Proceedings of the 1996 Winter Simulation Conference
ed. J. M. Charnes, D. J. Morrice, D. T. Brunner, and J. J. Swain

A FRAMEWORK FOR RARE EVENT SIMULATION OF STOCHASTIC PETRI NETS USING "RESTART"

Christian Kelling

Technische Universität Berlin • Institut für Technische Informatik
Franklinstr. 28/29 • 10587 Berlin • Germany
email: ck@cs.tu-berlin.de

ABSTRACT

For the performability evaluation of complex systems simulation remains often the only feasible method. Unfortunately, simulation experiments tend to be very time consuming if rare events have to be considered. This paper describes an algorithmic approach for fast simulation of rare events applied in a Petri net modeling environment. The technique is based on the RESTART method, which is applicable for rare events in a wide range of simulation models, and has the potential to reduce the simulation overhead extremely. The paper presents selection and refinement techniques for the most important input parameters of RESTART. The techniques allow an efficient execution of RESTART simulations especially in a flexible evaluation tool. The results show run length reductions up to six orders of magnitude.

1 INTRODUCTION

Model-based performance and dependability evaluation of real-world systems usually requires large and complex models to be considered. Most analytical evaluation methods need the generation of the complete state space of the model. Despite several approaches to reduce the complexity of this task are known (Buchholz and Kemper 1995, Ciardo and Trivedi 1993, Ziegler and Szczerbicka 1995), state space explosion remains the most serious limitation of these techniques. In this situation simulation methods offer a valuable instrument.

The work presented here considers evaluation methods for models of so-called soft real-time systems. In those systems, rare events as failures, missed deadlines, or frame losses do not lead to a serious failure, but to degrading performance or limited quality of service, which can be tolerated in most cases. However, designers of applications like telephone switches, video transmission links, or data bases are interested in the probability of those rare events. For these reasons, the availability of simulation methods for rare events is of interest especially if they support well established modeling paradigms as Stochastic Petri Nets. Thus, the analyst is able to consider performability evaluation as a substantial part of system design. The Petri net framework here allows for both, the qualitative as well as the quantitative investigation of the models under study.

Traditional simulation techniques require a huge amount of computing time to achieve reliable estimates of performance measures if the probabilities of the related events are very small, say about 10^{-9} to 10^{-12}. The investigation of even small models by crude simulation under these constraints is simply impossible. To decrease the overhead of stochastic simulation, fast simulation techniques are applied. They offer the potential to reduce the simulation run length by some orders of magnitude.

On the other hand, most fast simulation techniques require individually adapted simulation algorithms representing special knowledge about the simulation model under study. This makes its incorporation into flexible, user-friendly evaluation tools difficult. In addition, for the most prominent technique, the importance sampling method, some serious constraints apply for the considered model. Stationary simulation using importance sampling is only possible if the model has regeneration points. Identification of regeneration points may become very costly in complex models. Moreover, a single model parameter which corresponds to the rare event of interest must be identified for the application of the likelihood ratios (Heidelberger 1995). Some promising results have been published applying importance sampling to the transient simulation of Stochastic Activity Networks in the tool UltraSAN (Obal and Sanders 1994).

A technique with higher flexibility is the RESTART method (REpetitive Simulation Trials After Reaching Thresholds (Villén-Altamirano 1991)). This method does not require the model to have regeneration points, moreover it is not necessary to identify a single model parameter for biasing. The basic idea is to repeat the simulation of those phases of the model evolution with higher probability which lead to the rare event under study. The starting points of the repetitions are called thresholds. At these points, the model state is saved and repetitive trials are simulated, each of them beginning at the threshold. Using this approach, the portion of simulation time spent in model phases which do not contribute to the estimate of the rare event probability is reduced significantly.

The developers of the method demonstrate for optimal parameters a speedup up to 10^8 for probabilities of about 10^{-11} in a queueing example (Villén-Altamirano 1994). However, this run length reduction can be achieved only if (nearly) optimal thresholds can be applied. Moreover, the probabilities related to the thresholds must be known to derive the corresponding number of retrials. The authors provide probabilities for optimal thresholds and the optimal number of retrials, but their determination requires to know the model behavior prior to the simulation study.

