9-27-2016

MAP/PH/1 queueing model with working vacation and crowdsourcing

Srinivas R. Chakravarthy
Kettering University, schakrav@kettering.edu

Serife Ozkar
Dokuz Eylul University, serife.ozkar@deu.edu.tr

MATHEMATICA APPLICANDA
Vol. 44(2) 2016, p. 263–294
doi: 10.14708/ma.v44i2.1244

Srinivas R. Chakravarthy (Flint)
Serife Ozkar (Izmir)

$MAP/PH/1$ queueing model with working vacation and crowdsourcing

Abstract *Crowdsourcing* has been used in different domains such as healthcare, computer science, environmental sciences, business and marketing. However, only recently, queueing models useful in the context of *crowdsourcing* have been studied. These studies involve queueing models of the type $M/M/c$, $MAP/PH/1$, and $MAP/PH/c$. The motivation behind these models came from the context of service sectors getting possible help from one group of customers who first receive service from them and then opt to execute similar services to another group of customers. For example, one type of customers visits the store to procure items while the other type of customers orders over some medium such as Internet and phone and expects them to be delivered. The store management can use the customers visiting them as couriers to "serve" the other type of customers. Not all in-store customers may be willing and in some cases not possible to act as servers on behalf of the store. Hence a probability is introduced for in-store customers to opt for servicing the other type. In this paper we introduce vacation and working vacation in the context of $MAP/PH/1$ with *crowdsourcing*. The matrix-analytic methods are employed to study the model in steady-state analysis. Through illustrative numerical examples we demonstrate the significant benefits in introducing this type of variants to the classical queueing models.

2010 Mathematics Subject Classification: 90B22 (60K25; 90B05).

Key words and phrases: Queueing • Crowdsourcing • Matrix-analytic method • Quasi-birth-and-death process • Markovian arrival process • Phase type.

1. Introduction *Crowdsourcing* has been used in different domains gaining significant exposure in many fields such as healthcare, computer science, environmental sciences, business and marketing and we refer the reader to a recent survey paper by Hosseini et al. [8]. Also, the meaning and interpretation of *crowdsourcing* is varied and despite its popularity among companies in many sectors it remains little understood (Djelassi and Decoopman [5]). We refer the reader to Quora [15] for a number of examples of *crowdsourcing* in many sectors.

Recently queueing models in the context of *Crowdsourcing*, a concept that has been tested and used by many industries, have been studied by Chakravarthy and Dudin [3], and Chakravarthy and Ozkar [4]. According to Howe [7], "Simply defined, *crowdsourcing* represents the act of a company or institution taking a function once performed by employees and outsourcing it to an undefined (and generally large) network of people in the form of an open call. This can take the form of peer-production (when the job is performed collaboratively), but is also often undertaken by sole individuals. The crucial prerequisite is the use of the open call format and the large network of potential laborers."

While in Chakravarthy and Dudin [3] a multi-server queueing model with Poisson arrivals and exponential services in the context of *crowdsourcing* was studied, in Chakravarthy and Ozkar [4] queuing models of the type $MAP/PH/1$ were studied, and the authors employ simulation tools to study *crowdsourcing* in $MAP/PH/c$-type models. The motivation behind these models came from the context of service sectors getting possible help from one group of customers who first receive service from them and then opt to execute similar services to another group of customers. For example, one type of customers visits the store to procure items while the other type of customers orders over some medium such as Internet and phone and expects them to be delivered. The store management can use the customers visiting them as couriers to "serve" the other type of customers. Not all in-store customers may be willing and in some cases not possible to act as servers on behalf of the store. Hence a probability is introduced for in-store customers to opt for servicing the other type. The matrix-analytic methods were employed in Chakravarthy and Dudin [3], Chakravarthy and Ozkar [4] to study the models in steady-state analysis. Additionally, simulation approach was used in Chakravarthy and Ozkar [4]. Through illustrative numerical examples they demonstrated the significant benefits in introducing this type of variants to the classical queueing models.

In this paper we introduce vacation and working vacation in the context of $MAP/PH/1$ with *crowdsourcing*. A number of papers have been studied in the literature on classical queueing models with vacations and working vacations (see e.g., Baba [1]; Banik et al. [2]; Kim et al. [9]; Li and Tian [11]; Servi and Finn [16]; Sreenivasan et al. [17]; Tian and Zhang [19]; Tian and Li [20]; Wu and Takagi [21]; Zang and Hou [22]). Recently, Sreenivasan et al. [17] studied a queueing model of the type $MAP/PH/1$ with working vacations, vacation interruptions and N policy. We study $MAP/PH/1$-type models in the context of *crowdsourcing* in this paper.

In the sequel we need the following notations. By $\boldsymbol{e}$ we will denote a column vector (of appropriate dimension) of 1's; $\boldsymbol{e}_i$ we will denote a unit column vector (of appropriate dimension) with 1 in the i^{th} position and 0 elsewhere; and I an identity matrix (of appropriate dimension). We will display the dimension should there be a need to emphasize it. For example, if there is a need to display the dimension of an identity matrix of order m, we will do so by writing I_m rather than I; a unit vector of dimension m will be denoted as $\boldsymbol{e}(m)$ rather than $\boldsymbol{e}$. The notation $\boldsymbol{S^0}$ is such that $S\boldsymbol{e}+\boldsymbol{S^0}=\mathbf{0}$. The symbols, $\otimes$ and $\oplus$, respectively, will stand for the Kronecker product and sum of matrices. Thus, if A is a matrix of order $m \times n$ and if 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 $a_{ij}B$; the Kronecker sum of two square matrices, say, G of order g and H of h, is given by $G \otimes I + I \otimes H$, a square matrix of order gh. For details and properties on Kronecker products and Kronecker sums we refer the reader to Graham [6]; Marcus and Minc [12]; Steeb and Hardy [18]. Finally, the notation $'$ denotes the transpose of a matrix.

The rest of the paper is organized as follows. In Section 2 the model under study is described. The steady-state analysis of the model is performed in Section 3 and in this section we also discuss (a) *crowdsourcing* with pure vacation (i.e., the server doesn't offer any services during vacation time); (b) the phase type representation of the duration of vacation and working vacation; (c) some special cases; and (d) some key system performance measures. Illustrative numerical examples, including an optimization problem, are discussed in Section 4. Some concluding remarks are given in Section 5.

2. Model Description We consider a single server queueing system in which two types of customers, say, Type 1 arrive according to a Markovian arrival process (MAP) with representation (D_0, D_1) of order m. Let λ_1 denote the arrival rate of Type 1 customers. An arriving Type 1 customer finding the server idle (i.e., on vacation) will get into service immediately but at a lower rate. Otherwise, the customer will enter into a finite buffer of size $L, 1 \leqslant L < \infty$, to be served on FCFS basis when the server becomes available. Thus, it is possible for a Type 1 customer to be lost at the time of arrival due to the buffer being full. The assumption of finite waiting room for Type 1 and infinite waiting room for Type 2 customers are justified as follows. Type 1 customers are visiting the service facility (like a store) to receive a service whereas Type 2 customers place orders over the phone/Internet/etc. Thus, the assumptions of finite waiting room for Type 1 and infinite waiting room for Type 2 make sense. A pictorial description of the model is displayed in Figure 1.

