Markov Regenerative SPN with Non-Overlapping Activity Cycles

Andrea Bobbio

Dipartimento di Elettronica
Università di Brescia
25123 Brescia, Italy
email: bobbio@icil64.cilea.it

Miklós Telek

Department of Telecommunications
Technical University of Budapest
Budapest, Hungary
email: telek@plan.hit.bme.hu

Abstract

The paper discusses a class of Markov Regenerative Stochastic Petri Nets (MRSPN) *characterized by the fact that the stochastic process subordinated to two consecutive regeneration time points is a semi-Markov reward process. This class of* SPN*'s can accommodate transitions with generally distributed firing time and associated memory policy of both enabling and age type, thus generalizing and encompassing all the previous definitions of* MRSPN. *An unified analytical procedure is developed for the derivation of closed form expressions for the transient and steady state probabilities.*

Key words: *Stochastic Petri Nets, semi-Markov Reward Models, Markov regenerative processes.*

1 Introduction

In the usual definition of Stochastic Petri Nets *(SPN)* all the timed transitions have associated an exponential random variable, so that their modeling power is confined to Markovian systems. The analysis of stochastic systems with non-exponential timing is of increasing interest in the literature and requires the development of suitable modeling tools. Recently, some effort has been devoted to generalize the concept of *SPN*, by allowing the firing times to be generally distributed.

An extensive discussion of the semantics of *SPN*'s with generally distributed firing times is in [1], where it is shown that each non-exponential transition should be assigned a *memory policy* chosen among three proposed alternatives: *resampling*, *enabling* and *age memory*. We refer to this model as *Generally Distributed Transition_SPN (GDT_SPN)*. In general, the stochastic process underlying a *GDT_SPN* is too complex to be analytically tractable, while a simulative solution has been investigated in [16].

With the aim of providing a *modeler's representation* able to automatically generate an *analytical representation* [17], various restrictions of the general *GDT_SPN* model have been discussed in the literature [5]. A classification of *SPN* models, based on the nature of the associated marking process, has been proposed by Ciardo et al. [9].

A particular case of non-Markovian *SPN*, is the class of *Deterministic and SPN (DSPN)* defined in [3]. A *DSPN* is a non-Markovian *SPN*, where all the transitions are exponential, but in each marking, at most one transition is allowed to have associated a deterministic firing time with enabling memory policy. Only the steady state analysis was elaborated in [3]. An improved steady state algorithm was presented in [20], and some structural extensions were investigated in [10]. Choi et al. [7] have recognized that the marking process underlying a *DSPN* is a Markov Regenerative Process [11] for which a closed form transient solution is available. This observation has opened a very fertile line of research aimed at the definition of solvable classes of models whose underlying marking process is a *Markov Regenerative Process (MRP)*, and therefore referred to as *Markov Regenerative Stochastic Petri Nets (MRSPN)*.

Following this line, Choi et al. [8] have investigated a class of models in which one transition with a generally distributed firing time and enabling memory policy is allowed to be enabled in each marking. German and Lindemann [15] have proposed a numerical solution of the same model based on the method of supplementary variables [12].

In the mentioned references, the generally distributed (or deterministic) transitions must be assigned a firing policy of enabling memory type [1]. The enabling memory policy means [1] that whenever the transition becomes enabled anew, its firing distribu-

[1] The enabling memory assumption is relaxed in [10] for vanishing markings only. Since vanishing markings are transversed in zero time, this assumption does not modify the behavior of the marking process versus time

tion is resampled and the time eventually spent without firing in prior enabling periods is lost. In the language of queueing systems the above mechanism is referred to as *preemptive repeat different (prd)* policy [14, 19].

The possibility of incorporating non-exponential transitions with associated age memory policy has been first explored in [6]. The age memory is able to capture preemptive mechanisms of resume *(prs)* type, where an interrupted activity is recovered by keeping memory of the work already performed, and upon restart, only the residual service needs to be completed. This modeling extension is crucial in connection with fault tolerant and dependable computing systems, where an interrupted task must be resumed from the point it was interrupted.

The paper investigates the nature of *GDT_SPN* with combined memory policies such that the underlying marking process is a *MRP*. The timed transitions of the *GDT_SPN* are partitioned into two subsets: the EXP transitions have an exponentially distributed firing time, while for the GEN transitions the firing time is any random variable (including the deterministic). The *activity cycle* of a GEN transition is the interval of time in which the transition has a non-null memory. We study the case of *MRSPN* with non overlapping activity cycles, such that the marking process subordinated to two consecutive regeneration time points is a semi-Markov reward process. The proposed model generalizes and encompasses all the previous formulations of *MRSPN*.

In Section 2, the conditions under which the marking process underlying a *GDT_SPN* is a Markov Regenerative process are set in very general terms. In Section 3, the influence of the memory policy on the activity cycle of a transition is discussed. In Section 4, the subordinated process in a *MRSPN* with non-overlapping activity cycles is characterized, and a unified analytical solution for the transient and steady state transition probability matrix is proposed in Section 5.

2 Markov Regenerative Stochastic Petri Nets

A marked Petri Net is a tuple $PN = (P, T, I, O, H, M)$, where: $P = \{p_1, p_2, \ldots, p_{np}\}$ is the set of places, $T = \{t_1, t_2, \ldots, t_{nt}\}$ is the set of transitions and I, O and H are the input, the output and the inhibitor functions, respectively. $M = \{m_1, m_2, \ldots, m_{np}\}$ is the marking. The generic entry m_i is the number of tokens in place p_i, in marking M.

Input and output arcs have an arrowhead on their destination, inhibitor arcs have a small circle. A transition is enabled in a marking if each of its ordinary input places contains at least as many tokens as the multiplicity of the input function I and each of its inhibitor input places contains fewer tokens than the multiplicity of the inhibitor function H. An enabled transition fires by removing as many tokens as the multiplicity of the input function I from each ordinary input place, and adding as many tokens as the multiplicity of the output function O to each output place. The number of tokens in an inhibitor input place is not affected.

A marking M' is said to be *immediately reachable* from M, when is generated from M by firing an enabled transition. The reachability set $\mathcal{R}(M_0)$ is the set of all the markings that can be generated from an initial marking M_0 by repeated application of the above rules. If the set T comprises both timed and immediate transitions, $\mathcal{R}(M_0)$ is partitioned into tangible (no immediate transitions are enabled) and vanishing markings. Since the effect of vanishing markings can be incorporated into the tangible ones, according to [2], we do not account in this paper for the presence of immediate transitions. Let $\mathcal{N}$ be the cardinality of the tangible subset of $\mathcal{R}(M_0)$.