Hence, the determination of these input parameters remains the most difficult part for the application of RESTART. Only little research has been done concerning the selection of suitable input parameters and the effectiveness of the practical application of the method in a flexible simulation environment. In this paper we enhance a recently proposed selection procedure for thresholds of the RESTART method using results of a crude pre-simulation study and some information derived automatically from the considered model (Kelling 1995b). Furthermore, a new technique is described, which estimates the threshold probability during the simulation run and performs both, the adaptation of the threshold location according to the known optimality criterions and the computation of the optimal number of retrials based on the estimates.

The presented technique is part of TimeNET (Timed Net Evaluation Tool (German et al. 1995)), a modeling environment for Stochastic Petri Nets. SPNs are the most general class of timed Petri nets and allow for generally distributed firing times without structural restrictions. Despite analytical solution techniques are available for a large subclass of SPNs, discrete event simulation remains the sole method to handle complex models or nets with special properties. However, all Petri net properties based on the net structure can be also derived for simulation models. These properties are usually part of a qualitative analysis and are applied to proof an intended model behavior. The analytical components of TimeNET can handle SPN models with up to 10^6 states. The simulation component of the tool (Kelling 1995a) provides a framework for sequential as well as parallel and distributed execution of simulation experiments. The RESTART module enhances the simulator that includes standard simulation for the transient as well as the stationary case and a variance reduction method using control variates. Simulation studies with SPNs containing several hundred transitions and places and thousands of tokens have been performed with TimeNET.

The paper is organized as follows. In the next section, a short review of the RESTART method is given. Section 3 provides some background information about the considered class of Petri nets, introduces an example model, and derives preconditions for the applicability of RESTART. The fast simulation with RESTART in TimeNET is described in section 4 and results in section 5 demonstrate its performance. Finally we conclude and give directions for future work.

2 THE "RESTART" METHOD

RESTART has been proposed by M. and J. Villén-Altamirano and is based on ideas originally published by Bayes 1972, Hopmans 1979, and Wilson 1984. A complete introduction can be found in (Villén-Altamirano 1991) and (Villén-Altamirano 1994). The short review given here is mainly based on these papers.

The technique is developed from the observation that rare events tend to occur more often in a simulation run if a model state is chosen as the starting point which is significantly different from the average and closer to the rare event of interest. These biased starting points are called thresholds. Using the term event, we refer to an event in the probabilistic sense, i.e. a subset of the sample space or a state of the model.

Figure 1 shows a RESTART simulation. Suppose a simulation where $s(\omega)$ is defined for all $\omega \in \Omega$ and a rare event A given by $A = \{\omega \in \Omega : s(\omega) \geq S_A\}$ whose probability is to be estimated. We assume m events C_i with $C_i = \{\omega \in \Omega : s(\omega) \geq S_i\}$. Thus, an event C_i is related to each of the thresholds S_i and $C_1 \subset C_2 \subset \ldots \subset C_m \subset A$.

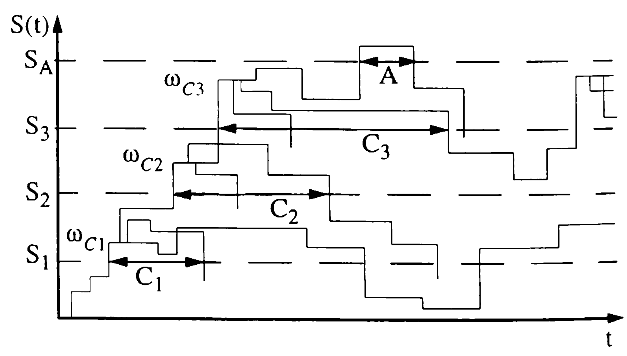

Figure 1: Simulation Run with RESTART

The RESTART algorithm comprises:

- If a transition from $\overline{C_i}$ to C_i occurs, the model state ω_{Ci} is saved.
- If a transition from C_i to $\overline{C_i}$ occurs, i.e. a state $\omega_{\overline{Ci}}$ is observed, the model state ω_{Ci} is restored and the

interval $[\omega_{C_i}, \omega_{\bar{C}_i})$ is simulated again using a different stream of random numbers.

