Preemptive Repeat Identical Transitions in Markov Regenerative Stochastic Petri Nets

Andrea Bobbio
Dipartimento di Elettronica per l'Automazione
Università di Brescia, 25123 Brescia, Italy

Vidyadhar G. Kulkarni
Department of Operations Research
The University of North Carolina, Chapel Hill, NC 27599-3180

Antonio Puliafito
Istituto di Informatica e Telecomunicazioni
Università di Catania, 95125 Catania, Italy

Miklós Telek
Department of Telecommunications, Technical University of Budapest
1521 Budapest, Hungary

Kishor S. Trivedi
Center for Advanced Computing and Communication
Department of Electrical Engineering
Duke University, Durham, NC 27708-0291

Abstract
The recent literature on Markov Regenerative Stochastic Petri Nets (MRSPN) assumes that the random firing time associated to each transition is resampled each time the transition fires or is disabled by the firing of a competitive transition. This modeling assumption does not cover the case of preemption mechanisms of repeat identical nature (pri). In this policy, an interrupted job must be repeated with an identical requirement so that its associated random variable must not be resampled. The paper investigates the implication of a pri policy into a MRSPN and describes an analytical procedure for the derivation of expressions for the transient probabilities.
Key words: Stochastic Petri Nets, Semi-Markov Reward Models, Markov regenerative processes, preemptive repeat identical policy.

1 Introduction

The analysis of stochastic systems with non-exponential timing is of increasing interest in the literature and requires the development of suitable modeling tools. Choi et al. have shown in [8] that the marking process underlying a *Stochastic Petri Net (SPN)*, where at most one generally distributed transition is enabled in each marking, belongs to the class of Markov Regenerative Stochastic Processes *(MRGPs)*. For this reason they referred to this new class of Petri nets as *Markov Regenerative Stochastic Petri Net (MRSPN)*. Following the line opened in [8], different approaches have been proposed to deal with non-exponential systems [11, 6, 14, 9].

The analysis technique proposed for this class of models, consists in identifying a sequence of time points at which it is possible to forget the past history of the process. These points, indicated as *regeneration points*, are such that the future evolution of the stochastic process only depends on the state entered when a regeneration time point occurs. Based on the sequence of the regeneration time points, an analytical formulation of the process is available [10, 8].

The models discussed in the previous references require that the generally distributed (or deterministic) transitions are assigned a firing policy of enabling memory type [1]. Bobbio and Telek [15] have introduced the class of *AgeMRSPN*, in which a general transition can be associated an age memory policy. The enabled/disabled status of the transition is marked by a binary reward variable, so that the process subordinated to two consecutive regeneration points can be a semi-Markov reward process. *AgeMRSPN* have proved to be useful in representing situations in which an interrupted job is resumed without loss of the previous work. In [16], a computationally effective approach to the steady state analysis of *AgeMRSPN* with subordinated *CTMC* is proposed.

The technique is applied to the steady-state solution of a preemptive *M/D/1/(n+m)/(n+m)* queueing system with two classes of customers.

In all the previous literature on *MRSPN* [9, 3], based on the semantics exposed in [1], it has been implicitly assumed that, for an enabling memory transition, the associated random firing time is resampled each time the transition fires or is disabled by the firing of a competitive transition. This assumption cannot include the case of a *pri* policy [13] in which an interrupted job must be repeated with an identical requirement.

The paper is aimed at investigating the conditions under which a *pri* policy can be modeled by means of the class of *MRSPN*. Finally, an analytical procedure for the derivation of expressions for the transient and steady state probabilities are provided.

A pratical situation, in which a *pri* policy is of value [7], arises in connection with the execution of jobs whose duration is a random variable of known distribution. Selecting a job corresponds to picking up a sample from the job duration distribution. If the execution is suspended and restarted a processing time identical to the one of the interrupted job is required.

The paper is organized as follows. Section 2 introduces the concept of a *pri* policy in connection with *MRSPN*. Section 3 characterizes the process subordinated to a dominant *pri* transition. Section 4 presents the transient solution of a preempitive *M/G/1/2/2* queueing system with two classes of customers, one of which behaves according to a *pri* mechanism.

2 pri memory policy

We adopt very standard notation. A marked Petri Net is a tuple $PN = (P, T_r, I, O, H, M_0)$, where: P is the set of places, Tr the set of transitions, I, O and H are the input, the output and the inhibitor functions, respectively, and M_0 is the initial marking. The reachability set $\mathcal{RS}(M_0)$ is the set of all the markings that can be generated from the initial marking M_0. The marking process $\mathcal{M}(t)$ denotes the marking occupied by the *PN* at time t.

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 the Markov property holds.*

To provide an analytical formulation of the stochastic process underlying a *MRSPN*, the following matrix valued functions ($\mathbf{V}(t) = [V_{ij}(t)], \mathbf{K}(t) = [K_{ij}(t)]$ and $\mathbf{E}(t) = [E_{ij}(t)]$) are defined on the reachability set $\mathcal{RS}(M_0)$ [10, 8]:

$$\begin{aligned} V_{ij}(t) &= Pr\{\mathcal{M}(t) = j \,|\, \mathcal{M}(\tau_0^*) = i\} \qquad (1)\\ K_{ij}(t) &= Pr\{\mathcal{M}(\tau_1^*) = j\,,\, \tau_1^* \le t | \,\mathcal{M}(\tau_0^*) = i\}\\ E_{ij}(t) &= Pr\{\mathcal{M}(t) = j\,,\, \tau_1^* > t | \,\mathcal{M}(\tau_0^*) = i\} \end{aligned}$$

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 marking 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 $\mathcal{M}(t)$ between 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) [10, 8]:

$$\mathbf{V}(t) = \mathbf{E}(t) + \mathbf{K} * \mathbf{V}(t) \qquad (2)$$

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

$$[\mathbf{K} * \mathbf{V}(t)]_{ij} = \sum_k \int_0^t dK_{ik}(y)\, V_{kj}(t-y) \qquad (3)$$

Equation (2) implies that the analysis of the whole process can be decomposed into the analysis of the marking process between any two successive regeneration points. The restriction of $\mathcal{M}(t)$ between two successive regeneration points is referred to as the subordinated process.