Definition 1 - *A stochastic GDT_SPN is a marked SPN in which [1]:*

- *To any timed transition $t_k \in T$ is associated a random variable γ_k, with cumulative distribution function $G_k(x)$, modeling the time needed by the activity represented by t_k to complete, when considered in isolation.*
- *Each timed transition t_k is attached a memory variable a_k and a memory policy; the memory policy specifies the functional dependence of the memory variable on the past enabling time of the transition.*
- *A initial probability is given on $\mathcal{R}(M_0)$.*

The memory variable a_k, associated to transition t_k, is a functional that depends on the time during which t_k has been enabled. The memory variables together with their memory policy univocally specify how the underlying stochastic process is conditioned upon its past history. The semantics of different memory policies has been discussed in [1] where three alternatives have been proposed and examined.

- *Resampling policy* - The memory variable a_k is reset to zero at any change of marking.
- *Enabling memory policy* - The memory variable a_k accounts for the elapsed time since the last

epoch in which t_k has been enabled. When transition t_k is disabled (even without firing) the corresponding enabling memory variable is reset.

- *Age memory policy* - The memory variable a_k accounts for the elapsed time since the last epoch in which t_k has been enabled without firing. The memory variable is reset only when t_k fires (and not when it is simply disabled).

At the entrance in a new tangible marking, the residual firing time is computed for each enabled timed transition given its memory variable, so that the next marking is determined by the minimal residual firing time among the enabled transitions (*race policy* [1]). Because of the memoryless property, the value of the memory variable is irrelevant in determining the residual firing time for exponential transitions, so that the three mentioned policies are completely equivalent in this case. Hence, for an exponential transition t_k, we assume, conventionally, that the corresponding memory variable is always identically zero. We can therefore partition the set of the transitions into EXP transitions with associated an exponential r.v. and identically zero memory variable, and GEN transition with associated any r.v. (including the deterministic case) and memory variable increasing in the enabling markings.

Definition 2 - *The stochastic process underlying a GDT_SPN is called the marking process $\mathcal{M}(x)$ ($x \geq 0$). $\mathcal{M}(x)$ is the marking of the GDT_SPN at time x.*

A single realization of the marking process $\mathcal{M}(x)$ can be written as:

$$\mathcal{R} = \{(\tau_0, M_0); (\tau_1, M_1); \ldots; (\tau_i, M_i); \ldots\}$$

where M_{i+1} is a marking immediately reachable from M_i, and $\tau_{i+1} - \tau_i$ is the sojourn time in marking M_i. With the above notation, $\mathcal{M}(x) = M_i$ for $\tau_i \leq x < \tau_{i+1}$.

Assertion 1 - *If at time τ_i^+ of entrance in a tangible marking M_i all the memory variables a_k ($k = 1, 2, \ldots, n_t$) are equal to zero, τ_i is a regeneration time point for the marking process $\mathcal{M}(x)$.*

In fact, if all the memory variables are equal to 0, the future of the marking process is not conditioned upon the past and depends only on the present state; hence, the Markov property holds.

Let us denote by τ_n^* the sequence of the regeneration time points embedded into a realization $\mathcal{R}$. The tangible marking $M_{(n)}$ entered at a regeneration time point τ_n^* is called a regeneration marking. The sequence $(\tau_n^*, M_{(n)})$ is a Markov renewal sequence and the marking process $\mathcal{M}(x)$ is a Markov regenerative process [11, 8, 9]. From Assertion 1 follows that:

i) if all the transitions are EXP all the memory variables are identically zero so that any instant of time is a regeneration time point, and the corresponding process is a *CTMC*;

ii) if at any firing all the memory variables of the GEN transitions are reset, the corresponding process reduces to a semi-Markov process.

iii) only GEN transitions are relevant to determine the occurrence of regeneration time points.

Definition 3 - *A GDT_SPN, for which an embedded Markov renewal sequence $(\tau_n^*, M_{(n)})$ exists, is called a Markov Regenerative Stochastic Petri Net (MRSPN)* [8].

Since $(\tau_n^*, M_{(n)})$ is a Markov renewal sequence, the following equalities hold:

$$\begin{aligned} & Pr\{M_{(n+1)} = j, (\tau_{n+1}^* - \tau_n^*) \leq x \mid \\ & \quad M_{(n)} = i, \tau_n^*, M_{(n-1)}, \tau_{n-1}^*, \ldots, M_{(0)}, \tau_0^*\} = \\ & Pr\{M_{(n+1)} = j, (\tau_{n+1}^* - \tau_n^*) \leq x \mid M_{(n)} = i, \tau_n^*\} = \\ & Pr\{M_{(1)} = j, \tau_1^* \leq x \mid M_{(0)} = i\} \end{aligned} \quad (1)$$

The first equality expresses the Markov property (i.e. in any regeneration time point the condition on the past is condensed in the present state). The second equality expresses the time homogeneity (i.e. the probability measures are independent of a translation along the time axis). According to [8, 11], we define the following matrix valued functions $\mathbf{V}(x) = [V_{ij}(x)]$, $\mathbf{K}(x) = [K_{ij}(x)]$ and $\mathbf{E}(x) = [E_{ij}(x)]$ (all of dimension $\mathcal{N} \times \mathcal{N}$), such that:

$$\begin{aligned} & V_{ij}(x) = Pr\{\mathcal{M}(x) = j \mid \mathcal{M}(\tau_0^*) = i\} \\ & K_{ij}(x) = Pr\{M_{(1)} = j, \tau_1^* \leq x \mid \mathcal{M}(\tau_0^*) = i\} \\ & E_{ij}(x) = Pr\{\mathcal{M}(x) = j, \tau_1^* > x \mid \mathcal{M}(\tau_0^*) = i\} \end{aligned} \quad (2)$$

$\mathbf{V}(x)$ is the transition probability matrix and provides the probability that the stochastic process $\mathcal{M}(x)$ is in marking j at time x given it was in i at $x = 0$. The matrix $\mathbf{K}(x)$ is the *global kernel* of the *MRP* and provides the cdf of the event that the next regeneration time point is τ_1^* and the next regeneration marking is $M_{(1)} = j$ given marking i at $\tau_0^* = 0$. Finally, the matrix $\mathbf{E}(x)$ is the *local kernel* since describes the behavior of the marking process $\mathcal{M}(x)$ inside two consecutive regeneration time points. The generic element $E_{ij}(x)$ provides the probability that the process stays in state j at time x starting from i at $\tau_0^* = 0$ before the next regeneration time point. From the above definitions:

$$\sum_j [K_{ij}(x) + E_{ij}(x)] = 1$$

The transient behavior of the *MRSPN* can be evaluated by solving the following generalized Markov renewal equation (in matrix form) [11, 8]:

$$\mathbf{V}(x) = \mathbf{E}(x) + \mathbf{K} * \mathbf{V}(x) \qquad (3)$$

where $\mathbf{K} * \mathbf{V}(x)$ is a convolution matrix, whose (i,j)-th entry is:

$$[\mathbf{K} * \mathbf{V}(x)]_{ij} = \sum_k \int_0^x dK_{ik}(y)\, V_{kj}(x-y) \qquad (4)$$

