Numerical Analysis of Generalized Semi-Markov Processes

Christoph Lindemann, University of Leipzig, Germany

Key Words: discrete-event stochastic systems, performance and reliability modeling, efficient techniques for transient and stationary analysis of stochastic processes

SUMMARY

This paper presents methodological results that allow the effective numerical analysis of finite-state generalized semi-Markov processes (GSMPs) with exponential and possibly concurrent deterministic events by an embedded general state space Markov chain (GSSMC). Key contributions constitute (i) the derivation of an algorithmic approach how elements of the transition kernel of the GSSMC can always be computed by appropriate summation of transient state probabilities of continuous-time Markov chains and (ii) the derivation of conditions under which kernel elements are constant. The exploitation of these properties is the key driver for the efficient and accurate transient and steady-state analysis of the considered class of GSMPs. The practical applicability of the presented approach is shown by an MMPP/D/2/K queueing system.

1 INTRODUCTION

The results presented in this paper were obtained in the years 2000 to 2002; i.e. concurrently to Kishor Trivedi's and his Ph.D. students' work on the analysis of Markov regenerative stochastic Petri nets and related modeling formalisms with an underlying stochastic process can be represented as a Markov regenerative process (see e.g. [2], [3]).

Since many activities associated with computer and communication systems have a constant duration, performance and dependability models of such systems should allow representation of both stochastic and deterministic timing. Activities of computer systems which have a constant duration include transfer times for data packets of fixed size, time-outs, and repair times of components. This paper deals with numerical methods for analysis of discrete-event systems with stochastic and deterministic timing. A discrete-event stochastic system makes state transitions when events associated with the occupied state occur; events occur only at an increasing sequence of random times. The underlying stochastic process of a discrete-event stochastic system records the state of the system as it evolves over continuous time. The usual model for this process is a generalized semi-Markov process (GSMP); see e.g, Glynn [6]. Although a GSMP constitutes a very general stochastic process, a rich body of theoretical results on monotonicity, regeneration, and continuity is available. Due to their generality, the analysis of GSMPs can be performed by discrete-event simulation only.

Previous work on the analysis of discrete-event stochastic systems with exponential and deterministic events considers the structurally restricted case, that deterministic events are not concurrently enabled. The analysis was mainly done in the context of deterministic and stochastic Petri nets (DSPNs, [1]). Recently, Ciardo and Li considered the approximate transient analysis of DSPNs with only a single deterministic transition that cannot get cancelled [3]. For both stationary and transient analysis of DSPNs, approaches based on Markov renewal theory (see e.g. [1], [2]) and on the method of supplementary variables (see e.g. [5]) have been considered. Unfortunately, the practical applicability of the supplementary variables approach is severely limited because it requires, already in the restricted case, numerical solution of a system of partial differential equations. The analysis of non-Markovian models under the assumption that only one non-exponential transition can be enabled at a time was studied by de Souza e Silva, Gail, and Muntz [15], [16]. Their approach is based on a discrete-time Markov chain embedded at starting or completion times of non-exponential events. The analysis of non-Markovian models with different preemption policies of non-exponential transitions was studied in [17].

Considering finite-state GSMPs with exponential and possibly concurrent deterministic events, Lindemann and Shedler introduced the first cost-effective numerical method for the analysis of such processes. Their approach is based on a general state space Markov chain (GSSMC) embedded at equidistant time points *nD* (*n=1, 2, ...*) of the continuous-time GSMP [10]. This numerical approach consists of two main steps: the derivation of the transition kernel and the solution of a system of multidimensional Fredholm integral equations. These integral equations constitute the time-dependent and stationary equations of the GSMP and have been presented in [10], [11] and [12], respectively. To make this GSSMC approach effectively applicable in performance and dependability modeling projects at large, the remaining open problem constitutes the algorithmic generation of the simplest form of the transition kernel of this GSSMC given the building blocks of the GSMP. The transition kernel of the GSSMC specifies one-step jump probabilities from a given state *s* at instant of time *nD* to all reachable new states *s'* at instant of time (*n*+1)*D*. In general, elements of the transition kernel of a GSSMC are functions of clock readings associated with the old state *s* and intervals of clock readings associated with the new state *s'*.

This paper presents two theorems that provide the foundation for such an effective algorithmic generation of the transition kernel. Key contributions constitute (i) the derivation of an algorithmic approach how kernel elements can always be computed by summation of transient state probabilities of continuous-time Markov chains (Theorem 1) and (ii) the derivation of conditions on the building blocks of the GSMP under which kernel elements are constant; i.e., are not functions of clock readings (Theorem 2).