A transition tr_g is associated with a memory variable a_g [1]. a_g is a functional that depends on the time during which tr_g has been enabled and keeps track of the amount of the elapsed time. The functional dependence of the memory variable on the past enabling time of the transition is named the *memory policy*. The semantics of different memory policies has been discussed in [1] where three alternatives have been proposed referred to respectively as *Resampling memory*, *Enabling memory* and *Age memory policy*.

Resampling memory, and *Enabling memory* policies can be classified as repeat type policies since the age variable is reset when the transition is disabled. On the contrary, the *Age memory* policy is a resume type policy since the memory variable is reset only when the transition fires while its value is maintained if the transition is disabled and then enabled again. In order to track the enabling/disabling condition of a generally distributed transition tr_g, in [5] a binary reward variable was introduced according to the following values:

- $r_k^g = 1$ if tr_g is enabled in marking k;
- $r_k^g = 0$ if tr_g is not enabled in marking k.

In this setting, the value of the memory variable can be computed as the accumulated reward and the firing of a transition can be formulated as a completion time problem [13, 2]. A transition fires when the elapsed time accumulated in the corresponding memory variable reaches a *threshold* γ_g equal to the value of the random firing time initially sampled from its cdf. Therefore, in order to completely define the firing conditions of a transition tr_g at a given time t, two elements must be known: the value of the memory variable a_g at time t, and the value of the threshold γ_g.

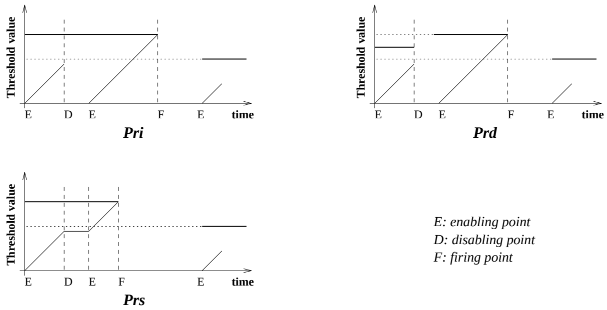


Figure 1: Pictorial representation of different firing time sampling policies

All the previous literature on *MRSPN* [9, 3], was based on the semantics proposed in [1] where it was implicitly assumed that in connection with both repeat type policies (*resampling* and *enabling*), the threshold γ_g is resampled each time the memory variable is reset (either because the corresponding transition fires or is disabled). The resulting *MRSPN* model cannot keep memory of the threshold value (firing time) of any transition beyond its current enabling period. According to this semantics, the repeat memory policies is suited to represent a *preemptive repeat different (prd)* execution mechanism in which the threshold value is sampled *each time* the transition fires or is disabled. The age memory policy is suited to represent a *preemptive resume (prs)* execution mechanism where the threshold value is sampled *only when* the transition fires, and, if the transition is disabled before the threshold is reached, then the value of the age variable is maintained.

This modeling framework was, however, inconsistent with the *pri* preemption policy. If a *pri* policy needs to be modeled, the threshold value must be maintained identical across successive enabling/disabling cycles, until the transition fires.

Figure 1 gives a pictorial description of the introduced firing time sampling strategies. In the picture, we indicate with *E*, *D*, *F* the enabling, disabling and firing time points of a transition, respectively. In the *prd* and *prs* cases the memory variable and the memory policy completely define the firing process of the transition. In the *pri* case, instead, the knowledge of these quantities is not enough, because, the value of the previously sampled threshold must be remembered after the transition is disabled (time point *D*).

In order to cover this case, the rule by which the threshold of a transition has to be sampled must be specified. The *sampling* policy is relevant only in connection with repeat type policies, and we consider the two classical alternatives [13]:

- *prd_sampling* - the threshold (firing time) is sampled each time the age variable is reset;
- *pri_sampling* - the threshold (firing time) is sampled only after the firing of the transition.

The combination of the memory policy with the sampling policy completely specifies the execution of the net. At the entrance in a new tangible marking, the completion time is computed for each enabled transition, given the memory variable and the sampled threshold. The transition with minimal completion time is the one which fires.

Definition 1 - *A transition is* dormant *in those markings in which the corresponding age variable is equal to zero and is* active *in those markings in which the age variable is greater than zero. The* activity cycle *of a transition is the period of time in which a transition is active between two dormant periods.*

Definition 2 - *A transition is* sampled *if the threshold value of its random firing time has already been set up. The* sampled cycle *of a transition is the period of time during which the threshold value is not resampled.*

For a *prd_sampling* policy, sampled and activity cycles are coincident. For a *pri_sampling* policy sampled and activity cycles are different: the sampled cycle is the interval of time between the first time the transition is enabled after it was fired, and the time instant in which it fires again. The sampled cycle strictly contains the activity cycle. A transition can be dormant but its threshold can be sampled, thus causing a conditional dependence in the underlying marking process.