By denoting the Laplace Stieltjes transform *(LST)* of a function $F(x)$ by $F^{\sim}(s) = \int_0^\infty e^{-sx}\, dF(x)$, Equation (3) becomes in the *LST* domain:

$$\mathbf{V}^{\sim}(s) = \mathbf{E}^{\sim}(s) + \mathbf{K}^{\sim}(s)\, \mathbf{V}^{\sim}(s) \qquad (5)$$

whose solution is:

$$\mathbf{V}^{\sim}(s) = [\mathbf{I} - \mathbf{K}^{\sim}(s)]^{-1}\, \mathbf{E}^{\sim}(s) \qquad (6)$$

If the steady state solution exists, it can be evaluated as $\lim_{s \to 0} \mathbf{V}^{\sim}(s)$.

As specified by (2), $\mathbf{K}(x)$ and $\mathbf{E}(x)$ depend on the evolution of the marking process between two consecutive regeneration time points. By virtue of the time homogeneity property (1), we can always define the two successive regeneration time points to be $x = \tau_0^* = 0$ and $x = \tau_1^*$.

Definition 4 - *The stochastic process subordinated to state i (denoted by $\mathcal{M}^i(x)$) is the restriction of the marking process $\mathcal{M}(x)$ for $x \le \tau_1^*$ given $\mathcal{M}(\tau_0^*) = i$:*

$$\mathcal{M}^i(x) = [\mathcal{M}(x) : x \le \tau_1^*, \mathcal{M}(\tau_0^*) = i]$$

According to Definition 4, $\mathcal{M}^i(x)$ describes the evolution of the *PN* starting at the regeneration time point $x = 0$ in the regeneration marking i, up to the next regeneration time point τ_1^*. Therefore, $\mathcal{M}^i(x)$ includes all the markings that can be reached from state i before the next regeneration time point. The entries of the i-th row of the matrices $\mathbf{K}(x)$ and $\mathbf{E}(x)$ are determined by $\mathcal{M}^i(x)$.

3 Non-Overlapping Activity Cycles

The analytical tractability of the marking process depends on the structure of the subordinated processes which, in turns, is related to the topology of the *PN* and to the memory policies of the GEN transitions.

Definition 5 - *A GEN transition is dormant in those markings in which the corresponding memory variable is equal to zero and is active in those markings in which the memory variable is greater than zero. The* activity cycle *of a GEN transition is the period of time in which a transition is active between two dormant periods.*

Let us consider a single generic GEN transition t_g. The activity cycle of t_g is influenced by its memory policy, and can be characterized in the following way.

Resampling Memory - If t_g is a resampling memory transition, its activity cycle starts as soon as t_g becomes enabled, and ends at the first subsequent firing of any transition (including t_g itself). Therefore, during the activity cycle of a resampling memory transition no change of marking is possible.

Enabling Memory - If t_g is an enabling memory transition its activity cycle starts as soon as t_g becomes enabled when dormant, and ends either when t_g fires, or when it becomes disabled by the firing of a competitive transition. During the activity cycle the marking can change inside the enabling subset of t_g (where the enabling subset is defined as the subset of connected markings in which t_g is enabled). The memory variable associated to t_g grows continuously during the activity cycle starting from 0. We associate a reward variable equal to 1 to all the states in the enabling subset, so that the value of the memory variable is represented by the total accumulated reward.

Age Memory - If t_g is an age memory transition, its activity cycle starts as soon as t_g becomes enabled when dormant, and ends only at the firing of t_g itself. During the activity cycle of an age memory transition there is no restriction on the markings reachable by the marking process. The age memory policy is the only policy in which a transition can be active even in markings in which it is not enabled. During the activity cycle, the memory variable is non-decreasing in the sense that it increases continuously in those markings in which t_g is enabled and maintains its constant positive value in those markings in which t_g is not enabled. In order to track the enabling/disabling condition of t_g during its activity cycle, we introduce a reward (indicator) variable which is equal to 1 in those markings in which t_g is enabled and equal to 0 in those mark-

Table I - Characterization of the activity cycle of a GEN transition t_g

Memory policy	*Resamp.*	*Enabling*	*Age*
start of activity cycle	t_g enabled	t_g enabled when dormant	t_g enabled when dormant
end of activity cycle	firing of any transition	firing or disabling of t_g	firing of t_g
reachable markings	starting marking only	markings in enabling subset	any reachable marking
memory variable	increasing	increasing	increasing or constant

ings in which t_g is not enabled. The memory variable corresponds to the total accumulated reward.

The above features are summarized in Table 1. By virtue of Assertion 1, a regeneration time point for the marking process occurs when a firing causes all the active GEN transitions to become dormant.

Definition 6 - *A transition is dominant if its activity cycle strictly contains the activity cycles of all the active transitions.*

Definition 7 - *A MRSPN with non-overlapping activity cycles is a MRSPN in which all the regeneration periods are dominated by a single transition: any two successive regeneration time points correspond to the start and to the end of the active cycle of the dominant transition.*

Definition 7, includes the possibility that the active cycles of GEN transitions are completely contained into the active cycle of the dominant one, hence allowing the simultaneous enabling of different GEN transitions inside the same subordinated process. In order to make the whole process analytically solvable, we further restrict the subordinated process inside any non-overlapping activity cycle to be semi-Markov.

Assertion 2 - *The subordinated process underlying any non-overlapping activity cycle is semi-Markov if at any firing inside the activity cycle of the dominant transition all the memory variables of the GEN transition are reset. This fact happens if the transitions can be partitioned into three classes (exclusive, competitive and concurrent) and only exclusive or competitive transitions are allowed to be GEN [13].*

For a regeneration period without internal state transitions (Markovian or semi-Markovian regeneration period) any of the enabled transitions can be chosen to be the dominant one.

4 The Subordinated Process

At $x = \tau_0^* = 0$ a dominant GEN transition t_g (with memory variable a_g and firing time γ_g) starts its activity cycle in state i ($a_g = 0$). The successive regeneration time point τ_1^* is the end of the activity cycle of t_g according to the rules summarized in Table I.

