Volume 22 (2), pp. 131–154
http://www.orssa.org.za

ORiON
ISSN 0529-191-X

Modelling $M/G/1$ queueing systems with server vacations using stochastic Petri nets

K Ramanath* P Lakshmi†

Received: 12 November 2005; Revised: 16 July 2006; Accepted: 6 August 2006

Abstract

The theory of non-Markovian stochastic Petri nets is employed in this paper to derive an alternative method for studying the steady state behaviour of the $M/G/1$ vacation queueing system with a limited service discipline. Three types of vacation schemes are considered, and sytems with both a finite population and those with an infinite population (but finite capacity) are considered. Simple numerical examples are also provided to illustrate the functionality of the methods and some useful performance measures for the system are obtained.

Key words: Markov regenerative stochastic Petri nets, vacation queueing models, limited service discipline, single, multiple and hybrid vacation schemes.

1 Introduction

Queueing systems in which the server is sometimes inactive (taking vacations or working elsewhere) while customers are waiting for service may find many applications in the performance modelling of computer and communication systems. For example, in many digital systems, the processor is multiplexed among a number of jobs and is hence not available all the time for a single job type. So, if we take any one job type as a reference point, the processor is alternatively busy handling that job type and absent doing work elsewhere [12].

Server vacation has been used successfully to model polling systems [13], multiclass priority scheduling systems [18], maintenance models [17] and processor failure [2]. For a survey on the use of vacation models, see Doshi [7].

Most of the papers on vacation models assume the source population of the queueing system to be infinite. However, Trivdei *et al.* [11] have studied several vacation models in which they have taken the population to be finite. They assume an exponentially

*Corresponding author: School of Mathematics, Madurai Kamaraj University, Madurai 625021, India, email: `kasturey@rediffmail.com`

†School of Mathematics, Madurai Kamaraj University, Madurai 625021, India

distributed service time. This paper is an extension of their work to the case where the service times are generally distributed.

We obtain Markov regenerative stochastic Petri net (MRSPN) representations of the $M/G/1/N$ queueing systems with server vacations under a limited service discipline. For an excellent introduction to the theory of $M/G/1$ queueing systems, see Gross and Harris [9] and Takagi [14]. The book by Takagi also contains information about queueing systems with server vacations. We consider three types of vacation schemes, namely the multiple vacation scheme, the single vacation scheme and the hybrid vacation scheme. We also show how the representations may be extended to the case where the population is infinite, but the system has a finite capacity.

The chief advantages of an MRSPN representation are:

1. The representation is aesthetically appealing and simple to understand, even for a layman.

2. Several other features of vacation queueing systems may be included in this representation, whose inclusion in other classical models would have led to analytically intractable solutions.

3. A considerable amount of research has been reported over the past two decades to develop software packages for stochastic Petri nets. As far as Markovian stochastic Petri nets are concerned, researchers have succeeded in developing the packages SPNP [15] and GreatSPN [5], which automatically generate the reachability graph and give efficient and reliable numerical solutions. The user only has to fit the model. For non-Markovian SPNs, several packages have been developed and a considerable amount of research is still ongoing in this direction. Two such packages are the WEBSPN [3] and SHARPE [16].

The rest of the paper is organised as follows: In §2 we give a brief outline of the theory of MRSPNs. In §3 we present MRSPN representations for the $M/G/1$ vacation queueing systems with a limited service discipline. We use the notation $M/G/1/V_l$ to denote such queueing systems in what follows. We consider the multiple, single vacation schemes and vacation schemes with a hybrid policy. We take up both the cases of a finite population and an infinite population with a finite system capacity. In §4 we illustrate the representation given in §3 with the help of simple numerical examples. Finally, in §5 we present a few concluding remarks.

2 Markov regenerative stochastic Petri nets

For the basics of the Petri net formalism, see Ajmone-Marsen [1] and German [8]. We begin with the theory of MRSPNs in this section.

Transitions with an exponentially distributed firing time are referred to as EXP transitions. Transitions with a generally distributed firing time are referred to as GEN transitions. If the Petri net contains GEN transitions, the underlying marking process $M(t)$ is not a continuous time Markov chain and it cannot be studied assuming the Markov chain theory.

As originally observed by Choi *et al.* [6], the underlying marking process is a Markov regenerative process under the conditions that:

1. there is at most one GEN transition enabled in any marking, and
2. the firing time value is sampled at the time the transition is enabled; it cannot change until the transition either fires or is disabled; a new firing time is obtained when the transition becomes enabled again.

Definition 1 *A Markov regenerative process (MRGP) is a continuous-time discrete state stochastic process with an embedded sequence of regenerative time points (RTPs), at which the process enjoys the Markov property.* ■

Definition 2 *A Markov regenerative stochastic Petri net (MRSPN) is a stochastic Petri net in which the underlying marking process is a MRGP.* ■

MRSPNs may be studied by using the theory of Markov regenerative processes. Based on the concept of memory in a stochastic Petri net, the regeneration time points may be defined. The memory of a transition in a stochastic Petri net is given by a pair of random variables (a, i). Here a is the age variable which keeps track of the amount of time elapsed since the enabling of the transition. The resampling indicator i is set equal to one whenever the firing time of the transition is to be resampled. Otherwise it is set equal to zero. Thus, a transition has non zero memory at any instant of time only if at least one of the variables a or i is non-zero. In the pre-emptive repeat different policy considered in this paper, the resampling variable and the age variable are reset as zero each time the transition is either disabled or is fired.

Definition 3 *A Regeneration Time Point in the marking process $\{\mathcal{M}(t) : t \geq 0\}$ is an instant of time at which all the transitions in the Petri net have zero memory.* ■

Because of the memoryless property of the MRGP at the RTPs, the analysis of a MRSPN can be split into independant subproblems wherein the subordinated processes between any two consecutive RTPs are examined. The probability functions necessary for the analysis of a MRSPN are commonly referred to as the global and local kernels.

The global kernel is defined as

$$K_{ij}(t) = \text{Prob}\{M(T_1+) = j,\ T_1 \leq t \mid M(0) = i\},$$

where $M(T_1+)$ represents the marking immediately after the end of the regeneration cycle, where $M(0)$ is the initial marking and where T_1 is the duration of the regeneration cycle. The local kernel is defined as

$$E_{ij}(t) = \text{Prob}\{M(t) = j,\ T_1 > t \mid M(0) = i\}.$$

In order to obtain the steady state probabilities of the MRSPN, we need to obtain the steady state probabilities of the embedded Markov chain (EMC) at the RTPs. These are obtained as the solution of $v = vP$, where $P = K(\infty)$ is the matrix whose ij^{th} entry is $K_{ij}(\infty)$. After calculating $v = (v_1, v_2, \cdots)$, we compute

$$\alpha_{ij} = \int_0^{\infty} E_{ij}(t)\mathrm{d}t.$$

Define $\mu_j = \sum_{k \in \Omega} \alpha_{jk}$. Then the steady state probabilities P_j of the MRSPN are given by the expressions

$$P_j = \lim_{t \to \infty} \mathrm{Prob}\{M(t) = j \mid M(0) = i\} = \frac{\sum_{k \in \Omega} \alpha_{kj} v_k}{\sum_{k \in \Omega} \mu_k v_k}.$$

For more details, the reader is referred to Choi *et al.* [6].

3 MRSPN representation of the $M/G/1/V_l$ queueing system

In this section we consider a queueing system with a single service facility which becomes unavailable from time to time. Arrivals to the system are in accordance with a Poisson process with a parameter λ. The service times of the individual customers are independent identically distributed (i.i.d.) random variables with a distribution function $B(t)$. The arrival process is assumed to be independant of the service mechanism. We assume that $\beta(s)$ is the Laplace Stieltjes transform of the service time distribution $B(t)$. The expected service time of a customer in the system is given by $\mu = -\beta^{'}(0)$. The duration of the server vacation is assumed to be exponentially distributed with a parameter γ.

3.1 A multiple vacation scheme

Under the multiple vacation scheme, the server goes on a vacation whenever there are no customers waiting for service in the system, that is, the system is empty. However, it is also allowed to go away after completion of a fixed number of, say k services, even if there are customers present in the system.

3.1.1 Systems with a finite population.

We first consider the case of a finite population of size N. The MRSPN representation of the system is given in Figure 1.

There are five places P_I, P_B, P_s, P_D and P_v, and five transitions t_a, t_v, t_s, S and S_1. The transition t_a whose firing indicates the arrival of a customer from the population P_I to the system is an EXP transition. The transition t_v whose firing represents the end of the server vacation is also an EXP transition. The transition t_s whose firing represents the completion of a service is a GEN transition. The transitions S and S_1 are immediate transitions which fire in zero time.