In either cases, the definition of the *sampling* policies assures that at the time point in which a threshold is resampled, the memory variable is zero. In the light of the previous discussion, we can particularize Proposition 1 by stating that a regeneration time point τ_n^* is the epoch of entrance in a marking M_n in which all the memory variables are zero and all the thresholds are not sampled.

The *prd_sampling* policy is the one implicitly assumed in the previous literature and its behavior has been completely characterized in [5]. The combination of the repeat type memory policies with the *pri_sampling* policy is described in the following:

- *pri_Resampling Memory* - The activity cycle ends at the first firing of any transition (including tr_g itself), while the sampled cycle ends only when tr_g fires. Thus the same sampled value is maintained identical over successive activity cycles.

- *pri_Enabling Memory* - The activity cycle starts as soon as tr_g becomes enabled, and ends either when tr_g fires, or when it becomes disabled by the firing of a competitive transition. The sampled cycles ends only when tr_g fires.

The above features can be compared with *prd* and *prs* policies in the following table:

Memory policy	Resam. prd	Enab. prd	Age (prs)	Resam. pri	Enab. pri
end act. cycle	every firing	firing / disabl. tr_g	firing of tr_g	every firing	firing / disabl. tr_g
sampled cycle end	every firing	firing / disabl. tr_g	firing of tr_g	firing of tr_g	firing of tr_g
reach. markings	init. reg. marking	markings in ena. subset	any reach. marking	any reach. marking	any reach. marking
memory variable	contin. increas.	contin. increas.	increas. / constant	increas. & restart	increas. & restart

Table I - Characterization of the activity cycle of transition tr_g

For a transition with exponentially distributed firing time the *prd_Resampling Memory*, the *prd_Enabling Memory* and the *Age Memory* policies have the same effect, due to the memoryless property. We denote these transitions as EXP transitions. Instead, even for exponential firing times the *pri* policies behave differently from the corresponding *prd* ones. The *pri_Resampling Memory* and the *pri_Enabling Memory* destroy the Markov property (except at very special sampling points) due to the further requirement that the threshold must be remembered. In the following we indicate as general (GEN) transitions both the generally distributed transitions (including the deteministic ones) and the exponentially distributed transitions of *pri* type.

3 Transient analysis of pri enabling type transition

In order to deal with a class of solvable models, we focus on *MRSPN* with non overlapping sampled cycles.

Definition 3 - *Sampled cycles are non-overlapping if there exists a dominant transition whose sampled cycle strictly contains the sampled cycles of all the active transitions.*

Definition 4 - *A MRSPN with non-overlapping sampled 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 sampled cycle of the dominant transition.*

Definition 4, includes the possibility that the sampled cycles of GEN transitions are completely contained into the sampled cycle of the dominant one, hence allowing the simultaneous enabling of different GEN transitions inside the same subordinated process. However, an analytic derivation is possible if the subordinated process is restricted to be a semi-Markov reward process. Assuming that a dominant transition exists, the *prd* enabling type and *age* type dominant transition cases have been addressed for the first time in [8, 5], respectively.

In the following we concentrate on the *pri_Enabling Memory* type dominant transition case and provide the transient analysis solution, assuming a subordinated semi-Markov reward process. The described method includes the analysis of the *pri_Resampling Memory* dominant transition as a special case.

Let us suppose that a regeneration period starts at time $t = 0$ from marking i and is dominated by a transition tr_g with memory variable a_g and random firing time γ_g. Transition tr_g is *pri_Enabling memory*, so that the next regeneration point is the firing time of tr_g itself. The process subordinated to the dominant transition is a semi-Markov process. Since the global and local kernels $\mathbf{K}(t)$ and $\mathbf{E}(t)$ can be evaluated row by row, given the above assumptions, we provide an analytical procedure for determining the non-zero entries of the i-th row.

To better understand the developed mathematical formalism, we summarize the notation:

- Ω: reachability set $\mathcal{RS}(M_0)$;
- n: cardinality of the reachability set;
- $\underline{r}_i$: vector grouping the reward rates associated to tr_g during its sampled cycle;
- R^i: subset of Ω grouping the states reachable from state i inside the sampled cycle of tr_g in which tr_g is enabled: for any $k \in R^i$, the reward rate is equal to 1 and a_g is stricly increasing;
- h: cardinality of R^i;
- R^{ci}: subset of Ω in which tr_g is not enabled, but still sampled: for any $k \in R^{ci}$, the reward rate is equal to 0 and a_g is not increasing;
- m: cardinality of R^{ci};
- R^{si}: subset of Ω in which tr_g is not enabled and not sampled: $\Omega = R^i + R^{ci} + R^{si}$;
- $Z^i(t)$: right-continuous subordinated semi-Markov process defined over $R^i + R^{ci}$;
- $\mathbf{Q}^i(t) = [q^i_{k\ell}(t)]$: kernel of the subordinated semi-Markov process;
- w: threshold value sampled from the firing time r.v. γ_g associated with transition tr_g.

$Z^i(t)$ starts at time $t = 0$ in marking M_i with probability 1, 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 except entry i, which equals 1).

For notational convenience we renumber the states in Ω so that the states numbered $1, 2, \ldots, h$ belong to the subset R^i, in which the dominant *pri* GEN transition tr_g is enabled and the states numbered $h+1, h+2, \ldots, m+h$ belong to R^{ci} in which tr_g is disabled. By this ordering of states $\mathbf{Q}^i(t)$ can be partitioned into the following submatrices $\mathbf{Q}^i(t) = \begin{array}{|c|c|}\hline \mathbf{Q_1}(t) & \mathbf{Q_2}(t) \\ \hline \mathbf{Q_3}(t) & \mathbf{Q_4}(t) \\ \hline \end{array}$ where $\mathbf{Q_1}(t)$ describes the transitions inside R^i, $\mathbf{Q_2}(t)$ from R^i to R^{ci}, $\mathbf{Q_3}(t)$ from R^{ci} to R^i and $\mathbf{Q_4}(t)$ inside R^{ci}. We denote by H the time duration until the first embedded time point in $Z^i(t)$ starting from state k at time $t=0$, and by $Q^i_k(t)$ the cdf of H ($Q^i_k(t) = \sum_{j=1}^{h+m} Q^i_{kj}(t)$).