Let $Z^i(x)$ ($x \geq 0$) be the process defined over the states reachable from i during the activity cycle of t_g, and $\underline{r}^i$ the corresponding binary reward vector. We assume in the following that $Z^i(x)$ is a semi-Markov process according to Assertion 2. The subordinated process $\mathcal{M}^i(x)$ (Definition 4) coincides with $Z^i(x)$ when the initial state is state i with probability 1 ($Pr\{Z^i(0) = i\} = 1$). The memory variable a_g increases at a rate r_j^i (which is either equal to 0 or to 1) when $\mathcal{M}^i(x) = j$.

We consider separately the following cases depending whether the dominant transition t_g is of enabling or age memory type.

4.1 Enabling type dominant transition

The dominant GEN transition t_g is of enabling type. The state space of the subordinated process is partitioned into two subsets: R^i contains the states in which t_g is continuously enabled, and R^{ci} contains the states in which t_g becomes disabled by the firing of a competitive transition. The reward vector is equal to 1 for $j \in R^i$ and 0 elsewhere. The next regeneration time point occurs because one of the following two mutually exclusive events:

- t_g fires: this event can be formulated as a completion time problem [4] when the accumulated reward (memory variable) a_g reaches an absorbing barrier equal to the firing requirement γ_g.

- t_g is disabled: this event can be formulated as a

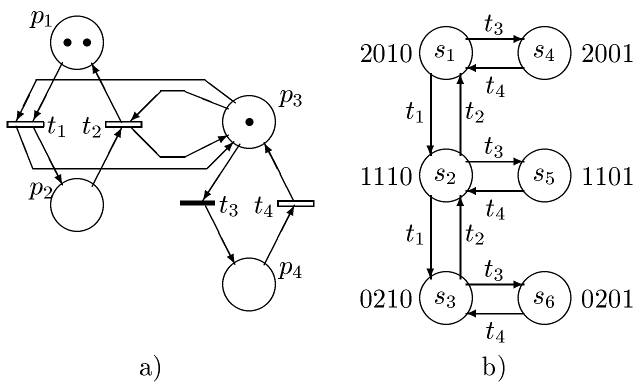

Figure 1 - a) PN of the periodically self tested M/M/1/k; b) corresponding reachability graph.

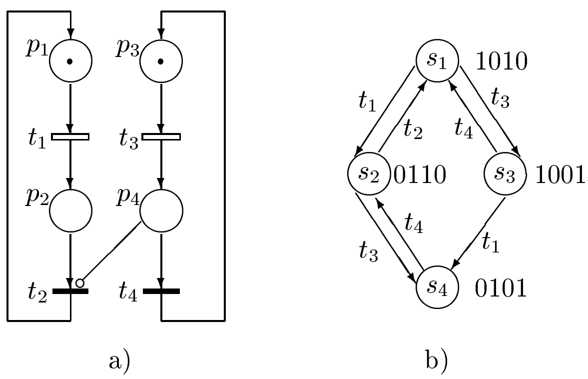

Figure 2 - Preemptive M/G/1/2/2 queue with two classes of customers.

first passage time in the subset R^{ci}, and therefore R^{ci} is made absorbing in the subordinated process.

We further particularize the following two cases:

CASE A - no other GEN transitions are activated during the activity cycle of t_g. The subordinated process $Z^i(x)$ is a CTMC.

Case A is the one considered in the *DSPN* model defined in [3, 7, 20], and in the successive extensions to general distributions elaborated in [8, 15]. All the examples reported in the mentioned papers belong to this case.

CASE B - during the activity cycles of t_g, Assertion 2 is satisfied and the subordinated process is a semi-Markov process.

The Markovian (semi-Markovian) regeneration period belongs to Case A (Case B), where R^i contains only the initial state. The steady state analysis of a *MRSPN* with semi-Markovian subordinated process has been considered in [9].

Example 1 - *A periodically self-tested system.*
A system is executing tasks according to a M/M/1/k queue (Figure 1a). Place p_1 represents user thinking and p_2 is the queue including the task under service. t_1 is the exponential submitting time with marking dependent rate $m_1\,\lambda$, and t_2 is the exponential service time with rate μ. p_3 represents the system waiting for the test and p_4 the system under test. t_3 is the deterministic testing interval, and t_4 the exponentially distributed test duration with rate δ. When t_3 fires the execution of the M/M/1/k queue is frozen until the test is completed (t_4 fires). The state space of the *PN* of Figure 1a with $k = 2$ customers is in Figure 1b. All the states can be regeneration states, but not all the transitions provide regeneration time points. States s_4, s_5 and s_6 are always regeneration states from which a single EXP transition is enabled (Case A). States s_1 or s_2 or s_3 are regeneration states only when entered by firing t_4, i.e. when the activity cycle of the dominant GEN transition t_3 starts. During the activity cycle of t_3, the subordinated process can move among s_1, s_2 and s_3 which therefore form the subordinated *CTMC* (Case A).

If transitions t_1 and t_2 are GEN with enabling memory policy, the features of states s_4, s_5 and s_6 do not change, while the subordinated process during the activity cycle of t_3 becomes semi-Markovian thus representing a Case B example.

4.2 Age type dominant transition

The situation in which the dominant GEN transition t_g is of age type has been addressed for the first time in [6]. The state space of the subordinated process R^i contains all the states reachable during the activity cycle of t_g, and the disabling subset R^c is empty (the only criterion for the termination of the activity cycle is the firing of t_g). The reward vector is equal to 1 for the states $j \in R^i$ in which t_g is enabled and 0 for the states $j \in R^i$ in which t_g is not enabled. The firing of t_g can be formulated as a completion time problem [4] when the accumulated reward (memory variable) a_g reaches the firing requirement γ_g. We further particularize the following two cases:

CASE C - During the activity cycle of t_g no other GEN transitions are activated and the subordinated process is a reward CTMC.

CASE D - during the activity cycles of t_g, Assertion 2 is satisfied and the subordinated process is a Reward semi-Markov process.

