Applied Mathematical Modelling 37 (2013) 3879–3893

Contents lists available at SciVerse ScienceDirect

Applied Mathematical Modelling

journal homepage: www.elsevier.com/locate/apm

MAP/*PH*/1 queue with working vacations, vacation interruptions and *N* policy

C. Sreenivasan [a], Srinivas R. Chakravarthy [b], A. Krishnamoorthy [c,*]

[a] *Department of Mathematics, Government College, Chittur, Palakkad 678104, India*
[b] *Department of Industrial Manufacturing Engineering, Kettering University, Flint, MI 48504, USA*
[c] *Department of Mathematics, Cochin University of Science and Technology, Cochin 682022, India*

ARTICLE INFO

Article history:
Received 27 September 2011
Received in revised form 1 June 2012
Accepted 5 July 2012
Available online 1 September 2012

Keywords:
Working vacation
Vacation interruption
Markov
Phase type distribution
Algorithmic probability

ABSTRACT

In this paper we study a *MAP*/*PH*/1 queueing model in which the server is subject to taking vacations and offering services at a lower rate during those times. The service is returned to normal rate whenever the vacation gets over or when the queue length hits a specific threshold value. This model is analyzed in steady state using matrix analytic methods. An illustrative numerical example is discussed.

© 2012 Elsevier Inc. All rights reserved.

1. Introduction

Queues with vacations have been extensively studied by several authors. We refer the reader to the paper by Doshi [1] for earlier works (prior to 1985) on vacation models and to the book by Tian and Zhang [2] for works through 2006. Queueing models with vacations under different scenarios such as (a) exhaustive clearance where the server clears all the work in the system before going on a vacation and returns to work only after completing the current vacation; (b) limited clearance in which the server proceeds on a vacation after completing a fixed number of services or after a fixed period of time; and (c) gated clearance in which the server returning from a vacation serves only those who are waiting at that instant before going on another vacation. The server can take single or multiple vacations at a time. For example, in the case of single vacation the server remains in the system even if there is no one waiting, whereas in the multiple vacations the server will start another vacation when the system is empty whenever coming back from a vacation.

Servi and Finn [3] introduced a working vacation model with the idea of offering services but at a lower rate whenever the server is on vacation. Their model was generalized to the case of *M*/*G*/1 in ([4,5]), and to *GI*/*M*/1 model in [6]. A survey of working vacation models with emphasis on the use of matrix analytic methods is given in Tian et al. [7]. Working vacation models have a number of applications in practice. Two such examples are given in [7].

Recently, Li and Tian [8] studied an $M/M/1$ queue with working vacations in which vacationing server offers services at a lower rate for the first customer arriving during a vacation. Upon completion of the service at a lower rate the server will (a) continue the current vacation (if not finished) or take another vacation (if the working vacation expired) if there are no

* Corresponding author. Tel.: +91 484 2577447.
E-mail addresses: sreenikoonathara@gmail.com (C. Sreenivasan), schakrav@kettering.edu (S.R. Chakravarthy), achyuthacusat@gmail.com (A. Krishnamoorthy).

0307-904X/$ - see front matter © 2012 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.apm.2012.07.054

customers waiting or (b) resume at a normal rate (irrespective of whether the vacation expired or not) if there are customers waiting. Resuming services at a normal rate while the vacation is still in progress corresponds to the vacation being interrupted.

Very recently, Zhang and Hou [9] studied a $MAP/G/1$ queue with working vacations and vacation interruption using supplementary variable method. In this model, the authors assume that the vacation times are exponentially distributed and that the server gets back to normal service mode when at the service (offered during a vacation) completion the system has at least one customer waiting in the queue. The server is allowed to take multiple vacations. In this paper we extend the work of [8] in the following way. First we assume a more versatile point process to model the arrivals. Secondly, we use phase type services which generalizes some of the well-known distributions such as exponential, generalized Erlang, and hyperexponential. Thirdly, we introduce a threshold, say, $1 \leqslant N < \infty$, such that the server offering services (at a lower rate) during a vacation will have the vacation interrupted, the moment the queue size hits N.

In this paper, we consider a single server queueing model in which customers arrive according to a versatile point process, namely, Markovian arrival process (MAP). A *MAP* is a tractable class of Markov renewal processes. It should be noted that by appropriately choosing the parameters of the *MAP* the underlying arrival process can be made as a renewal process. The *MAP* is a rich class of point processes that includes many well-known processes such as Poisson, PH-renewal processes, and Markov-modulated Poisson process. One of the most significant features of the *MAP* is the underlying Markovian structure and fits ideally in the context of matrix-analytic solutions to stochastic models. Matrix-analytic methods were first introduced and studied by Neuts [10]. As is well known, Poisson processes are the simplest and most tractable ones used extensively in stochastic modelling. The idea of the *MAP* is to significantly generalize the Poisson processes and still keep the tractability for modelling purposes. Furthermore, in many practical applications, notably in communications engineering, production and manufacturing engineering, the arrivals do not usually form a renewal process. So, *MAP* is a convenient tool to model both renewal and non-renewal arrivals. While *MAP* is defined for both discrete and continuous times, here we will need only the continuous time case.

The *MAP* in continuous time is described as follows. Let the underlying Markov chain be irreducible and let Q^* be the generator of this Markov chain. At the end of a sojourn time in state i, that is exponentially distributed with parameter λ_i, one of the following two events could occur: with probability $p_{ij}(1)$ the transition corresponds to an arrival and the underlying Markov chain is in state j with $1 \leqslant i,\ j \leqslant m$; with probability $p_{ij}(0)$ the transition corresponds to no arrival and the state of the Markov chain is j, $j \neq i$. Note that the Markov chain can go from state i to state i only through an arrival. Define matrices $D_0 = (d_{ij}^0)$ and $D_1 = (d_{ij}^1)$ such that $d_{ii}^0 = -\lambda_i$, $1 \leqslant i \leqslant m$, $d_{ij}^0 = \lambda_i p_{ij}(0)$, for $j \neq i$ and $d_{ij}^1 = \lambda_i p_{ij}(1)$, $1 \leqslant i,\ j \leqslant m$. By assuming D_0 to be a nonsingular matrix, the interarrival times will be finite with probability one and the arrival process does not terminate. Hence, we see that D_0 is a stable matrix. The generator Q^* is then given by $Q^* = D_0 + D_1$.

Thus, D_0 governs the transitions corresponding to no arrival and D_1 governs those corresponding to an arrival. It can be shown that *MAP* is equivalent to Neuts' versatile Markovian point process. The point process described by the *MAP* is a special class of semi-Markov processes with transition probability matrix given by

$$\int_0^x e^{D_0 t}\, dt D_1 = [I - e^{D_0 x}](-D_0)^{-1} D_1, \quad x \geqslant 0.$$

For use in sequel, let $\boldsymbol{e}(r)$, $\boldsymbol{e}_j(r)$ and I_r denote, respectively, the (column) vector of dimension r consisting of 1's, column vector of dimension r with 1 in the jth position and 0 elsewhere, and an identity matrix of dimension r. When there is no need to emphasize the dimension of these vectors we will suppress the suffix. Thus, $\boldsymbol{e}$ will denote a column vector of 1's of appropriate dimension. The notation $\otimes$ will stand for the Kronecker product of two matrices. Thus, if A is a matrix of order $m \times n$ and B is a matrix of order $p \times q$, then $A \otimes B$ will denote a matrix of order $mp \times nq$, whose (i,j)th block matrix is given by $a_{ij}B$. For more details on Kronecker products and sums, we refer the reader to [11].

Let $\boldsymbol{\pi}$ be the stationary probability vector of the Markov process with generator Q^*. That is, $\boldsymbol{\pi}$ is the unique (positive) probability vector satisfying.

$$\boldsymbol{\pi} Q^* = \mathbf{0}, \quad \boldsymbol{\pi}\boldsymbol{e} = 1. \tag{1}$$

Let $\boldsymbol{\xi}$ be the initial probability vector of the underlying Markov chain governing the *MAP*. Then, by choosing $\boldsymbol{\xi}$ appropriately we can model the time origin to be (a) an arbitrary arrival point; (b) the end of an interval during which there are at least k arrivals; and (c) the point at which the system is in specific state such as the busy period ends or busy period begins. The most interesting case is the one where we get the stationary version of the *MAP* by $\boldsymbol{\xi} = \boldsymbol{\pi}$. The constant $\lambda = \boldsymbol{\pi} D_1 \boldsymbol{e}$, referred to as the **fundamental rate** gives the expected number of arrivals per unit of time in the stationary version of the *MAP*.