The remainder of this paper is organized as follows. To make the paper self-content, we recapitulate in Section 2 the methodology for numerical analysis of GSMPs with exponential and concurrent deterministic events and introduce the notation. Section 3 presents two theorems on properties of the GSSMC which constitute the main results of this paper. To illustrate the practical applicability of the methodological results, we consider in Section 4 an MMPP/D/2/K queueing system. Finally, concluding remarks are given.

2 THE ANALYSIS METHODOLOGY

2.1 *The Embedded General State Space Markov Chain*

A generalized semi-Markov process (GSMP) is a continuous-time stochastic process that makes a state transition when one or more "*events*" associated with the occupied state occur. Events associated with a state compete to trigger the next state transition, and each set of trigger events has its own distribution for determining the next state. At each state transition of the GSMP, *new* events may be scheduled. For each of these new events, a clock indicating the time until the event is scheduled to occur is set according to an independent (stochastic) mechanism, i.e., for each new event a clock reading is generated according to its *clock setting distribution*. For each scheduled event which does not trigger a state transition but is still scheduled in the next state, its clock *continues* to run. If an event is no longer scheduled in the next state, it is *canceled*, and the corresponding clock reading is discarded.

In this paper, we consider finite-state, time-homogeneous GSMPs with exponential and deterministic clock setting distributions. Let $E = \{e_1, e_2,..., e_K\}$ be a finite set of events and $S = \{s_1, s_2,..., s_N\}$ be a finite set of states. For a state $s \in S$, let $s \longmapsto E(s)$ be a mapping from the set S to a nonempty subsets of E; $E(s)$ denotes the set of all events that are scheduled to occur when the process is in state s. Denote the probability that the new state is s' given that the event $e^* \in E(s)$ occurs in state s by $p(s',s,e^*)$. For each $s \in S$ and $e^* \in E(s)$, we assume that $p(\cdot,s,e^*)$ is a probability mass function (pmf). We divide the set of events $E = E_{exp} \cup E_{det}$ and enumerate the deterministic events by $e_1, e_2, ..., e_M$. Subsequently, we define D_m to be the firing delay of event e_m ($1 \leq m \leq M$). For the analysis of this class of GSMPs, in [10] a discrete-time general state space Markov chain (GSSMC) has been introduced. According to [10], we define $D = min \{D_1, D_2, ..., D_M\}$. To derive this GSSMC, we define a discrete-time process $X_n = \{X(nD): n \geq 0\}$ by observing the GSMP at a sequence $\{nD: n \geq 0\}$ of fixed times, i.e., $X_n = (S_n, C_{n,1}, C_{n,2}, ..., C_{n,M})$. Here, S_n represents the state of the GSMP and $C_{n,m}$ represents the clock reading of deterministic event e_m ($1 \leq m \leq M$) at instant of time nD. When deterministic event e_m is not enabled in state S_n, we set $C_{n,m} = 0$. The memoryless property of the exponential distribution implies that X_n is a GSSMC.

For ease of exposition, we restrict the discussion to GSMPs in which at most two deterministic events may be concurrently enabled. Furthermore, we assume that deterministic events cannot get canceled. However, we would like to point out that from a theoretical point of view the presented methods can be generalized in a straight-forward way such that none of these restrictions is necessary; in practice the analysis gets more time- and space-consuming and numerical accuracy suffers. Nevertheless, we describe how to deal with deterministic events that can get canceled at several places in the text. According to [12], we divide the set of states $S = S_{exp} \cup S_{det1} \cup S_{det2}$ and denote the subset of states in which only exponential events are enabled by $S_{exp} =\{s_1, s_2,..., s_{N1}\}$. Similarly, the subsets of states in which one deterministic event and two deterministic events are (concurrently) enabled are denoted by $S_{det1} = \{s_{N1+1}, s_{N1+2}, ..., s_{N1+N2}\}$ and $S_{det2} = \{s_{N1+N2+1}, s_{N1+N2+2},..., s_N\}$, respectively. We denote the index of the deterministic event enabled in a state $s_i \in S_{det1}$ by $l(i)$ and the corresponding clock reading is denoted by c_1. For $s_i \in S_{det2}$ we denote the indices of the enabled deterministic events by $l(i)$ and $m(i)$, with $l(i) < m(i)$, and the clock readings of $e_{l(i)}$ and $e_{m(i)}$ are denoted by c_1 and c_2, respectively. Other zero-valued clock readings are neglected.