The place P_D is used to keep a record of the number of service completions. As soon as k services are completed, *i.e.* k tokens are deposited in P_D, further service is suspended

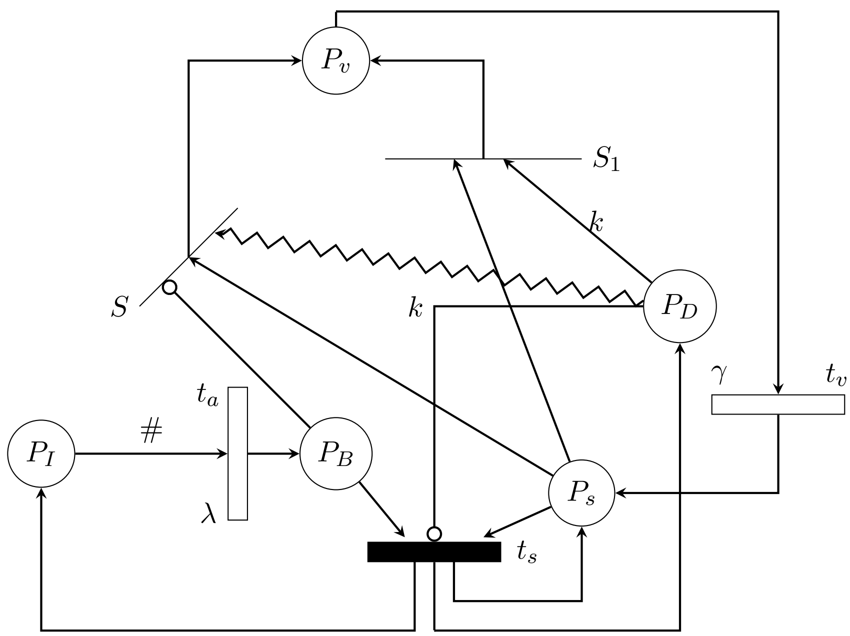

Figure 1: *MRSPN model of the multiple vacation $M/G/1/N/V_l$ queueing system.*

in the system. This is represented by an inhibitor arc of multiplicity k from the place P_D to the GEN transition t_s. The server then proceeds on vacation which is effected by the immediate transition S_1. An inhibitor arc from P_B to the transition S prevents the server from proceeding on a vacation when there is even a single customer in the system. The firing of S flushes out all the tokens in P_D, so that the system can start afresh, after the return of the server from the vacation. The marking

$$M(t) = \{M(P_I), M(P_B), M(P_s), M(P_D), M(P_v)\}$$

represents the state of the system at any instant of time t, where $M(P_I), M(P_B), M(P_s)$, $M(P_D), M(P_v)$ are the number of tokens in P_I, P_B, P_s, P_D, P_v respectively, at time t. Thus, the initial marking of the system, at time $t = 0$, is $M(0) = (N, 0, 1, 0, 0)$. This however, is a vanishing marking and is merged with its successor marking. Thus, $M(0) = (N, 0, 0, 0, 1)$.

When the number of tokens in P_I is $N - r$, the number of tokens in P_B is r. Again, when the number of tokens in P_s is 1, the number of tokens in P_v is 0. It is therefore enough to signify a marking by the number of tokens in the places $P_B,\ P_s$ and P_D. Hence

$$M(t) = \{M(P_B), M(P_s), M(P_D)\}$$

represents the marking at any instant of time t. The state space of the system is

$$\Omega = \{(r, 0, 0) : 0 \leq r \leq N\} \cup \{(r, 1, n) : 1 \leq r \leq N,\ 0 \leq n \leq k - 1\}.$$

Since there is only one GEN transition in the model, the associated marking process is an MRGP. The transitions $t_a,\ t_s$ and t_v are not conflicting. Thus, the question of pre-emption does not arise. The results of Choi *et al.* [6] may therefore be used to obtain the transient and steady state probabilities of the system.

The RTPs are defined as follows:

1. Let $T_0 = 0$.
2. If, at the n^{th} RTP T_n, the system is in the state $(r, 0, 0)$, the GEN transition t_s, is not enabled. Therefore T_{n+1} is the time at which either of the two EXP transitions t_a or t_v fires.
3. If, at the n^{th} RTP T_n, the system is in the state $(r, 1, n)$ with $1 \leq r \leq N$, $0 \leq n \leq k-1$, the GEN transition t_s is enabled. The next RTP T_{n+1} is the time at which t_s fires.

In this case, the subordinated process is a *continuous time Markov chain* (CTMC) with state space $\Omega(r, 1, n) = \{(s, 1, n) : r \leq s \leq N\}$. The transient and steady state solution of this CTMC are obtained by determining the rate matrix, which is known as the infinitisimal generator (see Medhi [10, p. 195]). The infinitesimal generator of the subordinated CTMC is given by $Q(r, 1, n) = [q_{ij}]$, where

$$q_{ij} = \begin{cases} 0 & \text{if } j \neq i, i+1, \\ (N-i-r+1)\lambda & \text{if } j = i, \\ -(N-i-r+1)\lambda & \text{if } j = i+1 \end{cases}$$

The expressions for the local kernel and the global kernel may now be obtained.

Local kernel

When the regeneration cycle starts in the state $(r, 0, 0)$ with $0 \leq r \leq N$,

$$\begin{aligned} E_{(r,0,0)\ (r',0,0)}(t) &= 0, \text{ for } r \neq r', \\ E_{(r,0,0)\ (r',1,n)}(t) &= 0, \text{ for } 1 \leq r' \leq N,\ 0 \leq n \leq k-1, \\ E_{(r,0,0)\ (r,0,0)}(t) &= e^{-[(N-r)\lambda+\gamma]t}. \end{aligned}$$

When the regeneration cycle starts in the state $(r, 1, n)$, with $1 \leq r \leq N$, and $0 \leq n \leq k-1$,

$$\begin{aligned} E_{(r,1,n)\ (r',0,0)}(t) &= 0, \text{ for } 0 \leq r' \leq N, \\ E_{(r,1,n)\ (r',1,n')}(t) &= 0, \text{ for } (r', 1, n') \notin \Omega(r, 1, n), \\ E_{(r,1,n)\ (r',1,n)}(t) &= [e^{Q(r,1,n)t}]_{(1,r'-r+1)}(1 - B(t)), \text{ for } r \leq r' \leq N. \end{aligned}$$

Here, $[e^{Q(r,1,n)t}]_{(1,r'-r+1)}$ denotes the element in the first row and $(r' - r + 1)^{th}$ column of the matrix $e^{Q(r,1,n)t}$.

Global kernel

When the regeneration cycle starts in the state $(r, 0, 0)$ with $0 < r < N$,

$$\begin{aligned}
K_{(r,0,0)\ (r',0,0)}(t) &= 0,\ \text{for } r' \neq r+1,\\
K_{(r,0,0)\ (r+1,0,0)}(t) &= \frac{(N-r)\lambda}{(N-r)\lambda+\gamma}(1-e^{-[(N-r)\lambda+\gamma]t}),\\
K_{(r,0,0)\ (r',1,n)}(t) &= 0,\ \text{for } (r',1,n) \neq (r,1,0),\\
K_{(r,0,0)\ (r,1,0)}(t) &= \frac{\gamma}{(N-r)\lambda+\gamma}(1-e^{-[(N-r)\lambda+\gamma]t}).
\end{aligned}$$

When $r=0$,

$$\begin{aligned}
K_{(0,0,0)\ (r',0,0)}(t) &= 0,\ \text{for } r' \neq 0,1,\\
K_{(0,00)\ (0,0,0)}(t) &= \frac{\gamma}{N\lambda+\gamma}(1-e^{-[N\lambda+\gamma]t}),\\
K_{(0,0,0)\ (1,0,0)}(t) &= \frac{N\lambda}{N\lambda+\gamma}(1-e^{-[N\lambda+\gamma]t}),\\
K_{(0,0,0)\ (r,1,n)}(t) &= 0,\ \text{for } 1 \leq r \leq N,\ 0 \leq n \leq k-1.
\end{aligned}$$

When $r=N$,

$$\begin{aligned}
K_{(N,0,0)\ (r,0,0)}(t) &= 0,\ \text{for } 0 \leq r \leq N,\\
K_{(N,0,0)\ (r,1,n)}(t) &= 0,\ \text{for } (r,1,n) \neq (N,1,0),\\
K_{(N,0,0)\ (N,1,0)}(t) &= 1-e^{-\gamma t}.
\end{aligned}$$

When the regeneration cycle begins in the state $(1,\ 1,\ n)$ with $0 \leq n \leq k-1$,

$$K_{(1,1,n)(0,0,0)}(t) = \int_0^t [e^{Q(1,\ 1,\ n)x}]_{(1,\ 1)} \mathrm{d}B(x).$$

For $0 \leq n \leq k-2$,

$$\begin{aligned}
K_{(1,1,n)(r,0,0)}(t) &= 0,\ \text{for } r \neq 0,\\
K_{(1,1,n)(r,1,n')}(t) &= 0,\ \text{for } n' \neq n+1,\\
K_{(1,1,n)(r,1,n+1)}(t) &= \int_0^t [e^{Q(1,\ 1,\ n)x}]_{(1,\ r+1)} \mathrm{d}B(x),\ \text{for } 1 \leq r < N,\\
K_{(1,1,n)(N,1,n+1)}(t) &= 0.
\end{aligned}$$