Often, in model comparisons, it is convenient to select the time scale of the *MAP* so that λ has a certain value. That is accomplished, in the continuous *MAP* case, by multiplying the coefficient matrices D_0 and D_1, by the appropriate common constant. For further details on *MAP* and their usefulness in stochastic modelling, we refer to [12–14] and for a review and recent work on *MAP* we refer the reader to [15,16].

This paper is organized as follows. In Section 2 we provide a description of the queueing model under study. In Section 3 the steady state analysis of the model is presented. In Section 4 we discuss an illustrative numerical example.

2. Mathematical model

We consider a single server queueing system in which customers arrive according to a Markovian arrival process with parameter matrices D_0 and D_1 of dimension m. An arriving primary customer finding the server free (i.e., on vacation) gets into service immediately but at a lower rate. On the other hand an arriving customer finding the server busy gets into a buffer (of infinite capacity) for the server to become available. The service times follow a phase type distribution with representation $(\boldsymbol{\alpha}, T)$ of order n. When the system becomes empty at the time of a completion of a service, the server will go on a vacation. The duration of a vacation is assumed to be exponentially distributed with parameter η. A vacation is interrupted when a customer arrives during that time. However, the server offers services to those customers arriving during a vacation at a lower rate as compared to the other (regular) customers. We assume that the service times of those customers (served at a lower rate) are also of phase type but with representation $(\boldsymbol{\alpha}, \theta T)$, with $0 < \theta < 1$. The server continues to serve at this rate until either the vacation expires or the queue length hits a pre-determined threshold, say, N, $1 \leqslant N < \infty$. At this instant, the server instantaneously switches over to the normal rate and continues to serve at this rate until the system becomes empty. At the end of a vacation if there is no customer waiting for service, the server takes another vacation. Let μ denote the regular service rate. It is easy to verify that $\mu = [\boldsymbol{\alpha}(-T)^{-1}\boldsymbol{e}]^{-1}$ and the vacation mode of service has rate $\theta\mu$.

3. steady-state analysis

In this section we will discuss the steady-state analysis of the model under study.

3.1. The QBD process

The model described in Section 2 can be studied as a quasi-birth-and-death (*QBD*) process. First, we set up necessary notations. Define $N(t)$ to be the number of customers in the system at time t,

$$S_1(t) = \begin{cases} 0, & \text{if the service is in vacation mode,} \\ 1, & \text{if the service is normal,} \end{cases}$$

$S_2(t)$, the phase of the service process when the server is busy, and $M(t)$ to be the phase of the arrival process at time t. It is easy to verify that $\{(N(t), S_1(t), S_2(t), M(t)) : t \geqslant 0\}$ is a quasi-birth-and-death process (*QBD*) with state space

$$\Omega = \bigcup_{i=0}^{\infty} l(i),$$

where

$$l(0) = \{(0,1), (0,2), \ldots (0,m)\},$$

and for $i \geqslant 1$,

$$l(i) = \{(i, j_1, j_2, k) : j_1 = 0 \text{ or } 1, 1 \leqslant j_2 \leqslant n, 1 \leqslant k \leqslant m\}.$$

Note that when $N(t) = 0$, $S_1(t)$ and $S_2(t)$ do not play any role and will not be tracked. In this case only the state, $M(t)$, of the arrival process needs to be accounted.

The generator, Q, of the *QBD* process under consideration is of the form

$$Q = \begin{pmatrix} D_0 & C_0 & & & & & & & \\ C_2 & B_1 & I \otimes D_1 & & & & & & \\ & \ddots & \ddots & \ddots & & & & & \\ & & B_2 & B_1 & I \otimes D_1 & & & & \\ & & & B_2 & B_1 & \boldsymbol{e} \otimes I \otimes D_1 & & & \\ & & & & \boldsymbol{e}_2'(2) \otimes \boldsymbol{T}^0 \boldsymbol{\alpha} \otimes I & A_1 & A_0 & & \\ & & & & & A_2 & A_1 & A_0 & \\ & & & & & & \ddots & \ddots & \ddots \end{pmatrix}, \tag{2}$$

where the (block) matrices appearing in Q are as follows.

$$C_0 = [\boldsymbol{\alpha} \otimes D_1 \; O], \quad C_2 = \begin{bmatrix} \theta \boldsymbol{T}^0 \otimes I \\ \boldsymbol{T}^0 \otimes I \end{bmatrix},$$

$$B_1 = \begin{bmatrix} \theta T \oplus D_0 - \eta I & \eta I \\ O & T \oplus D_0 \end{bmatrix}, \quad B_2 = \begin{bmatrix} \theta \boldsymbol{T}^0 \boldsymbol{\alpha} \otimes I & O \\ O & \boldsymbol{T}^0 \boldsymbol{\alpha} \otimes I \end{bmatrix},$$

$$A_0 = I \otimes D_1, \; A_1 = T \oplus D_0, \quad A_2 = \boldsymbol{T}^0 \boldsymbol{\alpha} \otimes I. \tag{3}$$

3.2. The steady-state probability vector

Defining $A = A_0 + A_1 + A_2$ and $\boldsymbol{\delta}$ to be the steady-state probability vector of the irreducible matrix A, it is easy to verify that the vector $\boldsymbol{\delta}$ satisfying

$$\boldsymbol{\delta} A = \mathbf{0}, \quad \boldsymbol{\delta}\mathbf{e} = 1, \tag{4}$$

is given by

$$\boldsymbol{\delta} = (\mu\boldsymbol{\alpha}(-T)^{-1} \otimes \boldsymbol{\pi}), \tag{5}$$

where $\boldsymbol{\pi}$ as given in (1).

The condition $\boldsymbol{\delta}A_0\mathbf{e} < \boldsymbol{\delta}A_2\mathbf{e}$ required for the stability of the queueing model under study (see [10]) reduces to $\lambda < \mu$.

Let $\boldsymbol{x}$ be the steady-state probability vector of Q. Partition this vector as:

$$\boldsymbol{x} = (\boldsymbol{x}_0, \boldsymbol{x}_1, \boldsymbol{x}_2 \ldots, \ldots, \boldsymbol{x}_N, \boldsymbol{x}_{N+1}, \ldots),$$

where $\boldsymbol{x}_0$ is of dimension m, $\boldsymbol{x}_1,\ \boldsymbol{x}_2,\ \ldots,\ \boldsymbol{x}_N$ are of dimension $2mn$ and $\boldsymbol{x}_{N+1},\ \boldsymbol{x}_{N+2},\ \ldots$ are of dimension mn.

Under the condition that $\lambda < \mu$, the steady-state probability vector $\boldsymbol{x}$ is obtained (see, e.g. [10]) as follows:

$$\boldsymbol{x}_{N+i} = \boldsymbol{x}_{N+1}R^{i-1}, i \geqslant 1, \tag{6}$$

where the matrix R is the minimal nonnegative solution to the matrix quadratic equation:

$$R^2A_2 + RA_1 + A_0 = 0, \tag{7}$$

and the vectors $\boldsymbol{x}_0,\ \ldots,\ \boldsymbol{x}_{N+1}$ are obtained by solving

$$\begin{aligned}
&\boldsymbol{x}_0D_0 + \boldsymbol{x}_1C_2 = \mathbf{0},\\
&\boldsymbol{x}_0C_0 + \boldsymbol{x}_1B_1 + \boldsymbol{x}_2B_2 = \mathbf{0},\\
&\boldsymbol{x}_{i-1}(I \otimes D_1) + \boldsymbol{x}_iB_1 + \boldsymbol{x}_{i+1}B_2 = \mathbf{0}, \quad 2 \leqslant i \leqslant N-1,\\
&\boldsymbol{x}_{N-1}(I \otimes D_1) + \boldsymbol{x}_NB_1 + \boldsymbol{x}_{N+1}(\mathbf{e}_2'(2) \otimes \boldsymbol{T}^0\boldsymbol{\alpha} \otimes I) = \mathbf{0},\\
&\boldsymbol{x}_N(\mathbf{e} \otimes I \otimes D_1) + \boldsymbol{x}_{N+1}(A_1 + RA_2) = \mathbf{0},
\end{aligned} \tag{8}$$

subject to the normalizing condition

$$\sum_{i=0}^{N}\boldsymbol{x}_i\mathbf{e} + \boldsymbol{x}_{N+1}(I-R)^{-1}\mathbf{e} = 1. \tag{9}$$

The computation of the R matrix can be carried out using a number of well-known methods such as logarithmic reduction. We will list only the main steps involved in the logarithmic reduction algorithm for the computation of R. For full details of the logarithmic reduction algorithm we refer the reader to [17].

