Steady State Analysis of Markov Regenerative SPN with Age Memory Policy

Miklós Telek[1], Andrea Bobbio[2], László Jereb[1]
Antonio Puliafito[3], Kishor S. Trivedi[4]

[1] Department of Telecommunications,
Technical University of Budapest, 1521 Budapest, Hungary
[2] Dipartimento di Elettronica per l'Automazione
Università di Brescia, 25123 Brescia, Italy
[3] Istituto di Informatica e Telecomunicazioni
Università di Catania, 95125 Catania, Italy
[4] Department of Electrical Engineering
Duke University, Durham NC 27708-0291, USA

Abstract. *Non-Markovian Stochastic Petri Nets (SPN) have been developed as a tool to deal with systems characterized by non exponentially distributed timed events. Recently, some effort has been devoted to the study of SPN with generally distributed firing times, whose underlying marking process belongs to the class of Markov Regenerative Processes (MRGP). We refer to this class of models as Markov Regenerative SPN (MRSPN). In this paper, we describe a computationally effective algorithm for the steady state solution of MRSPN with age memory policy and subordinated Continuous Time Markov Chain (CTMC).*

Key words: Stochastic Petri Nets, Generally distributed transitions, Markov regenerative processes, Preemptive resume policies.

1 Introduction

In the last decade several classes of *SPN*'s have been developed to deal with non-exponentially distributed events [7, 3]. The main reason for this is the observation that it is possible to identify a wide number of real situations in which deterministic or generally distributed events occur. Phase-type distributions [9] can be used to approximate the occurrence time of a generally distributed event, but the main drawback of this approach is in the often prohibitive state space size of the expanded Continuous Time Markov Chain *(CTMC)*.

The analysis of stochastic systems with non-exponential timing requires the development of suitable modeling tools. In [6] Markov Regenerative Stochastic Processes *(MRGPs)* have been proved to be an effective way to capture the behavior of the stochastic process underlying a *SPN* where at most one generally distributed transition is enabled in each marking. This class of *SPN* has been named as *MRSPN**. The solution proposed in [6] is based on the derivation of the time-dependent transition probability matrix in the Laplace transform domain, followed by a numerical inversion. German and Lindemann [11] have proposed to derive the steady state solution of the same model by resorting to the method of supplementary variables. The possibility of applying the methodology of supplementary variables to the transient analysis of a restricted set of *DSPN*'s is explored in [10]. Extensions to multiple simultaneous general transitions have been investigated in [7] and [5], under some structural restrictions.

A severe limitation of the models discussed in the previous references is that the generally distributed (or deterministic) transitions must be assigned a firing policy of enabling memory type. In some cases, a more realistic assumption is to adopt an age memory policy, according to which the amount of work already done by a running activity is not lost if the activity is temporarily suspended. Many practical situations behave according to this modeling extension. Consider for example, a fault tolerant, parallel computing system, where a single task may be interrupted either during a fault/recovery cycle or for the execution of a higher priority task, but when the cause originating the interruption is ceased, the dormant task is resumed from the point its interruption occurred.

To overcome the previous restriction, Bobbio and Telek defined the class of *MRSPN* with non-overlapping activity cycles [4]. The main characterization of this class is that the subordinated processes between any two successive regeneration time points are dominated by a single general transition whose memory policy can be either age or enabling type.

In this paper, we propose a computationally effective approach to the steady state analysis of the class of *MRSPN* with non-overlapping activity cycles whose subordinated processes are *CTMC*'s. The proposed method contains the existing results for the steady state analysis of *DSPN* and *MRSPN** as special cases.

The paper is organized as follows. Section 2 recalls some concepts, definitions and notation for *MRSPN*'s with non-overlapping activity cycles. In Section 3, a motivating example, based on a preemptive $M/D/1/\nu+\sigma/\nu+\sigma$ queueing system with two classes of customers, is proposed; its numerical analysis is postponed to Section 7. In Section 4 the transient analysis result is presented. Section 5 focuses on the steady state analysis when the subordinated process is a *CTMC*, and the related computational method is introduced in Section 6.

2 Markov Regenerative Stochastic Petri Nets

A standard technique for the probabilistic study of a *SPN* is to examine the marking process, $\{\mathcal{M}(t), t > 0\}$ defined over the *Reachability Set* $\mathcal{RS}(M_0)$, generated from a given initial marking M_0. If the transitions are assigned generally distributed firing times, the behaviour of the marking process depends on its past history. It has been proposed in [1], to represent the past history of the process by associating to each generally distributed transition tr_g a memory variable a_g and a memory policy. The memory variable a_g is a functional that depends on the time during which tr_g has been enabled and the memory policy defines when the memory variable is reset as a function of the enabling/disabling status of the corresponding transition. The semantics of different memory policies has been discussed in [1] where three alternatives have been proposed.

- *Resampling policy* - The memory variable a_g is reset to zero at any change of marking.
- *Enabling memory policy* - The memory variable a_g accounts for the work performed by the activity corresponding to tr_g from the last epoch in which tr_g has been enabled. When transition tr_g is disabled (even without firing) the corresponding enabling memory variable is reset.
- *Age memory policy* - The memory variable a_g accounts for the work performed by the activity corresponding to tr_g from its last firing up to the current epoch and is reset only when tr_g fires.