For $n=k-1$,

$$\begin{aligned}
K_{(1,1,k-1)(r,0,0)}(t) &= \int_0^t [e^{Q(1,\ 1,\ k-1)x}]_{(1,\ r+1)} \mathrm{d}B(x),\ \text{for } 1 \leq r \leq N-1,\\
K_{(1,1,k-1)(r',1,n')}(t) &= 0,\ \text{for } 1 \leq r' \leq N,\ 0 \leq n^{'} \leq k-1,\\
K_{(1,1,k-1)(N,0,0)}(t) &= 0.
\end{aligned}$$

When the regeneration cycle begins in the state $(r,1,n)$ with $1 < r \leq N$, $0 \leq n < k-1$,

$$\begin{aligned}
K_{(r,1,n)(r',0,0)}(t) &= 0,\ \text{for } 0 \leq r' \leq N,\\
K_{(r,1,n)(r',1,n')}(t) &= 0,\ \text{for } n' \neq n+1 \text{ or } r' < r-1 \text{ or } r' = N,\\
K_{(r,1,n)(r',1,n+1)}(t) &= \int_0^t [e^{Q(r,\ 1,\ n)x}]_{(1,r'-r+2)} \mathrm{d}B(x),\ \text{for } r-1 \leq r' < N.
\end{aligned}$$

When $n = k-1$,

$$\begin{aligned}
K_{(r,1,k-1)(r',0,0)}(t) &= 0, \text{ for } r' < r-1 \text{ or } r' = N, \\
K_{(r,1,k-1)(r',0,0)}(t) &= \int_0^t [e^{Q(r,\ 1,\ k-1)x}]_{(1,r'-r+2)} \mathrm{d}B(x), \text{ for } r-1 \leq r' < N, \\
K_{(r,1,k-1)(r',1,n')}(t) &= 0, \text{ for } 1 \leq r' \leq N,\ 0 \leq n' \leq k-1.
\end{aligned}$$

Assuming ergodicity of the system, the steady state probabilities $P_{(r,\ 0,\ 0)}$ for $0 \leq r \leq N$ and $P_{(r,1,n)}$ for $1 \leq r \leq N$ and $0 \leq n \leq k-1$ may be derived using the formulae given in §2. The performance measures of the system are the effective arrival rate (denoted by λ_{eff}), the expected number of customers in the system (denoted by L_s), the expected waiting time of a customer in the system (denoted by W_s), the expected number of vacations per unit time (denoted by μ_v) and the fraction of busy time during the interval $[0,t]$ of the server (denoted by W_t). It follows that

$$\begin{aligned}
\lambda_{\text{eff}} &= \lambda \left\{ \sum_{r=1}^{N-1} (N-r) \left[P_{(r,0,0)} + \sum_{n=0}^{k-1} P_{(r,1,n)} \right] + NP_{(0,0,0)} \right\} \text{ and} \\
L_s &= \sum_{r=1}^{N} r \left\{ P_{(r,0,0)} + \sum_{n=0}^{k-1} P_{(r,1,n)} \right\}.
\end{aligned}$$

Furthermore by Little's formula, (see, for example, Gross and Harris [9, p. 79]),

$$\begin{aligned}
W_s &= \frac{L_s}{\lambda_{\text{eff}}}, \\
\mu_v &= \gamma \sum_{r=0}^{N} P_{(r,0,0)} \text{ and} \\
W_t &= t \sum_{r=1}^{N} \sum_{n=0}^{k-1} P_{(r,1,n)}.
\end{aligned}$$

3.1.2 Systems with a finite capacity

In this section, we assume that the population may be infinite, but that the capacity, denoted by N, is finite. Figure 1 may also be used as the MRSPN representation of this queueing model, except for two changes:

1. An inhibitor arc from the place P_B to the arrival transition t_a of multiplicity N prevents further arrivals when there are N customers in the system.
2. Since the source population is infinite, the arrival rate is no longer marking dependant. Thus, the symbol # placed next to t_a in Figure 1 is now not necessary.

Again, since there is only one GEN transition and no pre-emption occurs, all the conditions necessary for the marking process $\{M(t) : t \geq 0\}$ to be a MRGP, are satisfied. The RTPs and state space Ω are as in §3.1.1. The subordinated process in a regeneration cycle starting in the state $(r,1,n)$ with $1 \le r \le N$, $0 \le n \le k-1$ is a CTMC with state space $\Omega(r,1,n)$, where $\Omega(r,1,n)$ is as defined in §3.1.1. Here, the infinitesimal generator is $Q(r,1,n) = [q_{ij}]$, where

$$q_{ij} = \begin{cases} 0 & \text{if } j \neq i,\ i+1, \\ -\lambda & \text{if } j = i, \\ \lambda & \text{if } j = i+1, \end{cases}$$

where $1 \le i \le N-r$ and the $(N-r+1)^{th}$ row contains only zero's. Expressions for the local and global kernels may now be obtained.

Local kernel
When the regeneration cycle starts in the state $(r,0,0)$ with $0 \le r < N$,

$$\begin{aligned} E_{(r,0,0)\ (r',0,0)}(t) &= 0, \text{ for } r \neq r', \\ E_{(r,0,0)\ (r,0,0)}(t) &= e^{-[\lambda+\gamma]t}, \\ E_{(r,0,0)\ (r',1,n)}(t) &= 0, \text{ for } 1 \le r' \le N,\ 0 \le n \le k-1. \end{aligned}$$

When the regeneration cycle starts in the state $(N,0,0)$,

$$\begin{aligned} E_{(N,0,0)\ (r,0,0)}(t) &= 0, \text{ for } r \neq N, \\ E_{(N,0,0)\ (N,0,0)}(t) &= e^{-\gamma t}, \\ E_{(N,0,0)\ (r,1,n)}(t) &= 0, \text{ for } 1 \le r \le N,\ 0 \le n \le k-1. \end{aligned}$$

When the regeneration cycle starts in the state $(r,1,n)$, the expressions for the local kernel are as in §3.1.1.

Global kernel
If the regeneration cycle starts in the state $(r,0,0)$ with $0 < r < N$,

$$\begin{aligned} K_{(r,0,0)\ (r',0,0)}(t) &= 0, \text{ for } r' \neq r+1, \\ K_{(r,0,0)\ (r+1,0,0)}(t) &= \frac{\lambda}{\lambda+\gamma}(1 - e^{-[\lambda+\gamma]t}), \\ K_{(r,0,0)\ (r',1,n)}(t) &= 0, \text{ for } (r',1,n) \neq (r,1,0), \\ K_{(r,0,0)\ (r,1,0)}(t) &= \frac{\gamma}{\lambda+\gamma}(1 - e^{-[\lambda+\gamma]t}). \end{aligned}$$

When the regeneration cycle begins in the state $(0,0,0)$,

$$\begin{aligned} K_{(0,0,0)\ (r,0,0)}(t) &= 0, \text{ for } r \neq 0,1, \\ K_{(0,0,0)\ (0,0,0)}(t) &= \frac{\gamma}{\lambda+\gamma}(1 - e^{-[\lambda+\gamma]t}), \\ K_{(0,0,0)\ (1,0,0)}(t) &= \frac{\lambda}{\lambda+\gamma}(1 - e^{-[\lambda+\gamma]t}), \\ K_{(0,0,0)\ (r,1,n)}(t) &= 0, \text{ for } 1 \le r \le N,\ 0 \le n \le k-1. \end{aligned}$$

When the regeneration cycle begins in the state $(N, 0, 0)$,

$$\begin{aligned} K_{(N,0,0)\ (r,0,0)}(t) &= 0, \text{ for } 0 \le r \le N, \\ K_{(N,0,0)\ (r,1,n)}(t) &= 0, \text{ for } (r,1,n) \ne (N,1,0), \\ K_{(N,0,0)\ (N,1,0)}(t) &= 1 - e^{-\gamma t}. \end{aligned}$$

When the regeneration cycle begins in the state $(r, 1, n)$ with $1 \le r \le N$, $0 \le n \le k-1$, the expressions for the global kernel are as in §3.1.1.

As in §3.1.1, if $P_{(r,0,0)}$ for $0 \le r \le N$ and $P_{(r,1,n)}$ for $1 \le r \le N$, $0 \le n \le k-1$ denote steady state probabilities of the system, the expressions for the performance measures of the system are the same as in §3.1.1, except that the effective arrival rate is now given by

$$\lambda_{\text{eff}} = \lambda \left[1 - P_{(N,0,0)} - \sum_{n=0}^{k-1} P_{(N,1,n)} \right].$$

3.2 A single vacation scheme

In this section, it is assumed that the server goes on a vacation if the system is empty, provided that at least one service completion has occurred. However, if the server has completed a fixed number, say k services, it can proceed on a vacation even if the system is not empty. The MRSPN representation of the system is given in Figure 2.