Logarithmic Reduction Algorithm for R:

Step 0: $H \leftarrow (-A_1)^{-1}A_0$, $L \leftarrow (-A_1)^{-1}A_2$, $G = L$, and $T = H$.

Step 1:

$$\begin{aligned}
&U = HL + LH\\
&M = H^2\\
&H \leftarrow (I-U)^{-1}M\\
&M \leftarrow L^2\\
&L \leftarrow (I-U)^{-1}M\\
&G \leftarrow G + TL\\
&T \leftarrow TH
\end{aligned}$$

Continue Step 1 until $||\mathbf{e} - G\mathbf{e}||_\infty < \epsilon$.

Step 2: $R = -A_0(A_1 + A_0G)^{-1}$.

The computation of the vectors $\boldsymbol{x}_0,\ \ldots,\ \boldsymbol{x}_{N+1}$ can be carried out by exploiting the special structure of the coefficient matrices and the details are omitted. For use in the sequel, we partition $\boldsymbol{x}_i = (\boldsymbol{u}_i,\ \boldsymbol{v}_i),\ 1 \leqslant i \leqslant N$, where $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$ are of dimension mn.

3.3. The stationary waiting time distribution in the queue

The stationary waiting time distribution in the queue of a customer is derived here. We obtain this by conditioning on the fact that at an arrival epoch the server is serving in normal mode or in vacation mode. First note that an arriving customer will enter into service immediately (at a lower service rate) when the server is on vacation. Otherwise, the customer has to wait before getting into service (either at a lower rate or normal rate).

3.3.1. Conditional waiting time in the queue (normal mode)

Here we condition that an arriving customer finds the server busy serving in normal mode. First note that in this case, the waiting time is always positive. We now define $\boldsymbol{z}_{i,j}$ to be the steady-state probability that an arrival will find the server busy in normal mode with the current service in phase j, and the number of customers in the system including the current arrival to be i, for $1 \leqslant j \leqslant n, i \geqslant 2$. Let $\boldsymbol{z}_i = (\boldsymbol{z}_{i,1}, \boldsymbol{z}_{i,2}, \ldots, \boldsymbol{z}_{i,n})$ and $\boldsymbol{z} = (0, \boldsymbol{z}_2, \boldsymbol{z}_3, \ldots)$. Then it is easy to verify that

$$\boldsymbol{z}_i = \begin{cases} \frac{1}{\lambda}\boldsymbol{v}_{i-1}(I \otimes D_1\mathbf{e}), & 2 \leqslant i \leqslant N, \\ \frac{1}{\lambda}(\boldsymbol{u}_N + \boldsymbol{v}_N)(I \otimes D_1\mathbf{e}), & i = N+1, \\ \frac{1}{\lambda}\boldsymbol{x}_{i-1}(I \otimes D_1\mathbf{e}), & i \geqslant N+2. \end{cases}$$

The waiting time may be viewed as the time until absorption in a Markov chain with a highly sparse structure. The state space (that includes the arriving customer in its count) of this Markov chain is given by $\Omega_1 = \{*\} \cup \{(i,j) : i \geqslant 2, 1 \leqslant j \leqslant n\}$. The state $*$ corresponds to the absorbing state indicating the completion of waiting for the service. It is easy to verify that the generator, $\tilde{Q}_1$, of this Markov process is of the form

$$\tilde{Q}_1 = \begin{pmatrix} 0 & O & & & \\ \boldsymbol{T}^0 & T & & & \\ & \boldsymbol{T}^0\boldsymbol{\alpha} & T & & \\ & & \boldsymbol{T}^0\boldsymbol{\alpha} & T & \\ & & & \ddots & \ddots \end{pmatrix}. \tag{10}$$

Define $W(t), t > 0$ to be the probability that an arriving customer will enter into service no later than time t conditioned on the fact that the service is in normal mode. Let $\widetilde{W}_{normal}(s)$ denote the Laplace–Stieltjes transform of the conditional stationary waiting time in the queue of an arriving customer during the normal service mode. Using the structure of $\tilde{Q}_1$ it can readily be verified that the following result holds good.

Theorem 1. *The LST of the conditional waiting time distribution of an arriving customer, finding the server busy in normal mode, is given by*

$$\widetilde{W}_{normal}(s) = c\sum_{i=2}^{\infty}\boldsymbol{z}_i(sI - T)^{-1}\boldsymbol{T}^0[\boldsymbol{\alpha}(sI - T)^{-1}\boldsymbol{T}^0]^{i-2}, \quad Re(s) \geqslant 0, \tag{11}$$

where the normalizing constant c is given by

$$c = \left[\sum_{i=2}^{\infty}\boldsymbol{z}_i\mathbf{e}\right]^{-1}. \tag{12}$$

Note. The conditional mean waiting time, μ'_{normal}, in the queue of an arrival finding the server to be busy in normal mode soon after the arrival is calculated as

$$\mu'_{normal} = -\widetilde{W}'(0) = c\sum_{i=2}^{\infty}\boldsymbol{z}_i(-T)^{-1}\mathbf{e} + \frac{c}{\mu}\sum_{i=2}^{\infty}(i-2)\boldsymbol{z}_i\boldsymbol{e}.$$

Using the expression for $\boldsymbol{z}_i$, the conditional mean waiting time, μ'_{normal}, can be simplified as

$$\begin{aligned}\mu'_{normal} = {} & \frac{c}{\lambda}\left[\sum_{i=1}^{N}v_i + \boldsymbol{u}_N + \boldsymbol{x}_{N+1}(I-R)^{-1}\right]\left[(-T)^{-1}\mathbf{e} \otimes D_1\mathbf{e}\right] \\ & + \frac{c}{\lambda\mu}\left[\sum_{i=1}^{\infty}\boldsymbol{v}_i + (N-1)\boldsymbol{u}_N + N\boldsymbol{x}_{N+1}(I-R)^{-1} + \boldsymbol{x}_{N+1}R(I-R)^{-2}\right][\mathbf{e} \otimes D_1\mathbf{e}].\end{aligned} \tag{13}$$

3.3.2. Conditional stationary waiting time in the queue (vacation mode)

The conditional stationary waiting time in the queue of an arriving customer given that the server is busy in vacation mode is derived here. First, observe that the waiting time in the queue of an arriving customer is zero with probability $z_0 = \frac{1}{\lambda}\boldsymbol{x}_0 D_1\mathbf{e}$. Let $w_{i,j_2,k},\ 1 \leqslant i \leqslant N,\ 1 \leqslant j_2 \leqslant n,\ 1 \leqslant k \leqslant m$, denote the steady-state probability that immediately after the arrival the customer will find the server busy serving in vacation mode with the service in phase j_2 and the number of customers in the system (including the current arrival) to be i, and the arrival process to be in phase k. Let $\boldsymbol{w}_i = (w_{i,1,1}, \ldots, w_{i,n,m})$. It is easy to verify that

$$\boldsymbol{w}_i = \begin{cases} \frac{1}{\lambda}\boldsymbol{x}_0(\boldsymbol{\alpha} \otimes D_1), & i = 1, \\ \frac{1}{\lambda}\boldsymbol{u}_{i-1}(I \otimes D_1), & 2 \leqslant i \leqslant N. \end{cases}$$

Observe that the conditional waiting time in the queue of an arriving customer finding the server busy in vacation mode soon after the arrival depends on the future arrivals due to the threshold placed on the system for bringing back the service rate to normal. Thus, we need to keep track of the phase of the arrival process up until the service rate becomes normal due either to meeting the threshold or the vacation expiring. Towards this end, we define the following set of states.

Let $(i,j,j_2,k),\ 1 \leqslant i \leqslant N-1,\ 1 \leqslant j \leqslant i,\ 1 \leqslant j_2 \leqslant n,\ 1 \leqslant k \leqslant m$, denote the state that corresponds to the server being in vacation mode with i customers in the queue; the arriving customer's position in the queue is j; the current service is in phase j_2, and the arrival process is in phase k. Define $(i^*,j_2) : 1 \leqslant i^* \leqslant N-1, 1 \leqslant j_2 \leqslant n$ to be the state that corresponds to the server serving in normal mode and the position of the tagged customer in the queue being i^* and the current service in phase j_2.