Let D, defined by $D = D_0 + D_1$, govern the underlying Markov chain of the MAP such that D_0 accounts for the transitions corresponding to no arrival; D_1 governs those corresponding to an arrival of a Type 1 customer. On the other hand, the arrivals of Type 2 customers are assumed to follow a Poisson process with rate λ_2. There is no restriction on how many Type 2 customers can be in the system. That is, there is an infinite buffer space for Type 2 customers. An arriving Type 2 customer finding the server idle will have to wait until the server becomes available to offer service.

For both types of customers the service times follow a phase type distribution with representation $(\boldsymbol{\beta}, S)$ 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 dura-

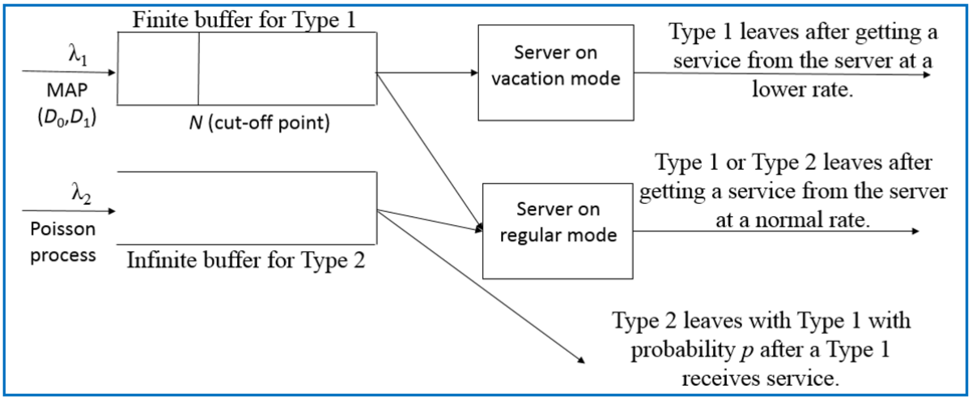

Figure 1: Pictorial description of the model

tion of a vacation is assumed to be exponentially distributed with parameter τ. A vacation is interrupted only when a Type 1 customer arrives during that time. However, the server offers services to those (Type 1) customers arriving during a vacation at a lower rate. Note that only Type 1 customers can be served during a vacation mode. We assume that the service times of those customers (served at a lower rate) are also of phase type but with representation $(\boldsymbol{\beta}, \theta S)$ of order n, where $0 \leqslant \theta \leqslant 1$. The server continues to serve at this rate until either the vacation expires or the number of Type 1 customers in the system hits a pre-determined threshold, say, $N, 1 \leqslant N \leqslant L+1$, whichever occurs first. At this instant, the server instantaneously switches over to the normal rate and continues to serve at this rate until the system becomes empty. Note that once the service resumes at a normal rate (either through vacation getting over or when the number of Type 1 customers in the system hits N) the server will serve both types of customers. 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 $\mu = [\boldsymbol{\beta}(-S)^{-1}\boldsymbol{e}]^{-1}$ and the vacation mode of service has rate $\theta\mu$. Note that the two extreme cases, $\theta = 0$ and $\theta = 1$, are of interest themselves as those cases will not reduce to any of the known models (with crowdsourcing) studied so far. We will elaborate more on this in Subsection 3.5.

While Type 1 customers are to be served by the server, Type 2 customers may be served by a Type 1 customer having already been served and also available to act as a server or by the (system) server. A Type 2 customer getting serviced by a Type 1 customer depends on the following conditions. First, that Type 1 customer should have just finished getting a service and opts to service a Type 2 customer. Secondly, at the time of opting to serve there is at least one Type 2 customer waiting to get a service. That is, this Type 2 customer should not have started getting a service from the server.

We assume that a served Type 1 customer will be available to act as a server for a Type 2 customer under the conditions mentioned above with probability $p, 0 \leqslant p \leqslant 1$. With probability $q = 1 - p$, the served Type 1 customer will leave the system. Upon completion of a service the free server will offer service to a Type 1 customer on a $FCFS$ basis; however, if there are no Type 1 customers waiting, the server will serve a Type 2 customer if there is one present in the queue. We assume that Type 1 customers have a non-preemptive priority over Type 2 customers. If a Type 1 customer decides to serve a Type 2 customer, for our analysis purposes that Type 2 customer will be removed from the system immediately. This is due to the fact that the system no longer needs to track that Type 2 customer.

Let $\boldsymbol{\eta}$ be the steady state probability vector of the Markov process with irreducible generator D. That is, $\boldsymbol{\eta}$ is the unique probability vector satisfying

$$\boldsymbol{\eta} D = 0, \ \boldsymbol{\eta}\boldsymbol{e} = 1. \tag{1}$$

3. The steady-state analysis Let $N_1(t)$,$N_2(t)$,$N_3(t)$,$N_4(t)$, and $N_5(t)$ denote, respectivly, the number of Type 2 customers in the queue, the number of Type 1 customers in the system, the phase of the service (if any), the phase of the arrival process and the state of the server, at time t. The state of the server is given by

$$N_5(t) \quad = \quad \begin{cases} 0_v \ , & \text{if the server is on vacation mode,} \\ 0_w \ , & \text{if the server is on working vacation mode,} \\ 1 \ , & \text{if the server is on regular mode and busy with Type 1 customer,} \\ 2 \ , & \text{if the server is on regular mode and busy with Type 2 customer.} \end{cases}$$

The process $\{(N_1(t), N_2(t), N_3(t), N_4(t), N_5(t)) : \ t \geqslant 0\}$ is a continuous-time Markov chain with state spacce given by

$$\begin{aligned} \Omega \quad = \quad & \{\{(i, k_2, 0_v), \ 1 \leqslant k_2 \leqslant m\} \\ & \bigcup \{(i, j, k_1, k_2, 0_w), \ 1 \leqslant j \leqslant N-1, \ 1 \leqslant k_1 \leqslant n, \ 1 \leqslant k_2 \leqslant m\} \\ & \bigcup \{(i, j, k_1, k_2, 1), \ 1 \leqslant j \leqslant L+1, \ 1 \leqslant k_1 \leqslant n, \ 1 \leqslant k_2 \leqslant m\} \\ & \bigcup \{(i, j, k_1, k_2, 2), \ 0 \leqslant j \leqslant L, \ 1 \leqslant k_1 \leqslant n, \ 1 \leqslant k_2 \leqslant m\}, \ i \geqslant 0\}, \end{aligned}$$

which is rewritten in terms of set of states as

$$\Omega = \{\underline{i}, \ i \geqslant 0\},$$

with the set of states, $\underline{i}$, partitioned as

$$\underline{i} = \{\underline{i_v}\} \bigcup \{\underline{i_w}\} \bigcup \{\underline{i_1}\} \bigcup \{\underline{i_2}\}.$$

Note that the level $\underline{i_v}$ is of dimension m and corresponds to the case when the server is on vacation mode with i Type 2 customers waiting in the queue and the arrival process is in one of m phases; the level $\underline{i_w}$, of dimension $(N-1)mn$, corresponds to the case when the server is on working vacation mode with i Type 2 customers waiting in the queue, j, $1 \leqslant j \leqslant N-1$, Type 1 customers in the system, the service is in one of n phases and the arrival process is in one of m phases; the level $\underline{i_1}$, of dimension $(L+1)mn$, corresponds to case when the server is on regular mode serving a Type 1 customer with i Type 2 customers waiting in the queue, j, $1 \leqslant j \leqslant L+1$, Type 1 customers in the system, the service is in one of n phases and the arrival process is in one of m phases; the level $\underline{i_2}$ is of dimension $(L+1)mn$ and corresponds to the case when the server is on regular mode and busy serving a Type 2 customer with i Type 2 customers waiting in the queue, j, $0 \leqslant j \leqslant L$, Type 1 customers in the system, the service is in one of n phases and the arrival process is in one of m phases.