- This procedure is repeated R_i times (R_i is the number of retrials according to threshold i). The starting state is always ω_{C_i}, but in general the trials end up with different states $\omega_{\bar{C}i1}, \omega_{\bar{C}i2}, \ldots, \omega_{\bar{C}iR_i}$.
- While a retrial is performed at level i, a transition from C_i to C_{i+1} may occur. In this case, R_{i+1} retrials are performed at level $i+1$ before the run proceeds at level i.
- If the trial R_i is complete, the simulation proceeds without retrials.

Given this scenario, the measure of interest can be computed by

$$\hat{P}(A) = \delta_A \left(t_{\text{first}} \prod_{i=1}^{m} R_i \right)^{-1}$$

where δ_A is the model time during A is observed in all retrials and t_{first} is the model time spent in the first retrials.

As the authors of RESTART show in (Villén-Altamirano 1991), the variance of this estimator can be computed, if the correlation between successive samples is known. The autocorrelation has to be taken into account, since the events A are no longer independent due to the forcing. However, to estimate the covariances requires much more overhead than the evaluation of the rare event itself. Our experiences with RESTART demonstrate that, with well known techniques to reduce the influence of autocorrelation (i.e. independent replications, batching), the method provides a very high accuracy, even if a standard variance estimation is applied. Detailed results are given in section 5 of this paper.

The main input parameter of this simulation technique is the location of the thresholds. Let $P_i = P(C_i | C_{i-1})$ ($1 \le i \le m$) be the conditional probability that a threshold S_i is reached given all states in which threshold S_{i-1} is already reached. To maximize the gain, under some simple assumptions optimal thresholds should be located so that $P_i = e^{-2}$ and the optimal number of retrials after crossing one of the thresholds is $R_i = P_i^{-1}$, which in the case of optimal locations obviously leads to $R_i = e^2$. In discrete models thresholds might be selected only from some discrete values. In this situation, the optimal number of retrials is derived from the thresholds probabilities by $R_i = (P_i \cdot P_{i+1})^{-1/2}$. Thus, the threshold selection criterion does not depend on the model under study. This result makes the incorporation of RESTART into a flexible simulation tool much more easier.

The remaining problem is the determination of suitable thresholds and the computation of the corresponding number of retrials. Before we turn to these new techniques, in the next section the relation between performance measures in Petri nets and the RESTART method is described and some requirements for its applicability are derived.

3 [RELATION TO STOCHASTIC PETRI NETS AND REQUIREMENTS

This part describes the links between measures derived from Petri net models and the RESTART method. A review of the considered class of timed Petri nets we use for designing and evaluating our simulation models is omitted. A detailed definition is given e.g. in (Ciardo et al. 1994).

Performance and reliability measures of timed Petri nets are usually expressed by tokens to be in a particular place. We distinguish between two typical kinds of estimators, the mean number of tokens in a place (E-measure, expectation) and the probability for a logical condition to be true (P-measure), in which this logical condition is also derived from a number of tokens to be in a place. Consider the example model of an ATM switch given in figure 2. A typical application of those ATM systems is the transmission of video data under real-time constraints. The arrival process is described by a Markov-modulated Poisson process. These processes are used for the modeling of superposed data and video traffic consisting of small packets (cells) in ATM systems (Lucantoni 1993). The model can be solved analytically, thus the simulation results can be validated. Simulation becomes necessary, if a transmission link with more than one switch and additional traffic sources is investigated.

The parameters of the SPN model are similar to (Blondia 1992). The transitions T1 and T2 model the modulator which can be in one of two states. A token in place P1 models a phase with a high cell arrival rate. The length of this phase is given by the firing delay of transition T1. The delay of T3 is the interarrival time in this case. When T1 fires, a phase with a lower cell traffic intensity starts. The arrivals are now determined by the delay of transition T4, the length of the period is given by T2. All transition delays in the modulator are exponentially distributed. The service in the queue is modelled by the subnet consisting of place P4 and transition T5. The service time is deterministic. A buffer is provided by place P4. We assume that the buffer capacity is limited to 50 cells. If the buffer is full and a new cell arrives, this cell gets lost (t2 fires, since the inhibitor arc from place P5 is without effect). This event corresponds to a frame loss in the video transmission.