Let $\boldsymbol{i} = \{(i,j,j_2,k), 1 \leqslant j \leqslant i, 1 \leqslant j_2 \leqslant n, 1 \leqslant k \leqslant m\},\ 1 \leqslant i \leqslant N-1$, and $\boldsymbol{i}^* = \{(i^*,j_2), 1 \leqslant j_2 \leqslant n\},\ 1 \leqslant i^* \leqslant N-1$.

Before we formally state the result we need the following notations. Define:

- I_r is an identity matrix of dimension r.
- $\widehat{I}_r$ is a matrix of dimension $r \times N-1$ of the form

$$\hat{I}_r = (I_r \quad 0), \quad 1 \leqslant r \leqslant N-1.$$

- $\bar{I}_r$ is a matrix of dimension $r \times r-1$ of the form

$$\bar{I}_r = \begin{pmatrix} 0 \\ I_{r-1} \end{pmatrix}, \quad 2 \leqslant r \leqslant N-1.$$

- $\widetilde{I}_r$ is a matrix of dimension $r \times r+1$ of the form

$$\widetilde{I}_r = (I_r \quad 0), \quad 1 \leqslant r \leqslant N-2.$$

Let

$$L_{1,1} = \begin{pmatrix} T & & & & \\ \boldsymbol{T}^0\boldsymbol{\alpha} & T & & & \\ & \boldsymbol{T}^0\boldsymbol{\alpha} & T & & \\ & \ddots & \ddots & & \\ & & & \boldsymbol{T}^0\boldsymbol{\alpha} & T \end{pmatrix}, \quad L_{2,1} = \begin{pmatrix} \eta\widehat{I}_1 \otimes I \otimes \boldsymbol{e} \\ \eta\widehat{I}_2 \otimes I \otimes \boldsymbol{e} \\ \vdots \\ \eta\widehat{I}_{N-2} \otimes I \otimes \boldsymbol{e} \\ I_{N-1} \otimes (\eta I \otimes \boldsymbol{e} + I \otimes D_1\boldsymbol{e}) \end{pmatrix}, \tag{14}$$

$$L_{2,2} = \begin{pmatrix} \widetilde{B}_1 & \widetilde{I}_1 \otimes I \otimes D_1 & & & \\ \theta\bar{I}_2 \otimes \boldsymbol{T}^0\boldsymbol{\alpha} \otimes I & I_2 \otimes \widetilde{B}_1 & \widetilde{I}_2 \otimes I \otimes D_1 & & \\ & \theta\bar{I}_3 \otimes \boldsymbol{T}^0\boldsymbol{\alpha} \otimes I & I_3 \otimes \widetilde{B}_1 & \widetilde{I}_3 \otimes I \otimes D_1 & \\ & & \ddots & \ddots & \\ & & & \theta\bar{I}_{N-1} \otimes \boldsymbol{T}^0\boldsymbol{\alpha} \otimes I & I_{N-1} \otimes \widetilde{B}_1 \end{pmatrix}, \tag{15}$$

and

$$\widetilde{B}_1 = (\theta T \oplus D_0) - \eta I. \tag{16}$$

Under this setup, one can readily verify the following result.

Theorem 2. *The conditional waiting time distribution in the queue of an arriving customer finding the server in vacation mode soon after the arrival is of phase type with representation* $(\boldsymbol{\gamma}, L)$ *of order* $[(N-1)n + 0.5N(N-1)mn]$, *where*

$$\boldsymbol{\gamma} = d(\boldsymbol{0}, \boldsymbol{w}_2, \boldsymbol{e}_2'(2) \otimes \boldsymbol{w}_3, \boldsymbol{e}_3'(3) \otimes \boldsymbol{w}_4, \cdots, \boldsymbol{e}_{N-1}'(N-1) \otimes \boldsymbol{w}_N), \tag{17}$$

and

$$L = \begin{pmatrix} L_{1,1} & 0 \\ L_{2,1} & L_{2,2} \end{pmatrix}, \tag{18}$$

where the normalizing constant is given by $d = \left[\sum_{i=1}^{N} \boldsymbol{w}_i\boldsymbol{e}\right]^{-1}$.

Note. The conditional mean waiting time, $\mu'_{vacation}$, in the queue of an arrival finding the server to be busy during vacation mode soon after the arrival is calculated as $\mu'_{vacation} = \boldsymbol{\gamma}(-L)^{-1}\boldsymbol{e}$. The computation of this mean is achieved by exploiting the special structure of $\boldsymbol{\gamma}$ and L. We will briefly present the steps involved in this.

Defining

$$\boldsymbol{\gamma}(-L)^{-1} = (\boldsymbol{a}, \boldsymbol{b}),$$

and partitioning the vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ as

$$\boldsymbol{a} = (\boldsymbol{a}_1, \cdots, \boldsymbol{a}_{N-1}),$$

$$\boldsymbol{b} = (\boldsymbol{b}_{1,1}, \boldsymbol{b}_{2,1}, \boldsymbol{b}_{2,2}, \ldots, \boldsymbol{b}_{N-1,1}, \ldots, \boldsymbol{b}_{N-1,N-1}),$$

where $\boldsymbol{a}_i,\ 1 \leqslant i \leqslant N-1$, is of dimension n and $\boldsymbol{b}_{i,j},\ 1 \leqslant j \leqslant i,\ 1 \leqslant i \leqslant N-1$, is of dimension of mn, the mean $\mu'_{vacation}$ is given by

$$\mu'_{vacation} = \sum_{i=1}^{N-1} \left[\boldsymbol{a}_i \boldsymbol{e} + \sum_{j=1}^{i} \boldsymbol{b}_{i,j} \boldsymbol{e} \right].$$

The vectors $\boldsymbol{a}_i, 1 \leqslant i \leqslant N-1$, and $\boldsymbol{b}_{i,j},\ 1 \leqslant j \leqslant i,\ 1 \leqslant i \leqslant N-1$, are ideally suited for solving using any of the well-known methods such as (block) Gauss–Seidel. The necessary equations are as follows:

$$\boldsymbol{a}_1 = \boldsymbol{a}_2 \boldsymbol{T}^0 \boldsymbol{\alpha} (-T)^{-1} + \eta \sum_{r=1}^{N-1} \boldsymbol{b}_{r,1} (-T^{-1} \otimes \boldsymbol{e}) + \boldsymbol{b}_{N-1,1} (-T^{-1} \otimes D_1 \boldsymbol{e}),$$

$$\boldsymbol{a}_i = \boldsymbol{a}_{i+1} \boldsymbol{T}^0 \boldsymbol{\alpha} (-T)^{-1} + \eta \sum_{r=i}^{N-1} \boldsymbol{b}_{r,i} (-T^{-1} \otimes \boldsymbol{e}) + \boldsymbol{b}_{N-1,i} (-T^{-1} \otimes D_1 \boldsymbol{e}), \quad 2 \leqslant i \leqslant N-2,$$

$$\boldsymbol{a}_{N-1} = \eta \boldsymbol{b}_{N-1,N-1} (-T^{-1} \otimes \boldsymbol{e}) + \boldsymbol{b}_{N-1,N-1} (-T^{-1} \otimes D_1 \boldsymbol{e}),$$

$$\boldsymbol{b}_{1,1} = [\boldsymbol{w}_2 + \theta \boldsymbol{b}_{2,2} (\boldsymbol{T}^0 \boldsymbol{\alpha} \otimes I)](-\widetilde{B}_1)^{-1},$$

$$\boldsymbol{b}_{i,1} = [\boldsymbol{b}_{i-1,1} (I \otimes D_1) + \theta \boldsymbol{b}_{i+1,2} (\boldsymbol{T}^0 \boldsymbol{\alpha} \otimes I)](-\widetilde{B}_1)^{-1}, \quad 2 \leqslant i \leqslant N-2,$$

$$\boldsymbol{b}_{i,j} = [\boldsymbol{b}_{i-1,j} (I \otimes D_1) + \theta \boldsymbol{b}_{i+1,j+1} (\boldsymbol{T}^0 \boldsymbol{\alpha} \otimes I)](-\widetilde{B}_1)^{-1}, \quad 2 \leqslant j \leqslant i-1,\ 3 \leqslant i \leqslant N-2,$$

$$\boldsymbol{b}_{i,i} = [\boldsymbol{w}_{i+1} + \theta \boldsymbol{b}_{i+1,i+1} (\boldsymbol{T}^0 \boldsymbol{\alpha} \otimes I)](-\widetilde{B}_1)^{-1}, \quad 2 \leqslant i \leqslant N-2,$$

$$\boldsymbol{b}_{N-1,j} = \boldsymbol{b}_{N-2,j} (I \otimes D_1)(-\widetilde{B}_1)^{-1}, \quad 1 \leqslant j \leqslant N-2,$$

$$\boldsymbol{b}_{N-1,N-1} = \boldsymbol{w}_N (-\widetilde{B}_1)^{-1},$$

subject to the condition

$$\boldsymbol{a}_1 \boldsymbol{T}^0 + \theta \sum_{i=1}^{N-1} \boldsymbol{b}_{i,1} (\boldsymbol{T}^0 \otimes \boldsymbol{e}) = 1 - d \boldsymbol{w}_1 \boldsymbol{e}.$$

3.3.3. The stationary waiting time in the queue

From the knowledge of conditional stationary waiting time in the queue, one can get the (unconditional) stationary waiting time in the queue and the details are omitted.