Example 2 - *Preemptive M/G/1/2/2 with different customers*

In this example, Cases C and D are mixed in a single *PN* [5]. The *PN* of Figure 2a models a M/G/1/2/2 queue in which the jobs submitted by customer 2 have higher priority and preempt the jobs submitted by customer 1. The server has a *prs* service discipline. Place p_1 (p_3) represents customer 1 (2) thinking, while place p_2 (p_4) represent job 1 (2) under service. Transitions t_1 and t_3 are EXP and represent the submission of a job of type 1 or 2, respectively. t_2 and t_4 are GEN transitions, and represent the completion of service of a job of type 1 or 2, respectively. A *prs* service discipline is modeled by assigning to t_2 and t_4 an age memory policy. The inhibitor arc from p_4 to t_2 models the described preemption mechanism: as soon as a type 2 job joins the queue the type 1 job eventually under service is interrupted. The reachability graph of the *PN* of Figure 2a is in Figure 2b. Under a *prs* service, after completion of the type 2 job, the interrupted type 1 job is resumed continuing the new service period from the point reached just before the last interruption. From Figure 2b, it is easily recognized that s_1, s_2 and s_3 can all be regeneration states, while s_4 can never be a regeneration state (in s_4 a type 2 job is always in execution so that its corresponding memory variable a_2 is never 0). Only exponential transitions are enabled in s_1 and the next regeneration states can be either s_2 or s_3 depending whether t_1 or t_3 fires first. From state s_3 the next regeneration marking can be either state s_1 or s_2 depending whether during the execution of the type 2 job a type 1 job does require service (but remains blocked until completion of the type 2 job) or does not. The subordinated process is a *CTMC*, and belongs to Case C. From s_2 the next regeneration state can be only s_1, but multiple cycles (s_2 - s_4) can occur depending whether type 2 jobs arrive to interrupt the execution of the type 1 job. The subordinated process is a *SMP* (t_4 is GEN), and belongs to case D.

5 Unified Transient Analysis

The global and local kernels $\mathbf{K}(x)$ and $\mathbf{E}(x)$ can be evaluated row by row. In this section, we provide an unified analytical procedure for determining in closed form the entries of a generic row i, given that i is a regeneration marking whose subordinated process is a semi-Markov reward process as described in the previous section.

Let $\mathbf{Q}^i(x) = [Q^i_{k\ell}(x)]$ be the kernel of the subordinated semi-Markov process ($Z^i(x)$). $Z^i(x)$ starts in marking M_i ($Z^i(0) = i$), so that the initial probability vector is $\underline{V}^i_0 = [0, 0, \ldots, 1_i, \ldots, 0]$ (a vector with all the entries equal to 0 but entry i equals to 1). For notational convenience we do not renumber the states in $Z^i(x)$ so that all the subsequent matrix functions have the dimensions $(\mathcal{N} \times \mathcal{N})$ (cardinality of $\mathcal{R}(M_0)$), but with the significant entries located in position (k, ℓ) only, with $k, \ell \in R^i \cup R^{ci}$. We denote by H the time duration until the first embedded time point in $Z^i(x)$ from time $x = 0$.

Let us fix the value of the firing requirement $\gamma_g = w$, and let us define the following matrix functions $\mathbf{P}^i(x,w)$, $\mathbf{F}^i(x,w)$, $\mathbf{D}^i(x,w)$ and $\mathbf{\Delta}^i$:

$$
\begin{aligned}
P^i_{k\ell}(x,w) &= Pr\{Z^i(x) = \ell \in R^i\,,\, \tau^*_1 > x \mid \\
&\qquad Z^i(0) = k \in R^i\,,\, \gamma_g = w\} \\
F^i_{k\ell}(x,w) &= Pr\{Z^i(\tau^{*\,-}_1) = \ell \in R^i\,,\, \tau^*_1 \le x,\, t_g \text{ fires} \mid \\
&\qquad Z^i(0) = k \in R^i\,,\, \gamma_g = w\} \\
D^i_{k\ell}(x,w) &= Pr\{Z^i(\tau^*_1) = \ell \in R^{c\,i}\,,\, \tau^*_1 \le x \mid \\
&\qquad Z^i(0) = k \in R^i\,,\, \gamma_g = w\} \\
\Delta^i_{k\ell} &= Pr\{\text{ next tangible marking is } \ell \mid \\
&\qquad \text{current marking is } k,\ t_g \text{ fires}\}
\end{aligned}
\tag{7}
$$

By the above definitions, the entries $P^i_{k\ell}(x,w)$ and $F^i_{k\ell}(x,w)$ are significant only for $k, \ell \in R^i$ and are 0 otherwise; the entries $D^i_{k\ell}(x,w)$ are significant for $k \in R^i$ and $\ell \in R^{ci}$, and are 0 otherwise.

- $P^i_{k\ell}(x,w)$ is the probability of being in state $\ell \in R^i$ at time x before absorption either at the barrier w or in the absorbing subset $R^{c\,i}$, starting in state $k \in R^i$ at $x = 0$.
- $F^i_{k\ell}(x,w)$ is the probability that t_g fires from state $\ell \in R^i$ (hitting the absorbing barrier w in ℓ) before x, starting in state $k \in R^i$ at $x = 0$.
- $D^i_{k\ell}(x,w)$ is the probability of first passage from a state $k \in R^i$ to a state $\ell \in R^{c\,i}$ before hitting the barrier w, starting in state $k \in R^i$ at $x = 0$.
- $\mathbf{\Delta}^i$ is the branching probability matrix and represents the successor tangible marking ℓ that is reached by firing t_g in state $k \in R^i$ (the firing of t_g in the subordinated process $\mathcal{M}^i(x)$, can only occur in a state k in which $r^i_k = 1$).

From (7), it follows for any x:

$$\sum_{\ell \in R^i \cup R^{ci}} [\, P^i_{k\ell}(x,w) \,+\, F^i_{k\ell}(x,w) \,+\, D^i_{k\ell}(x,w)\,] \,=\, 1$$

Given that $G_g(w)$ is the cumulative distribution function of the r.v. γ_g associated to the transition t_g, the elements of the i-th row of matrices $\mathbf{K}(x)$ and $\mathbf{E}(x)$ can be expressed as follows, as a function of the matrices $\mathbf{P}^i(x,w)$, $\mathbf{F}^i(x,w)$ and $\mathbf{D}^i(x,w)$:

$$\begin{aligned} K_{ij}(x) &= \int_{w=0}^{\infty} [\sum_{k \in R^i} F^i_{ik}(x,w)\, \Delta^i_{kj} + \\ &\qquad D^i_{ij}(x,w)]\, dG_g(w) \\ E_{ij}(x) &= \int_{w=0}^{\infty} P^i_{ij}(x,w)\, dG_g(w) \end{aligned} \tag{8}$$

In order to avoid unnecessarily cumbersome notation in the following derivation, we neglect the explicit dependence on the particular subordinated process $Z^i(x)$, by eliminating the superscript i. It is however tacitly intended, that all the quantities $\underline{r}$, $\mathbf{Q}(x)$, $\mathbf{P}(x,w)$, $\mathbf{F}(x,w)$, $\mathbf{D}(x,w)$, $\mathbf{\Delta}$, R and R^c refer to the specific process subordinated to the regeneration period starting from state i.

5.1 Derivation of $\mathbf{P}(x,w)$, $\mathbf{F}(x,w)$ and $\mathbf{D}(x,w)$

The derivation of these matrix functions is described in more detail in [21, 6] and follows the same pattern of the completion time analysis presented in [19, 4].