For graphical representation, we introduce the notion of the *state transition graph* of a GSMP, which is defined as a directed multigraph with a set of vertices S, i.e., the states of the GSMP. The states are connected by labeled edges representing state transitions. An edge corresponding to a state transition from state s to s' which is triggered by the deterministic event $e \in E_{det}$ is denoted by a triple (s,s',e). Edges representing exponential state transitions are triples (s,s',λ) with $s, s' \in S$ and λ the rate of the exponential event that triggers the corresponding state transition. A general (weighted) edge representing both cases is denoted by (s,s',w).

2.2 *General Form of the Transition Kernel*

Recall that a GSSMC is completely specified by a transition kernel and an initial distribution at time $t = 0$. The transition kernel of the GSSMC specifies one-step jump probabilities from a given state at instant of time nD to all reachable new states at instant of time $(n+1)D$. As for an ordinary discrete-time Markov chain, for all states s_j not reachable from s_i corresponding jump probabilities $p_{ij}(.)$ are zero. In general, elements of the transition kernel of the GSSMC are functions of clock readings associated with the current state s_i and the new state s_j. The transition kernel of the GSSMC $X_n = \{(S_n,C_n): n \geq 0\}$ constitutes a functional matrix of the form $P(c,A)$ with elements

$$p_{ij}(c,A) = P\left[X_{n+1} \in \{s_j\} \times A \middle| X_n = (s_i,c)\right] \quad (1)$$

Restricting the discussion to GSMPs with at most two deterministic events concurrently enabled, the vector of old clock readings c and the set A for intervals of new clock readings are given by:

$$c = c(s_i) = \begin{cases} \emptyset & , s_i \in S_{exp} \\ c_1 & , s_i \in S_{det1} \\ (c_1, c_2) & , s_i \in S_{det2} \end{cases}$$

$$and\, A = A(s_j) = \begin{cases} \emptyset & , s_j \in S_{exp} \\ (0, a_1] & , s_j \in S_{det1} \\ (0, a_1] \times (0, a_2] & , s_j \in S_{det2} \end{cases} \tag{2}$$

Thus, for GSMPs with at most two deterministic events concurrently enabled, the transition kernel of the GSSMC can be expressed by a functional matrix *P(c₁,c₂,a₁,a₂)*. Subsequently, an element of this kernel $p_{ij}(\cdot)$ is in general a function in four variables c_1, c_2, a_1, and a_2. However, we will observe that a large number of kernel elements are constant (Theorem 2), i.e., $p_{ij}(c_1,c_2,a_1,a_2) = p_{ij}$. Furthermore, for most functional kernel elements new clock readings need not be considered, i.e., $p_{ij}(c_1,c_2,a_1,a_2) = p_{ij}(c_1,c_2)$.

$$P(c_1,c_2,a_1,a_2)=\begin{array}{cc} \left(\begin{array}{c|c|c} P_{11} & P_{12}(a_1) & P_{13}(a_1,a_2) \\ \hline P_{21}(c_1) & P_{22}(c_1,a_1) & P_{23}(c_1,a_1,a_2) \\ \hline P_{31}(c_1,c_2) & P_{32}(c_1,c_2,a_1) & P_{33}(c_1,c_2,a_1,a_2) \end{array}\right) & \begin{array}{c} 1 \\ \vdots \\ N_1 \\ \hdashline N_1+1 \\ \vdots \\ N_1+N_2 \\ \hdashline N_1+N_2+1 \\ \vdots \\ N \end{array} \\ 1 \qquad\qquad N_1 \,\vdots\, N_1+1 \quad N_1+N_2 \,\vdots\, N_1+N_2+1 \quad N & \end{array} \tag{3}$$

Eq. (3) shows the general form of the kernel $P(c_1,c_2,a_1,a_2)$ as a composition of nine submatrices $P_{ij}(\cdot)$ of appropriate dimension using (1) and (2). In (3), the submatrix P_{11} represents state transitions among states of S_{exp}. State transitions from states of S_{exp} to states of S_{det1} and S_{det2} are represented by the submatrices $P_{12}(a_1)$ and $P_{13}(a_1,a_2)$. Furthermore, submatrix $P_{22}(c_1,a_1)$ represents state transitions among states of S_{det1} and $P_{21}(c_1)$ represents state transitions from states of S_{det1} to states of S_{exp}. The submatrix $P_{23}(c_1,a_1,a_2)$ represents state transitions from states of S_{det1} to states of S_{det2}, respectively. State transitions from states of S_{det2} to states of S_{det1} and S_{exp} are represented by the submatrices $P_{32}(c_1,c_2,a_1)$ and $P_{31}(c_1,c_2)$. The submatrix $P_{33}(c_1,c_2,a_1,a_2)$ represents state transitions among states of S_{det2}.