Note. The (unconditional) mean, μ'_{WTQ}, waiting time of a customer in the queue is obtained as

$$\mu'_{WTQ} = \frac{1}{\lambda} \left[\sum_{i=1}^{N} \boldsymbol{v}_i + \boldsymbol{u}_N + x_{N+1} (I-R)^{-1} \right] [(-T)^{-1} \boldsymbol{e} \otimes D_1 \boldsymbol{e}] + \frac{1}{\lambda \mu} \left[\sum_{i=1}^{\infty} \boldsymbol{v}_i + (N-1) \boldsymbol{u}_N + N x_{N+1} (I-R)^{-1} + x_{N+1} R (I-R)^{-2} \right] [\boldsymbol{e} \otimes D_1 \boldsymbol{e}] + \frac{1}{d} \sum_{i=1}^{N-1} \left[\boldsymbol{a}_i \boldsymbol{e} + \sum_{j=1}^{i} \boldsymbol{b}_{i,j} \boldsymbol{e} \right]. \tag{19}$$

3.4. Analysis of slow service mode

In this section we will discuss the duration of the server spending in slow service mode as well as the number of visits to level 0 before hitting normal service mode.

3.4.1. The duration in slow service mode

The duration, T_{slow}, in slow service mode is defined as the time the server starts in slow service mode (through initiating a working vacation) until either the server takes another vacation or the server gets back to normal mode through the working

vacation expiring. In this section we will show that the random variable T_{slow} can be studied as the time until absorption in a finite state continuous time Markov chain with two absorbing states. We first define

$$\boldsymbol{\gamma}_M = c_1(\boldsymbol{\alpha} \otimes \boldsymbol{x}_0 D_1, \boldsymbol{0}),$$

$$M = \begin{pmatrix} \widetilde{B}_1 & I \otimes D_1 & & & \\ \theta(\boldsymbol{T}^0\boldsymbol{\alpha} \otimes I) & \widetilde{B}_1 & I \otimes D_1 & & \\ & \theta(\boldsymbol{T}^0\boldsymbol{\alpha} \otimes I) & \widetilde{B}_1 & I \otimes D_1 & \\ & \ddots & \ddots & & \\ & & & \theta(\boldsymbol{T}^0\boldsymbol{\alpha} \otimes I) & \widetilde{B}_1 \end{pmatrix},$$

$$\boldsymbol{M}_1^0 = \begin{pmatrix} \theta(\boldsymbol{T}^0 \otimes \boldsymbol{e}) \\ \boldsymbol{0} \\ \vdots \\ \boldsymbol{0} \end{pmatrix}, \quad \boldsymbol{M}_2^0 = \begin{pmatrix} \eta\boldsymbol{e} \\ \eta\boldsymbol{e} \\ \vdots \\ \eta\boldsymbol{e} \\ \eta\boldsymbol{e} + (\boldsymbol{e} \otimes D_1\boldsymbol{e}) \end{pmatrix},$$

where $c_1 = [\boldsymbol{x}_0 D_1 \boldsymbol{e}]^{-1}$ is the normalizing constant and $\widetilde{B}_1$ is as given in (16). The matrix M is of dimension Nmn. First note that the probability, p_{slow}, that the server will serve only in slow mode before taking another vacation is given by $p_{slow} = \boldsymbol{\gamma}_M(-M)^{-1}\boldsymbol{M}_1^0$. We now have the following result.

Theorem 3. *The (conditional) probability density function of T_{slow}, conditioned on the fact that the slow service mode ends through the server taking another vacation, is given by*

$$f_{T_{slow}}(y) = \frac{1}{p_{slow}} \boldsymbol{\gamma}_M e^{My} \boldsymbol{M}_1^0, \quad y \geqslant 0. \tag{20}$$

Given that the slow service mode ends through the server taking another vacation, the (conditional) mean time spent in slow mode can be calculated as

$$\mu'_{SM} = \frac{1}{p_{slow}} \boldsymbol{\gamma}_M (-M)^{-2} \boldsymbol{M}_1^0. \tag{21}$$

Note. 1. The special structure of $\boldsymbol{\gamma}_M$, M, and $\boldsymbol{M}_1^0$ is to be exploited when computing this mean. The details are similar to the computation of $\mu'_{vacation}$ and hence omitted.

2. By a similar argument we can get the (conditional) probability density function of T_{slow} and the conditional mean, conditioned on the fact that the server ends the slow service mode by entering into the normal rate. The details are omitted.

3.4.2. The distribution of the number of visits to level 0 before hitting normal service mode

We consider the queueing system at an arrival epoch that finds the server in vacation mode. At this instant the service will start in slow mode. The quantity that is of interest here is the probability mass function, $\{p_k, k \geqslant 0\}$, of the number of visits to level **0** before hitting normal service mode. This mass function and its associated measures such as mean and standard deviation, play an important role in the qualitative study of the model under consideration. Using the set up in 3.4.1 it can easily be verified that

$$p_k = \boldsymbol{\gamma}_M (-M)^{-1} B^k \boldsymbol{M}_2^0, k \geqslant 0, \tag{22}$$

where

$$B = \theta\Big[(\boldsymbol{e}_1\boldsymbol{e}_1' \otimes \boldsymbol{T}^0\boldsymbol{\alpha} \otimes (-D_0)^{-1}D_1)\Big](-M)^{-1}. \tag{23}$$

Note. It is easy to see that the mean number of visits, μ_{NVZ}, to level **0** before hitting level $N+1$ is obtained as

$$\mu_{NVZ} = \boldsymbol{\gamma}_M (-M)^{-1} B (I - B)^{-2} M_2^0. \tag{24}$$

The computation of μ_{NVZ} can be carried out by exploiting the special structure of $\boldsymbol{\gamma}_M$, M, and B. Below, we will outline only the main steps. Towards this end, we first define

$$\boldsymbol{\gamma}_M (-M)^{-1} = (d_1, \ldots, d_N),$$

where the vectors $d_i,\ 1 \leqslant i \leqslant N$, are of dimension nm, and their computation is very similar to the one discussed in finding $\mu'_{vacation}$. From Eq. (23) it is clear that B is of the form

$$B = \begin{pmatrix} B_1 & B_2 & B_N \\ 0 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 0 & 0 & 0 \end{pmatrix},$$

where the matrices $B_i,\ 1 \leqslant i \leqslant N$, of order nm are obtained by solving the following equations that are ideally suited for any of the well-known methods such as (block) Gauss–Seidel.

$$B_1 = \theta[B_2(T^0\boldsymbol{\alpha} \otimes I) + (T^0\boldsymbol{\alpha} \otimes (-D_0)^{-1}D_1)](-\widetilde{B}_1)^{-1},$$

$$B_i = [B_{i-1}(I \otimes D_1) + \theta B_{i+1}(T^0\boldsymbol{\alpha} \otimes I)](-\widetilde{B}_1)^{-1}, \quad 2 \leqslant i \leqslant N-1,$$

$$B_N = B_{N-1}(I \otimes D_1)(-\widetilde{B}_1)^{-1},$$

subject to the condition

$$\theta B_1(T^0 \otimes e) + B_N(e \otimes D_1 e) + \eta\sum_{i=1}^{N} B_i e = \theta(T^0 \otimes e),$$

and $\widetilde{B}_1$ is as given in (16).