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

$$\begin{aligned} P^i_{k\ell}(t,w) &= Pr\{Z^i(t) = \ell\,,\ \tau_1^* > t \\ &\qquad \mid Z^i(0) = k\,,\ \gamma_g = w\} \end{aligned} \tag{4}$$

$$\begin{aligned} F^i_{k\ell}(t,w) &= Pr\{Z^i(\tau_1^{*-}) = \ell\,,\ \tau_1^* \le t,\ tr_g \text{ fires} \\ &\qquad \mid Z^i(0) = k\,,\ \gamma_g = w\} \end{aligned} \tag{5}$$

$$\begin{aligned} \Delta^i_{k\ell} &= Pr\{\text{ next tangible marking is } \ell \\ &\qquad \mid \text{current marking is } k,\ tr_g \text{ fires }\} \end{aligned} \tag{6}$$

By the above definitions

- $P^i_{k\ell}(t,w)$ is the probability of being in state ℓ at time t before absorption at the barrier w, starting in state k at $t=0$.
- $F^i_{k\ell}(t,w)$ is the probability that tr_g fires from state ℓ (hitting the absorbing barrier w in ℓ) before t, starting in state k at $t=0$. For $\ell \in R^{ci}$, $F^i_{k\ell}(t,w) = 0$.
- $\mathbf{\Delta}^i$ 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 $r^i_k = 1$).

From (5) and (6), it follows that for any t:

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

Given that $G_g(w)$ is the cumulative distribution function of the firing time r.v. γ_g associated with 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 $\mathbf{P}^i(t,w)$ and $\mathbf{F}^i(t,w)$:

$$\begin{aligned} K_{(ij)}(t) &= \int_{w=0}^{\infty} \sum_{k \in R^i} F^i_{ik}(t,w)\, \Delta^i_{kj}\; dG_g(w) \\ E_{(ij)}(t) &= \int_{w=0}^{\infty} P^i_{ij}(t,w)\, dG_g(w) \end{aligned} \tag{7}$$

where the notation $K_{(ij)}(t)$ and $E_{(ij)}(t)$ refers to the modified numbering of the states.

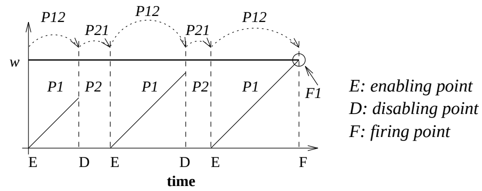


Figure 2: A sampled path of the subordinated process $Z(t)$

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

3.1 Derivation of $\mathbf{P}(t,w)$ and $\mathbf{F}(t,w)$

A regeneration period of a *pri_Enabling Memory* type transition (tr_g) starts from a state (k) in which tr_g is enabled ($k \in R$) and completes when tr_g fires. The reward rate is equal to 1 for all the states inside R and the age variable a_g is continuously increasing. a_g is reset to 0 when $Z(t)$ exits R to enter R^c.

The subordinated process consists of a random number of unsuccessful activity cycles inside R, each one followed by period inside R^c in which tr_g is disabled, finally concluded by a successful activity cycle, at the end of which tr_g fires. A subordinated process of this kind can be considered as a *MRSPN* with a dominant deterministic transition of *prd_Enabling Memory* type whose firing time equals the sampled work requirement of tr_g.

Figure 2 shows a possible realization of $Z(t)$, assuming that the sampled period starts at time $t=0$ and tr_g is enabled. The following cases can occur:

- the process is inside an active cycle R;
- tr_g has already fired;
- tr_g is disabled and $Z(t)$ enters R^c;
- $Z(t)$ is in R^c ;
- $Z(t)$ leaves R^c and enters R (tr_g is enabled again).

The subordinated *MRGP* can thus be described by the following matrix functions, where T_1 denotes the time point until $Z(t)$ visits R and T_2 denotes the time point until $Z(t)$ visits R^c.

$$\begin{aligned} P1_{k\ell}(t,w) &= Pr\{Z(t) = \ell \in R\,,\ \tau_1^* > t\,,\ T_1 > t \\ &\qquad \mid Z(0) = k \in R\,,\ \gamma_g = w\} \end{aligned} \tag{8}$$

$$\begin{aligned} F1_{k\ell}(t,w) &= Pr\{Z(\tau_1^{*-}) = \ell \in R\,,\ \tau_1^* \le t\,,\ T_1 > \tau_1^* \\ &\qquad \mid Z(0) = k \in R\,,\ \gamma_g = w\} \end{aligned} \tag{9}$$

$$P12_{k\ell}(t,w) \;=\; Pr\{Z(T_1)=\ell\in R^c\,,\,\tau_1^* > T_1\,,\,T_1<t \;|\; Z(0)=k\in R\,,\,\gamma_g = w\} \tag{10}$$

$$P2_{k\ell}(t) \;=\; Pr\{Z(t)=\ell\in R^c\,,\,T_2>t \,|\, Z(0)=k\in R^c\} \tag{11}$$

$$P21_{k\ell}(t) \;=\; Pr\{Z(T_2)=\ell\in R\,,\,T_2<t \,|\, Z(0)=k\in R^c\} \tag{12}$$