Recall that an element $p_{ij}(c_1,c_2,a_1,a_2)$ of the transition kernel constitutes the conditional one-step jump probability of been in state s_i with deterministic clock readings c_1 and c_2 and jumping to state s_j with clock readings less or equal than a_1 and a_2, respectively. To solve the time-dependent and stationary equations of the GSSMC, kernel elements have to be unconditioned by applying the low of total probability resulting in the system of Fredholm integral equations [10], [12].

3 THEOREMS ON PROPERTIES OF THE TRANSITION KERNEL

3.1 *Computation of the Transition Kernel*

In recent work, the concept of subordinated Markov chains (SMCs) has been applied for the efficient algorithmic computation of the probability matrix P of the discrete-time Markov chain embedded in the Markov regenerative process underlying a discrete-event stochastic system without concurrent deterministic events (see e.g. [9], [11]). The SMC of state s_i is a continuous-time Markov chain (CTMC) whose state space is given by the transitive closure of all states reachable from s_i via a (possible empty) sequence of exponential events and corresponding next state probabilities $p(s',s_i,e^*)$ of the GSMP. For such a sequence of exponential events from s_i to s_j, we write $s_i \xrightarrow{exp*} s_j$. We define a SMC for each state of the GSMP, i.e., also for states in which only exponential events are enabled.

Definition (Subordinated Markov chain): The continuous-time Markov chain $\{X_i(t): t \geq 0\}$ with state transitions corresponding to the occurrence of exponential events, state space $SMC_i = \left\{s \in S \mid s_i \xrightarrow{exp*} s\right\}$, and initial distribution $P\{X_i(0) = s_i\} = 1$ is called the *subordinated Markov chain (SMC)* of state s_i.

The following provides an intuitive explanation why elements of the transition kernel of a GSSMC can always be determined by appropriate sums of transient state probabilities of continuous-time Markov chains. Assuming the GSMP is at time *nD* in state s_i with two deterministic events $e_{l(i)}$ and $e_{m(i)}$ concurrently enabled. Thus, the GSSMC resides in a state, say (s_i,c_1,c_2) with $c_1 \leq c_2$, where c_1 and c_2 are clock readings associated with deterministic events $e_{l(i)}$ and $e_{m(i)}$, respectively. Noting that the state of the GSMP at time *(n+1)D* given the state at time *nD* is determined by a (possibly empty) sequences of exponential events in the subintervals *((nD,nD+c₁]*, *(nD+c₁,nD+c₂]* and *(nD+c₂,(n+1)D]* and the occurrence of the deterministic events $e_{l(i)}$ and $e_{m(i)}$ at instants of time $nD+c_1$ and $nD+c_2$, respectively. Thus, using the property that the GSMP is time-homogeneous and by decomposing the time interval *(0,D]* into three subintervals *(0,c₁]*, *(c₁,c₂]*, and *(c₂,D]*, the GSMP behaves in each subinterval as a CTMC. Each of these three CTMCs is given by an SMC as defined above. Subsequently, the kernel elements of the embedded GSSMC can be computed as summations of transient state probabilities of SMCs. It is important to note, that this holds irrespective of the number of deterministic events enabled in states s_i and s_j. The following theorem summarizes the discussion above and constitutes one of the main results of this paper.

Theorem 1 (Numerical computation of the transition kernel): Let $\{X(t): t \geq 0\}$ be a finite-state GSMP with exponential and deterministic events. Then, all elements $p_{ij}(.)$ of the transition kernel $P(c_1,c_2,a_1,a_2)$ of the embedded GSSMC $\{X_n: n \geq 0\}$ can be computed simply by summation of transient state probabilities of continuous-time Markov chains.

Proof: See [14].

Note that for the case if $S = S_{det2}$ the analysis simplifies considerably, since in each state of the GSSMC both deterministic clock readings are enabled. Therefore, if additionally deterministic events cannot get canceled, clock readings of the GSMP once set in the initial state are the same at every embedding time point *nD* of the GSSMC. As a consequence, the transition kernel $P(c_1,c_2,a_1,a_2)$ of the GSSMC reduces to a state transition matrix $P_{\tilde{c}_1,\tilde{c}_2}$, $0 < \tilde{c}_1, \tilde{c}_2 \leq D$, of an ordinary DTMC for every fixed initial clock readings $c_1 = \tilde{c}_1$

and $c_2 = \tilde{c}_2$. This provides a probabilistic explanation why an embedded DTMC for a structurally restricted class of deterministic and stochastic Petri nets, recently studied by German [4], exists.