Using the facts that

$$p_{slow} = \theta d_1(T^0 \otimes e) \quad \text{and} \quad \mu_{NVZ} = \gamma_M(-M)^{-1}(I-B)^{-2}M\begin{smallmatrix}0\\2\end{smallmatrix} - 1,$$

and the special form of B, it can easily be verified that

$$\mu_{NVZ} = \theta d_1(I - B_1)^{-1}(T^0 \otimes e).$$

3.4.3. The uninterrupted vacation time

The uninterrupted vacation time is defined as the duration that begins with the server becoming idle (and thus starts a vacation) until a new arrival interrupts the vacation. It is easy to verify that this duration is of phase type with representation (ξ, D_0) of dimension m, where $\xi = c_2(\theta u_1 + \nu_1)(T^0 \otimes I)$ and c_2 is the normalizing constant given by $c_2 = [(\theta u_1 + \nu_1)(T^0 \otimes e)]^{-1}$. The mean, μ_{UIV}, is calculated as $\mu_{UIV} = \xi(-D_0)^{-1}e$.

3.5. Key system performance measures

In this section we list a number of key system performance measures to bring out the qualitative aspects of the model under study. Note that these are in addition to the ones such as the conditional mean waiting times and mean waiting time listed above. The measures are listed below along with their formulae for computation:

1. The probability that the system is idle: $P_{IDLE} = x_0 e$.
2. The probability that the server is serving at a lower rate: $P_{LR} = \sum_{i=1}^{N} u_i e$.
3. The probability that the server is serving at a normal rate rate: $P_{NR} = \sum_{i=1}^{N} \nu_i e + x_{N+1}(I-R)^{-1}e$.
4. The mean number of customers in the system: $\mu_{NS} = \sum_{i=1}^{N} i(u_i + \nu_i)e + Nx_{N+1}(I-R)^{-1}e + x_{N+1}(I-R)^{-2}e$.

4. Numerical results

In order to bring out the qualitative nature of the model under study, we present a few representative examples in this section. For the arrival process we consider the following five sets of matrices for D_0 and D_1:

1. Erlang (*ERA*)

$$D_0 = \begin{pmatrix} -5 & 5 & & & & \\ & -5 & 5 & & & \\ & & -5 & 5 & & \\ & & & -5 & 5 & \\ & & & & -5 & 5 \\ & & & & & -5 \end{pmatrix} \quad D_1 = \begin{pmatrix} & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \\ & & & & & \\ 5 & & & & & \end{pmatrix}$$

2. Exponential (*EXA*)

$$D_0 = (-1), \quad D_1 = (1)$$

3. Hyperexponential (*HEA*)

$$D_0 = \begin{pmatrix} -10 & 0 \\ 0 & -1 \end{pmatrix}, \quad D_1 = \begin{pmatrix} 9 & 1 \\ 0.9 & 0.1 \end{pmatrix}$$

4. MAP with negative correlation (*MNA*)

$$D_0 = \begin{pmatrix} -2 & 2 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -450.5 \end{pmatrix}, \quad D_1 = \begin{pmatrix} 0 & 0 & 0 \\ 0.02 & 0 & 1.98 \\ 445.995 & 0 & 4.505 \end{pmatrix}$$

5. MAP with positive correlation (*MPA*)

$$D_0 = \begin{pmatrix} -2 & -2 & 0 \\ 0 & -2 & 0 \\ 0 & 0 & -450.5 \end{pmatrix}, \quad D_1 = \begin{pmatrix} 0 & 0 & 0 \\ 1.98 & 0 & 0.02 \\ 4.505 & 0 & 445.995 \end{pmatrix}.$$

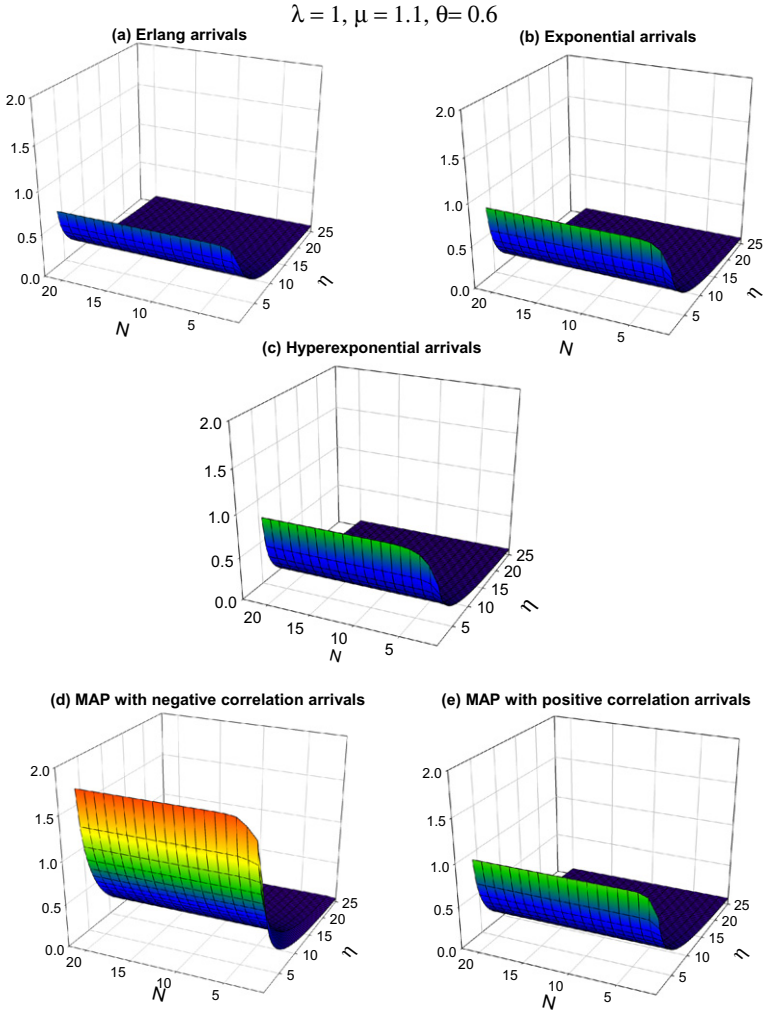

Fig. 1. Mean duration in slow mode – Erlang services.

All these five *MAP* processes are normalized so as to have an arrival rate of 1. However, these are qualitatively different in that they have different variance and correlation structure. The first three arrival processes, namely *ERA*, *EXA*, and *HEA*, correspond to renewal processes and so the correlation is 0. The arrival process labelled *MNA* has correlated arrivals with correlation between two successive inter-arrival times given by −0.4889 and the arrival process corresponding to the one labelled *MPA* has a positive correlation with value 0.4889. The ratio of the standard deviations of the inter-arrival times of these five arrival processes with respect to *ERA* are, respectively, 1, 2.2361, 5.0194, 3.1518, and 3.1518.

For the service time distribution we consider the following two phase type distributions.

1. Erlang (*ERS*)

$$\boldsymbol{\alpha} = (1,0), \quad T = \begin{pmatrix} -2 & 2 \\ 0 & -2 \end{pmatrix}$$

2. Hyperexponential (*HES*)

$$\boldsymbol{\alpha} = (0.9,0.1), \quad T = \begin{pmatrix} -1.90 & 0 \\ 0 & -0.19 \end{pmatrix}$$

These above two distributions will be normalized to have a specific mean in our illustrative example. Note that these are qualitatively different in that they have different variances. The ratio of the standard deviation of *HES* to that of *ERS* is 3.1745.