By the above definitions it follows that:

- $P1_{k\ell}(t,w)$ is the probability of being in state $\ell\in R$ at time t before absorption at the barrier w or leaving R, starting in state $k\in R$ at $t=0$.
- $F1_{k\ell}(t,w)$ is the probability that tr_g fires from state $\ell\in R$ before t, suppose that the subordinated process never left R up to t, starting in state $k\in R$ at $t=0$.
- $P12_{k\ell}(t,w)$ is the probability that the subordinated process left R before time t and before absorption at the barrier w to reach state $\ell\in R^c$ (i.e. ℓ is the first visited state once entered R^c), assuming to start in state $k\in R$ at $t=0$.
- $P2_{k\ell}(t)$ is the probability of being in state $\ell\in R^c$ at time t before leaving R^c, starting in state $k\in R^c$ at $t=0$.
- $P21_{k\ell}(t)$ is the probability that the subordinated process left R^c before time t and ℓ is the first visited state in R, starting in state $k\in R^c$ at $t=0$.

According to the previous definitions, the following equalities hold:

$$P1_{k\ell}(t,w)+P12_{k\ell}(t,w)+F1_{k\ell}(t,w)=1$$

$$P2_{k\ell}(t)+P21_{k\ell}(t)=1$$

Theorem 1 - The LST transform of the matrix function $\mathbf{P}(t,w)$ satisfies the following equation:

$$\mathbf{P}^{\sim}(s,w) \;=\; \left[\mathbf{I} - \mathbf{P12}^{\sim}(s,w)\,\mathbf{P21}^{\sim}(s)\right]^{-1} \left[\mathbf{P1}^{\sim}(s,w)+\mathbf{P12}^{\sim}(s,w)\,\mathbf{P2}^{\sim}(s)\right] \tag{13}$$

where the matrix functions in the r.h.s are derived in the following lemmas.

Lemma 1 - The double *LST-LT* transform of the probability function $P1_{k\ell}(t,w)$ satisfies the following equation:

$$P1^{\sim *}_{k\ell}(s,v) \;=\; \delta_{k\ell}\,\frac{s\,[1\,-\,Q^{\sim}_k(s\,+\,v)]}{v(s\,+\,v)} \;+\; \sum_{u\in R} Q^{\sim}_{ku}(s\,+\,v)\,P1^{\sim *}_{u\ell}(s,v) \tag{14}$$

Proof of Lemma 1 - The proof of the lemma follows the same procedural line developed in [13, 2] for the analysis of the distribution of the completion time in a semi-Markov reward process. Conditioning on $H=h$, and $\gamma_g=w$ let us define:

$$P1_{k\ell}(t,w\,|\,H=h \;=\; \begin{cases} \delta_{k\ell}\,[U(t)\,-\,U(t-w)] \\ \quad \text{if}: h\,\geq\,w \\ \\ \delta_{k\ell}\,[U(t)-U(t-h)]\,+ \\ \displaystyle\sum_{u\in R}\frac{dQ_{ku}(h)}{dQ_k(h)}\,P1_{u\ell}(t-h,w-h) \\ \\ \quad \text{if}: h\,<\,w \end{cases} \tag{15}$$

where $U(t)$ denotes the unit step function.

If $h\,<\,w$ then a transition occurs to state u with probability $dQ_{ku}(h)/dQ_k(h)$ and the residual firing time $(w\,-\,h)$ should be accomplished starting from state u at time $(t\,-\,h)$. Taking the LST transform with respect to t (denoting the transform variable by s), the LT transform with respect to w (denoting the transform variable by v) of (15) and unconditioning with respect to H, (15) becomes (15). □

Lemma 2 - The double *LST-LT* transform of the probability function $P12_{k\ell}(t,w)$ satisfies the following equation:

$$P12^{\sim *}_{k\ell}(s,v) = \frac{1}{v}Q^{\sim}_{k\ell}(s+v)\,+\sum_{u\in R} Q^{\sim}_{ku}(s+v)\,P12^{\sim *}_{u\ell}(s,v) \tag{16}$$

Proof of Lemma 2 - Conditioning on $H=h$, and $\gamma_g=w$ let us define:

$$P12_{k\ell}(t,w\,|\,H=h) \;=\; \begin{cases} 0 \qquad\qquad \text{if}: h\,\geq\,w \\ \\ \displaystyle\frac{dQ_{k\ell}(h)}{dQ_k(h)}\,U(t-h)\,+ \\ \displaystyle\sum_{u\in R}\frac{dQ_{ku}(h)}{dQ_k(h)}\,P12_{u\ell}(t-h,w-h) \\ \qquad\qquad \text{if}: h\,<\,w \end{cases} \tag{17}$$

The derivation of $\mathbf{P12}(t,w)$ based on (17) follows the same pattern [17] as for $\mathbf{P1}(t,w)$ in *Lemma 1.* □

Lemma 3 - The LST transform of the probability function $P2_{k\ell}(t)$ satisfies the following equation:

$$P2^{\sim}_{k\ell}(s) \;=\; \delta_{k\ell}\,[1-Q^{\sim}_k(s)] \;+\; \sum_{u\in R^c} P2^{\sim}_{u\ell}(s)\,Q^{\sim}_{ku}(s) \tag{18}$$

Proof of Lemma 3 - Lemma 3 can be directly derived from Lemma 1, by substituting subset R with R^c and assuming a threshold w equal to infinity. □