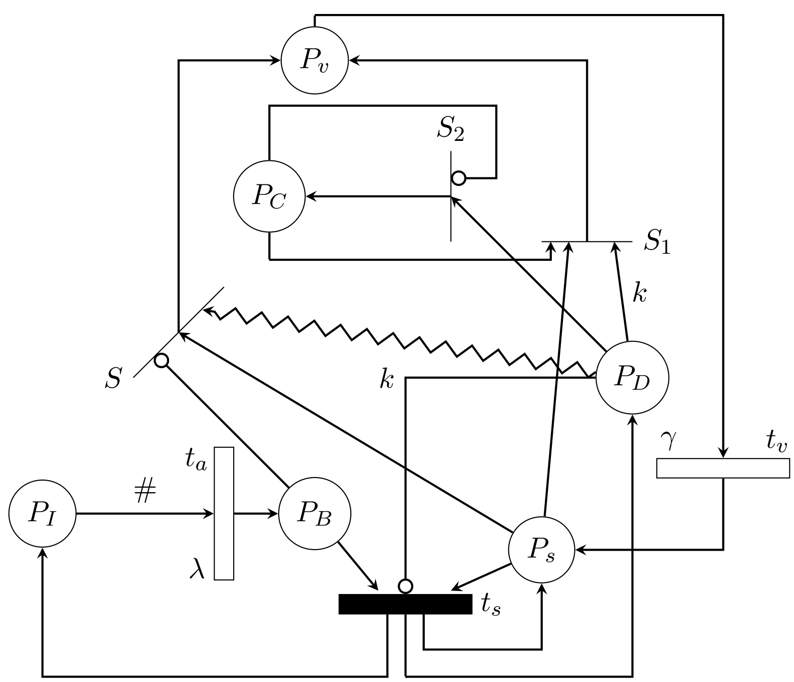

Figure 2: *MRSPN model of the single vacation $M/G/1/N/V_l$ queueing system.*

There is an additional place P_C in this model to record the fact that the server has completed at least one service before going away on a vacation. The immediate transition S_2, when enabled, places a token in P_C, thus keeping a record of the first service completion.

An inhibitor arc from P_C to S_2 ensures that this transition is not enabled again. The other services are recorded in P_D. The immediate transition S whose enabling allows the server to proceed on vacation when the system is empty, now requires a input arc from P_C. The multiplicity of the input arc from P_D to S_1 is $k-1$. The enabling of S_1 also requires an input arc from P_C. An inhibitor input arc from P_D to t_s of multiplicity $k-1$ prevents the enabling of t_s when there are $k-1$ tokens in P_D. That is, a total of k service completions means that the service is terminated for some time. A zig-zag arc from P_D to S flushes out all the tokens from P_D when S fires (including when there are zero tokens in P_D). As in the previous section, the state of the system at any instant of time is given by

$$M(t) = (M(P_B), M(P_s), M(P_C), M(P_D))$$

where $M(P_B), M(P_s), M(P_C)$ and $M(P_D)$ denote the number of tokens in the places P_B, P_s, P_C, P_D respectively. The state space of the system is given by

$$\begin{aligned}\Omega &= \{(r,0,0,0) : 0 \le r \le N\} \cup \{(r,1,0,0) : 0 \le r \le N\} \\ &\quad \cup \{(r,1,1,n) : 1 \le r \le N : 0 \le n \le k-2\}.\end{aligned}$$

The RTPs are defined as follows:

1. $T_0 = 0$.
2. If at the n^{th} RTP T_n, the process is in the state $(r,0,0,0)$ with $0 \le r \le N$, the GEN transition t_s is not enabled. The next RTP T_{n+1} is the time at which either of the two EXP transitions t_a or t_v fires.
3. If at T_n, the process is in the state $(0,1,0,0)$, the next RTP is the time at which t_a fires.
4. If at T_n, the process is in the state $(r,1,s,n)$ where $1 \le r \le N$, $0 \le n \le k-2$ and s is either 0 or 1, the GEN transition t_s is enabled. The next regeneration time point is the time at which t_s fires.

In this case, the subordinated process is a CTMC with state space

$$\Omega(r,1,s,n) = \{(r',1,s,n) : r \le r' \le N\}.$$

The infinitesimal generator $Q(r,1,s,n) = [q_{ij}]$ is as in the previous sections. The expressions for the local and global kernels may now be obtained.

Local kernel

When the regeneration cycle starts in the state $(r,0,0,0)$ with $0 \le r \le N$,

$$\begin{aligned}E_{(r,0,0,0)\ (r',0,0,0)}(t) &= 0,\ \text{for } r \ne r', \\ E_{(r,0,0,0)\ (r,0,0,0)}(t) &= e^{-[(N-r)\lambda+\gamma]t}, \\ E_{(r,0,0,0)\ (r',1,s,n)}(t) &= 0,\ \text{for } 1 \le r' \le N,\ s \in \{0,1\},\ 0 \le n \le k-2, \\ E_{(r,0,0,0)\ (0,1,0,0)}(t) &= 0.\end{aligned}$$

When the regeneration cycle starts in the state $(0, 1, 0, 0)$,

$$\begin{aligned}
E_{(0,1,0,0)\ (r,0,0,0)}(t) &= 0, \text{ for } 0 \le r \le N,\\
E_{(0,1,0,0)\ (r,1,0,0)}(t) &= 0, \text{ for } r \ne 0,\\
E_{(0,1,0,0)\ (0,1,0,0)}(t) &= e^{-N\lambda t},\\
E_{(0,1,0,0)\ (r,1,1,n)}(t) &= 0, \text{ for } 1 \le r \le N,\ 0 \le n \le k-2.
\end{aligned}$$

When the regeneration cycle starts in the state $(r, 1, s, n)$ with $1 \le r \le N$, $s \in \{0, 1\}, 1 \le n \le k-2$,

$$\begin{aligned}
E_{(r,1,s,n)\ (r',0,0,0)}(t) &= 0, \text{ for } 0 \le r' \le N,\\
E_{(r,1,s,n)\ (r',1,s',n')}(t) &= 0, \text{ for } (s', n') \ne (s, n),\\
E_{(r,1,s,n)\ (r',1,s,n)}(t) &= 0, \text{ for } r' \le r-1,\\
E_{(r,1,s,n)\ (r',1,s,n)}(t) &= [e^{Q(r,1,s,n)t}]_{(1,r'-r+1)}(1 - B(t)), \text{ for } r \le r' \le N.
\end{aligned}$$

Global kernel

When the regeneration cycle starts in the state $(r, 0, 0, 0)$ with $0 \le r < N$,

$$\begin{aligned}
K_{(r,0,0,0)\ (r',0,0,0)}(t) &= 0, \text{ for } r' \ne r+1,\\
K_{(r,0,0,0)\ (r+1,0,0,0)}(t) &= \frac{(N-r)\lambda}{(N-r)\lambda+\gamma}(1 - e^{-[(N-r)\lambda+\gamma]t}),\\
K_{(r,0,0,0)\ (r',1,0,0)}(t) &= 0, \text{ for } r' \ne r,\\
K_{(r,0,0,0)\ (r,1,0,0)}(t) &= \frac{\gamma}{(N-r)\lambda+\gamma}(1 - e^{-[(N-r)\lambda+\gamma]t}),\\
K_{(r,0,0,0)\ (r',1,1,n)}(t) &= 0, \text{ for } 1 \le r' \le N,\ 0 \le n \le k-2.
\end{aligned}$$

When the regeneration cycle starts in the state $(N, 0, 0, 0)$,

$$\begin{aligned}
K_{(N,0,0,0)\ (r',0,0,0)}(t) &= 0, \text{ for } 0 \le r' \le N,\\
K_{(N,0,0,0)\ (r,1,0,0)}(t) &= 0, \text{ for } r \ne N,\\
K_{(N,0,0,0)\ (N,1,0,0)}(t) &= (1 - e^{-\gamma t}),\\
K_{(N,0,0,0)\ (r,1,1,n)}(t) &= 0, \text{ for } 1 \le r \le N,\ 0 \le n \le k-2.
\end{aligned}$$

When the regeneration cycle begins in the state $(0, 1, 0, 0)$,

$$\begin{aligned}
K_{(0,1,0,0)\ (r,0,0,0)}(t) &= 0, \text{ for } 0 \le r \le N,\\
K_{(0,1,0,0)\ (r,1,0,0)}(t) &= 0, \text{ for } r \ne 1,\\
K_{(0,1,0,0)\ (1,1,0,0)}(t) &= (1 - e^{-N\lambda t}),\\
K_{(0,1,0,0)\ (r,1,1,n)}(t) &= 0, \text{ for } 1 \le r \le N,\ 0 \le n \le k-2.
\end{aligned}$$

When the regeneration cycle begins in the state $(r, 1, 0, 0)$ with $1 < r \le N$,

$$\begin{aligned}
K_{(r,1,0,0)\ (r',0,0,0)}(t) &= 0, \text{ for } 0 \le r' \le N,\\
K_{(r,1,0,0)\ (r',1,0,0)}(t) &= 0, \text{ for } 0 \le r' \le N,\\
K_{(r,1,0,0)\ (r',1,1,n)}(t) &= 0, \text{ for } n \ne 0 \text{ or for } n = 0,\ r' < r-1 \text{ or } r' = N,\\
K_{(r,1,0,0)\ (r',1,1,0)}(t) &= \int_0^t [e^{Q(r,1,0,0)x}]_{(1,\ r'-r+2)} \mathrm{d}B(x), \text{ for } r-1 \le r' < N.
\end{aligned}$$