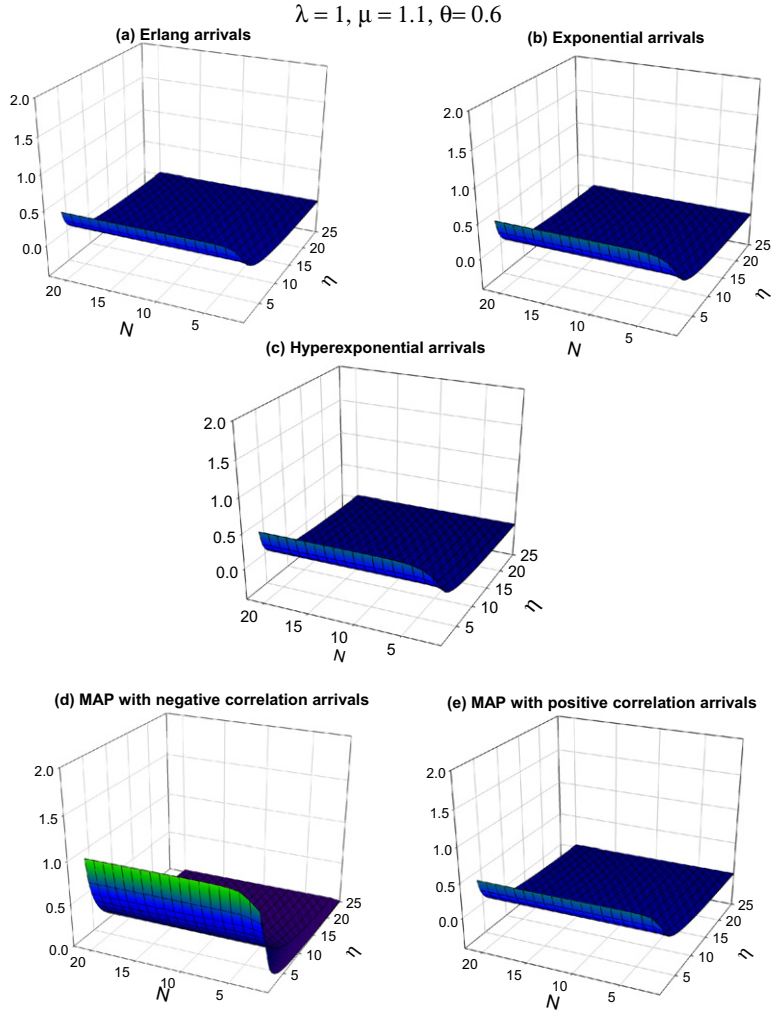

Fig. 2. Mean duration in slow mode – hyperexponential services.

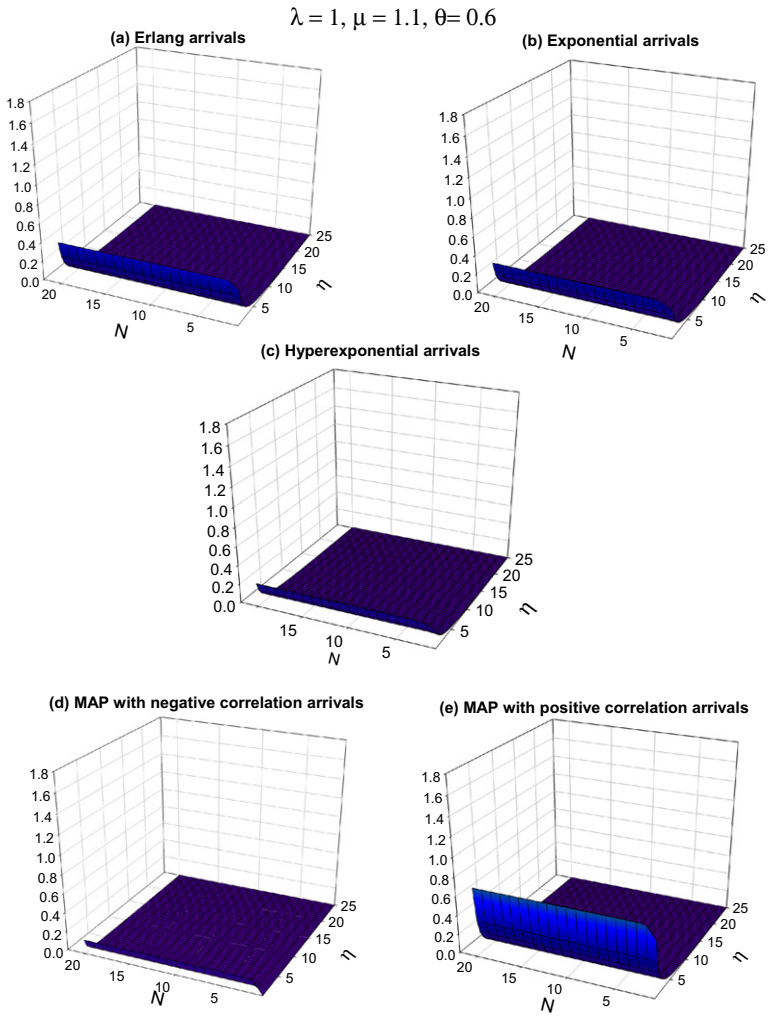

Fig. 3. Mean number of visits to level zero – Erlang services.

Illustrative example: The purpose of this example is to see how various system performance measure behave under different scenarios. We fix $\lambda = 1,\ \mu = 1.1$, and $\theta = 0.6$. First we look at the effect of varying N and η on the performance measures: (conditional) mean duration of service in slow mode which ends in the server taking another vacation and the mean number of visits to level zero before hitting the normal service mode. Due to space restriction, we will display only a few figures and others are available upon request from the authors. In the following we summarize the observations based on various graphs of these performance measures.

- An increase in η leads to a decrease in the mean duration of vacation. Hence a switch from the lower service rate to the normal one occurs more frequently. Once the service rate is brought back to normal, the server clears out the customers at a faster rate. So the measure, P_{idle}, appears to increase as η increases. This is true for all values of N and for all combinations of arrival and service processes under study. As N increases the duration of vacation mode of service gets extended, as is expected. Due to the slow service rate the customers get accumulated faster. So P_{idle} decreases until the service rate gets to normal. Also note that the probability, P_{LR}, that the server is serving at a low rate increases as N is increased (for fixed η) for all combinations of arrival and service times. This in turn will cause the probability, P_{NR}, of the server serving under normal mode to decrease as N increases. As expected, the measure P_{NR} appears to increase with increasing η. When comparing the mean duration of service in slow mode (see Figs. 1 and 2), we notice (for fixed N and η) that *HES* services yield a lower value as opposed to *ERS* services. This is the case for all five arrival processes considered.
- Referring to Figs. 3 and 4, we note that as η increases, the measure μ_{NVZ} appears to decrease in all cases, as expected, for any fixed N. Among renewal arrivals,those with larger variation yields a smaller value for this measure. That is, *HEA* has a smaller value compared to *EXA* and *EXA* has a smaller value compared to *ERA*. Among correlated arrivals,

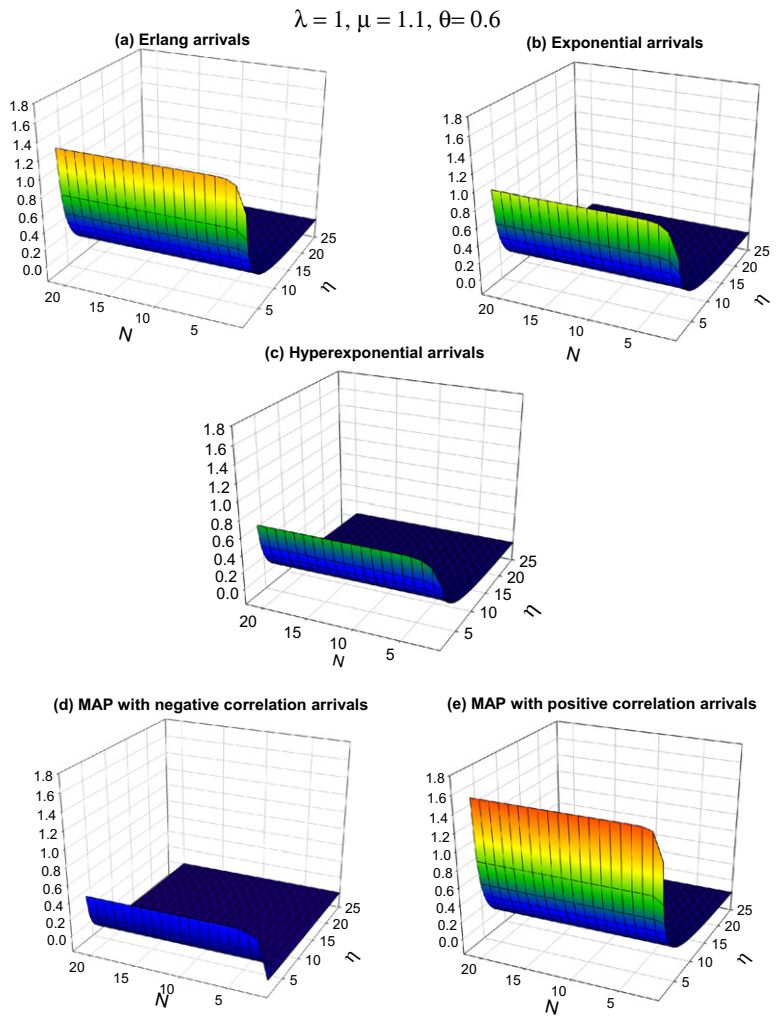

Fig. 4. Mean number of visits to level zero – hyperexponential services.