Lemma 4 - The LST transform of the probability function $P21_{k\ell}(t)$ satisfies the following equation:

$$P21^{\sim}_{k\ell}(s) \;=\; Q^{\sim}_{k\ell}(s) \;+\; \sum_{u\in R^c} Q^{\sim}_{ku}(s)\,P21^{\sim}_{u\ell}(s) \quad (19)$$

Proof of Lemma 4 - Lemma 4 can be directly derived from Lemma 2, by substituting subset R with R^c and assuming a threshold w equal to infinity. □
Proof of Theorem 1 - The event that the process is resident in R at time t before firing of tr_g can be decomposed into the mutually exclusive events that the process is resident in R continuously from $t = 0$ or after 1, 2, ... passages through R^c (see Figure 2); hence,

$$\begin{aligned}\mathbf{P}^{\sim}(s,w) \;=\;& \mathbf{P1}^{\sim}(s,w) + \mathbf{P12}^{\sim}(s,w)\,\mathbf{P2}^{\sim}(s) + \\ & \mathbf{P12}^{\sim}(s,w)\,\mathbf{P21}^{\sim}(s)\cdot \\ & [\mathbf{P1}^{\sim}(s,w) + \mathbf{P12}^{\sim}(s,w)\,\mathbf{P2}^{\sim}(s)] + \\ & [\mathbf{P12}^{\sim}(s,w)\,\mathbf{P21}^{\sim}(s)]^2\cdot \\ & [\mathbf{P1}^{\sim}(s,w) + \mathbf{P12}^{\sim}(s,w)\,\mathbf{P2}^{\sim}(s)] \\ & + \ldots \qquad (20) \\ =\;& \sum_{u=0}^{\infty} [\mathbf{P12}^{\sim}(s,w)\,\mathbf{P21}^{\sim}(s)]^u \cdot \\ & [\mathbf{P1}^{\sim}(s,w) + \mathbf{P12}^{\sim}(s,w)\,\mathbf{P2}^{\sim}(s)] \qquad (21)\end{aligned}$$

The expression (21) is obtained by applying $\sum_{i=0}^{\infty} \mathbf{M}^i = [\mathbf{I} - \mathbf{M}]^{-1}$. □

Since only the *pri_Resapmling* and *pri_Enabling* transitions can have more than one activity cycle in their sampled cycles, their analysis is more complicated than the analysis of all the other cases, because the arbitrary number of sampled cycles up to the firing of the transition has to be considered. Instead the analysis of the other general transitions requires only the evaluation of one sampled cycle, and only the matrix functions like $\mathbf{P1}(t,w)$ and $\mathbf{F1}(t,w)$ has to be evaluated.

Theorem 2 - The *LST* transform of the firing probability matrix $\mathbf{F}(t,w)$ satisfies the following equation:

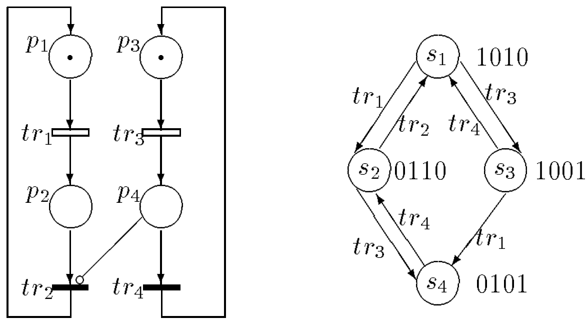
$$\mathbf{F}^{\sim}(s,w) \;=\; [\mathbf{I} \;-\; \mathbf{P12}^{\sim}(s,w)\,\mathbf{P21}^{\sim}(s)]^{-1}\,\mathbf{F1}^{\sim}(s,w) \quad (22)$$

Where $\mathbf{F1}(t,w)$ is derived in Lemma 5.

Lemma 5 - The double *LST-LT* transform of the firing probability $F1_{k\ell}(t,w)$ satisfies the following equation:

$$\begin{aligned}F1^{\sim *}_{k\ell}(s,v) \;=\;& \delta_{k\ell}\,\frac{1 - Q^{\sim}_k(s+v)}{s+v} \;+ \\ & \sum_{u\in R} Q^{\sim}_{ku}(s+v)\,F1^{\sim *}_{u\ell}(s,v) \quad (23)\end{aligned}$$

Proof of Lemma 5 - Conditioning on $H = h$ and $\gamma_g = w$, let us define:



a) b)

Figure 3: Preemptive M/G/1/2/2 queue with two classes of customers.

$$F1_{k\ell}(t,w\,|\,H=h) \;=\; \begin{cases} \delta_{k\ell}\,U(t-w) & \text{if}: h \ge w \\ \displaystyle\sum_{u\in R} \frac{dQ_{ku}(h)}{dQ_k(h)}\cdot & \\ F1_{u\ell}(t-h, w-h) & \text{if}: h < w \end{cases} \quad (24)$$

The derivation of the matrix function $\mathbf{F1}(t,w)$ based on (15) follows the same pattern mentioned for the function $\mathbf{P1}(t,w)$ in *Lemma 1*[17]. □
Proof of Theorem 2 - The event that tr_g fires in R at time t can be decomposed into the mutually exclusive events that tr_g fires during the first sojourn in R starting at $t = 0$ or after 1, 2, ... passages through R^c (see Figure 2); hence,

$$\begin{aligned}\mathbf{F}^{\sim}(s,w) \;=\;& \mathbf{F1}^{\sim}(s,w) \;+\; \mathbf{P12}^{\sim}(s,w)\,\mathbf{P21}^{\sim}(s)\mathbf{F1}^{\sim}(s,w) \;+ \\ & [\mathbf{P12}^{\sim}(s,w)\,\mathbf{P21}^{\sim}(s)]^2\,\mathbf{F1}^{\sim}(s,w) \;+\; \ldots \\ =\;& \sum_{u=0}^{\infty} [\mathbf{P12}^{\sim}(s,w)\,\mathbf{P21}^{\sim}(s)]^u\;\; \mathbf{F1}^{\sim}(s,w)\end{aligned}$$