When the regeneration cycle begins in the state $(1, 1, 0, 0)$,

$$\begin{aligned}
K_{(1,1,0,0)(0,0,0,0)}(t) &= \int_0^t [e^{Q(1,1,0,0)x}]_{(1,1)} \mathrm{d}B(x), \\
K_{(1,1,0,0)(r,0,0,0)}(t) &= 0, \text{ for } r \neq 0, \\
K_{(1,1,0,0)(r,1,0,0)}(t) &= 0, \text{ for } 0 \leq r \leq N, \\
K_{(1,1,0,0)(r,1,1,n)}(t) &= 0, \text{ for } n \neq 0 \text{ or for } n = 0 \text{ and } r = N, \\
K_{(1,1,0,0)(r,1,1,0)}(t) &= \int_0^t [e^{Q(1,\ 1,\ 0,\ 0)x}]_{(1,r+1)} \mathrm{d}B(x), \text{ for } 1 \leq r < N.
\end{aligned}$$

When the regeneration cycle starts in the state $(r, 1, 1, n)$ with $1 < r \leq N$ and $0 \leq n < k - 2$,

$$\begin{aligned}
K_{(r,1,1,n)(r',0,0,0)}(t) &= 0, \text{ for } 0 \leq r' \leq N, \\
K_{(r,1,1,n)(r',1,0,0)}(t) &= 0, \text{ for } 0 \leq r' \leq N, \\
K_{(r,1,1,n)(r',1,1,n')}(t) &= 0, \text{ for } n' \neq n+1 \text{ or for } n' = n+1 \text{ and } r' < r-1 \text{ or } r' = N, \\
K_{(r,1,1,n)(r',1,1,n+1)}(t) &= \int_0^t [e^{Q(r,\ 1,\ 1,\ n)x}]_{(1,r'-r+2)} \mathrm{d}B(x), \text{ for } r-1 \leq r' < N.
\end{aligned}$$

When $r = 1$,

$$\begin{aligned}
K_{(1,1,1,n)(r',0,0,0)}(t) &= 0, \text{ for } r' \neq 0, \\
K_{(1,1,1,n)(0,0,0,0)}(t) &= \int_0^t [e^{Q(1,1,1,n)x}]_{(1,1)} \mathrm{d}B(x), \\
K_{(1,1,1,n)(r',1,0,0)}(t) &= 0, \text{ for } 0 \leq r' \leq N, \\
K_{(1,1,1,n)(r',1,1,n')}(t) &= 0, \text{ for } n' \neq n+1 \text{ or } n' = n+1 \text{ and } r' = N, \\
K_{(1,1,1,n)(r',1,1,n+1)}(t) &= \int_0^t [e^{Q(1,1,1,n)x}]_{(1,r'+1)} \mathrm{d}B(x), \text{ for } 1 \leq r' < N.
\end{aligned}$$

When the regeneration cycle starts in the state $(r, 1, 1, k-2)$ with $1 \leq r \leq N$,

$$\begin{aligned}
K_{(r,1,1,k-2)(r',0,0,0)}(t) &= 0, \text{ for } r' < r-1 \text{ or } r' = N, \\
K_{(r,1,1,k-2)(r',0,0,0)}(t) &= \int_0^t [e^{Q(r,\ 1,\ 1,\ k-2)x}]_{(1,r'-r+2)} \mathrm{d}B(x), \text{ for } r-1 \leq r' < N, \\
K_{(r,1,1,k-2)(r',1,0,0)}(t) &= 0, \text{ for } 0 \leq r' \leq N, \\
K_{(r,1,1,k-2)(r',1,1,n')}(t) &= 0, \text{ for } 1 \leq r' \leq N,\ 0 \leq n' \leq k-2.
\end{aligned}$$

Assuming ergodicity of the system, the steady state probabilities $P_{(r,0,0,0)}$ for $0 \leq r \leq N$ and $P_{(r,1,0,0)}$ for $0 \leq r \leq N$ as well as $P_{(r,1,1,n)}$ for $1 \leq r \leq N,\ 0 \leq n \leq k-2$ may be derived using the formulae in §2. The performance measures of the system are

$$\begin{aligned}
\lambda_{\text{eff}} &= \lambda\left\{\sum_{r=1}^{N-1}(N-r)\left[P_{(r,\ 0,\ 0,\ 0)} + P_{(r,1,0,0)} + \sum_{n=0}^{k-2} P_{(r,1,1,n)}\right]\right. \\
&\quad \left. + N[P_{(0,\ 0,\ 0,\ 0)} + P_{(0,\ 1,\ 0,\ 0)}]\right\}, \\
L_s &= \sum_{r=1}^{N} r\left\{P_{(r,0,0,0)} + P_{(r,1,0,0)} + \sum_{n=0}^{k-2} P_{(r,1,1,n)}\right\}, \\
\mu_v &= \gamma \sum_{r=0}^{N} P_{(r,0,0,0)} \text{ and} \\
W_t &= t\sum_{r=1}^{N}\left[\sum_{n=0}^{k-2} P_{(r,1,1,n)} + P_{(r,1,0,0)}\right],
\end{aligned}$$

where the symbols λ_{eff}, L_s, μ_v and W_t have the same meaning as in §3.1.1. As in the previous sections, the above discussion may be extended to the case where the population is infinite, but the system has a finite capacity.

3.3 A hybrid scheme

In this section it is assumed that the server can proceed on a vacation after the completion of a fixed number of services, as in the previous sections. However, if the system is empty, it can proceed on a vacation after waiting for a random period of time. This random amount of time is assumed to be exponentially distributed with parameter α. The MRSPN representation of the system is given in Figure 3.

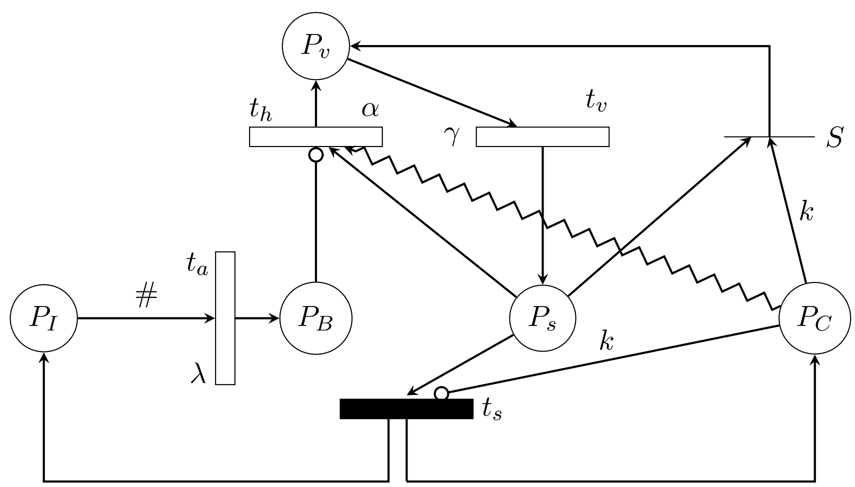

Figure 3: *MRSPN representation of the $M/G/1/V_l$ queueing system with a hybrid scheme.*

The places P_I, P_B, P_s, P_v, and the transitions t_a, t_s, t_v have the same meaning as in the §3.1 and §3.2. The place P_C keeps a record of the number of service completions. The transition t_h allows the server to proceed on a vacation when the system is empty. The inhibitor arc from P_B to t_h prevents enabling of t_h when the system is not empty. A zig-zag arc from P_C to t_h flushes out all the tokens in P_C (including zero), when t_h fires. The immediate transition S is enabled, when there are k tokens in P_C. As in the previous section, there is an inhibitor arc of multiplicity k from P_C to t_s. As before, the marking $M(t)$ at any instant of time t is specified by the number of tokens in P_B, P_s and P_C.

Since there is only one GEN transition t_s, which does not conflict with any of the EXP transitions t_a, t_h and t_v, the underlying marking process is a MRGP. The RTPs may be defined as in the previous section. The state space of the system is given by

$$\Omega = \{(r,0,0) : 0 \le r \le N\} \cup \{(r,1,n) : 0 \le r \le N, 0 \le n \le k-1\}.$$

The subordinated processes, which start in the state $(r,1,n)$ with $1 \le r \le N$ and $0 \le n \le k-1$, is a CTMC with state space

$$\Omega(r,1,n) = \{(s,1,n) : r \le s \le N\}.$$

The infinitesimal generator is the same as that in the previous section. The expressions for the local and global kernels may now be obtained.