Theorem 1 - *For the firing probability $F_{k\ell}(x,w)$ the following double transform equation holds:*

$$\begin{aligned} F^{\sim *}_{k\ell}(s,v) &= \delta_{k\ell}\, \frac{r_k\,[1 - Q^{\sim}_k(s + v\, r_k)]}{s + v\, r_k} + \\ &\sum_{u \in R} Q^{\sim}_{ku}(s + v\, r_k)\, F^{\sim *}_{u\ell}(s,v) \end{aligned} \tag{9}$$

Proof - Conditioning on $H = h$ and $\gamma_g = w$, let us define:

$$F_{k\ell}(x,w \,|\, H = h) = \begin{cases} \delta_{k\ell}\, U\left(x - \dfrac{w}{r_k}\right) & \text{if} : h\, r_k \geq w \\ \displaystyle\sum_{u \in R} \frac{dQ_{ku}(h)}{dQ_k(h)} \cdot F_{u\ell}(x - h, w - hr_k) & \text{if} : h\, r_k < w \end{cases} \tag{10}$$

In (10), two mutually exclusive events are identified. If $r_k \neq 0$ and $h\, r_k \geq w$, a sojourn time equals to w is accumulated before leaving state k, so that the firing time (next regeneration time point) is $\tau^*_1 = w/r_k$. If $h\, r_k < w$ then a transition occurs to state u with probability $dQ_{ku}(h)/dQ_k(h)$ and the residual service $(w - hr_k)$ should be accomplished starting from state u at time $(x - h)$. Taking the LST transform with respect to x (denoting the transform variable by s), the LT transform with respect to w (denoting the transform variable by v) of (10) and unconditioning with respect to H, (10) becomes (9). □

Theorem 2 - *The state probability $P_{k\ell}(x,w)$ satisfies the following double transform equation:*

$$\begin{aligned} P^{\sim *}_{k\ell}(s,v) &= \delta_{k\ell}\, \frac{s\,[1 - Q^{\sim}_k(s + v\, r_k)]}{v(s + v\, r_k)} + \\ &\sum_{u \in R} Q^{\sim}_{ku}(s + v\, r_k)\, P^{\sim *}_{u\ell}(s,v) \end{aligned} \tag{11}$$

Proof - Conditioning on $H = h$, and $\gamma_g = w$ let us define:

$$P_{k\ell}(x,w \,|\, H = h) = \begin{cases} \delta_{k\ell}\left[U(x) - U\left(x - \dfrac{w}{r_k}\right)\right] & \text{if} : h\, r_k \geq w \\ \delta_{k\ell}\,[U(x) - U(x-h)] + \displaystyle\sum_{u \in R} \frac{dQ_{ku}(h)}{dQ_k(h)}\, P_{u\ell}(x - h, w - hr_k) & \text{if} : h\, r_k < w \end{cases} \tag{12}$$

The derivation of the matrix function $\mathbf{P}(x,w)$ based on (12) follows the same pattern as for the function $\mathbf{F}(x,w)$ [21]. □

Theorem 3 - *The probability $D_{k\ell}(x,w)$ of first passage into R^c satisfies the following double transform equation:*

$$\begin{aligned} D^{\sim *}_{k\ell}(s,v) &= \frac{1}{v} Q^{\sim}_{kl}(s + v\, r_k) + \\ &\sum_{u \in R} Q^{\sim}_{ku}(s + v\, r_k)\, D^{\sim *}_{u\ell}(s,v) \end{aligned} \tag{13}$$

Proof - Conditioning on $H = h$, and $\gamma_g = w$ let us define:

$$D_{k\ell}(x,w \,|\, H=h) \; = $$

$$\begin{cases} 0 & \text{if}: h\, r_k \;\geq\; w \\ \dfrac{dQ_{k\ell}(h)}{dQ_k(h)}\, U(x-h) + \\ \quad \displaystyle\sum_{u \in R} \frac{dQ_{ku}(h)}{dQ_k(h)}\, D_{u\ell}(x-h, w-hr_k) \\ & \text{if}: h\, r_k \;<\; w \end{cases} \tag{14}$$

The derivation of the matrix function $\mathbf{D}(x,w)$ based on (14) follows the same pattern as for the function $\mathbf{F}(x,w)$ [21]. □

5.2 The subordinated process is a Reward CTMC

Let us consider the particular case in which the subordinated process $Z(x)$ is a reward *CTMC* with infinitesimal generator $\mathbf{A} \;=\; \{a_{k\ell}\}$. Let us suppose that the states numbered $1, 2, \ldots, m$ belong to R $(1, 2, \ldots, m \in R)$ and the states numbered $m+1, m+2, \ldots, n$ belong to R^c $(m+1, m+2, \ldots, n \in R^c)$. By this ordering of states $\mathbf{A}$ can be partitioned into the following submatrices $\mathbf{A} = \begin{array}{|c|c|} \hline \mathbf{B} & \mathbf{C} \\ \hline \mathbf{U_1} & \mathbf{U_2} \\ \hline \end{array}$ where $\mathbf{B}$ contains the intensity of the transitions inside R, and $\mathbf{C}$ contains the intensity of the transitions from R to R^c. $\mathbf{U_1}$ and $\mathbf{U_2}$ refer to the portion of the state space not involved in the current subordinated marking process, and are, thus, not influential for the problem at hand. For this reason, their entries can be assumed equal to zero.

Corollary 4 - *The entries of the matrix functions $P_{k\ell}(x,w)$, $F_{k\ell}(x,w)$ and $D_{k\ell}(x,w)$, in double transform domain, take the following expression:*

$$\begin{aligned} (s + v r_k)\, F^{\sim *}_{k\ell}(s,v) \;&=\; \delta_{k\ell}\;\; r_k \;\; + \sum_{u \in R} a_{ku}\, F^{\sim *}_{u\ell}(s,v) \\ (s + v r_k)\, P^{\sim *}_{k\ell}(s,v) \;&=\; \delta_{k\ell}\;\; \frac{s}{v} \;\; + \sum_{u \in R} a_{ku}\, P^{\sim *}_{u\ell}(s,v) \\ (s + v r_k)\, D^{\sim *}_{k\ell}(s,v) \;&=\; \frac{a_{k\ell}}{v} \;\; + \sum_{u \in R} a_{ku}\, D^{\sim *}_{u\ell}(s,v) \end{aligned} \tag{15}$$