Typical measures are related to the buffer. The mean number of cells in the buffer is expressed by the E-measure E{#P4}. A more important measure is the probability of a frame loss which occurs whenever a new cell arrives and the buffer is full. The loss probability is one of the service qualities an ATM provider has to ensure, thus

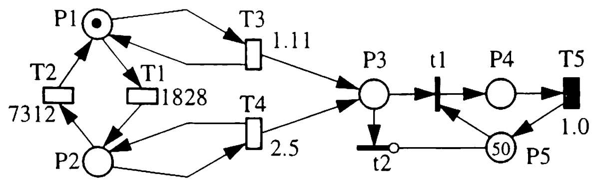

Figure 2: SPN of the MMPP/D/1/50-Queue

video transmission is a typical soft real-time application in this context. The measure is derived from a certain number of cells waiting for service and therefore a typical P-measure. The measure definition P{#P4>=c} describes that the number of waiting cells in the buffer reaches or exceeds the value *c*.

For two special cases of P-measures the RESTART method can be used. Our implementation deals with the probability that the number of tokens in the considered place reaches a certain limit. This type of measure usually corresponds to rare events in the modeled system, e.g. to buffer overflows or to losses as in the example above. From those results, probabilities for deadline violations due to failures can be derived.

The technique presented here determines suitable thresholds for one measure of a Petri net model which needs to be associated to only one place. The measure can either represent an upper limit of a number of tokens in a place (as the loss probability in the example) or a lower limit of that number. Therefore, subject of the evaluation is the rare event that a predefined number of tokens in the place can be observed.

Some requirements for the efficient application of RESTART must be met. First, for the measure monotonous behavior is required, i.e. given the rare event *A*, no events which are a subset of *A* may have a smaller probability than *A*. For measures which represent lower limits, an upper bound of the number of tokens must exist for the place of interest. This is a Petri net property which can be derived from so-called P-invariants. These invariants are one of the structural characteristics that can easily be determined for the underlying Petri net exploiting its structure. In addition, the number of tokens that can be added to or removed from the place of interest at a given time must be known. This can also be determined easily from the Petri net. The last precondition ensures that at a given time only the next threshold can be reached.

4 FAST SIMULATION WITH RESTART IN THE TOOL TimeNET

This section introduces the application of the RESTART technique. Despite fast simulation methods in general offer a huge potential for the acceleration of experiments, their applicability is often limited due to a lack of an user-friendly modeling framework or absent tool support or both. Often for each model an individually adapted simulation software is required.

The approach presented here allows each model to be evaluated automatically in a Petri net modeling tool using the RESTART technique as long as the rare event of interest can be expressed in a certain form (cf. section 3). Therefore the simulation technique is available for the users of the tool without programming overhead or model modification. The modeling framework of SPNs supports an automatic parameter selection. Finally, RESTART is the only fast simulation method, which is applicable to the transient and stationary simulation of non-regenerative models.

As shown above, the main problem with the application of RESTART is the selection of appropriate thresholds as starting points for the retrials and the determination of the number of these retrials. According to (Villén-Altamirano 1994) thresholds should be selected so that $P_i = e^{-2}$. Obviously, the optimal selection of the thresholds is only possible if some results of the model under study are known. Therefore, as a first step a pre-selection of thresholds prior to the simulation run is required. The next section reviews a new technique for the determination of suitable thresholds originally proposed in (Kelling 1995b).

4.1 Determination of Thresholds

The algorithm for the threshold selection is shown in figure 3. Suppose, for a considered place a rare event is defined exceeding a number of tokens *c*. Then, a maximum number of M ($M < c$) P-measures

$$p_i = P\{\#P_x \geq \left\lfloor \frac{ci}{M} \right\rfloor\},\ i \in \{1 \ldots (M-1)\}$$

is defined. A pre-simulation with predefined length is started estimating the probability of the measures p_i. This pilot run yields reliable estimates for the measures of high probability (small i). From those measures (nearly) optimal thresholds can be selected according to the optimality criterion (cf. section 2). Assume that $M' < M$ thresholds could be determined, then we have to derive the missing thresholds in the interval $(\lfloor c/M' \rfloor \ldots c)$. This interval is divided in equal sections defining approximate locations for those remaining thresholds.