The infinitesimal generator of the Markov chain governing the system is given by

$$Q = \begin{pmatrix} B_1 & A_0 & & & \\ B_2 & A_1 & A_0 & & \\ A_3 & A_2 & A_1 & A_0 & \\ & A_3 & A_2 & A_1 & A_0 \\ & & \ddots & \ddots & \ddots & \ddots \end{pmatrix}, \tag{2}$$

where the matrices appearing in Q are as follows (first in partitioned form and then the form of the blocks in those partitioned matrices).

$$B_1 = \begin{pmatrix} B_{11}^{(1)} & B_{12}^{(1)} & 0 & 0 \\ B_{21}^{(1)} & B_{22}^{(1)} & B_{23}^{(1)} & 0 \\ B_{31}^{(1)} & 0 & B_{33}^{(1)} & 0 \\ B_{41}^{(1)} & 0 & B_{43}^{(1)} & B_{44}^{(1)} \end{pmatrix},$$

with

$$B_{11}^{(1)} = D_0 - \lambda_2 I, \quad B_{21}^{(1)} = \boldsymbol{e}_1(N-1) \otimes (\theta \boldsymbol{S^0} \otimes I), \quad B_{31}^{(1)} = B_{41}^{(1)} = \boldsymbol{e}_1(L+1) \otimes (\boldsymbol{S^0} \otimes I),$$

$$B_{12}^{(1)} = \boldsymbol{e}_1^{'}(N-1) \otimes (\boldsymbol{\beta} \otimes D_1),$$

$$B_{22}^{(1)} = \begin{pmatrix} \mathfrak{D}_0(\theta,\tau) & \mathfrak{D}_1 & & & \\ \mathfrak{S}^0(1,\theta,\beta) & \mathfrak{D}_0(\theta,\tau) & \mathfrak{D}_1 & & \\ & \ddots & \ddots & \ddots & \\ & & \mathfrak{S}^0(1,\theta,\beta) & \mathfrak{D}_0(\theta,\tau) & \mathfrak{D}_1 \\ & & & \mathfrak{S}^0(1,\theta,\beta) & \mathfrak{D}_0(\theta,\tau) \end{pmatrix},$$

where $\mathfrak{D}_0(\theta,\tau) = (\theta S \oplus D_0) - \tau I - \lambda_2 I$, $\mathfrak{D}_1 = I \otimes D_1$, and $\mathfrak{S}^0(z,\theta,\beta) = \theta z \boldsymbol{S^0}\boldsymbol{\beta} \otimes I$.

$$B_{33}^{(1)} = \begin{pmatrix} \mathfrak{D}_0(1,0) & \mathfrak{D}_1 & & & \\ \mathfrak{S}^0(1,1,\beta) & \mathfrak{D}_0(1,0) & \mathfrak{D}_1 & & \\ & \ddots & \ddots & \ddots & \\ & & \mathfrak{S}^0(1,1,\beta) & \mathfrak{D}_0(1,0) & \mathfrak{D}_1 \\ & & & \mathfrak{S}^0(1,1,\beta) & (S \oplus D) - \lambda_2 I \end{pmatrix},$$

$$B_{44}^{(1)} = \begin{pmatrix} \mathfrak{D}_0(1,0) & \mathfrak{D}_1 & & & \\ & \mathfrak{D}_0(1,0) & \mathfrak{D}_1 & & \\ & & \ddots & \ddots & \\ & & & \mathfrak{D}_0(1,0) & \mathfrak{D}_1 \\ & & & & (S \oplus D) - \lambda_2 I \end{pmatrix},$$

$$B_{43}^{(1)} = \begin{pmatrix} 0 & & & & \\ \mathfrak{S}^0(1,1,\beta) & 0 & & & \\ & \ddots & \ddots & & \\ & & \mathfrak{S}^0(1,1,\beta) & 0 & \\ & & & \mathfrak{S}^0(1,1,\beta) & 0 \end{pmatrix},$$

$$B_{23}^{(1)} = \begin{pmatrix} \tau I & & & & & \\ & \tau I & & & & \\ & & \ddots & & & \\ & & & \tau I & \mathfrak{D}_1 & 0 & \cdots \end{pmatrix}.$$

$$B_2 = \begin{pmatrix} 0 & 0 & 0 & B_{14}^{(2)} \\ B_{21}^{(2)} & B_{22}^{(2)} & 0 & 0 \\ B_{31}^{(2)} & 0 & B_{33}^{(2)} & B_{34}^{(2)} \\ 0 & 0 & 0 & B_{44}^{(2)} \end{pmatrix},$$

with

$$\begin{aligned} B_{14}^{(2)} &= \boldsymbol{e}_1^{'}(L+1) \otimes (\tau \boldsymbol{\beta} \otimes I), \\ B_{21}^{(2)} &= \boldsymbol{e}_1(N-1) \otimes (\theta p \boldsymbol{S^0} \otimes I), \\ B_{31}^{(2)} &= \boldsymbol{e}_1(L+1) \otimes (p \boldsymbol{S^0} \otimes I), \\ B_{34}^{(2)} &= \boldsymbol{e}_1(L+1) \otimes \boldsymbol{e}_1^{'}(L+1) \otimes (\mathfrak{S}^0(q,1,\beta)), \\ B_{44}^{(2)} &= \boldsymbol{e}_1(L+1) \otimes \boldsymbol{e}_1^{'}(L+1) \otimes (\mathfrak{S}^0(1,1,\beta)), \\ B_{22}^{(2)} &= \begin{pmatrix} 0 & & & & \\ \mathfrak{S}^0(p,\theta,\beta) & 0 & & & \\ & \ddots & \ddots & & \\ & & \mathfrak{S}^0(p,\theta,\beta) & 0 & \\ & & & \mathfrak{S}^0(p,\theta,\beta) & 0 \end{pmatrix}, \\ B_{33}^{(2)} &= \begin{pmatrix} 0 & & & & \\ \mathfrak{S}^0(p,1,\beta) & 0 & & & \\ & \ddots & \ddots & & \\ & & \mathfrak{S}^0(p,1,\beta) & 0 & \\ & & & \mathfrak{S}^0(p,1,\beta) & 0 \end{pmatrix}. \end{aligned}$$

$$A_1 = \begin{pmatrix} A_{11} & B_{12}^{(1)} & 0 & 0 \\ A_{21} & A_{22} & B_{23}^{(1)} & 0 \\ 0 & 0 & A_{33} & 0 \\ 0 & 0 & B_{43}^{(1)} & B_{44}^{(1)} \end{pmatrix},$$

with

$$A_{11} = D_0 - \tau I - \lambda_2 I, \qquad A_{21} = \boldsymbol{e}_1(N-1) \otimes (\theta q \boldsymbol{S^0} \otimes I),$$

$$A_{22} = \begin{pmatrix} \mathfrak{D}_0(\theta,\tau) & \mathfrak{D}_1 & & \\ \mathfrak{S}^0(q,\theta,\beta) & \mathfrak{D}_0(\theta,\tau) & \mathfrak{D}_1 & \\ & \ddots & \ddots & \ddots \\ & \mathfrak{S}^0(q,\theta,\beta) & \mathfrak{D}_0(\theta,\tau) & \mathfrak{D}_1 \\ & & \mathfrak{S}^0(q,\theta,\beta) & \mathfrak{D}_0(\theta,\tau) \end{pmatrix},$$