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 the exponential transitions, and can be assumed identically zero. Hence, the set of the transitions can be partitioned 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. Since the history of the marking process $\mathcal{M}(t)$ is accounted for through the age variables associated to the GEN transitions we can assert the following:

Proposition 1. *A regeneration time point τ_n^* in the marking process $\mathcal{M}(t)$ is the epoch of entrance in a marking M_n in which all the memory variables are equal to 0.*

The embedded sequence of regeneration time points $(\tau_n^*,\, M_n)$ if exists is a Markov renewal sequence and the marking process $\mathcal{M}(t)$ is a Markov regenerative process MRGP [8, 6, 7] (or semi-regenerative process).

Definition 1. *A SPN with GEN transitions, for which an embedded Markov renewal sequence $(\tau_n^*,\, M_n)$ exists, is a Markov Regenerative Stochastic Petri Nets (MRSPN) [6].*

To provide an analytical formulation of the stochastic process underlying a *MRSPN* [6, 8], we define the following matrix valued functions (of dimension $n \times n$), where n is the cardinality of $\mathcal{RS}(M_0)$.
Matrix $\mathbf{V}(t)$ is the transition probability matrix and provides the probability that the stochastic process $\mathcal{M}(t)$ is in marking j at time t given it was in i at $t = 0$. The matrix $\mathbf{K}(t)$ is the *global kernel* of the *MRGP* 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 $\mathbf{E}(t)$ is the *local kernel* since it describes the behavior of the marking process $\mathcal{M}(t)$ inside

two consecutive regeneration time points. The generic element $E_{ij}(t)$ provides the probability that the process is found in state j at time t starting from i at $\tau_0^* = 0$ before the next regeneration time point.
The transient behavior of the *MRSPN* can be evaluated by solving the following generalized Markov renewal equation (in matrix form) [8, 6]:

$$\mathbf{V}(t) \;=\; \mathbf{E}(t) \;+\; \mathbf{K} \,*\, \mathbf{V}(t) \tag{1}$$

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

$$[\mathbf{K} \,*\, \mathbf{V}(t)]_{ij} \;=\; \sum_k \int_0^t dK_{ik}(y)\, V_{kj}(t-y) \tag{2}$$

By denoting the Laplace Stieltjes transform *(LST)* of a function $F(t)$ by $F^{\sim}(s) = \int_0^{\infty} e^{-st}\, dF(t)$, Equation (1) becomes in the *LST* domain:

$$\mathbf{V}^{\sim}(s) \;= [\mathbf{I} \,-\, \mathbf{K}^{\sim}(s)]^{-1} \;\mathbf{E}^{\sim}(s) \tag{3}$$

The steady state solution can be evaluated as:

$$\lim_{t\to\infty} V_{ij}(t) = \lim_{t\to\infty} Pr\{\mathcal{M}(t) = j \mid \mathcal{M}(0) = i\} = \lim_{s\to 0} V_{ij}^{\sim}(s) \tag{4}$$

$\mathbf{K}(t)$ and $\mathbf{E}(t)$ depend on the evolution of the marking process between two consecutive regeneration time points. By virtue of the time homogeneity property, we can always define the two successive regeneration time points to be $\tau_0^* = 0$ and τ_1^*.

Definition 2. *The stochastic process subordinated to state i (denoted by $\mathcal{M}^i(t)$) is the restriction of the marking process $\mathcal{M}(t)$ for $t \leq \tau_1^*$ given $\mathcal{M}(\tau_0^*) = i$:*

$$\mathcal{M}^i(t) \;=\; [\mathcal{M}(t) \,:\, t \leq \tau_1^*,\, \mathcal{M}(\tau_0^*) = i]$$

According to Definition 2, $\mathcal{M}^i(t)$ describes the evolution of the *PN* starting at the regeneration time point $\tau^* = 0$ in the marking i, up to the next regeneration time point τ_1^*. Therefore, $\mathcal{M}^i(t)$ 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 $\mathbf{K}(t)$ and $\mathbf{E}(t)$ are determined by $\mathcal{M}^i(t)$.

The class of *MRSPN* addressed in this paper, is a direct generalization of the *MRSPN** class defined in [6]. In each marking almost a single GEN transition of either enabling or age memory type can be enabled, being all the other transitions EXP. This definition implies that the subordinated process is a *CTMC*.

Under the age memory policy, if a GEN transition tr_g is disabled without firing, the memory variable maintains its constant positive value. The memory variable a_g can only be reset by the firing of transition tr_g, and the enabling/disabling condition of tr_g during its firing cycle is tracked by introducing a reward (indicator) variable which is set to 1 in those markings in which tr_g is enabled and set to 0 in those markings in which tr_g is not enabled. With this assignment, the value of the age variable versus time can be computed as the total accumulated reward.

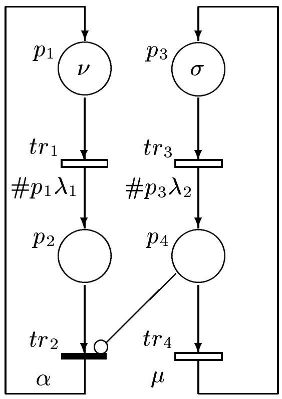

Figure 1 - Preemptive M/D/1/$\nu + \sigma/\nu + \sigma$ queue with two classes of customers.

3 A motivating example

The *PN* of Figure 1 models an M/D/1/$\nu + \sigma/\nu + \sigma$ queue with two classes of customers and limited capacity. The system supports at most ν jobs submitted by customer 1 and at most σ jobs submitted by customer 2. Jobs submitted by customer 2 have higher priority and preempt the jobs submitted by customer 1. This model, with $\nu = \sigma = 1$, has been addressed for the first time in [3,4] where the service times were assumed to be both deterministic with age memory policy.