Proof - The kernel (transition probability matrix) of the given *CTMC* can be written as:

$$Q_{k\ell}(x) \;=\; \begin{cases} \dfrac{a_{k\ell}}{-a_{kk}}\,(1 - e^{a_{kk}\,x}) & \text{if}: k \;\neq\; \ell \\ 0 & \text{if}: k \;=\; \ell \end{cases} \tag{16}$$

and in *LST* domain:

$$Q^{\sim}_{k\ell}(s) \;=\; \begin{cases} \dfrac{a_{k\ell}}{s \;-\; a_{kk}} & \text{if}: k \;\neq\; \ell \\ 0 & \text{if}: k \;=\; \ell \end{cases} \tag{17}$$

with $a_{kk} = -\sum_{\ell \in R^i \cup R^{ci}\,,\, \ell \neq k} a_{k\ell}$

By substituting (17) into (11), (9) and (13), the corollary is proved.□

Equations (15) can be rewritten in matrix form:

$$\begin{aligned} \mathbf{F}^{\sim *}(s,v) \;&=\; (s\mathbf{I} + v\mathbf{R} - \mathbf{B})^{-1}\mathbf{R} \\ \mathbf{P}^{\sim *}(s,v) \;&=\; \frac{s}{v}\,(s\mathbf{I} + v\mathbf{R} - \mathbf{B})^{-1} \\ \mathbf{D}^{\sim *}(s,v) \;&=\; \frac{1}{v}\,(s\mathbf{I} + v\mathbf{R} - \mathbf{B})^{-1}\mathbf{C} \end{aligned}$$

where $\mathbf{I}$ is the identity matrix and $\mathbf{R}$ is the diagonal matrix of the reward rates (r_k); the dimensions of $\mathbf{I}$, $\mathbf{R}$, $\mathbf{B}$, $\mathbf{F}$ and $\mathbf{P}$ are $(m \times m)$, and the dimensions of $\mathbf{C}$ and $\mathbf{D}$ are $(m \times (n-m))$.

6 Numerical Results

A numerical derivation of the transient state probabilities of the M/D/1/2/2 system described in Example 2 of Section 4.2 is provided. We consider in details the particular case in which the GEN transitions t_2 and t_4 are assumed to be deterministic with duration α, while t_1 and t_3 are EXP with parameter λ [6]. The reachability graph in Figure 2b comprises 4 states. Let us build up the $\mathbf{K}^{\sim}(s)$ and $\mathbf{E}^{\sim}(s)$ matrices row by row, taking into consideration that state s_4 can never be a regeneration marking since a type 2 job with nonzero age memory is always active.

i) - The starting regeneration state is s_1 - No deterministic transitions are enabled: the state is Markovian and the next regeneration state can be either state s_2 or s_3. The nonzero elements of the 1-st row of matrices $\mathbf{K}^{\sim}(s)$ and $\mathbf{E}^{\sim}(s)$ take the form:

$$K^{\sim}_{12}(s) \;=\; \frac{\lambda}{s \;+\; 2\lambda} \qquad ; \qquad K^{\sim}_{13}(s) \;=\; \frac{\lambda}{s \;+\; 2\lambda}$$

$$E^{\sim}_{11}(s) \;=\; \frac{s}{s \;+\; 2\lambda} \qquad ;$$

ii) - The starting regeneration state is s_2 - Transition t_2 is deterministic so that the next regeneration time point is the epoch of firing of t_2. The subordinated process $\mathcal{M}^2(x)$ comprises states s_2 and s_4 and is a semi-Markov process (Case D) since t_4 is deterministic. The kernel of the semi-Markov process is:

$$Q^{\sim}(s) = \begin{vmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \dfrac{\lambda}{s+\lambda} \\ 0 & 0 & 0 & 0 \\ 0 & e^{-\alpha s} & 0 & 0 \end{vmatrix}$$

The reward vector is $\underline{r}^{(2)} = [0, 1, 0, 0]$, and the only nonzero entry of the branching probability matrix is $\Delta_{21}^{(2)} = 1$. Applying Equations (9) and (11) we obtain the following results for the nonzero entries:

$$\begin{aligned} F_{22}^{\sim *}(s,w) &= \frac{1}{s+w+\lambda-\lambda e^{-s\alpha}} \\ P_{22}^{\sim *}(s,w) &= \frac{s/w}{s+w+\lambda-\lambda e^{-s\alpha}} \\ P_{24}^{\sim *}(s,w) &= \frac{\lambda(1-e^{-s\alpha})/w}{s+w+\lambda-\lambda e^{-s\alpha}} \end{aligned}$$

Applying (8), and after inverting the *LT* transform with respect to w, the *LST* matrix functions $\mathbf{K}^{\sim}(s)$ and $\mathbf{E}^{\sim}(s)$ become:

$$\begin{aligned} K_{21}^{\sim}(s) &= e^{-\alpha(s+\lambda-\lambda e^{-\alpha s})} \\ E_{22}^{\sim}(s) &= \frac{s[1-e^{-\alpha(s+\lambda-\lambda e^{-\alpha s})}]}{s+\lambda-\lambda e^{-\alpha s}} \\ E_{24}^{\sim}(s) &= \frac{\lambda(1-e^{-\alpha s})\,[1-e^{-\alpha(s+\lambda-\lambda e^{-\alpha s})}]}{s+\lambda-\lambda e^{-\alpha s}} \end{aligned}$$

iii) - The starting regeneration state is s_3 - The subordinated process $\mathcal{M}^3(x)$ is a *CTMC* (Case C), hence the results of Section 5.2 apply. The infinitesimal generator of the *CTMC* is:

$$\mathbf{A} = \begin{vmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & -\lambda & \lambda \\ 0 & 0 & 0 & 0 \end{vmatrix}$$

and the reward vector is $\underline{r}^{(3)} = [0, 0, 1, 1]$. The branching probabilities arising from the firing of t_4 are $\Delta_{31}^{(3)} = 1$ and $\Delta_{42}^{(3)} = 1$. Applying the first and second equation in (15), the nonzero entries take the form:

$$\begin{aligned} F_{33}^{\sim *}(s,w) &= \frac{1}{s+\lambda+w} \\ F_{34}^{\sim *}(s,w) &= \frac{\lambda}{(s+w)(s+\lambda+w)} \\ P_{33}^{\sim *}(s,w) &= \frac{s}{w(s+\lambda+w)} \\ P_{34}^{\sim *}(s,w) &= \frac{\lambda s}{w(s+w)(s+\lambda+w)} \end{aligned}$$

Inverting the above equations with respect to w, taking into account the branching probabilities, yields:

$$\begin{aligned} K_{31}^{\sim}(s) &= e^{-\alpha(s+\lambda)} \\ K_{32}^{\sim}(s) &= e^{-\alpha s}(1-e^{-\alpha\lambda}) \\ E_{33}^{\sim}(s) &= \frac{s}{s+\lambda}(1-e^{-\alpha(s+\lambda)}) \\ E_{34}^{\sim}(s) &= \frac{\lambda}{s+\lambda} - (1-\frac{s}{s+\lambda}e^{-\alpha\lambda})e^{-\alpha s} \end{aligned}$$

The time domain probabilities are calculated by first deriving matrix $\mathbf{V}^{\sim}(s)$ from (6) using a standard package for symbolic analysis (e.g. MATHEMATICA), and then numerically inverting the resulting *LST* expressions resorting to the Jagerman's method [18]. The plot of the state probabilities versus time for states s_1 and s_4 is reported in Figure 3, for a deterministic service duration $\alpha = 1$ and for two different values of the submitting rate $\lambda = 0.5$ and $\lambda = 2$.

