (1996:2)) by

$$\mathrm{Var}\, \hat{\gamma} = \gamma^2 \sum_{k=1}^m \frac{1 - p_k}{\prod_{j=1}^k p_j n_j} = \gamma^2 \sum_{k=1}^m \frac{1 - p_k}{p_k \, \mathrm{E}\, r_k}. \tag{7}$$

The last equality follows from the fact that $\mathrm{E}\, r_k = n_k \, \mathrm{E}\, R_{k-1}$.

When we compare (7) with (3) we see that for the FS method to be as efficient as the FE method we should choose $m \approx -\log(\gamma)/2$, $p_k \approx e^{-2}$, and $\mathrm{E}\, r_k = r/m$, so that $n_k \approx 1/p_k \approx e^2$. Moreover, the FE and FS implementations yield approximately the same variance for some (large) fixed expected total simulation effort r.

2.3 RESTART with dependent runs

We now address the question *how* the indicator random variables are generated from samples of the Markov process X.

Assume, as before, that the interval $[0, L]$ is divided into levels $0 = L_0, L_1, \ldots, L_m = L$. The general RESTART simulation procedure is as follows. From level 0 we run r_1 (fixed) independent copies of X (and Z), and define $I_j^{(1)}$ as the indicator that the jth copy of Z reaches level L_1 before visiting 0, $j = 1, \ldots, r_1$. At the first stage, we save the *entrance states* of all paths that reach level L_1. More precisely, for every copy of Z which crosses level L_1 we remember the state of the corresponding X at the time crossing. After that, r_2 new copies of Z are started, each copy from a certain saved state (two or more copies may share the same saved state), and we generate Bernoullis $I_j^{(2)}, j = 1, \ldots, r_2$, such that $I_j^{(2)}$ indicates whether the jth copy of Z (Z starting from level L_1 and X from a saved state) reaches level L_2 before 0. This process repeats itself at all the subsequent stages $3, \ldots, m$. In the Fixed Effort (FE) implementation, r_k is fixed at every stage k. In the Fixed Splitting (FS) implementation $r_k = R_{k-1} n_k$, where n_k is fixed and R_{k-1} is the number of successful hits of level L_k before 0.

A typical outcome of the simulation can thus be viewed as a "tree" of Z-paths. We start with r_1 *roots*. Whenever one of the roots reaches a threshold, it generates *offspring*, which in turn generate offspring when they hit the next level, etc.

Notice that in general the indicators $\{I_i^{(k)}\}$ are not independent; the success probability of an indicator depends typically on the state from which X restarts. Let $p_k(x)$ be the probability that Z, starting from level L_{k-1}, reaches level L_k before 0, when X starts from state x. Also, let μ_k be the conditional distribution of X at the time when Z crosses L_k, given that this happens before Z returns to 0. Finally, let S_k be a random variable with distribution μ_k. Then, obviously

$$\mathrm{E}\, p_k(S_k) = p_k,\ k = 1, \ldots, m.$$

Now, consider the estimator (2) for the FE method. At every stage $k \in \{1, \ldots, m\}$ we have

$$\mathrm{E}\, R_k = r_k \, \mathrm{E}\, p_k(S_k) = r_k \, p_k.$$

Consequently, by first conditioning on R_m, then on R_{m-1}, etc., we find that also in this case (2) is an unbiased estimator for γ. For the FS method we can prove similarly that (6) is an unbiased estimator for γ.

To investigate the variance of $\hat{\gamma} = \hat{p}_1 \cdots \hat{p}_m$, let us assume that the total simulation effort in every stage is large, so that $\hat{p}_k$ is approximately distributed as $p_k + \sigma_k V_k$, where σ_k is the (small) standard deviation of $\hat{p}_k$ and V_k has a standard normal distribution. Because the σ_k's are assumed to be small, we have

$$\mathrm{Var} \prod_{k=1}^m (p_k + \sigma_k V_k) \approx \mathrm{Var} \gamma \sum_{k=1}^m \frac{\sigma_k V_k}{p_k} = \gamma^2 \, \mathrm{Var} \sum_{k=1}^m \frac{\hat{p}_k}{p_k}.$$

Let Y_i denote the total number of paths that hit level L_k of all the paths that start from the ith initial state, $i = 1, \ldots, R_{k-1}$. The $\{Y_i\}$ and also $\{(Y_i, Y_j)\}_{i \neq j}$ are identically distributed. We can show that

$$\begin{aligned} \mathrm{Var}\, \hat{p}_k &= \frac{p_k(1 - p_k)}{r_k} \\ &+ \frac{\mathrm{E}\left\{ R_{k-1}(R_{k-1} - 1) \left(Y_1 \, Y_2 - (\frac{R_k}{R_{k-1}})^2 \right) \right\}}{r_k^2} \end{aligned}$$