Local kernel

When the regeneration cycle starts in the state $(r,0,0)$ with $0 \le r \le N$,

$$\begin{aligned} E_{(r,0,0)(r',0,0)}(t) &= 0, \text{ for } r' \ne r, \\ E_{(r,0,0)(r,0,0)}(t) &= e^{-[(N-r)\lambda+\gamma]t}, \\ E_{(r,0,0)(r',1,n)}(t) &= 0, \text{ for } 0 \le r' \le N \text{ and } 0 \le n \le k-1. \end{aligned}$$

When the regeneration cycle starts in the state $(0,1,n)$ with $0 \le n \le k-1$,

$$\begin{aligned} E_{(0,1,n)(r,0,0)}(t) &= 0, \text{ for } 0 \le r \le N, \\ E_{(0,1,n)(r,1,n')}(t) &= 0, \text{ for } (r,n') \ne (0,n), \\ E_{(0,1,n)(0,1,n)}(t) &= e^{-[N\lambda+\alpha]t}. \end{aligned}$$

When the regeneration cycle starts in the states $(r,1,n)$ with $1 \le r \le N$ and $0 \le n \le k-1$,

$$\begin{aligned} E_{(r,1,n)(r',0,0)}(t) &= 0, \text{ for } 0 \le r' \le N, \\ E_{(r,1,n)(r',1,n')}(t) &= 0, \text{ for } (r',1,n') \notin \Omega(r,1,n), \\ E_{(r,1,\ n)(r',1,n)}(t) &= [e^{Q(r,\ 1,\ n)t}]_{(1,r'-r+1)}(1-B(t)), \text{ for } r \le r' \le N. \end{aligned}$$

Global kernel

When the regeneration cycle starts in the state $(r,0,0)$ with $0 \le r < N$,

$$\begin{aligned} K_{(r,0,0)(r',0,0)}(t) &= 0, \text{ for } r' \ne r+1, \\ K_{(r,0,0)(r+1,0,0)}(t) &= \frac{(N-r)\lambda}{(N-r)\lambda+\gamma}[1-e^{-[(N-r)\lambda+\gamma]t}], \\ K_{(r,0,0)(r',1,n)}(t) &= 0, \text{ for } (r',n) \ne (r,0), \\ K_{(r,0,0)(r,1,0)}(t) &= \frac{\gamma}{(N-r)\lambda+\gamma}[1-e^{-[(N-r)\lambda+\gamma]t}]. \end{aligned}$$

When $r = N$,

$$\begin{aligned} K_{(N,0,0)(r',0,0)}(t) &= 0,\ \text{for}\ \ 0 \le r' \le N, \\ K_{(N,0,0)(r',1,n)}(t) &= 0,\ \text{for}\ \ (r',\ n) \ne (N,\ 0), \\ K_{(N,0,0)(N,1,0)}(t) &= 1 - e^{-\gamma t}. \end{aligned}$$

When the regeneration cycle starts in the state $(0, 1, n)$ with $0 \le n \le k-1$,

$$\begin{aligned} K_{(0,1,n)(r,0,0)}(t) &= 0,\ \text{for}\ r \ne 0, \\ K_{(0,1,n)(0,0,0)}(t) &= \frac{\alpha}{N\lambda+\alpha}[1 - e^{-[N\lambda+\alpha]t}], \\ K_{(0,1,n)(r',1,n')}(t) &= 0,\ \text{for}\ (r', n') \ne (1, n), \\ K_{(0,1,n)(1,1,n)}(t) &= \frac{N\lambda}{N\lambda+\alpha}[1 - e^{-[N\lambda+\alpha]t}]. \end{aligned}$$

When the regeneration cycle starts in the state $(r, 1, n)$ with $1 < r \le N$ and $0 \le n < k-1$,

$$\begin{aligned} K_{(r,1,n)(r',0,0)}(t) &= 0,\ \text{for}\ 0 \le r' \le N, \\ K_{(r,1,n)(r',1,n')}(t) &= 0,\ \text{when either}\ r' < r-1\ \text{or}\ r' = N\ \text{or}\ n' \ne n+1, \\ K_{(r,1,n)(r',1,n+1)}(t) &= \int_0^t [e^{Q(r,\ 1,\ n)x}]_{(1,r'-r+2)} \mathrm{d}B(x),\ \text{for}\ r-1 \le r' \le N-1. \end{aligned}$$

When $r = 1$,

$$\begin{aligned} K_{(1,1,n)(r',0,0)}(t) &= 0,\ \text{for}\ r' \ne 0, \\ K_{(1,1,n)(0,0,0)}(t) &= \int_0^t [e^{Q(1,\ 1,\ n)x}]_{(1,\ 1)} \mathrm{d}B(x), \\ K_{(1,1,n)(r',1,n')}(t) &= 0,\ \text{when either}\ r' = N\ \text{or}\ n' \ne n+1, \\ K_{(1,1,n)(r',1,n+1)}(t) &= \int_0^t [e^{Q(1,\ 1,\ n)x}]_{(1,\ r'+1)} \mathrm{d}B(x),\ \text{for}\ 1 \le r' < N. \end{aligned}$$

When the regeneration cycle starts in the state $(r, 1, k-1)$ with $1 \le r \le N$,

$$\begin{aligned} K_{(r,1,k-1)(r',0,0)}(t) &= 0,\ \text{for}\ r' < r-1\ \text{or}\ r' = N, \\ K_{(r,1,k-1)(r',0,0)}(t) &= \int_0^t [e^{Q(r,\ 1,\ k-1)x}]_{(1,r'-r+2)} \mathrm{d}B(x),\ \text{for}\ r-1 \le r' \le N-1, \\ K_{(r,1,k-1)(r',1,n')}(t) &= 0,\ \text{for}\ 0 \le r' \le N,\ 0 \le n \le k-1. \end{aligned}$$

Assuming ergodicity of the system considered, the steady state probabilities $P_{(r,0,0)}$ for $0 \le r \le N$ and $P_{(r,1,n)}$ for $0 \le r \le N$, $0 \le n \le k-1$ may be evaluated by means of the formulae given in §2. In this case performance measures are

$$\begin{aligned} \lambda_{\text{eff}} &= \lambda \sum_{r=0}^{N} (N-r) \left[P_{(r,0,0)} + + \sum_{n=0}^{k-1} P_{(r,1,n)} \right], \\ L_s &= \sum_{r=1}^{N} r \left\{ P_{(r,0,0)} + \sum_{n=0}^{k-1} P_{(r,1,n)} \right\}, \end{aligned}$$

$$
\begin{aligned}
\mu_v &= \gamma \sum_{r=0}^{N} P_{(r,0,0)} \quad \text{and} \\
W_t &= t \sum_{r=1}^{N} \sum_{n=0}^{k-1} P_{(r,1,n)},
\end{aligned}
$$

where the symbols λ_{eff}, L_s, μ_v and W_t have the same meaning as in §3.1.1. As in the previous sections, the above discussion may be extended to the case where the population is infinite, but the system has a finite capacity.

4 Numerical Examples

In this section we present three numerical examples to indicate how the MRSPN representations of the queueing systems considered in §3 may be used to obtain the performance measures of the system. In all three examples we use $N = 3$.

4.1 Limited service discipline with multiple vacation scheme

In this example we consider the vacation queueing system with a limited service discipline and a multiple vacation scheme. The reachability graph is given in Figure 4. This graph contains ten tangible markings $\{M_j : 1 \leq j \leq 10\}$ and five vanishing markings. After removing the vanishing markings, the state space of the process is given by $\Omega = \{i : 1 \leq i \leq 10\}$ where i stands for the marking M_i. The local kernel is given by

$$
[E_{ij}(t)] = \begin{bmatrix} A' & 0_{4\times 3} & 0_{4\times 3} \\ 0_{3\times 4} & B' & 0_{3\times 3} \\ 0_{3\times 4} & 0_{3\times 3} & B' \end{bmatrix},
$$

where A' is a diagonal matrix of order 4 with $e^{-[(4-i)\lambda+\gamma]t}$ in the i^{th} row and where $B' = [b_{ij}]$ is an upper triangular matrix of order 3 containing the elements

$$
\begin{aligned}
b_{1j}(t) &= \begin{cases} e^{-2\lambda t}(1 - B(t)) & \text{for } j = 1, \\ 2(e^{-\lambda t} - e^{-2\lambda t})(1 - B(t)) & \text{for } j = 2, \\ (e^{-2\lambda t} - 2e^{-\lambda t} + 1)(1 - B(t)) & \text{for } j = 3, \end{cases} \\
b_{2j}(t) &= \begin{cases} e^{-\lambda t}(1 - B(t)) & \text{for } j = 2, \\ (1 - e^{-\lambda t})(1 - B(t)) & \text{for } j = 3, \end{cases} \\
b_{33}(t) &= 1 - B(t).
\end{aligned}
$$

The global kernel is given by

$$
[K_{ij}(t)] = \begin{bmatrix} C_1 & G & H_1 & 0_{3\times 3} \\ 0 & 0_{1\times 3} & R_1 & 0_{1\times 3} \\ C_2 & 0_{3\times 3} & 0_{3\times 3} & D_1 \\ C_2 & D_1 & 0_{3\times 3} & 0_{3\times 3} \end{bmatrix},
$$