3.2 *Detection of Constant Kernel Elements*

In this section, we state sufficient conditions on the building blocks of the GSMP under which kernel elements are constant because jump probabilities of the GSSMC are independent of clock readings. Examples of this case for which the entire kernel comprises of constant kernel elements constitute GSMPs underlying queueing systems with quasi birth-death arrival process, one or several deterministic servers, and infinite waiting room (i.e., MAP/D/c queues). This implies that for such GSMPs the corresponding GSSMC behaves in fact as an ordinary DTMC. As illustrated in Section 5, the GSSMC underlying queueing systems with finite waiting room behaves almost as a DTMC, i.e., almost all kernel elements are constant.

In order to decide whether kernel elements $p_{ij}(c_1,c_2)$ are also independent of old clock readings c_1 and c_2 we consider a path in the state transition graph of the GSMP. A *path* from state s_i to s_j is defined as a sequence of states (and corresponding enabled and occurring events) that can be temporarily hold by the GSMP when traversing from state s_i to s_j in time interval $[0,D]$. The *length* of a path is defined as the number of states traversed. The set of *all* feasible paths from state s_i to s_j is denoted by $PATH(s_i,s_j)$. Assuming that deterministic events cannot get canceled, Figure 3 depicts the general form of a path $\tau \in PATH(s_i,s_j)$ of length n from state $s_i \in S_{det1}$ to an arbitrary state s_j. The deterministic event $e_{l(i)}$ occurs in the k-th state z_k of the path. Furthermore, the rates of exponential events that are enabled and occur on this path are included in Figure 3. Note that in general the probability of traversing a path from s_i to s_j depends on the clock reading c_1 of the deterministic event $e_{l(i)}$. The key idea for detecting kernel elements that are independent of clock reading c_1 is to show that under some conditions the paths of $PATH(s_i,s_j)$ can be grouped into a (possibly infinite) number of classes Γ_m, $m = 1,2,...,$ such that the *proportionate jump probability* $p_{ijm}(c_1)$ of traversing the paths of Γ_m is constant. Furthermore we show, that the detection as well as the computation of constant kernel elements can be easily performed without explicitly computing the classes Γ_m.

Theorem 2 presents a sufficient condition under which kernel elements $p_{ij}(c_1)$ for $s_i, s_j \in S_{det1}$ are constant.

Theorem 2 (Constant kernel elements): Consider a GSMP with exponential and deterministic events. Let $P(c_1,c_2,a_1,a_2)$ be the transition kernel of its GSSMC $\{X_n : n \geq 0\}$ and s_i and $s_j \in S_{det1}$ with $p_{ij}(c_1,a_1)$ independent of a_1. Then, the corresponding kernel element $p_{ij}(c_1,a_1)$ is constant if the following condition holds:

$$(s, s', w) \in E_{pre}(s_i, s_j, e_{l(i)})$$
$$\Leftrightarrow (f_{l(i)}(s), f_{l(i)}(s'), w) \in E_{post}(s_i, s_j, e_{l(i)})$$

Proof: See [14].

The condition of Theorem 2 can be generalized to kernel elements of the form $p_{ij}(c_1,c_2)$ for s_i, $s_j \in S_{det2}$. For this case we have separate conditions for each clock reading c_1 and c_2:

$$P_{ij}(c_1, c_2) = p_{ij}(c_2) \; if \; (s, s', w) \in E_{pre}(s_i, s_j, e_{l(i)})$$
$$\Leftrightarrow (f_{l(i)}(s), f_{l(i)}(s'), w) \in E_{post}(s_i, s_j e_{l(i)}) \qquad (4)$$
$$P_{ij}(c_1, c_2) = p_{ij}(c_1) \; if \; (s, s', w) \in E_{pre}(s_i, s_j, e_{m(i)})$$
$$\Leftrightarrow (f_{m(i)}(s), f_{m(i)}(s'), w) \in E_{post}(s_i, s_j e_{m(i)}) \qquad (5)$$

The kernel element $p_{ij}(c_1,c_2,a_1,a_2) = p_{ij}(c_1,c_2)$ is constant if both, conditions (4) and (5) hold.

4 *PERFORMANCE RESULTS FOR THE MMPP/D/2/K QUEUE*

To illustrate the impact of the methodological results of the previous sections, we consider two application examples of high interest for communication network performance analysis. The experiments have been performed on a PC workstation with a 2.3 GHz processor and 2 GB main memory running the operating system Linux. For the performance tests the user CPU time has been measured with the system call times.