Note, that this algorithm for upper limits can also be applied to lower limits, but in this case the marking of the considered place needs to be bounded by *c*, what can be derived from the corresponding P-invariant.

Applying this heuristic, at least some optimal thresholds can be found in the pilot run. Depending on the computing time the experimenter is willing to spent with the pre-simulation, the number of optimally selected thresholds can be increased easily. In any case, some approximately chosen thresholds are added to the optimal ones. As stated in

(Villén-Altamirano 1994), RESTART is sufficiently robust to deal with non-optimal thresholds. However, the efficiency can be increased significantly with optimally located thresholds. But, we found in experiments that the gain drops dramatically if a non-optimal number of retrials is used. This leads to a second step, that refines both, the initially selected thresholds and the number of retrials during the simulation run according to the optimality criterion. This algorithm is described in the next section.

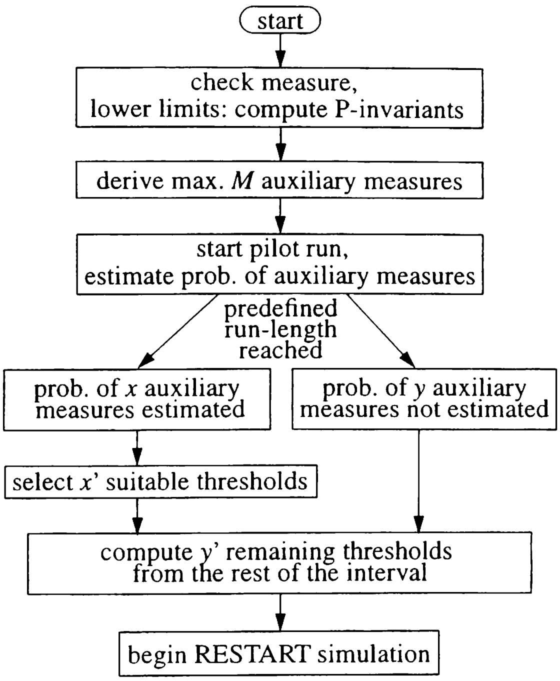

Figure 3: Threshold Determination for RESTART

4.2 Refinement of Threshold Location

As stated in section 2, the simulation proceeds without retrials whenever R_i retrials are complete. In the special case that R_1 retrials are done, the simulation behaves like a standard simulation, since no biased starting point was considered. That enables us to divide the whole RESTART run into K phases, in which the probability of the rare event can be estimated for each of the phases. The refinement method applies two main observations:

First, for each of the phases $1 \le k \le K$ different thresholds $S_i^{(k)}$ and different numbers of retrials $R_i^{(k)}$ can be applied. Each parameter set leads to one sample of the estimated rare event probability, which can be computed by

$$\hat{P}^{(k)}(A) = \delta_A^{(k)} \left(t_{\text{first}}^{(k)} \prod_{i=1}^{m^{(k)}} R_i^{(k)} \right)^{-1}.$$

Note, that the estimated sample of $P(A)$ is always correct, regardless which parameter set is applied (but it might be obtained inefficiently). By incorporating the samples from previous phases, the simulation results are not lost, if a modified parameter set is used.

Second, during each of the phases, not only the probability of the rare event A can be estimated, but additionally an estimate for each of the conditional probabilities P_i can be obtained. These intermediate results can justify the preselection of the threshold locations and the numbers of retrials. This is important for all thresholds for which no estimate could be found in the pilot run, and which therefore were established heuristically. The location will be examined by checking the optimality for the assumed $P_i^{(k)}$. Subsequently, new threshold locations can be used in the phase $k+1$, if the set of thresholds applied in phase k turns out to be inadequate. Additionally, the number of retrials for each threshold level is optimized using the estimated conditional probabilities P_i. This becomes necessary especially if constraints apply for the threshold values, as it is always the case in discrete models. Finally, the total number of thresholds is adjusted considering the first estimate of $P(A)$.