$$A_{33} = \begin{pmatrix} \mathfrak{D}_0(1,0) & \mathfrak{D}_1 & & \\ \mathfrak{S}^0(q,1,\beta) & \mathfrak{D}_0(1,0) & \mathfrak{D}_1 & \\ & \ddots & \ddots & \ddots \\ & \mathfrak{S}^0(q,1,\beta) & \mathfrak{D}_0(1,0) & \mathfrak{D}_1 \\ & & \mathfrak{S}^0(q,1,\beta) & (S \oplus D) - \lambda_2 I \end{pmatrix}.$$

$$A_2 = \begin{pmatrix} 0 & 0 & 0 & B_{14}^{(2)} \\ B_{21}^{(2)} & B_{22}^{(2)} & 0 & 0 \\ 0 & 0 & B_{33}^{(2)} & B_{34}^{(2)} \\ 0 & 0 & 0 & B_{44}^{(2)} \end{pmatrix}, \qquad A_0 = \lambda_2 I,$$

$$A_3 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \hat{A} \\ 0 & 0 & 0 & 0 \end{pmatrix}, \hat{A} = \boldsymbol{e}_1(L+1) \otimes \boldsymbol{e}_1'(L+1) \otimes (\mathfrak{S}^0(p,1,\beta)),$$

The generator Q in (2) is of the $GI/M/1$-type and we can apply the well-known matrix-analytic methods Neuts [13] to perform the steady-state analysis. In this set up the rate matrix, R, of dimension $m + (N-1)mn + 2(L+1)mn$, is the minimal nonnegative solution to a matrix cubic equation. However, by combining the set of states, $\{\underline{2i}, \underline{2i+1}\}$, for $i \geqslant 0$ as

$$\tilde{\underline{i}} = \{\underline{2i}, \underline{2i+1}\},$$

the model under study can be analyzed as a $QBD-$process(*quasi-birth-and-death*) with generator, $\tilde{Q}$, on the state space $\tilde{\Omega} = \{\tilde{\underline{i}},\ \tilde{\underline{i}} \geqslant 0\}$ given by

$$\tilde{Q} = \begin{pmatrix} \tilde{B}_0 & F_0 & & & \\ F_2 & F_1 & F_0 & & \\ & F_2 & F_1 & F_0 & \\ & & \ddots & \ddots & \ddots \end{pmatrix}, \tag{3}$$

where the block entries appearing in $\tilde{Q}$ are obtained from those of Q as follows.

$$\tilde{B}_0 = \begin{pmatrix} B_1 & A_0 \\ B_2 & A_1 \end{pmatrix}, \; F_0 = \begin{pmatrix} 0 & 0 \\ A_0 & 0 \end{pmatrix}, \; F_1 = \begin{pmatrix} A_1 & A_0 \\ A_2 & A_1 \end{pmatrix}, \; F_2 = \begin{pmatrix} A_3 & A_2 \\ 0 & A_3 \end{pmatrix}.$$

In the rest of the paper we will analyze the model under study in steady-state using QBD approach.

3.1. The stability condition Let $\boldsymbol{\xi} = (\boldsymbol{\xi}(1), \boldsymbol{\xi}(2))$ be the steady state probability vector of the generator
$F = F_0 + F_1 + F_2$. That is, $\boldsymbol{\xi}$ satisfies

$$\boldsymbol{\xi} F = \mathbf{0}, \boldsymbol{\xi}\boldsymbol{e} = 1. \tag{4}$$

On noting that

$$F = \begin{pmatrix} A_1 + A_3 & A_0 + A_2 \\ A_0 + A_2 & A_1 + A_3 \end{pmatrix}, \tag{5}$$

is a circulant matrix, the vector $\boldsymbol{\xi}$ is of the form given by

$$\boldsymbol{\xi} = 0.5(\boldsymbol{e'} \otimes \boldsymbol{\pi}), \tag{6}$$

where $\boldsymbol{\pi}$ satisfies (with $A = A_0 + A_1 + A_2 + A_3$)

$$\boldsymbol{\pi} A = \mathbf{0}, \boldsymbol{\pi}\boldsymbol{e} = 1. \tag{7}$$

First note that the matrix A is <u>*reducible*</u> and hence $\boldsymbol{\pi}$ is of the form
$\boldsymbol{\pi} = (\mathbf{0}, \boldsymbol{\pi_1}, \boldsymbol{\pi_2})$, with $\boldsymbol{\pi_1} = (\boldsymbol{\pi}_{11}, \boldsymbol{\pi}_{12}, \cdots, \boldsymbol{\pi}_{L+1})$ and $\boldsymbol{\pi_2} = (\boldsymbol{\pi}_{20}, \boldsymbol{\pi}_{21}, \cdots, \boldsymbol{\pi}_{2L})$.

We now establish a few results that are needed for later use as well as serve as internal accuracy checks in our numerical results.

Lemma 3.1 *We have*

$$\begin{aligned} \Big(\sum_{j=1}^{L+1} \boldsymbol{\pi}_{1j} + \sum_{j=0}^{L} \boldsymbol{\pi}_{2j}\Big)(I \otimes \boldsymbol{e}) &= \mu\boldsymbol{\beta}(-S)^{-1}, \\ \Big(\sum_{j=1}^{L+1} \boldsymbol{\pi}_{1j} + \sum_{j=0}^{L} \boldsymbol{\pi}_{2j}\Big)(\boldsymbol{e} \otimes I) &= \boldsymbol{\eta}, \end{aligned} \tag{8}$$

where $\boldsymbol{\eta}$ is as given in (1) *and μ is the service rate.*

Proof The steady state equations in (7) can be rewritten as

$$\begin{aligned} \boldsymbol{\pi}_{11}(S \oplus D_0) + (\boldsymbol{\pi}_{12} + \boldsymbol{\pi}_{21})(S^0\beta \otimes I) &= 0 \\ \boldsymbol{\pi}_{1j}(S \oplus D_0) + \boldsymbol{\pi}_{1,j-1}(\mathfrak{D}_1) + (\boldsymbol{\pi}_{1,j+1} + \boldsymbol{\pi}_{2j})(S^0\beta \otimes I) &= 0, \; 2 \leqslant j \leqslant L \\ \boldsymbol{\pi}_{1,L+1}(S \oplus D) + \boldsymbol{\pi}_{1L}(\mathfrak{D}_1) &= 0 \\ \boldsymbol{\pi}_{20}((S + S^0\beta) \oplus D_0) + \boldsymbol{\pi}_{11}(S^0\beta \otimes I) &= 0 \\ \boldsymbol{\pi}_{2j}(S \oplus D_0) + \boldsymbol{\pi}_{2,j-1}(\mathfrak{D}_1) &= 0, \; 1 \leqslant j \leqslant L-1 \\ \boldsymbol{\pi}_{2L}(S \oplus D) + \boldsymbol{\pi}_{2,L-1}(\mathfrak{D}_1) &= 0 \end{aligned} \tag{9}$$

with the normalizing condition

$$\Big(\sum_{j=1}^{L+1}\boldsymbol{\pi}_{1j}+\sum_{j=0}^{L}\boldsymbol{\pi}_{2j}\Big)\boldsymbol{e}=1. \tag{10}$$

Now adding the equations given in (9) we get

$$\Big(\sum_{j=1}^{L+1}\boldsymbol{\pi}_{1j}+\sum_{j=0}^{L}\boldsymbol{\pi}_{2j}\Big)[(S+S^0\beta)\oplus D]=0.$$

Post-multiplying the equation by $(e\otimes I)$ and using the uniqueness of the steady-state vector of the generator D along with the normalizing condition in (10) yields the first result in (8). Similarly, by post-multiplying the equation by $(I\otimes e)$ one can obtain the second result in (8) from the uniqueness of the steady-state vector of the generator $S+S^0\beta$ and the normalizing condition in (10). ■

The following theorem establishes the stability condition of the queueing system under study.