The number of tokens in place p_1 (p_3) represents the number of customers of type 1 (type 2) thinking. Tokens in place p_2 (p_4) represent the number of jobs of type 1 (type 2) queueing for service (included the one under service). Transitions tr_1 and tr_3 are EXP and represent the submission of a job of type 1 or 2, respectively. tr_2 is GEN and tr_4 is EXP, and represent the completion of service of a job of type 1 or 2, respectively. Customers of type 2 are never interrupted, while customers of type 1 are preempted as soon as a type 2 customer requests service. The inhibitor arc from p_4 to tr_2 models the described preemption mechanism: as soon as a type 2 job joins the queue the type 1 job under service is interrupted. Under a *prd* service discipline each time customer of type 1 is preempted the job already accumulated is lost and upon restart the complete task must be re-executed. This service discipline is modeled by assigning to transition tr_2 an enabling memory policy, and the corresponding model belongs to the *MRSPN** class considered in [6]. A more interesting case arises when the server has a *prs* service discipline, so that upon restart of an interrupted job of type 1 only the remaining time needs to be completed. A *prs* service discipline is modeled by assigning to tr_2 an age memory policy. When p_4 is marked (a task of type 2 is under service), tr_2 is disabled but the related age variable a_2 keeps memory of the previous enabling time. This situation requires the modeling framework proposed in this paper.

4 Unified transient analysis of MRSPN

In this Section, we briefly summarize the theory for the transient analysis of *MRSPN* with non-overlapping activity cycles and subordinated *CTMC*. An extension to subordinated semi-Markov processes and major details on the analytical procedure can be found in [4, 15]. In this section we show how to derive row by row the global kernel $\mathbf{K}(t)$ and the local kernel $\mathbf{E}(t)$. Hence, we assume to start a new regeneration period at time $t = 0$ in marking i, being tr_g the dominant GEN transition and being the subordinated process a reward *CTMC*. To better understand the developed mathematical formalism, we briefly describe the adopted notation:

- Ω: reachability set $\mathcal{RS}(M_0)$, of cardinality n;
- r_i: reward rate associated to the tagged GEN transition in state i;
- $\mathbf{R}$: diagonal matrix of reward rates;
- R^i: subset of Ω grouping the states reachable during the regeneration period starting from state i;
- m: cardinality of R^i;
- h: cardinality of the subset of R^i in which the age variable associated with tr_g is strictly increasing (the reward rate is equal to 1);
- R^{ci}: subset of Ω in which tr_g is not active. Namely, $R^{ci} = \Omega - R^i$;
- $\mathcal{M}^i(t)$: subordinated *CTMC* defined over Ω;
- $\mathbf{A^i} = [a^i_{k\ell}]$: generator matrix of the subordinated *CTMC* $Z^i(t)$.

Let us renumber the states in $\mathbf{A}^i$ in such a way that states numbered $1, 2, \ldots, m$ belong to R^i ($1, 2, \ldots, m \in R^i$) and the states numbered $m+1, m+2, \ldots, n$ belong to R^{ci} ($m+1, m+2, \ldots, n \in R^{ci}$). By this ordering of states $\mathbf{A}^i$ can be partitioned into the following submatrices[1] $\mathbf{A^i} = \begin{array}{|c|c|}\hline \mathbf{B} & \mathbf{C} \\ \hline \mathbf{0} & \mathbf{0} \\ \hline \end{array}$ where $\mathbf{B}$ contains the intensity of the transitions inside R^i, and $\mathbf{C}$ contains the intensity of the transitions from R^i to R^{ci}, being the other submatrices irrelevant.

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

$$P^i_{k\ell}(t,w) \;=\; Pr\{Z^i(t) = \ell \in R^i \,,\, \tau_1^* > t \,|\, Z^i(0) = k \in R^i \,,\, \gamma_g = w\}$$

$$F^i_{k\ell}(t,w) \;=\; Pr\{Z^i(\tau_1^{*\,-}) = \ell \in R^i \,,\, \tau_1^* \le t,\, tr_g \text{ fires} \,|\, Z^i(0) = k \in R^i \,,\, \gamma_g = w\}$$

$$D^i_{k\ell}(t,w) \;=\; Pr\{Z^i(\tau_1^*) = \ell \in R^{c\,i} \,,\, \tau_1^* \le t \,|\, Z^i(0) = k \in R^i \,,\, \gamma_g = w\}$$

$$\Delta^i_{k\ell} \;=\; Pr\{\text{ next tangible marking is } \ell \,|\, \text{current marking is } k,\; tr_g \text{ fires}\,\}$$

By the above definitions, $P^i_{k\ell}(t,w)$ and $F^i_{k\ell}(t,w)$ are 0 if $k \in R^{ic}$ or $\ell \in R^{ic}$; and $D^i_{k\ell}(t,w)$ is 0 if $k \in R^{ic}$ or $\ell \in R^i$.

[1] In notation of the particular submatrices we neglect the dependence on the particular subordinated process, and superscript i is omitted in the following.