Since the small probabilities related to heuristically established thresholds describe always a tail of the distribution density function, for the determination of the threshold location linear regression is applied to predict the tail of the function. For this region of the most common pdfs, linearity turns out to be a useful assumption. Each of the decisions derived from the estimates of the various probabilities is only be taken, if the estimates are obtained from a sufficient number of samples. Of course, the accuracy requirements can be assumed to be significantly lower than those for the final estimation of $P(A)$. Additional tests, which compare the probabilities of the first thresholds (obtained in the pre-simulation) with estimates of the same measures obtained in the RESTART simulation ensure further stability.

4.3 The RESTART Component of TimeNET

After preliminary studies, a RESTART algorithm has been implemented in the simulator TimeNET-Sim (Kelling 1995a), which is the simulation component of the tool TimeNET. It allows for complete simulation experiments to estimate rare event probabilities. User have to provide the model with the performance measure specified. As an additional input parameter the number of thresholds might be given. This value can be approximated using the expected order of magnitude of the probability to be estimated. If this parameter is not specified, the adaptive procedures start with default values and determine the optimal number automatically.

The experiment starts with a pre-simulation to obtain the initial location of thresholds as described in section 4.1. With these thresholds the RESTART algorithm will be executed. During the run, the parameters are optimized as described in section 4.2. The simulation may run in parallel, i.e. distributed in a workstation cluster. Each of the distributed replications obtains an estimate for the performance measure and a centralized variance estimation determines a confidence interval. The optimization is done in each of the replications independently in order to omit synchronization. The simulation stops when a predefined accuracy is reached.

5 SIMULATION RESULTS

In this section we demonstrate the performance of the method. The simulation technique is employed to the evaluation of the quality of service in a real-time transmission setting. The example in figure 2 describes an ATM switch, the model together with the measures considered here has been explained in section 3.

The small probabilities for different numbers of ATM cells in the buffer (place P4) are investigated. All measures were estimated with a maximum relative half-width of the confidence interval of 0.1 at 0.95 confidence level. The length of the pre-simulation was set to 10 minutes. Thus, at least three thresholds can be selected optimally. All runs were executed using 10 parallel and independent replications in a workstation cluster. For the first two measures, also a crude simulation was performed to derive the speedup ratios. The overhead of this experiment ranges from 208 to (fictitious) $1.6 \cdot 10^8$ minutes. All other speedups are approximations based on these values.

Table 1 shows the results for a first experiment. The total number of thresholds was initially set to a default value of 10. This is a non-optimal choice for probabilities different from about 10^{-10}, but a realistic situation, since the experimenter does not know the model behavior in advance. Without refinement during the simulation, in some cases the run length was disastrous. It can be shown that the refinement reduces the run length in all cases. Note that the improvement is better for higher run length of the first approach.

A second experiment addresses the improvement caused by the refinement of the threshold location and the number of retrials only. Here the number of thresholds was set to their optimal values. The results in table 2 attest that the heuristic for the selection of the initial thresholds works very well and in some cases no improvement can be achieved by the refinement. On the other hand a further run length reduction by about 60% is possible. A detailed study is needed here to develop conditions for the successful application of the refinement.

Table 1: Speedups for a Default Number of Thresholds

meas.	prob.	no refinement		with refinement	
		sim. time (min.)	speed-up	sim. time (min.)	speed-up
$b \geq 26$	$1.27 \cdot 10^{-6}$	> 2700	<<1	140	1.5
$b \geq 31$	$1.0 \cdot 10^{-7}$	644	4	156	17
$b \geq 35$	$1.34 \cdot 10^{-8}$	187	104	71	275
$b \geq 40$	$1.07 \cdot 10^{-9}$	61	4180	24	$1.1 \cdot 10^4$
$b \geq 44$	$1.40 \cdot 10^{-10}$	48	$4 \cdot 10^4$	39	$4.8 \cdot 10^4$
$b \geq 48$	$1.35 \cdot 10^{-11}$	126	$1.3 \cdot 10^5$	112	$1.5 \cdot 10^5$
$b = 50$	$1.59 \cdot 10^{-12}$	154	$1 \cdot 10^6$	146	$1.1 \cdot 10^6$