THEOREM 3.2 *The queuing system under study is stable if and only if the following condition is satisfied.*

$$\lambda_2<p\mu+q\sum_{j=0}^{L}\boldsymbol{\pi}_{2j}(S^0\otimes\boldsymbol{e}). \tag{11}$$

PROOF The queueing system under study with the QBD type generator given in (3) is stable if and only if $\boldsymbol{\xi}F_0\boldsymbol{e}<\boldsymbol{\xi}F_2\boldsymbol{e}$. From the entries of the generator $\tilde{Q}$ and the equation in (6), we see that

$$\boldsymbol{\xi}F_0\boldsymbol{e}=0.5\lambda_2 \text{ and } \boldsymbol{\xi}F_2\boldsymbol{e}=0.5[\boldsymbol{\pi}A_2\boldsymbol{e}+2\boldsymbol{\pi}A_3\boldsymbol{e}]. \tag{12}$$

It is easy to verify that

$$\begin{aligned}\boldsymbol{\pi}A_2\boldsymbol{e} &= q\boldsymbol{\pi}_{11}(S^0\beta\otimes I)\boldsymbol{e}+\sum_{j=2}^{L+1}p\boldsymbol{\pi}_{1j}(S^0\beta\otimes I)\boldsymbol{e}+\boldsymbol{\pi}_{20}(S^0\beta\otimes I)\boldsymbol{e}\\ \boldsymbol{\pi}A_3\boldsymbol{e} &= p\boldsymbol{\pi}_{11}(S^0\beta\otimes I)\boldsymbol{e}.\end{aligned} \tag{13}$$

Using equations (12) and (13), the stability condition becomes

$$\lambda_2<(\boldsymbol{\pi}_{11}+\boldsymbol{\pi}_{20})(S^0\otimes\boldsymbol{e})+\sum_{j=1}^{L+1}p\boldsymbol{\pi}_{1j}(S^0\otimes\boldsymbol{e}).$$

Now post-multiplying the last three equations in (9) by $\boldsymbol{e}$ and adding them we get

$$\boldsymbol{\pi}_{11}(S^0 \otimes \boldsymbol{e}) = \sum_{j=1}^{L} \boldsymbol{\pi}_{2j}(S^0 \otimes \boldsymbol{e}). \tag{14}$$

From (14) and the fact (from Lemma 3.1) that $(\sum_{j=0}^{L} \boldsymbol{\pi}_{2j} + \sum_{j=1}^{L+1} \boldsymbol{\pi}_{1j})(S^0 \otimes \boldsymbol{e}) = \mu$, the stated result follows immediately. ■

3.2. The steady-state probability vector. Let $\boldsymbol{x}$=($\boldsymbol{x}(0)$, $\boldsymbol{x}(1)$,...) denote the steady-state probability vector of $\tilde{Q}$. That is, $\boldsymbol{x}$ satisfies

$$\boldsymbol{x}\tilde{Q} = 0, \boldsymbol{x}\boldsymbol{e} = 1. \tag{15}$$

The vector $\boldsymbol{x}(i), i \geqslant 0$, of dimension $2m + 2(N-1)mn + 4(L+1)mn$, is further partitioned as $\boldsymbol{x}(i) = (\hat{\boldsymbol{x}}(i), \tilde{\boldsymbol{x}}(i)), i \geqslant 0$. The vector $\hat{\boldsymbol{x}}(i)$, of dimension $m + (N-1)mn + 2(L+1)mn$, is further partitioned into

$$\hat{\boldsymbol{x}}(i) = (\hat{\boldsymbol{x}}_v(i), \hat{\boldsymbol{x}}_{w1}(i), ..., \hat{\boldsymbol{x}}_{w,N-1}(i), \hat{\boldsymbol{x}}_{11}(i), ..., \hat{\boldsymbol{x}}_{1,L+1}(i), \hat{\boldsymbol{x}}_{20}(i), ..., \hat{\boldsymbol{x}}_{2,L}(i)).$$

Note that $\hat{\boldsymbol{x}}_v(i)$, of dimension m, gives the steady-state probability that the server is on vacation with no Type 1 customers in the system and $2i$ Type 2 customers waiting in the queue with arrival process in one of the m phases; $\hat{\boldsymbol{x}}_{wj}(i)$, $1 \leqslant j \leqslant N-1$, of dimension mn, gives the steady-state probability that there are j Type 1 customers in the system and $2i$ Type 2 customers in the queue with the server busy with a Type 1 customer during a vacation, the arrival process in one of the m phases, and the service phase is in one of n phases; $\hat{\boldsymbol{x}}_{1j}(i), 1 \leqslant j \leqslant L+1$, of dimension mn, gives the steady-state probability that there are j Type 1 customers in the system, $2i$ Type 2 customers in the queue with the server on a regular mode busy with a Type 1 customer, and the arrival process in one of the m phases along with the service in one of n phases; $\hat{\boldsymbol{x}}_{2j}(i), 0 \leqslant j \leqslant L$, of dimension mn, gives the steady-state probability that there are j Type 1 customers in the system, $2i$ Type 2 customers in the queue with the server on a regular mode busy with a Type 2 customer, and the arrival process in one of the m phases along with the service in one of n phases. Similarly, the vector $\tilde{\boldsymbol{x}}(i)$, of dimension $m + (N-1)mn + 2(L+1)mn$, is partitioned into

$$\tilde{\boldsymbol{x}}(i) = (\tilde{\boldsymbol{x}}_v(i), \tilde{\boldsymbol{x}}_{w1}(i), ..., \tilde{\boldsymbol{x}}_{w,N-1}(i), \tilde{\boldsymbol{x}}_{11}(i), ..., \tilde{\boldsymbol{x}}_{1,L+1}(i), \tilde{\boldsymbol{x}}_{20}(i), ..., \tilde{\boldsymbol{x}}_{2,L}(i)).$$

The components of $\tilde{\boldsymbol{x}}(i)$ have similar interpretations except that the number of Type 2 customers in the queue is now given by $2i+1$.

Under the stability condition given in (11) the steady-state probability vector $\boldsymbol{x}$ is obtained as follows

$$\boldsymbol{x}(i) = \boldsymbol{x}(0)R^i, i \geqslant 0, \tag{16}$$

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

$$R^2F_2 + RF_1 + F_0 = 0,$$

and $\boldsymbol{x}(0)$ is obtained by solving

$$\boldsymbol{x}(0)[\tilde{B}_0 + RF_2] = 0,$$

subject to the normalizing condition

$$\boldsymbol{x}(0)(I - R)^{-1}\boldsymbol{e} = 1.$$

The computation of the R matrix can be carried out using a number of well-known methods such as logarithmic reduction and (block) Gauss-Seidel iterative by exploiting the special structure of the coefficient matrices F_0, F_1, and F_2, which are of dimension $2m+2(N-1)mn+4(L+1)mn$. This is very important especially when one is dealing with large values of L, N, m and n. For example, one can exploit the structure of R matrix:

$$R = \begin{pmatrix} 0 & 0 \\ R_2 & R_1 \end{pmatrix}.$$

Note that the above structure is due to the fact that F_0 has first $m+(N-1)mn+2(L+1)mn$ rows full of zeros. It is worth pointing out that $R_1 = R_2^2$ and that R_2 satisfies the matrix-cubic equation (in $GI/M/1$ set up as pointed out Section 3): $R_2^3A_3 + R_2^2A_2 + R_2A_1 + A_0 = 0$.