3 RESTART IMPLEMENTATIONS

In this section we have a closer look at the implementation issues concerning RESTART. In the previous section we have already encountered two different implementations: the FE method, which fixes the the effort per stage, and the FS method, which fixed the number of splits per reached state in a stage. Also parameter optimization, as discussed in the previous section, is in some sense an implementation issue. It seems reasonable to choose the parameters of any implementation such as suggested in Sections 2.1 and 2.2, even if the runs are not independent. Numerical experiments, based on the models of the next section, support this idea.

In what follows, we use the terminology of Section 2.3. The words *saved state*, *entrance state* and *starting state* are used interchangeably.

3.1 Single Step vs. Global Step

In any simulation experiment involving RESTART we have two choices: either we simulate "stage–by–stage" or "root–by–root". In the first case we complete all the paths starting from a certain stage before we move to the next one. This is called the *Single Step* approach. In the second case, we generate all the offspring originating from a single root before we move to the next root. We call this the *Global Step* approach.

Although the "classical" Global Step approach uses, in general, less memory than the Single Step approach, the latter method offers more flexibility in controlling the variance of the estimator. This was also demonstrated in Schreiber et al. (1996). We will therefore mainly use the Single Step approach in our experiments.

3.2 Fixed Effort vs. Fixed Splitting

In the *Fixed Splitting* (FS) method we create at every stage a fixed number of offspring from each saved state. In the *Fixed Effort* (FE) method we create at every stage a fixed total number of offspring.

We expect the FS method to perform less than optimal in multistage simulations, because the total number of simulations for each stage is uncontrollable. When the number of splits per stage is too small, we will see the simulation paths "die-out"; when on the other hand the number of splits per stage is too high the number of simulation paths will "explode". In the first case the variance in the estimator will become too large, in the second case the time spent on simulation will become too large. It is therefore of utmost importance to keep the underlying branching process "critical". Glasserman et al. have tried this principle in Glasserman et al. (1996:1) by *randomizing* the number of splits in order to ensure the critical nature of the simulation. This again has disadvantages as it needs a pilot run to determine the distribution for the generator of the number of splits pf each stage. The Fixed Effort method avoids these problems in a much better way because the number of simulations we will perform per stage is fixed in advance.

3.3 Fixed Assignment vs. Random Assignment

In stage k of the simulation we have to distribute the r_k sample paths we need to simulate over the given R_{k-1} entrance states (created by the successful hits of stage $k-1$). We could draw an entrance state *randomly* each time we need to generate a sample path in stage k. This seems sensible because we are then using the empirical entrance distribution into stage k for the starting states. We will call this the *Random Assignment* method.

An alternative approach is to distribute the R_{k-1} starting states evenly (deterministically) amongst the r_k runs. We call this the *Fixed Assignment* method.

We will analyse the variance generated in the second stage in a two-stage situation for both methods. Suppose we have R_1 starting states $S_1, \ldots, S_{R_1}$. R_1 is assumed here to be *fixed*. We wish to start a total of $r_2 := n_2 R_1$ new runs, where $n_2 \geq 1$ is some fixed integer. The success probability from some state s will be denoted by $p_2(s)$. Let Y_i be the number of successful runs (that reach the next level) starting from state S_i. Since we have only one intermediate stage, the S_i's are independent and identically distributed samples of the entrance state, and $\mathrm{E}\, p_2(S_i) = p_2$. Consequently,

$$\hat{p}_2 := \frac{1}{r_2} \sum_{i=1}^{R_1} Y_i$$

is an unbiased estimator for p_2.

Fixed Assignment: From each starting state we start n_2 independent paths. Then, *given* the vector of starting states $S := (S_1, \ldots, S_{R_1})$, Y_i has a Binomial distribution of size n_2 and success probability $p_2(S_i)$. Let us denote conditional expectation and variance with respect to S by E_S and Var_S, respectively. We have

$$\mathrm{Var}\, \hat{p}_2 \quad = \quad \frac{1}{r_2} \left\{ p_2(1-p_2) + (n_2 - 1) \mathrm{Var}\, p_2(S_1) \right\}.$$

Random Assignment: We distribute the R_1 starting states completely random amongst the $r_2 =$