Table 2: Effect of the Refinement of Threshold Location (Number Optimal)

meas.	prob.	no refinement		with refinement	
		sim. time (min.)	speed-up	sim. time (min.)	speed-up
$b \geq 26$	$1.27 \cdot 10^{-6}$	37	5	30	7
$b \geq 31$	$1.0 \cdot 10^{-7}$	40	65	52	50
$b \geq 35$	$1.34 \cdot 10^{-8}$	60	325	130	150
$b \geq 40$	$1.07 \cdot 10^{-9}$	61	4180	24	$1.1 \cdot 10^4$
$b \geq 44$	$1.40 \cdot 10^{-10}$	58	$3.3 \cdot 10^4$	39	$4.8 \cdot 10^4$
$b \geq 48$	$1.35 \cdot 10^{-11}$	50	$3.8 \cdot 10^5$	105	$1.6 \cdot 10^5$
$b = 50$	$1.59 \cdot 10^{-12}$	96	$1.7 \cdot 10^6$	206	$7.8 \cdot 10^6$

The last study comprises the sensitivity of the method, if initially a non-optimal number of thresholds is chosen. Table 3 summarizes the values. The RESTART approach without refinement shows drastically decreasing speedup, if the number of thresholds is smaller than the optimal one. Using the refinement, this effect can be compensated in most cases and a further run length reduction of more than one order of magnitude can be achieved.

The results show that the new simulation component offers excellent speedups. With the optimal number of thresholds given, the heuristic establishes thresholds as a set of input parameters for RESTART. Then, the simulation run reaches speedups up to $1.7 \cdot 10^6$ for probabilities of about 10^{-12}. Compared to the accurate numerical results, non of the relative errors was greater than 0.02. The optimization of the number of retrials turned out to have a greater impact than the refinement of the actual threshold location. The improvement by the factor 20 demonstrates that this feature is helpful when the optimal number of

Table 3: Effect of the Refinement if the Optimal Number of Thresholds Is Unknown

initial number of thresholds	run length (min.)	
	no refinement	with refinement
7	6000	305
8	100	177
9	61	69
10 (optimal)	70	24
11	93	73

thresholds is unknown prior to the simulation run. However, in some situation the refinement overhead is remarkable and simulations without this procedure are shorter. Thus, additional studies are required to discover the reasons for the increasing run length due to the refinement and to improve the algorithm.

The execution of the experiments in parallel provides independent samples for the variance estimation and yields an additional gain of about one order of magnitude.

6 SUMMARY

We have introduced an automatic simulation framework for fast simulation of rare events with RESTART. The implemented heuristic to determine the thresholds allows the incorporation of the RESTART acceleration into a Petri net simulation environment, runs transparently to the analyst and does not require detailed knowledge about the applied algorithm. Furthermore, an additional refinement to improve locations of thresholds and the number of retrials with respect to the optimality criterion is presented. The method conserves the simulation overhead done in previous phases by incorporating these samples into the final result estimation. Thus, no simulation time is lost, but the efficiency increases during the simulation run, if non-optimal parameters were selected at the beginning. A remarkable gain can be reached applying the refinement strategies, if the initially chosen parameters turn out to be significantly different from the optimal ones. Examples show that an investigation of probabilities as small as 10^{-12} is possible and that very accurate estimates can be obtained.

The techniques extend the work in (Villén-Altamirano 1994) by providing selection heuristics for thresholds and by proposing an adaptive algorithm for the refinement of the parameter setting during the simulation run. It presents the first integration of the RESTART method into a Petri net modeling tool.

Some additional studies are necessary for the detailed investigation of complex measures. Future work will address the reasons for the increasing run length due to the refinement and transient simulation with RESTART.