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

From the above definitions, it follows for any t:

$$\sum_{\ell \in \Omega^i} \left[P^i_{k\ell}(t,w) \;+\; F^i_{k\ell}(t,w) \;+\; D^i_{k\ell}(t,w) \right] \;=\; 1$$

Given that $G_g(w)$ is the cumulative distribution function of the r.v. γ_g associated to the transition tr_g, the elements of the i-th row of matrices $\mathbf{K}(t)$ and $\mathbf{E}(t)$ can be expressed as follows, as a function of the matrices $\mathbf{P}^i(t,w)$, $\mathbf{F}^i(t,w)$ and $\mathbf{D}^i(t,w)$ [4]:

$$\begin{aligned} K_{ij}(t) \;&=\; \int_{w=0}^{\infty} \left[\sum_{k \in R^i} F^i_{ik}(t,w)\, \Delta^i_{kj} \;+\; D^i_{ij}(t,w) \right] dG_g(w) \\ E_{ij}(t) \;&=\; \int_{w=0}^{\infty} P^i_{ij}(t,w)\, dG_g(w) \end{aligned} \tag{5}$$

The introduced matrix functions are derived in double Laplace transform domain based on the infinitesimal generator of the subordinated *CTMC* $\mathbf{A}^i$ and on the memory policy of the dominant GEN transition. A detailed derivation is in [15, 4]. The final expressions take the following matrix form.

$$\mathbf{F}^{i\sim *}(s,v) = (s\mathbf{I} + v\mathbf{R} - \mathbf{B})^{-1}\mathbf{R} \tag{6}$$

$$\mathbf{P}^{i\sim *}(s,v) = \frac{s}{v}\;(s\mathbf{I} + v\mathbf{R} - \mathbf{B})^{-1} \tag{7}$$

$$\mathbf{D}^{i\sim *}(s,v) = \frac{1}{v}\;(s\mathbf{I} + v\mathbf{R} - \mathbf{B})^{-1}\mathbf{C} \tag{8}$$

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

A numerical technique based on Equations (6-8), has been discussed in [4, 14]. The technique contains some computationally intensive steps, and can be applied to very small scale problems (less than 10 states) only.

5 Steady state analysis of MRSPN

For the steady state analysis of the *MRSPN* introduced in the previous sections, we propose a new, computationally effective, approach based on the procedure published by Ajmone Marsan and Chiola [2]. The procedure is extended for the first time to GEN transitions with age memory policy. Let us define:

$$\alpha_{ij} = \int_{t=0}^{\infty} E_{ij}(t)dt \quad ; \qquad \alpha_i = \sum_j \alpha_{ij} \tag{9}$$

the expected time the $\mathcal{M}^i(t)$ subordinated process spends in state j, and the expected duration of the activity cycle of transition tr_g.

The state transition probability matrix of the *DTMC* embedded into the regeneration time points is

$$\mathbf{\Pi} = \{\pi_{ij}\} = \lim_{t \to \infty} \mathbf{K}(t) \tag{10}$$

Let $P = \{p_i\}$ (row vector) be the unique solution of the set of equations:

$$P = P\mathbf{\Pi} \quad ; \qquad \sum_i p_i = 1 \tag{11}$$

The steady state probabilities of the *MRGP* can be evaluated based on α_{ij} and π_{ij} (or p_i by applying (11)) as follows [2]:

$$v_{ij} = \lim_{t \to \infty} Pr\{\mathcal{M}(t) = j \,|\, \mathcal{M}(0) = i\} = \frac{\sum_k p_k \alpha_{kj}}{\sum_k p_k \alpha_k} \tag{12}$$

Thus, the steady state solution can be easily obtained once the *time conversion factor* α_{ij} and the *state transition probabilities* π_{ij} of the *DTMC* embedded into the regeneration points are known. The memory policy for GEN transitions is taken into account in the evaluation of the local and global kernels.

5.1 Time conversion factor α_{ij}

Let us suppose that the states of R^i in which the dominant transition is enabled (reward rate $= 1$) are numbered as $1, 2, \ldots, h$ $(h \leq m)$ and the states of R^i in which the dominant transition is disabled (reward rate $= 0$) are numbered $h+1,\ h+2,\ \ldots,\ m$. By this ordering of the infinitesimal generator $\mathbf{A^i}$ of the subordinated *CTMC* starting from state i can be partitioned into the following submatrices

$$\mathbf{A^i} = \begin{array}{|c|c|} \hline \mathbf{B} & \mathbf{C} \\ \hline \mathbf{0} & \mathbf{0} \\ \hline \end{array} \quad = \quad \begin{array}{|cc|c|} \hline B_1 & B_2 & \\ B_3 & B_4 & \mathbf{C} \\ \hline \multicolumn{2}{|c|}{\mathbf{0}} & \mathbf{0} \\ \hline \end{array}$$

Theorem 1. *The expected time the subordinated CTMC $\mathcal{M}^i(t)$ spends in state $j \in R^i$ starting from state $i \in R^i$ is:*

$$\alpha_{ij}^i = \int_{w=0}^{\infty} \begin{bmatrix} L(w) & -L(w)\mathrm{B}_2{\mathrm{B}_4}^{-1} \\ -{\mathrm{B}_4}^{-1}\mathrm{B}_3 L(w) & {\mathrm{B}_4}^{-1} + {\mathrm{B}_4}^{-1}\mathrm{B}_3 L(w)\mathrm{B}_2{\mathrm{B}_4}^{-1} \end{bmatrix}_{ij} dG_g(w) \qquad (13)$$

where

$$\beta = \mathrm{B}_1 - \mathrm{B}_2{\mathrm{B}_4}^{-1}\mathrm{B}_3 \quad , \quad L(w) = \int_0^w e^{\beta w} dw.$$

and superscript i denotes the fact that the states are rearranged according to the enabled or disabled state of the tagged GEN transition.