$n_2 R_1$ new runs. Let K_i denote the number of runs that start from state S_i. Then, the vector $K := (K_1, \ldots, K_{R_1})$ has a multinomial distribution of size r_2 and with equal success probabilities $1/R_1$. Notice that the Y_i are identically distributed, but not independent. Also, each pair $(Y_i, Y_j), i \neq j$ has the same distribution. Moreover, given K, the Y_i's are independent; and given K *and* S, each Y_i has a Binomial distribution with size K_i and success probability $p_2(S_i)$. Finally, K and S are independent. By conditioning on K ad S and using the independencies we arrive at the following equation for the variance of p_2:

$$\operatorname{Var} \hat{p}_2 = \frac{1}{r_2} \left\{ p_2(1-p_2) + \frac{(r_2-1)\operatorname{Var} p_2(S_1)}{R_1} \right\}.$$

When we compare this with the variance for the Fixed Assignment case we see that the last formula always gives a higher variance, irrespective of the unknown constant $\operatorname{Var} p_2(S_1)$. This is a perhaps surprising result, which has been verified empirically also to hold for the multilevel case, see Section 4.

4 NUMERICAL EXPERIMENTS

To analyse the behaviour of the different RESTART implementations we conduct a series of simulation experiments using two different models. In particular, we compare the FS and FE implementations, both as Single Step methods. Also, the FE implementation will be evaluated in both the Random Assignment (RA) and the Fixed Assignment (FA) case.

The RESTART parameters (e.g., m, $L_1, \ldots, L_m$, etc.) are chosen in accordance with the values suggested in Sections 2.1 and 2.2. For example, we try to choose the levels such that success probabilities (of hitting the next level) are near the "optimal" value e^{-2}. Also, we have used a truncation procedure to discard unpromising trials, as in Glasserman et al. (1996:1).

In all tables γ denotes the rare event probability of interest. The estimate of γ is given by $\hat{\gamma}$. For each $\hat{\gamma}$ the corresponding estimate of the *Relative Error* (RE) is included. As a measure of the efficiency of the estimator $\hat{\gamma}$ we use the *Relative Time Variance product* (RTV), which we define as the simulation time (in seconds of CPU time used on a Sun Ultra 2 using Sun CC 2.1 with optimization level 5) multiplied by the squared (estimate of the) relative error of $\hat{\gamma}$. Notice that the RTV is equivalent to the "work-balanced variance" used in Glynn et al. (1992). Once a stable estimate of the variance is reached, the RTV becomes constant. This constant is smaller for more efficient simulation schemes. Practically, if scheme 1 gives a RTV which is half that of scheme 2, it would take twice as long to estimate γ within a certain accuracy via scheme 2 than via scheme 1.

4.1 Tandem queue

The first model is a 2-node tandem queue. Customers arrive at the first queue according to a Poisson process with rate λ. The service time of a customer at the first queue is exponential with rate μ_1, independent of the input process and the service time at the second node. The output process of the first queue forms the input process of the second queue. The service time of customer at the second queue is exponential with rate μ_2, also independent of every thing else. This model has received considerable attention in rare event probability estimation, e.g., in Glasserman et al. (1996:2) and Parekh et al. (1989). We wish to estimate the probability γ of the event that the number of customers in the the second queue reaches some (high) level L, before the system empties, starting from an empty system. See Figure 1 for a graphical illustration.

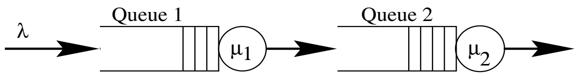

Figure 1: Tandem model

Let Y_t and Z_t be number of clients in the first and the second queue at time t, respectively (including the customers in service). Then $X := (Y_t, Z_t, t \geq 0)$ is the underlying Markov process in the general set-up of Section 2.

Remark 1 Notice that γ is defined as the probability of overflow before *both* buffers become empty, not just the second buffer. We therefore have a slightly different setting than described in Section 2. However, the RESTART procedure is easily adapted for this case.