The details of other exploitation will be omitted; however, the key steps in the logarithmic reduction are given below and for full details on this we refer the reader to Latouche and Ramaswami [10].

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:

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

Continue Step 1 until $||\boldsymbol{e} - G\boldsymbol{e}||_\infty < \epsilon$.
Step 2: $R = -A_0(A_1 + A_0G)^{-1}$.

For use in the sequel we define

$$\boldsymbol{a} = \boldsymbol{x}(0)(I-R)^{-1}, \qquad \boldsymbol{b} = \boldsymbol{x}(0)R(I-R)^{-1}.$$

We partition the $2m+2(N-1)mn+4(L+1)mn$-dimensional vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ as

$$\boldsymbol{a} = (\hat{\boldsymbol{a}}_v, \hat{\boldsymbol{a}}_{w1}, \cdots, \hat{\boldsymbol{a}}_{w,N-1}, \hat{\boldsymbol{a}}_{1,1}, \cdots, \hat{\boldsymbol{a}}_{1,L+1}, \hat{\boldsymbol{a}}_{2,0}, \cdots, \hat{\boldsymbol{a}}_{2,L}, \tilde{\boldsymbol{a}}_v, \tilde{\boldsymbol{a}}_{w1}, \cdots, \tilde{\boldsymbol{a}}_{w,N-1}, \tilde{\boldsymbol{a}}_{1,1}, \cdots, \tilde{\boldsymbol{a}}_{1,L+1}, \tilde{\boldsymbol{a}}_{2,0}, \cdots, \tilde{\boldsymbol{a}}_{2,L}),$$

$$\boldsymbol{b} = (\hat{\boldsymbol{b}}_v, \hat{\boldsymbol{b}}_{w1}, \cdots, \hat{\boldsymbol{b}}_{w,N-1}, \hat{\boldsymbol{b}}_{1,1}, \cdots, \hat{\boldsymbol{b}}_{1,L+1}, \hat{\boldsymbol{b}}_{2,0}, \cdots, \hat{\boldsymbol{b}}_{2,L}, \tilde{\boldsymbol{b}}_v, \tilde{\boldsymbol{b}}_{w1}, \cdots, \tilde{\boldsymbol{b}}_{w,N-1}, \tilde{\boldsymbol{b}}_{1,1}, \cdots, \tilde{\boldsymbol{b}}_{1,L+1}, \tilde{\boldsymbol{b}}_{2,0}, \cdots, \tilde{\boldsymbol{b}}_{2,L}),$$

We now list some useful results in Lemma 3.3 which are intuitively obvious and serve as accuracy checks in our numerical computation.

LEMMA 3.3 *We have*

$$\sum_{j=0}^{L}(\hat{\boldsymbol{a}}_{2j} + \tilde{\boldsymbol{a}}_{2j})(S^0 \otimes \boldsymbol{e}) + \sum_{j=1}^{L+1} p(\tilde{\boldsymbol{x}}_{1j}(0) + \hat{\boldsymbol{b}}_{1j} + \tilde{\boldsymbol{b}}_{1j})(S^0 \otimes \boldsymbol{e})$$

$$+ \sum_{j=1}^{N-1} p(\tilde{\boldsymbol{x}}_{wj}(0) + \hat{\boldsymbol{b}}_{wj} + \tilde{\boldsymbol{b}}_{wj})(\theta S^0 \otimes \boldsymbol{e}) = \lambda_2,$$

$$\sum_{j=1}^{N-1}(\hat{\boldsymbol{a}}_{wj} + \tilde{\boldsymbol{a}}_{wj})(\theta S^0 \otimes \boldsymbol{e}) + \sum_{j=1}^{L+1}(\hat{\boldsymbol{a}}_{1j} + \tilde{\boldsymbol{a}}_{1j})(S^0 \otimes \boldsymbol{e}) = \lambda_1(1 - P_{lost}),$$

where

$$P_{lost} = \frac{1}{\lambda_1}[(\hat{\boldsymbol{a}}_{1,L+1} + \tilde{\boldsymbol{a}}_{1,L+1}) + (\hat{\boldsymbol{a}}_{2,L} + \tilde{\boldsymbol{a}}_{2,L})](\boldsymbol{e} \otimes D_1\boldsymbol{e}).$$

3.3. $MAP/PH/1{-}CSQ$ **with pure vacation** First note that $MAP/PH/1$ queue with *crowdsourcing* with vacation and working vacation cannot be reduced to $MAP/PH/1$ queue with *crowdsourcing* by suitably choosing the values for θ and N, which appear in working vacation model. Secondly, we will compare the working vacation model with pure vacation model. Thus, we will very briefly outline the needed quantities for $MAP/PH/1$ *crowdsourcing* queue with vacation in this section.

The process $\{(N_1(t), N_2(t), N_3(t), N_4(t), N_5^*(t)) : \ t \geqslant 0\}$ is a continuous-time Markov chain, where the state of the server, $N_5^*(t)$, is now given by

$$N_5^*(t) = \begin{cases} v, & \text{if the server is on vacation with no Type 1 customer present,} \\ v1, & \text{if the server is on vacation with Type 1 customer present,} \\ 1, & \text{if the server is on regular mode and busy with Type 1 customer,} \\ 2, & \text{if the server is on regular mode and busy with Type 2 customer.} \end{cases}$$

The state space of the Markov chain is given by

$\Omega_v = \{\{(i, k_2, v),\ 1 \leqslant k_2 \leqslant m\}$
$\bigcup\{(i, j, k_2, v1),\ 1 \leqslant j \leqslant L,\ 1 \leqslant k_2 \leqslant m\}$
$\bigcup\{(i, j, k_1, k_2, 1),\ 1 \leqslant j \leqslant L+1,\ 1 \leqslant k_1 \leqslant n,\ 1 \leqslant k_2 \leqslant m\}$
$\bigcup\{(i, j, k_1, k_2, 2),\ 0 \leqslant j \leqslant L,\ 1 \leqslant k_1 \leqslant n,\ 1 \leqslant k_2 \leqslant m\},\ i \geqslant 0\}$,

which is rewritten in terms of set of states as

$$\Omega_v = \{\underline{i},\ i \geqslant 0\},$$

with the set of states, $\underline{i}$, partitioned as

$$\underline{i} = \{\underline{i_v}\} \bigcup \{\underline{i_{v1}}\} \bigcup \{\underline{i_1}\} \bigcup \{\underline{i_2}\}.$$

Note that the level $\underline{i_v}$ is of dimension m and corresponds to the case when the server is on vacation mode with i Type 2 customers and no Type 1 customer waiting in the queue, and the arrival process is in one of m phases; the level $\underline{i_{v1}}$ is of dimension Lm and corresponds to the case when the server is on vacation mode with i Type 2 customers and j, $1 \leqslant j \leqslant L$, Type 1 customers waiting in the queue, and the arrival process is in one of m phases.
The infinitesimal generator of the Markov chain governing the system is given by

$$Q_v = \begin{pmatrix} B_{1v} & A_{0v} & & & & \\ B_{2v} & A_{1v} & A_{0v} & & & \\ A_{3v} & A_{2v} & A_{1v} & A_{0v} & & \\ & A_{3v} & A_{2v} & A_{1v} & A_{0v} & \\ & & \ddots & \ddots & \ddots & \ddots \end{pmatrix},$$

where the matrices appearing in Q_v are as follows (first in partitioned form and then the form of the blocks in those partitioned matrices). Since most of the blocks are common with Q (see equation (2)), we will display only those that are new here.