ACKNOWLEDGEMENTS

I am thankful to my students Patrick Drescher and Henning Ziegler who implemented the RESTART techniques and did patiently most of the simulation experiments. My thanks go to Manuel and Jose Villén-Altamirano for inviting me to Madrid, for the fruitful discussion, and their interest in my work.

REFERENCES

Bayes, A.J. 1972. A Minimum Variance Sampling Technique for Simulation Models, *Journal of the ACM* 19: 734–741.

Blondia, C. 1992. Performance Evaluation on an M/1-Stage in an ATM Switching Element. *Performance Evaluation* 15(1): 1–20.

Buchholz, P. and P. Kemper 1995. Numerical Analysis of Stochastic Marked Graph Nets. In *Proc of the Sixt Int. Workshop on Petri Nets and Performance Models*, Durham, North Carolina, USA, IEEE Press: 32–41.

Ciardo, G. and K. S. Trivedi 1993. A Decomposition Approach for Stochastic Reward Net Models. *Performance Evaluation* 18(1): 37–59.

Ciardo, G., R. German, and C. Lindemann 1994. A Characterization of the Stochastic Process Underlying a Stochastic Petri Net. *Trans. on Softw. Eng.*, 20(7): 506–515.

German, R., C. Kelling, A. Zimmermann and G. Hommel 1995. TimeNET - A Toolkit for Evaluating Non-Markovian Stochastic Petri Nets. *Performance Evaluation* 24 (1995): 69–87.

Heidelberger, P.1995. Fast Simulation of Rare Events in Queueing and Reliability Models. *ACM Transactions on Modeling and Computer Simulation*, 5(1): 43–85.

Hopmans, A.C.M. and J.P.C. Kleijnen 1979. Importance Sampling in Systems Simulation: A Practical Failure. In *Mathematics and Computing in Simulation XXI*. North-Holland: 209–220.

Kelling, C. 1995(a). TimeNET-Sim - a Parallel Simulator for Stochastic Petri Nets. In *Proc. of the 28th Annual Simulation Symposium*, Phoenix, AZ, USA, IEEE Press: 250–258.

Kelling, C. 1995(b). Rare Event Simulation with RESTART in a Petri Net Modeling Environment. In *Proc of the 1995 European Simulation Symposium*, Erlangen, Germany. SCS: 370–374.

Lucantoni, D. 1993. The BMAP/G/1-Queue: A Tutorial. In *Models and Techniques for the Performance Evaluation of Computer and Communications Systems*. L. Donatiello and R. Nelson. Berlin, Springer Verlag: 330–358.

Obal, W. D. and W. H. Sanders 1994. Importance Sampling Simulation in UltraSAN. *Simulation* 62(2): 98–111.

Villén-Altamirano, M. and J. Villén-Altamirano 1991. RESTART: A Method for Accelerating Rare Event Simulations. In *Queueing Performance and Control in ATM (ITC-13)*. J. W. Cohen and C. D. Pack, Elsevier Science Publishers B. V. (North-Holland): 71–76.

Villén-Altamirano, M. and J. Villén-Altamirano 1994. Enhancement of the Accelerated Simulation Method RESTART by Considering Multiple Thresholds. In *ITC-14*. J. Labetoulle and J. W. Roberts, Elsevier Science Publishers B. V. (North-Holland): 797–810.

Wilson, J.R. 1984. Variance Reduction Techniques for Digital Simulation. *American Journal on Mathematics in Managment Science*, 4(3, 4): 277–312.

Ziegler, P. and H. Szczerbicka 1995. A Structure Based Decomposition Approach for GSPN. In *Proc of the Sixt Int. Workshop on Petri Nets and Performance Models*, Durham, North Carolina, USA, IEEE Press: 261–270.

AUTHOR BIOGRAPHY

CHRISTIAN KELLING is a research assistent in the Computer Science Department of the Berlin University of Techology in Germany. He received his Diploma degree in electrical engineering from the Humboldt-University of Berlin in 1992, and received his Ph. D. degree in 1995 from the Berlin University of Techology. His research interests focus on fast simulation techniques for design and evaluation of highly reliable systems as real-time systems and embedded architectures. He is a member of GI and SCS.