We compare the FE approach with the FS approach of Glasserman et al. (1996:1); and we do this for two different cases. In the first case the second buffer has the highest load, and in the second case the first buffer. As service rates we use $\mu_1 = 4, \mu_2 = 2$, in the first case, and $\mu_2 = 4/3, \mu_2 = 2$, in the second case. In both cases $\lambda = 1$. Two different levels are considered: $L = 20$ and 60. The intermediate levels will be chosen as multiples of 2, hence, $L_k := 2k, \quad k = 0, 1 \ldots, m-1$, where m

is 10 and 30, respectively. The number of samples were chosen as $r_k = 10^6, \forall k$ for the FE method and $n_1 = 5 \cdot 10^6, n_2 = 2, n_k = 4, \forall k \geq 3$ for the FS method to optimize comparability between the simulation methods. As in Glasserman et al. (1996:1), a cut-off technique has been used to reduce simulation time. The idea is to discard unpromissing paths which lead back to zero, since a lot of time is being spent simulating paths back to the empty system state. The simulation results for this model are found in Table 1, along with the exact probabilities which were obtained from Glasserman et al. (1996:1). For each estimate of γ we have simulated long enough to obtain relative errors of about 3%.

Table 1: Results for the tandem queue. Parameters $(\lambda, \mu_1, \mu_2) = (1, 4, 2)$ in top and $(1, \frac{4}{3}, 2)$ in bottom half

	L	$\hat{\gamma}$	RE	sec	RTV
Exact	20	1.27e-6			
FE	20	1.256e-6	6.4e-3	418	1.7e-2
FS	20	1.256e-6	9.6e-3	281	2.6e-2
Exact	60	1.16e-18			
FE	60	1.179e-18	1.2e-2	2195	3.0e-1
FS	60	1.128e-18	1.9e-2	1450	5.4e-1
Exact	20	3.82e-6			
FE	20	3.812e-6	3.6e-3	1521	1.9e-2
FS	20	3.811e-6	9.6e-3	1418	1.3e-1
Exact	60	3.47e-18			
FE	60	3.440e-18	9.4e-3	1817	1.6e-1
FS	60	3.398e-18	3.0e-1	4909	4.3e0

We conclude that also in this case the FE method is more efficient than the FS method. Note that, as observed in Glasserman et al. (1996:1), in the second case, where the first buffer is the bottleneck the RTV is much higher (for both the FE and FS implementation) than in the first case. The RTV seems to grow quadratically with L.

4.2 Flow line

The second model deals with a continuous flow line consisting of three machines and two intermediate buffers. Machine $i \in \{1, 2, 3\}$ can process the continuous flow products at some maximum rate ν_i, called the *machine speed*. Moreover, the machines are prone to failure. The life and repair times of the machine i are exponentially distributed with parameters λ_i and μ_i, respectively, and are independent of each other. The buffer capacities are C_1 and C_2. The system is depicted in Figure 2 .

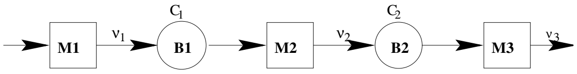

Figure 2: Flow line model

This model has been studied in Kroese et al. (1998), where an Importance Sampling procedure was described for the efficient estimation of the overflow probability γ in the second buffer $B2$ (defined as the probability that buffer $B2$ reaches level $L := C_2$ before it empties again, starting from an empty system). The translation into the RESTART set-up is the following. Let Y_t and Z_t be the level of the first and second buffer at time t, respectively; and let $\mathbf{M}_t \in \{0, 1\}^3$ denote the state of the machines at time t. Then obviously $X := (M_t, Y_t, Z_t, t \geq 0)$ is the basic Markov process of Section 2.

We wish to compare the performance of different RESTART methods with that of the Importance Sampling (IS) method in Kroese et al. (1998).

The model parameters are $\nu_1 = 3$, $\nu_2 = 2$, $\nu_3 = 1$; $\lambda_1 = 5$, $\lambda_2 = 2$; $\mu_1 = 1$ $\mu_2 = 1$ and $C_1 = 1$. The third machine is assumed to be perfectly reliable. In Kroese et al. (1998), a uniformization-based approach is used to implement the IS estimator. A uniformization rate of 70 was found to give the most accurate results and the number of cycles (runs) is 10^5. The overflow level L is taken to be 3; The intermediate levels for the RESTART methods are $L_k := k/2, \quad k = 1, \ldots, 6$. The number of runs is $1.5 \cdot 10^6$ for the first stage and 10^6 for the following stages. The total simulation time for each experiment is about one minute. See Table 2 for simulation results

Table 2: Results for the flow line model

	$\hat{\gamma}$	RE	sec	RTV
SS	8.20e-6	0.110	61	0.7408
IS	7.95e-6	0.0149	63	0.0140
RS,FA	7.85e-6	0.0070	62	0.0031
RS,RA	7.79e-6	0.0077	65	0.0038

Comparing the RTV's we observe that the RESTART method compares very well with IS. It is not safe to conclude that in general the RESTART method outperforms IS, since this would require a more careful consideration of their respective implementations. We found the RESTART method easier to implement, and requiring less critical optimization parameters, and thus a more robust estimator.