From which the theorem comes.□

4 Preemptive M/G/1/2/2 queue with different customers

The *PN* of Figure 3a models a M/G/1/2/2 queue in which the jobs submitted by customer 2 have higher priority and preempts the jobs submitted by customer 1. The server has a *pri Enabling Memory* service discipline, which means, that the service of a preempted lower priority job starts from the begining, when the server becomes available. Place p_1 (p_3) represents customer 1 (2) thinking, while place p_2 (p_4) represent job 1 (2) 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 and tr_4 are GEN transitions, and represent the completion of service of a job of type 1 or 2, respectively. A *pri* service discipline is modeled by assigning to tr_2 an *pri Enabling Memory* type. 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 (if any) is interrupted. The reachability graph of the *PN* of Figure 3a is in Figure 3b. Under a *pri* service, after completion of the type 2 job, the service of the interrupted type 1 job is restarted, and the same job (with an identical work requirement) has to be completed. From Figure 3b, 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.

Only EXP transitions are enabled in s_1 and the next regeneration states can be either s_2 or s_3 depending whether tr_1 or tr_3 fires first. 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 dominant transition is tr_2 with *pri_Enabling Memory* and the subordinated process is a reward *SMP* (tr_4 is GEN); hence the results of the previous section can be applied for the evaluation of the relevant row of the $\mathbf{E}$(t) and $\mathbf{K}$(t) matrices.

¿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*.

In order to evaluate $\mathbf{V}$(t) we need to compute the matrices $\mathbf{E}$(t) and $\mathbf{K}$(t). The i-th row of these matrices describes the behavior of the process subordinated to a regeneration period starting from state i (if state i can never be a regeneration state, the corresponding row is zero). Their entries corresponding to s_1 and s_3 have already been derived in [5], and we refer to that paper for a comprehensive description of the computational procedure. In the following we provide a detailed description of the new features introduced in this paper: i.e. how to evaluate the elements of $\mathbf{E}$(t) and $\mathbf{K}$(t), when s_2 is as a regeneration state dominated by a *pri* transition.

We denote the cumulative distribution function of the GEN transitions tr_2 and tr_4 by $G_2(t)$ and $G_4(t)$, respectively (and their Laplace-Stieltjes transform by $G_2^\sim(s)$ and $G_4^\sim(s)$), while tr_1 and tr_3 are EXP with parameter λ [4].

tr_2 is of *pri_Enabling Memory type* so that the next regeneration time point is the epoch of firing of tr_2. The subordinated process comprises states s_2 and s_4 and is a semi-Markov process since tr_4 is GEN. For the sake of simplicity we restrict the following analysis only to states s_2 and s_4 reachable during the considered subordinated process, an we renumber the states according to the previous section. In the following, state s_2 is mapped into state 1 and state s_4 into state 2. With this renumbering, the kernel of the semi-Markov process becomes:

$$Q^\sim(s) = \begin{vmatrix} 0 & \dfrac{\lambda}{s+\lambda} \\ G_4^\sim(s) & 0 \end{vmatrix}$$

The reward vector is $\underline{r} = [1, 0]$, and the firing of tr_2 leads to s_1 with probability 1. Applying Lemma 1 and 5 we obtain the following results for the nonzero entries in the double transform domain:

$$P1_{11}^{\sim *}(s,v) = \frac{s}{v(s+v+\lambda)};\ F1_{11}^{\sim *}(s,v) = \frac{1}{s+v+\lambda} \tag{25}$$

after inverse Laplace transforming with respect to v we have:

$$P1_{11}^{\sim}(s,w) = \frac{s\ a}{s+\lambda};\ F1_{11}^{\sim}(s,w) = 1-a \tag{26}$$

where $a = 1 - e^{-(s+\lambda)w}$. Applying Lemma 2:

$$P12_{12}^{\sim *}(s,v) = \frac{\lambda}{v(s+v+\lambda)};\ P12_{12}^{\sim}(s,w) = \frac{\lambda\ a}{s+\lambda} \tag{27}$$

Let us define the matrix functions to describe the subordinated process.

$$\mathbf{P1}^\sim(s,w) = \begin{vmatrix} \dfrac{sa}{s+\lambda} & 0 \\ 0 & 0 \end{vmatrix}$$

$$\mathbf{P12}^\sim(s,w) = \begin{vmatrix} 0 & \dfrac{\lambda s}{s+\lambda} \\ 0 & 0 \end{vmatrix}$$

$$\mathbf{P2}^\sim(s) = \begin{vmatrix} 0 & 0 \\ 0 & 1-G_4^\sim(s) \end{vmatrix}$$

$$\mathbf{P21}^\sim(s) = \begin{vmatrix} 0 & 0 \\ G_4^\sim(s) & 0 \end{vmatrix}$$

$$\mathbf{F1}^\sim(s,w) = \begin{vmatrix} 1-a & 0 \\ 0 & 0 \end{vmatrix}$$

Due to Theorems 1 and 2, the relevant entries (1st row) of the $\mathbf{P}^\sim(s,w)$ and the $\mathbf{F}^\sim(s,w)$ matrices are given by:

$$\begin{aligned} P_{11}^\sim(s,w) &= \frac{\dfrac{s\ a}{s+\lambda}}{1-G_4^\sim(s)\dfrac{\lambda\ a}{s+\lambda}} \\ P_{12}^\sim(s,w) &= \frac{(1-G_4^\sim(s))\dfrac{\lambda\ a}{s+\lambda}}{1-G_4^\sim(s)\dfrac{\lambda\ a}{s+\lambda}} \\ F_{11}^\sim(s,w) &= \frac{1-a}{1-G_4^\sim(s)\dfrac{\lambda\ a}{s+\lambda}} \end{aligned} \tag{28}$$