Proof.

$$\alpha_{ij}^i = \int_{t=0}^{\infty} E_{ij}(t)dt = \lim_{s\to 0} \frac{1}{s} E_{ij}^{\sim}(s) = \lim_{s\to 0} \frac{1}{s} \int_{w=0}^{\infty} P_{ij}^{i\sim}(s,w)\, dG_g(w) = \int_{w=0}^{\infty} \mathrm{LT}^{-1} \left[\frac{1}{v} \left|(v\mathbf{R}-\mathbf{B})^{-1}\right|_{ij} \right] dG_g(w) \qquad (14)$$

Let us consider the term $\mathrm{LT}^{-1}\left[1/v\ (v\mathbf{R}-\mathbf{B})^{-1}\right]$ separately. Based on the numbering of the states $\mathbf{R}$ has the form $\mathbf{R} = \begin{array}{|c|c|} \hline I_1 & 0 \\ \hline 0 & 0 \\ \hline \end{array}$ where I_1 is the identity matrix of dimension $(h \times h)$. Using this and Lemma 1 in the Appendix, the inverse Laplace transform satisfies the following equation:

$$\mathrm{LT}^{-1}\left[\frac{1}{v}(v\mathbf{R}-\mathbf{B})^{-1}\right] = \mathrm{LT}^{-1}\left\{\frac{1}{v}\begin{bmatrix} v\mathrm{I}_1 - \mathrm{B}_1 & -\mathrm{B}_2 \\ -\mathrm{B}_3 & -\mathrm{B}_4 \end{bmatrix}^{-1}\right\} =$$

$$\mathrm{LT}^{-1}\left\{\frac{1}{v}\begin{bmatrix} (v\mathrm{I}_1-\beta)^{-1} & -(v\mathrm{I}_1-\beta)^{-1}\mathrm{B}_2{\mathrm{B}_4}^{-1} \\ -{\mathrm{B}_4}^{-1}\mathrm{B}_3(v\mathrm{I}_1-\beta)^{-1} & {\mathrm{B}_4}^{-1} + {\mathrm{B}_4}^{-1}\mathrm{B}_3(v\mathrm{I}_1-\beta)^{-1}\mathrm{B}_2{\mathrm{B}_4}^{-1} \end{bmatrix}\right\} =$$

$$\begin{bmatrix} L(w) & -L(w)\mathrm{B}_2{\mathrm{B}_4}^{-1} \\ -{\mathrm{B}_4}^{-1}\mathrm{B}_3 L(w) & {\mathrm{B}_4}^{-1} + {\mathrm{B}_4}^{-1}\mathrm{B}_3 L(w)\mathrm{B}_2{\mathrm{B}_4}^{-1} \end{bmatrix} \qquad (15)$$

where $L(w) = \int_0^w e^{\beta w} dw$ (see Lemma 3 in the Appendix). From this the theorem follows.

5.2 State transition probabilities of the EMC π_{ij}

Theorem 2. *The state transition probabilities of the DTMC embedded into the regeneration time points are given by:*

$$\pi^i_{ij} = \int_{w=0}^{\infty} \sum_{k \le h} \begin{bmatrix} e^{\beta w} & 0 \\ -\mathrm{B_4}^{-1}\mathrm{B_3} e^{\beta w} & 0 \end{bmatrix}_{ik} \Delta^i_{kj}\, dG_g(w)$$
$$+ \int_{w=0}^{\infty} \sum_{k \le m} \begin{bmatrix} L(w) & -L(w)\mathrm{B_2}\mathrm{B_4}^{-1} \\ -\mathrm{B_4}^{-1}\mathrm{B_3} L(w) & \mathrm{B_4}^{-1} + \mathrm{B_4}^{-1}\mathrm{B_3} L(w) \mathrm{B_2}\mathrm{B_4}^{-1} \end{bmatrix}_{ik} C_{kj}\, dG_g(w) \qquad (16)$$

Proof.

$$\pi^i_{ij} = \lim_{t\to\infty} K_{ij}(t) = \lim_{s\to 0} K^{\sim}_{ij}(s) =$$
$$\lim_{s\to 0} \int_{w=0}^{\infty} \sum_{k \le h} F^{i}_{ik}{}^{\sim}(s,w) \Delta^i_{kj} + D^{i}_{ij}{}^{\sim}(s,w)\, dG_g(w) =$$
$$\int_{w=0}^{\infty} \sum_{k \le h} \mathrm{LT}^{-1} \left[(v\mathbf{R} - \mathbf{B})^{-1} \mathbf{R} \right]_{ik} \Delta^i_{kj}\, dG_g(w) \qquad (17)$$
$$+ \int_{w=0}^{\infty} \sum_{k \le m} \mathrm{LT}^{-1} \left[\frac{1}{v} (v\mathbf{R} - \mathbf{B})^{-1} \right]_{ik} C_{kj}\, dG_g(w)$$

Let us consider the term $\mathrm{LT}^{-1}\left[(v\mathbf{R}-\mathbf{B})^{-1}\mathbf{R}\right]$ separately. By the partitioned form of $\mathbf{R}$ and $\mathbf{B}$ and adopting Lemma 2 in the Appendix, the inverse Laplace transform satisfies the following equation:

$$\mathrm{LT}^{-1}\left[(v\mathbf{R}-\mathbf{B})^{-1}\mathbf{R}\right] = \mathrm{LT}^{-1}\left\{ \begin{bmatrix} v\mathrm{I_1} - \mathrm{B_1} & -\mathrm{B_2} \\ -\mathrm{B_3} & -\mathrm{B_4} \end{bmatrix}^{-1} \mathbf{R} \right\} =$$
$$\mathrm{LT}^{-1} \begin{bmatrix} (v\mathrm{I_1} - \beta)^{-1} & 0 \\ -\mathrm{B_4}^{-1}\mathrm{B_3}(v\mathrm{I_1} - \beta)^{-1} & 0 \end{bmatrix} = \begin{bmatrix} e^{\beta w} & 0 \\ -\mathrm{B_4}^{-1}\mathrm{B_3} e^{\beta w} & 0 \end{bmatrix} \qquad (18)$$

The term $\mathrm{LT}^{-1}\left[\frac{1}{v}(v\mathbf{R}-\mathbf{B})^{-1}\right]$ is known from (15).