We consider an MMPP/D/2/K queueing system comprising of two identical servers with constant service time $D = 1.0$ seconds. Arrivals occur according to a Poisson process which is controlled by an irreducible CTMC with two states, representing bursty and normal mode, i.e., non-bursty mode of arrivals of customers. The duration of bursty mode and normal

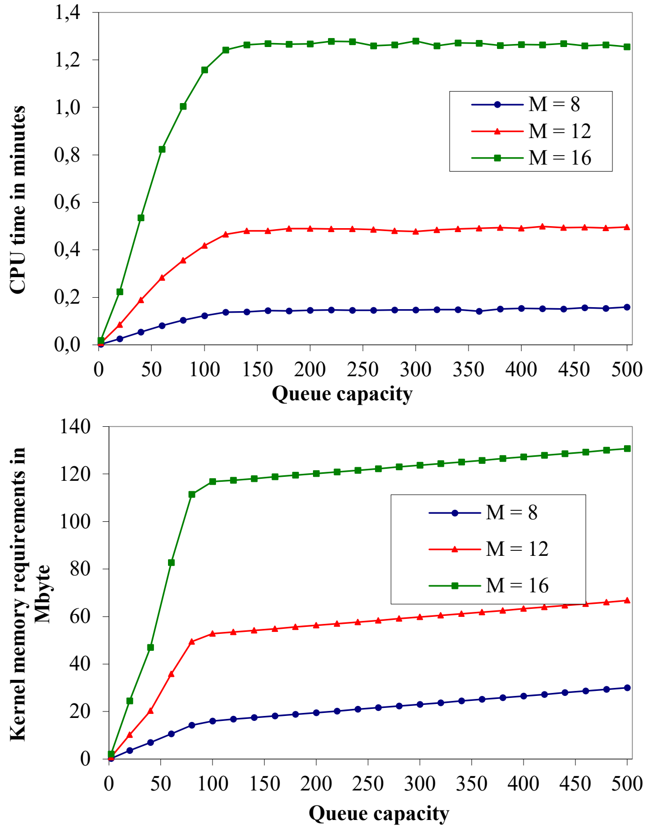

Figure 1. Generation of transition kernel: CPU time and memory requirements

mode is assumed to be 0.5 seconds and 100 seconds, respectively. The arrival rates in bursty mode and normal mode, denoted by λ_{bursty} and λ_{normal}, are determined from the average arrival rate $\lambda_{avg} = 0.5$ and the intensity of burstiness (burstfactor), denoted by $B = \lambda_{bursty}/\lambda_{normal}$.

The following experiment considers the performance of the algorithmic approach for computing the kernel generation. The left side of Figure 1 shows the CPU time needed to generate the transition kernel, i.e., transient analysis of SMCs using randomization (see e.g. [7], [8]) plus summation of transient state probabilities. The right side of Figure 1 shows the corresponding memory requirements of the transition kernel. The experiments are presented for varying model size and different numbers of discretization steps M. The remaining parameters of the model are kept fixed, i.e., $\lambda_{avg} = 0.5$ and B = 150. Note that model size depends on the queue capacity. That is, for queue capacity K the model consists of $2{\cdot}(K{+}2)$ states. As expected, the computation as well as the memory requirement is much more time and space consuming for increasing number of discretization steps M. Increasing the model size results in a different observation. For queue capacities $K = 2$ to $K = 120$ a significant increase in time and space is observed. For larger queue capacities only a very slightly further increase in time and space can be seen. This is due to the fact that for increasing queue capacity more than 98% of nonzero kernel elements are constant as shown in Figure 2. Furthermore, the kernel generation employs a dynamic sparsing method by setting both constant and functional kernel elements smaller than a given threshold $\varepsilon = 10^{-16}$ to zero. This results in an almost linear growth of the nonzero kernel elements and stagnation in number of functional kernel elements for this class of GSMP models. Figure 3 shows the CPU time required for solving the system of integral equations according to the iterative algorithm proposed in [12]. Results for 100 iterations of the “Picard algorithm” for transient solution are presented. As expected, the curves have a similar shape as the curves for kernel generation of Figure 1 since the time consumed for numerical solution of the system of Fredholm integral equations depends on the number of nonzero kernel elements.