With respect to the original state space numbering, the nonzero entries in the 2nd row of the *LST* matrix functions $\mathbf{K}^{\sim}(s)$ and $\mathbf{E}^{\sim}(s)$ have the following expression:

$$
\begin{aligned}
E^{\sim}_{s_2 s_2}(s) &= \int_{w=0}^{\infty} P^{\sim}_{11}(s,w)\; dG_2(w) \\
E^{\sim}_{s_2 s_4}(s) &= \int_{w=0}^{\infty} P^{\sim}_{12}(s,w)\; dG_2(w) \\
K^{\sim}_{s_2 s_1}(s) &= \int_{w=0}^{\infty} F^{\sim}_{11}(s,w)\; dG_2(w)
\end{aligned}
\tag{29}
$$

The time domain probabilities are calculated by first deriving $\mathbf{V}^{\sim}(s)$ from (2) using a standard package for symbolic analysis (e.g. MATHEMATICA), and then numerically inverting the resulting *LST* expressions resorting to the Jagerman's method [12]. The plot of the state probabilities versus time together with the steady state results (dotted line) is reported in Figure 4, for the following set of numerical parameters:

- submitting rate $\lambda = 0.5$;
- higher priority customer (transition tr_4) with an exponentially distributed service time with parameter $\mu = 1$ ($G^{\sim}_4(s) = \frac{\mu}{s+\mu}$);
- lower priority customer (transition tr_2) with a service time uniformly distributed between $\alpha = 0.5$ and $\beta = 1.5$ ($G_2(t) = \frac{1}{\beta-\alpha}(U(t-\alpha) - U(t-\beta))$). [1]

5 Conclusion

The semantics of the *SPN* model with generally distributed transitions discussed in [1], provides a natural environment for the definition of a class of analitycally tractable *MRSPN*'s. However, the proposed semantics, for repeat type memory policies, assumes a coincidence between the action of resetting the memory variable of a transition and the action of resampling its firing time threshold. Therefore, only *prd* policies can be modeled. This paper has shown that a *pri* policy can be considered if the instant at which the firing time is resampled is evaluated separately from the instant in which the memory variable is reset. The sampled period of a *pri* transition may contain several activity periods, and is the crucial factor in determining the sequence of the regeneration time points in the marking process. An analytical solution is given when the dominant transition is classified as *pri_Enabling Memry* type with subordinated reward semi-Markov process.

6 Acknowledgements

The authors are thankful to Sachin Garg for the fruitful discussions. A. Bobbio and K. S. Trivedi would like to thank NATO for grant No. CRG.940308. A. Puliafito would like to thank Italian CNR for grant No. 203.15.5. This research was carried out while M. Telek was visiting the Department of Operations Research of the University of North Carolina at Chapel Hill sponsored by the Hungarian OTKA (Grant No. W 015859).

[1] The cases when both the customers have the same exponentially, or uniformly distributed service time caused numerical instabilities with the implemented inverse Laplace transform method.

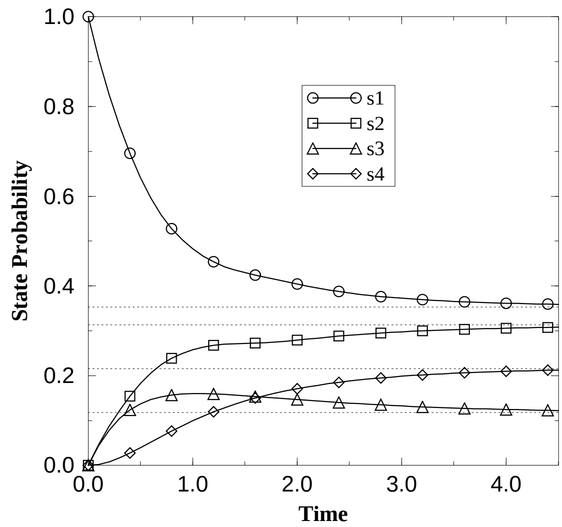


Figure 4: Transient behavior of the state probabilities for the preemptive M/G/1/2/2 system with different customers.

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

[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. 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.

[5] A. Bobbio and M. Telek. Markov regenerative SPN with non-overlapping activity cycles. In *International Computer Performance and Dependability Symposium - IPDS95*, April 1995.

[6] 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.

[7] P.F. Chimento and K.S. Trivedi. The completion time of programs on processors subject to failure and repair. *IEEE Transactions on Computers*, 42:1184–1194, 1993.

[8] Hoon 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. *IEEE Transactions on Software Engineering*, 20:506–515, 1994.

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

[11] 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.

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

[13] 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.

[14] D. Logothetis. *Transient Analysis of Communication Networks*. Ph.D. Thesis, Dept. of Electrical Eng., Duke University, 1994.

[15] 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.

[16] M. Telek, A. Bobbio, L. Jereb, A. Puliafito, and K. Trivedi. Steady state analysis of Markov regenerative SPN with age memory policy. Technical report, Department of Telecommunications - Technical University of Budapest, February 1995.

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