$$B_{1v} = \begin{pmatrix} B_{11}^{(1)} & B_{12}^{(1v)} & 0 & 0 \\ 0 & B_{22}^{(1v)} & B_{23}^{(1v)} & 0 \\ B_{31}^{(1)} & 0 & B_{33}^{(1)} & B_{34}^{(1)} \\ B_{41}^{(1)} & 0 & B_{43}^{(1)} & B_{44}^{(1)} \end{pmatrix},$$

with

$$B_{22}^{(1v)} = \begin{pmatrix} D_0 - \tau I - \lambda_2 I & D_1 & & & \\ 0 & D_0 - \tau I - \lambda_2 I & D_1 & & \\ & \ddots & \ddots & \ddots & \\ & & 0 & D_0 - \tau I - \lambda_2 I & D_1 \\ & & & 0 & D_0 - \tau I - \lambda_2 I \end{pmatrix},$$

$$B_{23}^{(1v)} = \begin{pmatrix} \tau(\beta \otimes I) & & & & \\ & \tau(\beta \otimes I) & & & \\ & & \ddots & & \\ & & & \tau(\beta \otimes I) & 0 & \cdots \end{pmatrix}, \quad B_{12}^{(1v)} = \boldsymbol{e}_1^{'}(L) \otimes D_1.$$

$$B_{2v} = \begin{pmatrix} 0 & 0 & 0 & B_{14}^{(2)} \\ 0 & 0 & 0 & 0 \\ B_{31}^{(2)} & 0 & B_{33}^{(2)} & B_{34}^{(2)} \\ 0 & 0 & 0 & B_{44}^{(2)} \end{pmatrix}, \qquad A_{0v} = \lambda_2 I,$$

$$A_{1v} = \begin{pmatrix} A_{11} & B_{12}^{(1v)} & 0 & 0 \\ 0 & B_{22}^{(1v)} & B_{23}^{(1v)} & 0 \\ 0 & 0 & A_{33} & 0 \\ 0 & 0 & B_{43}^{(1)} & B_{44}^{(1)} \end{pmatrix},$$

$$A_{2v} = \begin{pmatrix} 0 & 0 & 0 & B_{14}^{(2)} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & B_{33}^{(2)} & B_{34}^{(2)} \\ 0 & 0 & 0 & B_{44}^{(2)} \end{pmatrix}, \; A_{3v} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \hat{A} \\ 0 & 0 & 0 & 0 \end{pmatrix}.$$

Combining the set of states $\{\underline{2i}, \underline{2i+1}\}$, for $i \geqslant 0$, like earlier, this model under investigation can be studied as a $QBD-$process with generator, $\tilde{Q}_v$, on the state space $\tilde{\Omega}_v = \{\underline{\tilde{i}},\ \underline{\tilde{i}} \geqslant 0\}$ given by

$$\tilde{Q}_v = \begin{pmatrix} \tilde{B}_{0v} & F_{0v} & & & \\ F_{2v} & F_{1v} & F_{0v} & & \\ & F_{2v} & F_{1v} & F_{0v} & \\ & & \ddots & \ddots & \ddots \end{pmatrix},$$

where the block entries appearing in $\tilde{Q}_v$ are obtained from those of Q_v as follows.

$$\tilde{B}_{0v} = \begin{pmatrix} B_{1v} & A_{0v} \\ B_{2v} & A_{1v} \end{pmatrix},$$

$$F_{0v} = \begin{pmatrix} 0 & 0 \\ A_{0v} & 0 \end{pmatrix}, \; F_{1v} = \begin{pmatrix} A_{1v} & A_{0v} \\ A_{2v} & A_{1v} \end{pmatrix}, \; F_{2v} = \begin{pmatrix} A_{3v} & A_{2v} \\ 0 & A_{3v} \end{pmatrix}.$$

Let $\boldsymbol{y}$=($\boldsymbol{y}(0)$, $\boldsymbol{y}(1)$,...) denote the steady-state probability vector of $\tilde{Q}_v$. That is, $\boldsymbol{y}$ satisfies $\boldsymbol{y}\tilde{Q}_v = 0$ and $\boldsymbol{ye} = 1$. The vector $\boldsymbol{y}(i), i \geqslant 0$, of dimension $2(L+1)m + 4(L+1)mn$, is further partitioned as $\boldsymbol{y}(i) = (\hat{\boldsymbol{y}}(i), \tilde{\boldsymbol{y}}(i)), i \geqslant 0$. The vectors $\hat{\boldsymbol{y}}(i)$ and $\tilde{\boldsymbol{y}}(i)$, of dimension $(L+1)m + 2(L+1)mn$, are further partitioned into

$$\hat{\boldsymbol{y}}(i) = (\hat{\boldsymbol{x}}_v(i), \hat{\boldsymbol{y}}_{v1,1}(i), ..., \hat{\boldsymbol{y}}_{v1,L}(i), \hat{\boldsymbol{x}}_{11}(i), ..., \hat{\boldsymbol{x}}_{1,L+1}(i), \hat{\boldsymbol{x}}_{20}(i), ..., \hat{\boldsymbol{x}}_{2,L}(i)).$$

$$\tilde{\boldsymbol{y}}(i) = (\tilde{\boldsymbol{x}}_v(i), \tilde{\boldsymbol{y}}_{v1,1}(i), ..., \tilde{\boldsymbol{y}}_{v1,L}(i), \tilde{\boldsymbol{x}}_{11}(i), ..., \tilde{\boldsymbol{x}}_{1,L+1}(i), \tilde{\boldsymbol{x}}_{20}(i), ..., \tilde{\boldsymbol{x}}_{2,L}(i)).$$

Note that $\hat{\boldsymbol{y}}_{v1,j}(i), 1 \leqslant j \leqslant L$, of dimension m, gives the steady-state probability that there are j Type 1 customers in the queue and $2i$ Type 2 customers in the queue when the server is on vacation mode with arrival process in one of the m phases. The components $\tilde{\boldsymbol{y}}_{v1,j}(i), 1 \leqslant j \leqslant L$, have similar interpretations except that the number of Type 2 customers in the queue is now given by $2i + 1$.