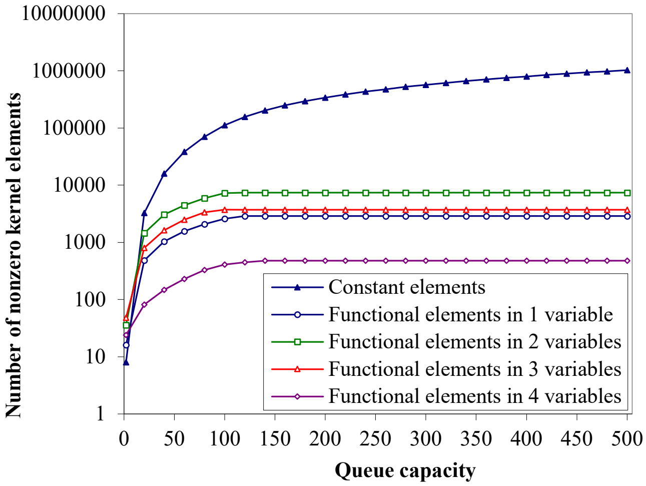

Figure 2. Classification of elements of the transition kernel

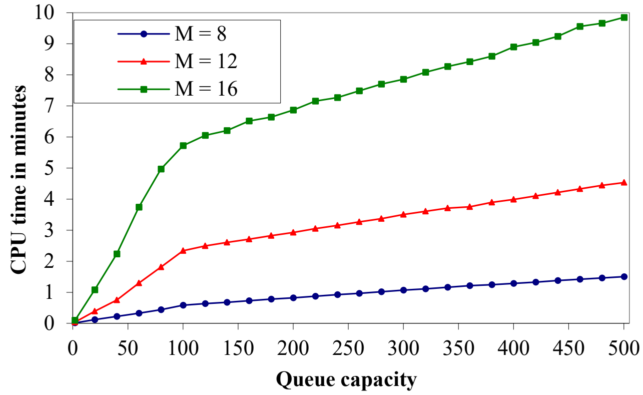

Figure 3. Picard iteration for solving system of integral equations

To show the scalability of the solution algorithm we consider the CPU time required for models with up to 15.000 states, i.e., $K = 7.498$. The left side of Figure 4 shows the time needed for transient analysis of the SMCs and the right side shows the time spend summing up the transient probabilities to compute constant and functional kernel elements.

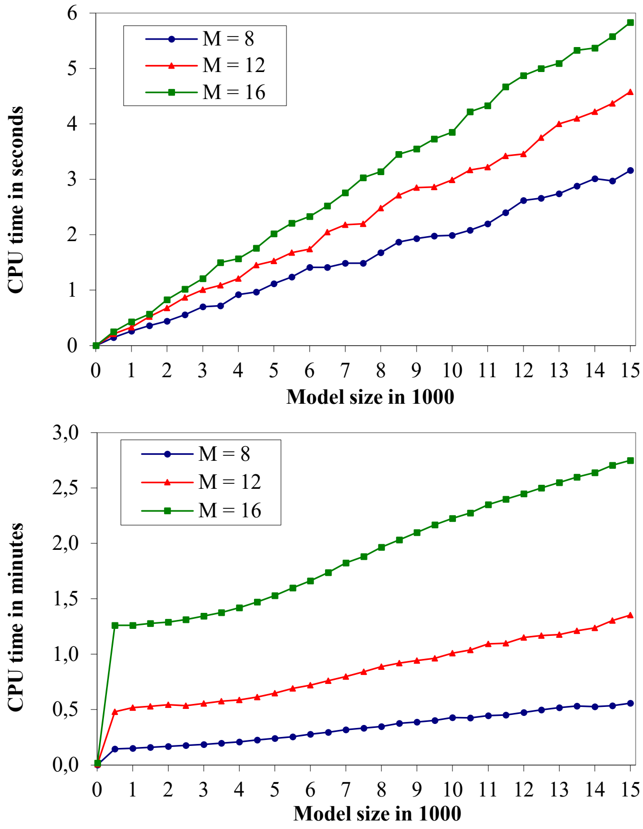

Figure 4. Generation of transition kernel: CPU time required for transient analysis of SMCs (first) and summation of transient state probabilities (second)

As in Figure 1, we observe that the CPU time grows nearly linear for increasing model size. Furthermore, for models with up to 15.000 states the generation of the transition kernel with

$M = 8$ discretization steps requires less than 30 seconds of CPU time.

This is due to the sparseness of the transition kernel and the exploitation of constant kernel elements according to Section 3.2. Thus, for such GSMPs underlying finite-capacity queueing systems with two deterministic servers, the exploitation of constant kernel elements is key for their highly efficient transient and steady-state analysis.

CONCLUSIONS

In this paper, we presented two theorems that provide the foundation for the effective algorithmic generation of the transition kernel of the general state space Markov chain (GSSMC) underlying a GSMP with exponential and possibly concurrent deterministic events. Key contributions constitutes the derivation of an algorithmic approach how kernel elements can always be computed by appropriate summation of transient state probabilities of continuous-time Markov chains (Theorem 1). Furthermore, we derived conditions on the building blocks of the GSMP under which kernel elements of its GSSMC are constant (Theorem 2). Thus, for such state transitions the GSSMC behaves like a discrete-time Markov chain. Applying Theorem 2, we showed that the considered queueing systems almost all kernel elements are constant.