6 Computational method

The relevant entries of the matrices $(\alpha^i_{ij}, \pi^i_{ij})$ are located in row i $(i \le h)$, because when the subordinated process of the dominant transition tr_g starts, transition tr_g is enabled. The expressions for α^i_{ij} and π^i_{ij} are evaluated in the following subsections assuming enabling and age type memory policy, respectively.

6.1 Enabling type dominant transition

In the case of enabling type dominant transition $h = m$ and $m < n$, since the GEN transition is enabled in all states reachable during the regeneration period, and $\beta = \mathrm{B}_1$ hence the partitioned form of the infinitesimal generator of the subordinated process is $\mathbf{A^i} = \begin{array}{|c|c|}\hline \mathrm{B}_1 & \mathbf{C} \\ \hline \mathbf{0} & \mathbf{0} \\ \hline \end{array}$ Note that $\mathbf{A^i}$ is a $(n \text{ x } n)$ matrix, and, in this case, B_1 is a $(h \text{ x } h)$ matrix.

$$\alpha^i_{ij} = \int_{w=0}^{\infty} [L(w)]_{ij}\, dG_g(w) = \int_{w=0}^{\infty} \int_{\tau=0}^{w} \left[e^{\tau \mathrm{B}_1}\right]_{ij} d\tau\, dG_g(w) \tag{19}$$

$$\pi^i_{ij} = \int_{w=0}^{\infty} \sum_{k \le h} \left[e^{\mathrm{B}_1 w}\right]_{ik} \Delta^i_{kj}\, dG_g(w) + \int_{w=0}^{\infty} \sum_{k \le m} [L(w)]_{ik}\, C_{kj}\, dG_g(w) \tag{20}$$

As can be observed, in evaluating $e^{A^i t}$ we just need to concentrate on the submatrix B_1. The complexity of the proposed algorithm is thus in the evaluation of $e^{\mathrm{B}_1 t}$ and its integral. This observation allows us to save computational time. Other algorithms, such as those proposed in [13, 6], evaluate $e^{A^i t}$, even if it is not the case.

6.2 Age type dominant transition

In the case of age type dominant transition the GEN transition can become disabled during the regenerative period, hence $h \le m$, but none of the state transitions, only the firing of tr_g can conclude the regeneration period, hence $m = n$. Thus the partitioned form of the infinitesimal generator of the subordinated process is $\mathbf{A^i} = \begin{array}{|c|c|}\hline B_1 & B_2 \\ \hline B_3 & B_4 \\ \hline \end{array}$ and $D^i_{ij}{}^{\sim}(s, w) = 0$.

$$\alpha^i_{ij} = \int_{w=0}^{\infty} \left[L(w)\middle|-L(w)\mathrm{B}_2{\mathrm{B}_4}^{-1}\right]_{ij} dG_g(w) \tag{21}$$

$$\pi^i_{ij} = \int_{w=0}^{\infty} \sum_{k \le h} \left[e^{\beta w}\right]_{ik} \Delta^i_{kj}\, dG_g(w) \tag{22}$$

The complexity of the algorithm for the age type dominant transition, is related to the evaluation of $e^{\beta t}$, where β is a $(h \text{ x } h)$ matrix, and to the evaluation of the inverse of matrix B_4 which is an $(m - h \text{ x } m - h)$ matrix.

7 Numerical example

The preemptive priority queue with two type of customers, introduced in Section 3, has been addressed for the first time in [3, 4] with only one customer per type; $\nu = \sigma = 1$. In the following we show how the algorithm presented in the previous section can handle more complex examples. To fit the requirement that only one

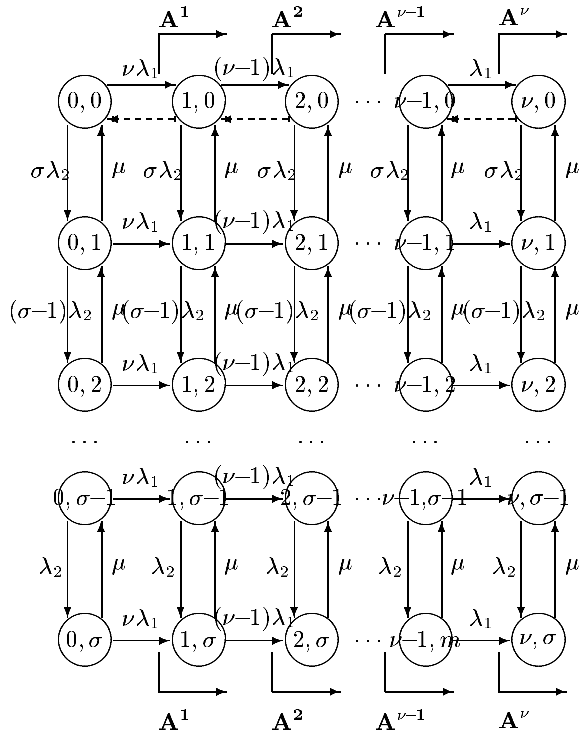

Figure 2 - State transition diagram of the preemptive M/D/1/$\nu+\sigma/\nu+\sigma$ queue with two classes of customers.

GEN transition is enabled in each marking, we assume that the service time of customer 2 is EXP, while it is deterministic for customer 1.