5 CONCLUSIONS

We have compared several implementations of the RESTART method and have found that the original

RESTART implementation (which we call the Fixed Splitting (FS) implementation), in which each path is split into a fixed number of copies, is in general not the most efficient one. It is better to fix the total simulation effort for each stage of the simulation (we call this the Fixed Effort (FE) implementation). In this way the number of paths that hit the next level will remain approximately the same, irrespective of how we chose the RESTART parameters. On the contrary, the FS implementation is very sensitive to the choice of the RESTART parameters; if we do not choose these parameters exactly right, the paths will either "die-out" or "explode", leading to excessive simulation time. For both methods (FE and FS) the "optimal" parameters are determined by making the success probabilities in each stage approximately e^{-2}, and the number of trials in each phase equal.

We also find that, if we use the FE method, it is better, for a given total effort per stage, to restart an equal number of times from each saved state, rather than to restart each time from a randomly (in this case uniformly) chosen saved state.

We note that if the entrance distribution is (approximately) known, we should sample from this (approximate) distribution, thus rendering the samples independent and reducing the variance of the estimator. The advantage of such an approach is currently being investigated. Another direction for future research is the estimation of *stationary* probabilities rather than overflow probabilities via the RESTART method.

6 ACKNOWLEDGEMENTS

The authors wish to thank Jos de Smit and Victor Nicola for helpful discussions and constructive comments.

REFERENCES

P. Glasserman, P. Heidelberger, P. Shahabuddin, T. Zajic, Multilevel Splitting for Estimating Rare Event Probabilities , *IBM Research Report* RC 20478 , 1996.

P. Glasserman, P. Heidelberger, P. Shahabuddin and T. Zajic, Splitting for Rare Event Simulation: Analysis of Simple Cases , *IBM Research Report* RC 20521 , 1996

P. Glasserman, P. Heidelberger, P. Shahabuddin, T. Zajic, A Large Deviations Perspective on the Efficiency of Multilevel Splitting ,*IBM Research Report* RC 20691 , 1997.

P.W. Glynn and W. Whitt, The asymptotic efficiency of simulation estimators. *Op. Research* 40, 505–520, 1992.

H. Kahn and T.E. Harris, Estimation of Particle Transmission by Random Sampling, *National Bureau of Standards Applied Mathematics Series* , 1951.

D.P. Kroese and V.F. Nicola, Efficient Simulation of Backlogs in Fluid Flow Lines, *Int. J. Electron. Commun. (AEÜ)* 52 (3), 165 – 171, 1998.

S. Parekh and J. Walrand, A Quick Simulation Method for Excessive Backlogs in Networks of Queues, *IEEE Trans. on Automatic Control* 34 , 54-66, 1989.

F. Schreiber and C. Görg, Rare Event Simulation: A Modified RESTART Method Using The LRE-Algorithm, *Proc. of the 1996 W. S. C.* , Coronado, California, 1996.

M. and J. Villén-Altamirano, RESTART: A Method For Accelerating Rare Event Simulations , *Proc. of the 13th Int. Teletraffic Congress, Queueing, Performance and Control in ATM* , ed. J.W. Cohen, 1991.

M. and J. Villén-Altamirano, Enhancement of the Accelerated Simulation Method RESTART by Considering Multiple Thresholds, *Proc. of the 14th Int. Teletraffic Congress, The Fundamental Role Of Teletraffic in the Evolution of Telecom. Networks* , Ed. J. Labetouille and J.W. Roberts, 1994.

M. and J. Villén-Altamirano, RESTART: An efficient and general method for fast simulation of rare events, *Technical Report* No. 7, Departamentado de Matemática Aplicada - Universidad Politécnica de Madrid, Julio 1997.

AUTHOR BIOGRAPHIES

M. J. J. GARVELS is a researcher at the Centre for Telematics and Information Technology at the University of Twente, The Netherlands. His research interests include simulation methods for queueing systems. He holds a degree in Mathematical Sciences from the University of Twente and is currently working towards his Ph.D. there.

D. P. KROESE holds a Ph.D degree in Mathematical Sciences from the University of Twente, The Netherlands, where he has been a Lecturer from 1990 to 1998. He has held visiting staff positions at Princeton University, U.S.A. and the University of Melbourne, Australia. Currently he holds a research staff position at the Teletraffic Research Centre in Adelaide, Australia. His interests include queueing theory, performance analysis, efficient simulation, point processes and fluid queues.