In [13], a mapping from system specifications described by time-enhanced state diagrams and activity diagrams of the Unified Modeling Language (UML) onto a GSMP with exponential and deterministic events has been introduced. The methodological results of this paper can be put into practice for the quantitative analysis of such system specifications in the time-enhanced UML.

REFERENCES

1. M. Ajmone Marsan and G. Chiola, “On Petri Nets with Deterministic and Exponentially Distributed Firing Times”, in: G. Rozenberg (Ed.), *Advances in Petri Nets 1986, Lecture Notes in Computer Science*, vol. 266, Springer, 1987, pp 132-145.
2. H. Choi, V.G. Kulkarni, and K.S. Trivedi, “Markov Regenerative Stochastic Petri Nets”, *Performance Evaluation*, vol. 20, 1994, pp 336-353.
3. G. Ciardo and G. Li, “Approximate transient analysis for subclasses of deterministic and stochastic Petri nets”, *Performance Evaluation*, vol. 35, 1999, pp 109-129.
4. R. German, “Cascaded Deterministic and Stochastic Petri Nets”*, Proc. 3rd Int. Workshop on the Numerical Solution of Markov Chains, Zaragoza, Spain*, 1999, pp 111-130.
5. R. German and A. Heindl, “A Fourth Order Algorithm with Automatic Stepsize Control for the Transient Analysis of DSPNs”, *IEEE Trans. Softw. Engin.*, vol.. 25, 1999, pp 194-206.
6. P.W. Glynn, “A GSMP Formalism for Discrete-Event Systems”, *Proc. of the IEEE,* vol. 77, 1989, pp. 14-23.
7. W.K. Grassmann, “Transient solutions in Markovian queueing systems”, *Comput. Oper. Res.*, vol. 4, 1977, pp 47-53.
8. D. Gross and D.R. Miller, “The Randomization Technique as a Modeling Tool and Solution Procedure for Transient Markov Processes”, *Operations Research*, vol. 32, 1984, pp 345-361.
9. C. Lindemann, “Exploiting Isomorphisms and Special Structures in the Analysis of Markov Regenerative Stochastic Petri Nets”, in: W.J. Stewart (Ed.), *Computations with Markov Chains*, Kluwer, 1995, pp 383-402.
10. C. Lindemann and G.S. Shedler, “Numerical Analysis of Deterministic and Stochastic Petri Nets with Concurrent Deterministic Transitions”, *Performance Evaluation*, vol. 27&28, 1996, pp 565-582.
11. C. Lindemann, *Performance Modelling with Deterministic and Stochastic Petri Nets*, John Wiley & Sons, 1998.
12. C. Lindemann and A. Thümmler, “Transient Analysis of Deterministic and Stochastic Petri Nets with Concurrent Deterministic Transitions”, *Performance Evaluation*, vol. 36&37, 1999, pp 35-54.
13. C. Lindemann, A. Thümmler, A. Klemm, M. Lohmann, and O. Waldhorst, “Performance Analysis of Time-enhanced UML Diagrams Based on Stochastic Processes”, *Proc. 3rd Int. Workshop on Software and Performance (WOSP), Rome, Italy*, 2002, pp 25-34.
14. C. Lindemann and A. Thümmler, “Numerical Analysis of Generalized Semi-Markov Processes”, *Technical Report*, University of Dortmund, 2005.
15. E. de Souza e Silva and H.R. Gail, “Calculating Availability and Performability Measures of Repairable Computer Systems using Randomization”, *Journal of the ACM*, vol. 36, 1989, pp 171-193.
16. E. de Souza e Silva, H.R. Gail, and R.R. Muntz, “Efficient Solution for a Class of Non-Markovian Models”, in: W.J. Stewart (Ed.), *Computations with Markov Chains*, Kluwer, 1995, pp 483-506.
17. M. Telek and A. Horváth, “Time Domain Analysis of Non-Markovian Stochastic Petri Nets with PRI Transitions”, *IEEE Trans. Softw. Engin.*, vol. 28, 2002, pp 933-943.

BIOGRAPHIES

Christoph Lindemann, Professor, Dr.

e-mail: cl@rvs.informatik.uni-leipzig.de

Christoph Lindemann is chaired professor for computer networks and distributed systems in the Department of Computer Science at the University of Leipzig, Germany. His current research interests lie in mobile applications and protocol support for wireless multihop networks as well as in stochastic modeling. In 2005, he served as General Co-Chair for the 11th International Conference on Mobile Computing and Networking, ACM MobiCom. He was general chair 26th International Symposium on Computer Performance, Modeling, Measurements, and Evaluation, Performance 2007 and program committee co-chair of the 10th International Symposium on Mobile Ad Hoc Networking and Computing, ACM MobiHoc 2010.