The reachability graph of the *PN* of Figure 1 is depicted in Figure 2. States in Figure 2 are labeled with two indices: the first index is the number of tokens in place p_2 ($\#p_2$), the second index is the number of tokens in place p_4 ($\#p_4$). Let us define a numbering of the states between 1 and $(\nu+1)\times(\sigma+1)$ based on the following rule: $i = 1 + \#p_2 + \#p_4 \times (\nu+1)$. For notational convenience we denote $[(i-1) \mod (\nu+1)]$ by i^* and $[(i-1) \text{ div } (\nu+1)]$ by i^+.

From Figure 2, it is easily recognized that all states can be regeneration states. Only exponential transitions are enabled in state 1 and states $\{i : i > \nu+1\}$. The elements of the $\mathbf{\Pi}$ and α matrices can be easily defined for these states:

for $i = 1$ and $i > \nu + 1$

$$\pi_{ij} = \begin{cases} \dfrac{(\nu - i^*)\lambda_1}{(\nu - i^*)\lambda_1 \,+\, (\sigma - i^+)\lambda_2 \,+\, \mu} & j = i+1 \\[2ex] \dfrac{(\sigma - i^+)\lambda_2}{(\nu - i^*)\lambda_1 \,+\, (\sigma - i^+)\lambda_2 \,+\, \mu} & j = i+\nu+1 \\[2ex] \dfrac{\mu}{(\nu - i^*)\lambda_1 \,+\, (\sigma - i^+)\lambda_2 \,+\, \mu} & j = i-\nu-1 \\[2ex] 0 & otherwise \end{cases} \qquad (23)$$

$$\alpha_{ii} = \alpha_i = \frac{1}{(\nu - i^*)\lambda_1 \,+\, (\sigma - i^+)\lambda_2 \,+\, \mu} \quad ; \qquad \alpha_{ij} = 0 \text{ if } j \neq i \qquad (24)$$

States $\{i : 1 < i \leq \nu + 1\}$ are age type regeneration states. The state space of the subordinated process R^i contains all the states reachable during the activity cycle of tr_2. The only criterion for the termination of the activity cycle is the firing of tr_2. The reward vector is equal to 1 for the states $j \in R^i$ in which tr_2 is enabled (states located in the first row on Figure 2) and 0 for the states $j \in R^i$ in which tr_2 is not enabled. As during the activity cycle of tr_g no other GEN transitions are activated, the subordinated process is a reward *CTMC*. Then we meet the structural conditions required by the steady state solution method described in the previous Sections.

The reachable states and the subordinated *CTMC* for state i is denoted by $\mathbf{A^i}$ in Figure 2. (The deterministic state transitions /dashed lines/ should be excluded.) Every relevant row of the branching probability matrix contains only one non-zero entry $\Delta_{i,i-1} = 1$.

Table 1 reports the steady-state results of the example of Figure 1 for a deterministic service duration $\alpha = 1$, an exponential service duration of mean $\mu = 1$, submitting rates $\lambda_1 = \lambda_2 = 0.25$ and for $n = 3$, $m = 2$. Columns from 3 to 5 illustrate the state probabilities assuming an age type memory policy. In detail, the third column shows the results obtained by implementing the proposed method through a a MATHEMATICA program. Columns 4 and 5 ,instead, provide the results obtained by approximating the deterministic service time tr_2 by an Erlang distribution of order 10 and 1000 respectively, and evaluating the resulting *PHSPN* model [3] utilizing the method described in [9]. Columns from 6 to 8 give the results of the same example when an enabling memory policy is assumed. More specifically, column 6 contains the results obtained through a MATHEMATICA implementation, while columns 7 and 8 provide the results obtained a through a PHSPN model. Two are the main conclusions deriving from the observation of Table 1. First, the proposed method provides the same accuracy of the Erlang approximation of order 1000, but with a much more limited state space: 12 states against 9003 states. Second, the comparison between age and enabling memory policies highlights how the assumption on the memory policy strongly affects the validity of the results. Finally, one more comparison was carried out solving the model of Figure 1 under the assumption of enabling type memory policy with DSPNexpress [12]. We observed that even if the final results were wery close to the ones listed in column 6, nevertheless the CPU time of the MATHEMATICA program was only some seconds, while DSPNexpress took some minutes on an IBM RISC 6000 computer.

state	marking	MRSPN age	PHSPN (10) age	PHSPN (1000) age	MRSPN enabling	PHSPN (10) enabling	PHSPN (1000) enabling
1	3020	0.17960	0.18209	0.17963	0.13142	0.13958	0.13218
2	2120	0.25286	0.24491	0.25278	0.22130	0.21818	0.22101
3	1220	0.14814	0.14977	0.14815	0.18433	0.18016	0.18394
4	0320	0.03478	0.03861	0.03482	0.07832	0.07744	0.07824
5	3011	0.04835	0.04902	0.04836	0.03538	9.03758	0.03558
6	2111	0.10493	0.10277	0.10491	0.08824	0.08839	0.08825
7	1211	0.10378	0.10348	0.10378	0.11014	0.10865	0.11000
8	0311	0.05061	0.05241	0.05064	0.07393	0.07306	0.07384
9	3002	0.00691	0.00700	0.00691	0.00505	0.00536	0.00508
10	2102	0.02094	0.02063	0.02094	0.01723	0.01741	0.01725
11	1202	0.02913	0.02894	0.02913	0.02892	0.02869	0.02890
12	0302	0.01994	0.02034	0.01994	0.02571	0.02543	0.02568

Table I - Steady state results of the preemptive $M/D/1/\nu+\sigma/\nu+\sigma$ queue ($\nu = 3$, $\sigma = 2$)

8 Conclusion

We considered *MRSPN* with age type memory policy. This kind of memory policy was motivated by the need of modeling systems in which the execution of tasks may follow a preemptive resume policy. We focused on the case in which the subordinated stochastic process is a CTMC and derived a computationally effective approach for the steady state analysis of Petri net with GEN transitions and associated age memory policy.