MPA has a higher value than *MNA*. It is worth pointing out that both *MNA* and *MPA* processes have the same mean and variance, but *MPA* has a positive correlation while *MNA* has a negative correlation. This indicates the significant role played by correlation. As N increases, this measure appears to increase monotonically to a limiting value (which depends on η as well as on the arrival and service time distributions). It should be noted that the rate of approach is higher for larger values of η. That is, the impact of N on this measure decreases as η increases. We notice that this measure appears to have a larger value when services are changed from Erlang to hyperexponential. When comparing this measure (for fixed N and η), we notice that *HES* services yield a higher value as opposed to *ERS* services. This is the case for all five arrival processes considered.

Now we look at the unconditional mean waiting time, μ'_{WTQ}, in the queue of a customer. The values of this measure as functions of N and η under different scenarios are displayed in Table 1. Some key observations are as follows.

- As is to be expected, the mean is a non-increasing function of η (for fixed N) and is a non-decreasing function of N (for fixed η). This is the case for all combinations of arrival and service processes. However, the rate of change is much smaller in the case of *MPA* arrivals as compared to the other arrivals.
- The mean is significantly large for *MPA* case indicating the role played by the (positively) correlated arrivals.
- For all except *MPA* arrivals, we notice the mean changes significantly as a function of η when N becomes large. This is due to the fact that for large N the mean waiting time can only be reduced through an increase in η (which will decrease the duration of the slow service period).

Table 1
The unconditional mean waiting time in the queue (μ'_{WTQ}).

N	η	Erlang services					Hyperexponential services				
		ERA	*EXA*	*HEA*	*MNA*	*MPA*	*ERA*	*EXA*	*HEA*	*MNA*	*MPA*
1	0.1	3.21	6.97	25.21	7.07	497.45	23.82	27.64	46.32	27.63	518.10
	0.2	3.20	6.96	25.21	7.07	497.44	23.81	27.63	46.31	27.63	518.08
	0.3	3.19	6.96	25.21	7.07	497.44	23.80	27.62	46.31	27.63	518.07
	0.4	3.18	6.95	25.20	7.07	497.43	23.79	27.61	46.30	27.63	518.06
	0.5	3.18	6.95	25.20	7.07	497.43	23.78	27.61	46.30	27.63	518.05
2	0.1	3.50	7.19	25.39	7.42	497.57	24.13	27.90	46.51	28.05	518.27
	0.2	3.46	7.16	25.37	7.40	497.55	24.08	27.86	46.48	28.02	518.22
	0.3	3.42	7.14	25.36	7.38	497.52	24.03	27.82	46.46	27.99	518.18
	0.4	3.39	7.12	25.34	7.36	497.50	23.99	27.79	46.44	27.96	518.15
	0.5	3.36	7.10	25.33	7.35	497.49	23.96	27.77	46.43	27.94	518.12
3	0.1	3.83	7.45	25.59	7.57	497.68	24.43	28.17	46.72	28.24	518.40
	0.2	3.73	7.38	25.55	7.52	497.61	24.31	28.08	46.66	28.17	518.31
	0.3	3.64	7.32	25.51	7.48	497.57	24.22	28.00	46.62	28.11	518.24
	0.4	3.56	7.27	25.48	7.44	497.54	24.14	27.94	46.58	28.06	518.19
	0.5	3.50	7.23	25.45	7.41	497.52	24.08	27.90	46.55	28.02	518.15
4	0.1	4.16	7.73	25.81	7.90	497.74	24.70	28.43	46.93	28.57	518.50
	0.2	3.96	7.59	25.72	7.79	497.65	24.50	28.27	46.83	28.42	518.36
	0.3	3.80	7.49	25.65	7.70	497.59	24.36	28.15	46.76	28.31	518.27
	0.4	3.68	7.40	25.60	7.63	497.55	24.24	28.06	46.70	28.23	518.21
	0.5	3.58	7.34	25.55	7.57	497.53	24.15	27.99	46.65	28.16	518.16
5	0.1	4.46	8.00	26.03	8.14	497.79	24.94	28.68	47.15	28.79	518.58
	0.2	4.14	7.78	25.89	7.96	497.67	24.66	28.44	46.99	28.57	518.40
	0.3	3.92	7.62	25.78	7.82	497.60	24.46	28.27	46.88	28.42	518.29
	0.4	3.75	7.50	25.70	7.71	497.56	24.31	28.15	46.79	28.30	518.22
	0.5	3.63	7.41	25.63	7.63	497.53	24.19	28.05	46.72	28.21	518.17
10	0.1	5.54	9.10	26.99	9.29	497.84	25.82	29.61	48.06	29.76	518.77
	0.2	4.61	8.36	26.48	8.59	497.69	25.08	28.94	47.55	29.11	518.46
	0.3	4.12	7.95	26.17	8.19	497.62	24.67	28.56	47.24	28.73	518.32
	0.4	3.85	7.70	25.96	7.95	497.57	24.42	28.32	47.04	28.50	518.23
	0.5	3.68	7.54	25.81	7.79	497.54	24.25	28.16	46.90	28.34	518.18

5. Concluding remarks

In this paper we extended the work of Li and Tian [8] to MAP arrivals and phase type services. We introduced the *N*-policy to return the server to normal mode from vacationing one. An illustrative numerical example to bring out the qualitative nature of the model was presented.

Acknowledgments

The authors thank the anonymous referees for their valuable suggestions and comments which greatly improved the presentation of the paper. C. Sreenivasan's research is supported by the University Grants Commission, Govt. of India, under Faculty Development Program (Grant No. F.FIP/11th Plan/KLCA042TF02).

References

[1] B.T. Doshi, Queueing systems with vacations – a survey, Queue. Syst. 1 (1986) 29–66.
[2] N. Tian, Z.G. Zhang, Vacation Queueing Models: Theory and applications, Springer Publishers, New York, 2006.
[3] L. Servi, S. Finn, *M*/*M*/1 queue with working vacations (*M*/*M*/1/*WV*), Perform. Eval. 50 (2002) 41–52.
[4] J. Kim, D. Choi, K. Chae, Analysis of queue-length distribution of the *M*/*G*/1 queue with working vacations, in: International Conference on Statistics and Related Fields, Hawaii, 2003.
[5] D. Wu, H. Takagi, *M*/*G*/1 queue with multiple working vacations, Perform. Eval. 63 (2006) 654–681.
[6] Y. Baba, Analysis of a *GI*/*M*/1 queue with multiple working vacations, Oper. Res. Lett. 33 (2005) 201–209.
[7] N. Tian, J. Li, Z.G. Zhang, Matrix-analytic method and working vacation queues – a survey, Int. J. Inform. Manage. Sci. 20 (2009) 603–633.
[8] J. Li, N. Tian, The *M*/*M*/1 queue with working vacations and vacation interruptions, J. Syst. Sci. Syst. Eng. 16 (1) (2007) 121–127.
[9] M. Zhang, Z. Hou, Performance analysis of *MAP*/*G*/1 queue with working vacations and vacation interruption, Appl. Math. Modell. 35 (2011) 1551–1560.
[10] M.F. Neuts, Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach, The Johns Hopkins University Press, Baltimore, MD, 1981 (1994 version is Dover Edition).
[11] M. Marcus, H. Minc, A Survey of Matrix Theory and Matrix Inequalities, Allyn and Bacon, Boston, MA, 1964.
[12] D.M. Lucantoni, New results on the single server queue with a batch Markovian arrival process, Stochastic Models 7 (1991) 1–46.
[13] M.F. Neuts, Structured Stochastic Matrices of *M*/*G*/1 Type and Their Applications, Marcel Dekker, NY, 1989.
[14] M.F. Neuts, Models based on the Markovian arrival process, IEICE Trans. Commun. E75B (1992) 1255–1265.

[15] S.R. Chakravarthy, The batch markovian arrival process: a review and future work, in: A. Krishnamoorthy, et al. (Eds.), Advances in Probability Theory and Stochastic Processes, Notable Publications, Inc., New Jersey, 2001, pp. 21–49.
[16] S.R. Chakravarthy, Markovian arrival processes, Wiley Encyclopedia of Operations Research and Management Science, 15 June 2010.
[17] G. Latouche, V. Ramaswami, Introduction to Matrix Analytic Methods in Stochastic Modeling, SIAM, 1999.