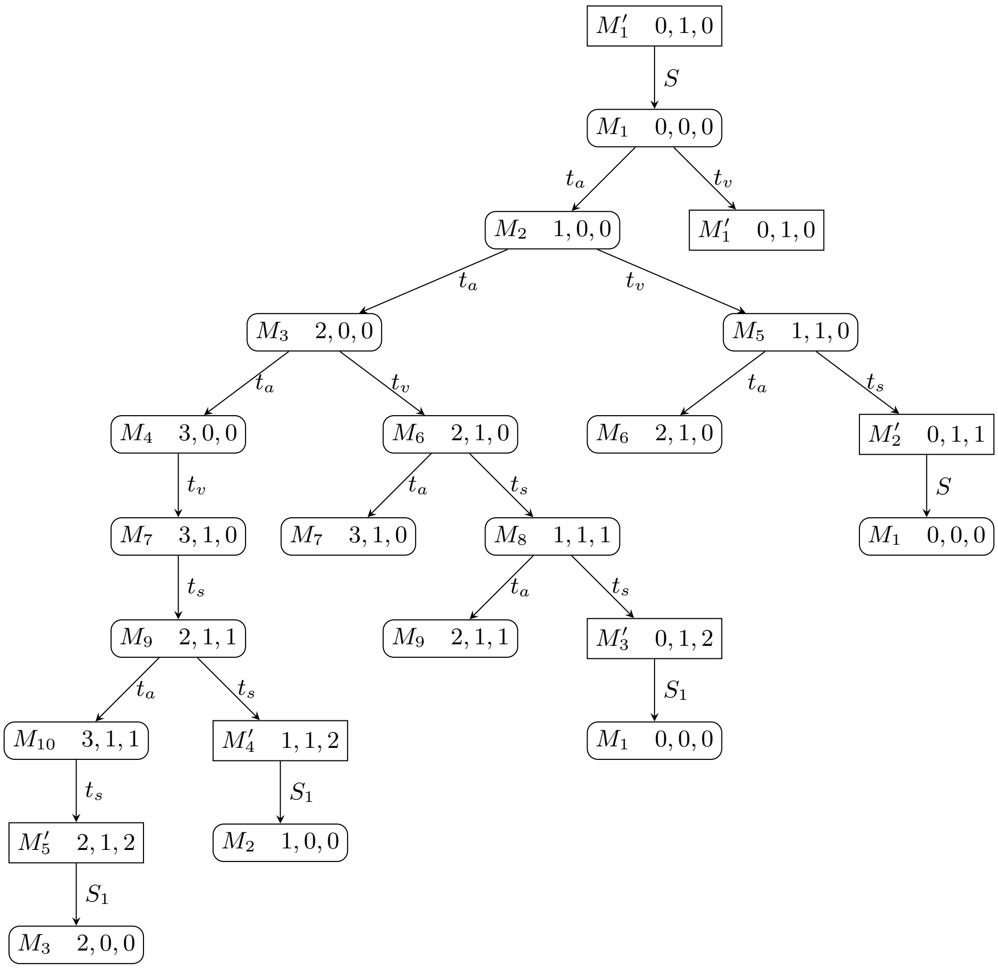

Figure 4: *Reachability graph for the MRSPN of Figure 1, with $N = 3$, $k = 2$.*

where G is a diagonal matrix of order 3 containing

$$\frac{(4-i)\lambda}{(4-i)\lambda+\gamma}(1-e^{-[(4-i)\lambda+\gamma]t}), i = 1, 2, 3$$

as the element in the i^{th} row. Furthermore $H_1 = [h_{ij}]$ is a matrix of order 3 with only two non zero elements, namely

$$\begin{aligned} h_{21}(t) &= \frac{\gamma}{2\lambda+\gamma}(1-e^{-(2\lambda+\gamma)t}) \text{ and} \\ h_{32}(t) &= \frac{\gamma}{\lambda+\gamma}(1-e^{-(\lambda+\gamma)t}). \end{aligned}$$

Moreover, the entries of the matrix $D_1 = [d_{ij}]$ are given by

$$
\begin{aligned}
d_{11}(t) &= \int_0^t 2(e^{-\lambda x} - e^{-2\lambda x})\mathrm{d}B(x), \\
d_{12}(t) &= \int_0^t (e^{-2\lambda x} - 2e^{-\lambda x} + 1)\mathrm{d}B(x), \\
d_{21}(t) &= \int_0^t e^{-\lambda x}\mathrm{d}B(x), \\
d_{22}(t) &= \int_0^t (1 - e^{-\lambda x})\mathrm{d}B(x), \quad \text{and} \\
d_{32}(t) &= B(t).
\end{aligned}
$$

Finally,

$$
\begin{aligned}
R_1 &= \begin{bmatrix} 0 & 0 & 1 - e^{-\gamma t} \end{bmatrix}, \\
C_1 &= \begin{bmatrix} c_{11} & 0 & 0 \end{bmatrix}^T, \\
C_2 &= \begin{bmatrix} c_{21} & 0 & 0 \end{bmatrix}^T, \\
c_{11}(t) &= \frac{\gamma}{3\lambda + \gamma}(1 - e^{-(3\lambda+\gamma)t}) \text{ and} \\
c_{21}(t) &= \int_0^t e^{-2\lambda x}\mathrm{d}B(x).
\end{aligned}
$$

Using the formulae in §2, the steady state probabilities $P_1, \ldots, P_{10}$ may be obtained as

$$
\begin{aligned}
P_1 &= \frac{\beta(\lambda)\beta(2\lambda)\gamma[R(1 + 2(\beta(\lambda) - \beta(2\lambda)) + D]}{M}, \\
P_2 &= \frac{\beta(\lambda)3\lambda R}{M}, \\
P_3 &= \frac{3\lambda D}{M}, \\
P_4 &= \frac{3\lambda^2 D}{\gamma M}, \\
P_5 &= \frac{3\beta(\lambda)(1 - \beta(\lambda))\gamma R}{2M}, \\
P_6 &= \frac{3\gamma[(1 - \beta(\lambda))D + \beta(\lambda)(\beta(2\lambda) - 2\beta(\lambda) + 1)R]}{M}, \\
P_7 &= 3\left(\mu\lambda[\gamma\beta(\lambda)R + D(\lambda + \gamma)] - \gamma D(1 - \beta(\lambda)) \right. \\
&\quad \left. + \frac{1}{2}\gamma\beta(\lambda)R(4\beta(\lambda) - 3 - \beta(2\lambda)) \right) \frac{1}{M},
\end{aligned}
$$

$$
\begin{aligned}
P_8 &= \frac{3\gamma\beta(\lambda)(1-\beta(2\lambda))[2(\beta(\lambda)-\beta(2\lambda))R+D]}{2M} \\
P_9 &= \frac{3}{M}\{R\gamma\beta(\lambda)(1-2\beta(\lambda)+\beta(2\lambda))(\beta(\lambda)-2\beta(2\lambda)+1) \\
&\quad +D\left[\gamma\beta(\lambda)(\beta(2\lambda)-\beta(\lambda))+(\lambda+\gamma)(1-\beta(\lambda))\right]\} \\
P_{10} &= \frac{3}{M}\{\gamma\beta(\lambda)R\left[(\beta(\lambda)-\beta(2\lambda))(2\beta(\lambda)-\beta(2\lambda))+\beta(2\lambda)(2-\beta(\lambda))-1\right] \\
&\quad +D\left[(\lambda+\gamma)(\beta(\lambda)-1)+\gamma\beta(\lambda)(\beta(\lambda)-\frac{\beta(2\lambda)}{2}-\frac{1}{2})\right] \\
&\quad +\lambda\mu\left[\gamma\beta(\lambda)R(1-\beta(2\lambda))+D(\lambda+\gamma)\right]\}
\end{aligned}
$$

In the above,

$$
\begin{aligned}
M &= R\beta(\lambda)[\gamma\beta(2\lambda)(1+2(\beta(\lambda)-\beta(2\lambda))+3\lambda+3\lambda\gamma(2-\beta(2\lambda))\mu] \\
&\quad + D[\beta(\lambda)\beta(2\lambda)\gamma+3\lambda+\frac{3\lambda^2}{\gamma}+6\lambda(\lambda+\gamma)\mu], \\
R &= (\beta(\lambda)-\beta(2\lambda))\gamma+(\lambda+\gamma) \text{ and} \\
D &= (2\lambda+\gamma)-\gamma[\beta(2\lambda)+\beta(\lambda)(1-\beta(2\lambda))+2(\beta(\lambda)-\beta(2\lambda))^2],
\end{aligned}
$$

where μ is the expected value of the service distribution. The same reachability graph may be used to obtain the steady state probabilities of the queueing system when the population is infinite and the system capacity is three.