Acknowledgements

Kishor Trivedi and Andrea Bobbio would like to thank NATO for grant No. CRG.940308. László Jereb and Miklós Telek would like to thank Hungarian OTKA for grant No. T-16637. Antonio Puliafito the Italian CNR for grant No. 203.15.5. Miklós Telek would like to thank Hungarian OTKA for grant No. W-015859.

Appendix

Lemma 1. *The inverse of the partitioned matrix* $\mathcal{M} = \begin{array}{|c|c|}\hline \mathcal{A} & \mathcal{B} \\ \hline \mathcal{C} & \mathcal{D} \\ \hline \end{array}$ *has the following form, when* $DET(\mathcal{M}) \neq 0$ *and* $DET(\mathcal{D}) \neq 0$ *:*

$$\mathcal{N} = \begin{vmatrix} (\mathcal{A} - \mathcal{B}\mathcal{D}^{-1}\mathcal{C})^{-1} & -(\mathcal{A} - \mathcal{B}\mathcal{D}^{-1}\mathcal{C})^{-1}\mathcal{B}\mathcal{D}^{-1} \\ -\mathcal{D}^{-1}\mathcal{C}(\mathcal{A} - \mathcal{B}\mathcal{D}^{-1}\mathcal{C})^{-1} & \mathcal{D}^{-1} + \mathcal{D}^{-1}\mathcal{C}(\mathcal{A} - \mathcal{B}\mathcal{D}^{-1}\mathcal{C})^{-1}\mathcal{B}\mathcal{D}^{-1} \end{vmatrix} \quad (25)$$

Proof. $\mathcal{N}\mathcal{M} = \mathbf{I}$; $\quad \mathcal{M}\mathcal{N} = \mathbf{I}$

Lemma 2. *Given matrix* A $(n \times n)$, *the inverse Laplace transform of* $(sI - A)^{-1}$ *has the following form:* $LT^{-1}(sI - A)^{-1} = e^{At}$

Proof. $LT(e^{At}) = (sI - A)^{-1}$

Lemma 3. *Given matrix* A $(n \times n)$, *the inverse Laplace transform of* $\frac{1}{s}(sI - A)^{-1}$ *has the following form:* $LT^{-1}\left[\frac{1}{s}(sI - A)^{-1}\right] = \int_0^{\tau} e^{At} dt = L(\tau)$

Proof. Given $Y(t) = \int_0^t F(\tau) d\tau$ by definition $LT\,[Y(t)] = Y^*(s) = 1/sF^*(s)$ where $F^*(s) = LT\,[F(t)]$ If $F(t) = e^{At}$ then: $LT\left[\int_0^t e^{A\tau} d\tau\right] = \frac{1}{s} LT(e^{A\tau}) = \frac{1}{s}\left(sI - A\right)^{-1}$

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 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.
3. 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), Lecture Notes in Computer Science*, volume 852, pages 131–148, 1994.
4. A. Bobbio and M. Telek. Markov regenerative SPN with non-overlapping activity cycles. In *International Computer Performance and Dependability Symposium - IPDS95*, April 1995.
5. V. Catania, A. Puliafito, M. Scarpa, and L. Vita. Concurrent generalized petri nets. In *Proceedings of Numerical Solution of Markov Chains*, pages 359–382, Raleigh, NC, 1995.
6. Hoon Choi, V.G. Kulkarni, and K. Trivedi. Markov regenerative stochastic Petri nets. *Performance Evaluation*, 20:337–357, 1994.
7. G. Ciardo, R. German, and C. Lindemann. A characterization of the stochastic process underlying a stochastic Petri net. *IEEE Transactions on Software Engineering*, 20:506–515, 1994.

8. E. Cinlar. *Introduction to Stochastic Processes.* Prentice-Hall, Englewood Cliffs, 1975.
9. A. Cumani. Esp - A package for the evaluation of stochastic Petri nets with phase-type distributed transition times. In *Proceedings International Workshop Timed Petri Nets*, pages 144–151, Torino (Italy), 1985. IEEE Computer Society Press no. 674.
10. R. German. *Transient Analysis of deterministic and stochastic Petri nets by the method of supplementary variables.* Internal Report Technische Universität Berlin (to be presented MASCOT'95), 1994.
11. R. German and C. Lindemann. Analysis of stochastic Petri nets by the method of supplementary variables. *Performance Evaluation*, 20:317–335, 1994.
12. C. Lindemann. DSPNexpress: a software package for the efficient solution of deterministic and stochastic Petri nets. In R. Pooley and J. Hilliston, editors, *Proceedings 6-th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation*, pages 15–29, Edinburgh, 1992.
13. C. Lindemann. An improved numerical algorithm for calculating steady-state solutions of deterministic and stochastic Petri net models. *Performance Evaluation*, 18:75–95, 1993.
14. M. Telek and A. Bobbio. Markov regenerative stochastic Petri nets with age type general transitions. In *16-th International Conference Application and Theory of Petri Nets*, June 1995.
15. Miklós Telek. *Some advanced reliability modelling techniques.* Phd Thesis, Hungarian Academy of Science, 1994.