7 Conclusion

The *GDT_SPN* model, whose semantics has been discussed in [1], provides a natural environment for the definition of a class of analytically tractable *MRSPN*'s. The paper has considered the case of *GDT_SPN* with non-overlapping activity cycles, such that the marking process subordinated to the activity cycle of the dominant transition is a reward semi-Markov process. The inclusion of a reward variable in the description of the subordinated process has proven to be very effective technique for extending the descriptive power of the model to age memory policies, and for providing a unified procedure for the analytical solution.

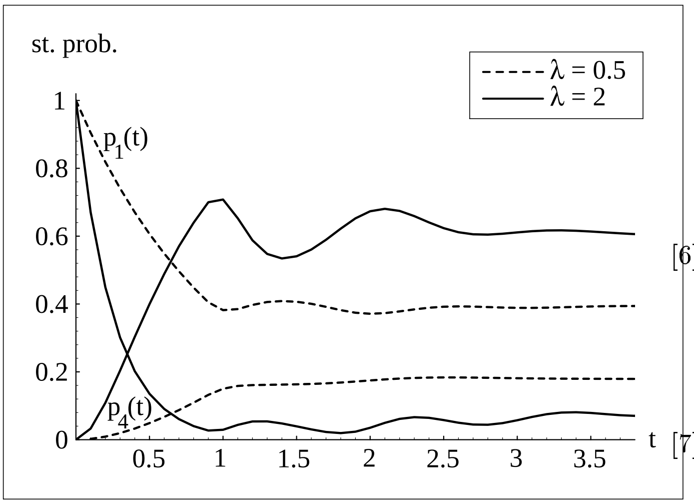

Figure 3 - Transient behavior of the state probabilities for the preemptive M/D/1/2/2 system with different customers.

Acknowledgements

This work has been partially supported by PHARE-ACCORD under grant No. H-9112-0353 and by OTKA under grant No. W-015859.

References

[1] M. Ajmone Marsan, G. Balbo, A. Bobbio, G. Chiola, G. Conte, and A. Cumani. The effect of execution policies on the semantics and analysis of stochastic Petri nets. *IEEE Transactions on Software Engineering*, SE-15:832–846, 1989.

[2] M. Ajmone Marsan, G. Balbo, and G. Conte. A class of generalized stochastic Petri nets for the performance evaluation of multiprocessor systems. *ACM Transactions on Computer Systems*, 2:93–122, 1984.

[3] M. Ajmone Marsan and G. Chiola. On Petri nets with deterministic and exponentially distributed firing times. In *Lecture Notes in Computer Science*, volume 266, pages 132–145. Springer Verlag, 1987.

[4] A. Bobbio and M. Telek. Task completion time. In *Proceedings 2nd International Workshop on Performability Modelling of Computer and Communication Systems (PMCCS2)*, 1993.

[5] A. Bobbio and M. Telek. Computational restrictions for SPN with generally distributed transition times. In D. Hammer K. Echtle and D. Powell, editors, *First European Dependable Computing Conference (EDCC-1)*, pages 131–148, 1994.

[6] A. Bobbio and M. Telek. Transient analysis of a preemptive resume M/D/1/2/2 through Petri nets. Technical report, Department of Telecommunications - Technical University of Budapest, April 1994.

[7] H. Choi, V.G. Kulkarni, and K. Trivedi. Transient analysis of deterministic and stochastic Petri nets. In *Proceedings of the 14-th International Conference on Application and Theory of Petri Nets*, Chicago, June 1993.

[8] H. Choi, V.G. Kulkarni, and K. Trivedi. Markov regenerative stochastic Petri nets. *Performance Evaluation*, 20:337–357, 1994.

[9] G. Ciardo, R. German, and C. Lindemann. A characterization of the stochastic process underlying a stochastic Petri net. In *Proceedings International Workshop on Petri Nets and Performance Models - PNPM93*, pages 170–179. IEEE Computer Society, 1993.

[10] G. Ciardo and C. Lindemann. Analysis of deterministic and stochastic Petri nets. In *Proceedings International Workshop on Petri Nets and Performance Models - PNPM93*, pages 160–169. IEEE Computer Society, 1993.

[11] E. Cinlar. *Introduction to Stochastic Processes.* Prentice-Hall, Englewood Cliffs, 1975.

[12] D.R. Cox. The analysis of non-markovian stochastic processes by the inclusion of supplementary variables. *Proceedings of the Cambridge Phylosophical Society*, 51:433–440, 1955.

[13] J. Bechta Dugan, K. Trivedi, R. Geist, and V.F. Nicola. Extended stochastic Petri nets: applications and analysis. In *Proceedings PERFORMANCE '84*, Paris, 1984.

[14] D.P. Gaver. A waiting line with interrupted service, including priorities. *Journal of the Royal Statistical Society*, B24:73–90, 1962.

[15] R. German and C. Lindemann. Analysis of stochastic Petri nets by the method of supplementary variables. *Performance Evaluation*, 20: 317–335, 1994.

[16] P.J. Haas and G.S. Shedler. Regenerative stochastic Petri nets. *Performance Evaluation*, 6: 189–204, 1986.

[17] B.R. Haverkort and K. Trivedi. Specification techniques for Markov Reward Models. *Discrete Event Dynamic Systems: Theory and Applications*, 3:219–247, 1993.

[18] D.L. Jagerman. An inversion technique for the Laplace transform. *The Bell System Technical Journal*, 61:1995–2002, October 1982.

[19] V.G. Kulkarni, V.F. Nicola, and K. Trivedi. On modeling the performance and reliability of multimode computer systems. *The Journal of Systems and Software*, 6:175–183, 1986.

[20] C. Lindemann. An improved numerical algorithm for calculating steady-state solutions of deterministic and stochastic Petri net models. *Performance Evaluation*, 18:75–95, 1993.

[21] Miklós Telek. *Some advanced reliability modelling techniques*. Phd Thesis, Hungarian Academy of Science, 1994.