P_n	μ	$\gamma=\frac{1}{3}$ finite population	finite capacity	$\gamma=\frac{1}{4}$ finite population	finite capacity	$\gamma=\frac{1}{5.0}$ finite population	finite capacity
1		0.0002	0.0051	0.0001	0.0034	0.0001	0.0025
2		0.0168	0.0323	0.0148	0.0287	0.0131	0.0256
3		0.1208	0.1167	0.1113	0.1084	0.1019	0.0998
4		0.3624	0.3501	0.4453	0.4336	0.5099	0.4990
5		0.0026	0.0083	0.0017	0.0055	0.0012	0.0039
6	1.5	0.0346	0.0349	0.0238	0.0242	0.0174	0.0177
7		0.2126	0.2062	0.1886	0.1824	0.1688	0.1655
8		0.0051	0.0095	0.0035	0.0065	0.0025	0.0047
9		0.1273	0.1232	0.1095	0.1067	0.0960	0.0940
10		0.1170	0.1131	0.1009	0.0982	0.0886	0.0867
1		0.0001	0.0040	0.0001	0.0027	0.0000	0.0020
2		0.0146	0.0281	0.0129	0.0251	0.0115	0.0225
3		0.1173	0.1138	0.1085	0.1060	0.0997	0.0978
4		0.3519	0.3414	0.4340	0.4240	0.4985	0.4890
5	1.6	0.0023	0.0074	0.0015	0.0050	0.0011	0.0036
6		0.0343	0.0347	0.0237	0.0241	0.0173	0.0177
7		0.2214	0.2155	0.1969	0.1929	0.1766	0.1736
8		0.0045	0.0085	0.0031	0.0058	0.0022	0.0043
9		0.1270	0.1236	0.1096	0.1073	0.0964	0.0948
10		0.1261	0.1226	0.1091	0.1067	0.0962	0.0944

Table 1: *Steady state probabilities for the finite population and finite capacity $M/G/1/3$ queueing systems.*

The values for the steady state probabilities in both the finite population and finite capacity cases are given in Table 1. A deterministic service time μ is used. It may be seen from the table that for these values of μ and γ, the minimum probability is P_1, corresponding to $M_1(0,0,0)$. Likewise, the maximum probability is P_4, corresponding to the state $M_4(3,0,0)$.

4.2 Performance measures of the $M/G/1/3/V_l$ queueing systems under three different vacation schemes

In this example, we compute some performance measures of the $M/G/1/3$ queueing system under a limited service discipline with $k = 2$. The values of the performance measures for the multiple, single and hybrid vacation schemes are given in Table 2. A deterministic service time distribution wirh service time μ is assumed. Also, the values $\lambda = 1.0$ and $\alpha = 3.0$ were used.

From Table 2 it may be seen that for fixed values of μ and γ, the mean response time W_s is the smallest in the case of a hybrid vacation scheme. The mean response time is smaller in the single vacation scheme than in the multiple vacation scheme. Similarly μ_v, the number of vacations per unit time is smallest in the hybrid case. Consequently, the fraction of busy time of the server W_t in the interval $[0,t]$ is the largest in the hybrid scheme. The single vacation scheme is better than the multiple vacation scheme from the customer's point of view.

μ		Hybrid		Single		Multiple	
		$\gamma = \frac{1}{3.0}$	$\gamma = \frac{1}{4.0}$	$\gamma = \frac{1}{3.0}$	$\gamma = \frac{1}{4.0}$	$\gamma = \frac{1}{3.0}$	$\gamma = \frac{1}{4.0}$
	w_s	8.0000	9.4999	8.0040	9.5040	8.0062	9.5058
1.5	μ_v	0.1667	0.1428	0.1667	0.1428	0.1667	0.1429
	w_t	2.9988	2.5706	2.9986	2.5704	2.9979	2.5700
	w_s	8.2999	9.7999	8.3029	9.8029	8.3046	9.8042
1.6	μ_v	0.1613	0.1389	0.1613	0.1389	0.1613	0.1389
	w_t	3.0959	2.6660	3.0957	2.6651	3.0952	2.6656

Table 2: *Performance measures for the $M/G/1/3/V_l$ queueing systems with multiple, single and hybrid schemes.*

4.3 Mean response time for the $M/G/1/3/V_l$ queueing system with three types of service time distributions

In this example, the mean response time associated with the three types of vacation schemes and service time distributions are compared for three types of service time distributions: a deterministic case, a three-phase Erlang distribution, and a hyper-exponential distribution. The expected service time in all three distributions is assumed to have the same value μ. Values of the mean response time for different values of μ, for the three distributions and for the three types of vacation schemes, are given in Table 3.

From Table 3, it may be seen that, as far as the single vacation and the multiple vacation schemes are concerned, the deterministic distribution gives the minimum values for the

Type	Distribution	$\mu = 1.5$	$\mu = 1.6$	$\mu = 1.7$
Multiple	Deterministic	8.00624	8.30460	8.60338
	Hyper-exponential	8.07054	8.36336	8.65723
	Three Phase Erlang	8.02091	8.31767	8.61500
Single	Deterministic	8.00407	8.30299	8.60220
	Hyper-exponential	8.04930	8.34410	8.63969
	Three Phase Erlang	8.01393	8.31174	8.60995
Hybrid	Deterministic	8.00340	8.30250	8.60184
	Hyper-exponential	8.03954	8.33544	8.63195
	Three Phase Erlang	7.91611	8.21745	8.51704

Table 3: *Mean response time for the $M/G/1/3/V_l$ queueing systems with three types of service time distributions.*

response time. The maximum response times are given by the hyper-exponential service time distribution. In the case of the hybrid scheme, the three-phase Erlang distribution gives smaller values of the response time than the deterministic case. In this case also, the hyper-exponential service time distribution gives the maximum response time.

5 Concluding Remarks

In this paper we have shown how MRSPN representations may be used as an easy and efficient method to obtain performance measures for queueing systems with different vacation schemes. The applicability of the results obtained in §3 are demonstrated in §4 for small values of N. However, with the help of software packages now available [3, 16], the representations obtained in §3 may be applied to a wide variety of practical problems with large values of N.

The authors hope to consider the applicability of these results in the modelling of wireless and communication networks in some detail. Another interesting extension would be to consider the feasibility of including repeated arrivals or retrials in the vacation queueing models considered.

References

[1] Ajmone-Marsan M, Balbo G, Conte G, Donatelli S & Franceschinis G, 1995, *Modelling with generalised stochastic Petri nets*, John Wiley and Sons, New York (NY).

[2] Altiok T, 1988/1989, *Queueing modeling of a single processor with failures*, Performance Evaluation, **9**, pp. 93–102.

[3] Bobbio A, Puliofito A, Scarpa M & Telek M, 1998, *Web SPN: A WEB accessible Petri net tool*, Proceedings of International Conference on Web-Based Modeling and Simulations, San Diego, Available from `http://mdslab.unime.it/webspn`

[4] Bruneel H, 1993, *Performance of discrete-time queueing system*, Computational Operations Research, **20**, pp. 303–320.

[5] Chiola G, Franceschinis G, Gaeta R & Ribaudo M, 1995, *GreatSPN 1.7: Graphical editor and analyzer for timed and stochastic Petri nets*, Performance Evaluation, **24**, pp. 137–159.

[6] Choi H, Kulkarni VG & Trivedi KS, 1994, *Markov regenerative stochastic Petri nets*, Performance Evaluation, **20**, pp. 337–357.

[7] Doshi BT, 1986, *Queueing system with vacations — A survey*, Queueing Systems, **1**, pp. 29–66.

[8] German R, 2000, *Performance analysis of communications system: Modeling with non-Markovian stochastic Petri nets*, Wiley, Chichester.

[9] Gross D & Harris CM, 1985, *Fundamentals of queueing theory*, John Wiley and Sons, New York (NY).

[10] Medhi J, 1982, *Stochastic processes*, John Wiley, New York (NY).

[11] Ibe OC & Trivedi KS, 1991, *Stochastic Petri net analysis of finite-population vacation queueing systems*, Queueing Systems Theory and Applications, **8(2)**, pp. 111–127.

[12] Stuck BW & Arthur E, 1985, *A computer and communications network performance analysis primer*, Prentice–Hall, Englewood Cliffs (NJ).

[13] Takagi H, 1987, *A survey of queueing analysis of polling models*, Proceedings of the 3rd IFIP International Conference on Data Communication Systems and their Performance, Rio de Janerio, pp. 5–25.

[14] Takagi H, 1991, *Queueing analysis*, Vol. I, Elsevier, Amsterdam.

[15] Trivedi KS, 1999, *SPNP user's manual*, Version 6.0, Technical report, Center for Advanced Computing and Communication, Department of Electrical and Computer Communication, Duke University, Durham (NC).

[16] Trivedi KS, 2002, SHARPHE software, *Symbolic hierarchical automated reliability and performance evaluation 544*, Available from `http://www.ee.duke.edu/kst/software_packages`

[17] Van der Duyn Schuouten FA & Vanneste SG, 1995, *Maintanence optimization of a production system with buffer capacity*, European Journal of Operational Research, **82**, pp. 232–338.

[18] Walraevens J, Steyaert B & Bruneel H, 2001, *Performance analysis of the system contents in a discrete-time non-preemptive priority queue with general service times*, Belgian Journal of Operations Research, Statistics and Computer Science, **40(1–2)**, pp. 91–103.