The rate matrix, R_v, of dimension $2(L+1)m+4(L+1)mn$ is the minimal nonnegative solution to the matrix quadratic equation $R_v^2F_{2v}+R_vF_{1v}+F_{0v} =$

0. The stability condition for this case is the same as that for working vacation (see equation (11)).

For use in the performance measures we define $a_v = y(0)(I-R_v)^{-1}$ and $b_v = y(0)R_v(I-R_v)^{-1}$. We partition the $2(L+1)m+4(L+1)mn$-dimensional vectors $\boldsymbol{a}_v$ and $\boldsymbol{b}_v$ as

$$\boldsymbol{a}_v = (\hat{\boldsymbol{a}}_v, \hat{\boldsymbol{a}}_{v1,1}, \cdots, \hat{\boldsymbol{a}}_{v1,L}, \hat{\boldsymbol{a}}_{1,1}, \cdots, \hat{\boldsymbol{a}}_{1,L+1}, \hat{\boldsymbol{a}}_{2,0}, \cdots, \hat{\boldsymbol{a}}_{2,L}, \tilde{\boldsymbol{a}}_v, \tilde{\boldsymbol{a}}_{w1}, \cdots, \tilde{\boldsymbol{a}}_{w,N-1}, \tilde{\boldsymbol{a}}_{1,1}, \cdots, \tilde{\boldsymbol{a}}_{1,L+1}, \tilde{\boldsymbol{a}}_{2,0}, \cdots, \tilde{\boldsymbol{a}}_{2,L}),$$

$$\boldsymbol{b}_v = (\hat{\boldsymbol{b}}_v, \hat{\boldsymbol{b}}_{v1,1}, \cdots, \hat{\boldsymbol{b}}_{v1,L}, \hat{\boldsymbol{b}}_{1,1}, \cdots, \hat{\boldsymbol{b}}_{1,L+1}, \hat{\boldsymbol{b}}_{2,0}, \cdots, \hat{\boldsymbol{b}}_{2,L}, \tilde{\boldsymbol{b}}_v, \tilde{\boldsymbol{b}}_{w1}, \cdots, \tilde{\boldsymbol{b}}_{w,N-1}, \tilde{\boldsymbol{b}}_{1,1}, \cdots, \tilde{\boldsymbol{b}}_{1,L+1}, \tilde{\boldsymbol{b}}_{2,0}, \cdots, \tilde{\boldsymbol{b}}_{2,L}).$$

3.4. Phase type representation for vacation/working vacation duration In this section we will prove that the duration of the vacation/working vacation can be modeled as a phase type distribution.
The generator matrix of a Markov chain with a single absorbing state is the form

$$Q^* = \begin{pmatrix} T & T^0 \\ 0 & 0 \end{pmatrix},$$

is further partitioned by considering two subsets (1: pure vacation & 2: working vacation) as

$$Q^* = \begin{pmatrix} T(1,1) & T(1,2) & T^0(1) \\ T(2,1) & T(2,2) & T^0(2) \\ 0 & 0 & 0 \end{pmatrix},$$

with

$$T^0(1) = 0, \qquad T^0(2) = (\tau\boldsymbol{e}, ..., \tau\boldsymbol{e},\ \boldsymbol{e}\otimes D_1\boldsymbol{e} + \tau\boldsymbol{e})',$$

$$T(1,1) = D_0, \qquad T(1,2) = \boldsymbol{e}_1^{'}(N-1)\otimes(\boldsymbol{\beta}\otimes D_1), \qquad T(2,1) = \boldsymbol{e}_1(N-1)\otimes(\theta\boldsymbol{S^0}\otimes I),$$

$$T(2,2) = \begin{pmatrix} \mathfrak{D}_0(\theta,\tau) & \mathfrak{D}_1 & & & \\ \mathfrak{S}^0(1,\theta,\beta) & \mathfrak{D}_0(\theta,\tau) & \mathfrak{D}_1 & & \\ & \ddots & \ddots & \ddots & \\ & & \mathfrak{S}^0(1,\theta,\beta) & \mathfrak{D}_0(\theta,\tau) & \mathfrak{D}_1 \\ & & & \mathfrak{S}^0(1,\theta,\beta) & \mathfrak{D}_0(\theta,\tau) \end{pmatrix},$$

where the square matrices $T(1,1)$ and $T(2,2)$ correspond to transitions within subsets 1 and 2 of transient states.The submatrices $T(1,2)$ and $T(2,1)$ correspond to transitions from set 1 to set 2 and vice versa.
The initial probability vector is written in the correspondingly partitioned form

$$\boldsymbol{\alpha} = (\boldsymbol{\alpha}_0, 0, 0),$$

with $\boldsymbol{\alpha}_0 = \delta(\boldsymbol{x}_{11}(0) + \boldsymbol{x}_{02}(0) + p\boldsymbol{x}_{11}(1))(\boldsymbol{S^0} \otimes I)$ where δ is the normalizing constant.
The total time spent in the subsets 1 and 2 prior to absorption (server works at normal rate) is of phase type with representation $(\boldsymbol{\alpha}, T)$ of order $m + (N-1)mn$.

$$\mu'_{v/wv} = \boldsymbol{\alpha}(-T)^{-1}\boldsymbol{e}.$$

The computation of this mean is achieved by exploiting the special structure of α and T. We will briefly outline the steps involved in this computation.
Defining $\alpha(-T)^{-1} = \boldsymbol{c}$ and partitioning the vector $\boldsymbol{c}$ as

$$\boldsymbol{c} = (\boldsymbol{c}_0, ..., \boldsymbol{c}_{N-1}),$$

where $\boldsymbol{c}_0$ is of dimension m and $\boldsymbol{c}_i, 1 \leqslant i \leqslant N-1$, is of dimension mn, the mean $\mu'_{v/wv}$ is given by

$$\mu'_{v/wv} = \sum_{i=0}^{N-1} \boldsymbol{c}_i\boldsymbol{e}.$$

The vector $\boldsymbol{c}$ is calculated using the following set of equations.

$$\boldsymbol{c}_0 D_0 + \boldsymbol{c}_1(\theta\boldsymbol{S^0} \otimes I) = -\boldsymbol{\alpha}_0,$$

$$\boldsymbol{c}_0(\boldsymbol{\beta} \otimes D_1) + \boldsymbol{c}_1(\theta S \oplus D_0 - \tau I) + \boldsymbol{c}_2(\theta\boldsymbol{S^0}\boldsymbol{\beta} \otimes I) = 0,$$

$$\boldsymbol{c}_{i-1}(\mathfrak{D}_1) + \boldsymbol{c}_i(\theta S \oplus D_0 - \tau I) + \boldsymbol{c}_{i+1}(\mathfrak{S}^0(1, \theta, \beta)) = 0, \qquad 2 \leqslant i \leqslant N-2,$$

$$\boldsymbol{c}_{N-2}(\mathfrak{D}_1) + \boldsymbol{c}_{N-1}(\theta S \oplus D_0 - \tau I) = 0.$$

subject to the condition

$$\tau\sum_{i=1}^{N-1} \boldsymbol{c}_i\boldsymbol{e} + \boldsymbol{c}_{N-1}(\boldsymbol{e} \otimes D_1\boldsymbol{e}) = 1.$$

The time spent *only* in the subset 1 prior to absorption is of phase type with representation $(\boldsymbol{\alpha}_0, K)$ of order m.

$$\mu'_{pv} = \boldsymbol{\alpha}_0(-K)^{-1}\boldsymbol{e},$$

where

$$K = T(1,1) + T(1,2)\big(-T(2,2)\big)^{-1}T(2,1).$$

Defining $T(1,2)\big(-T(2,2)\big)^{-1} = \boldsymbol{d}$ and partitioning the vector $\boldsymbol{d}$ as

$$\boldsymbol{d} = (\boldsymbol{d}_1, ..., \boldsymbol{d}_{N-1}).$$

where $\boldsymbol{d}_i, 1 \leqslant i \leqslant N-1$, is of dimension mn, K is rewritten by exploiting the special structure of $T(2,1)$ as $K = T(1,1) + \boldsymbol{d}_1(\theta\boldsymbol{S^0} \otimes I)$.
The vector $\boldsymbol{d}$ is calculated using the following system of equations.

$$\boldsymbol{d}_1(\tau I - (\theta S \oplus D_0)) - \boldsymbol{d}_2(\theta\boldsymbol{S^0}\boldsymbol{\beta} \otimes I) = (\boldsymbol{\beta} \otimes D_1),$$

$$\boldsymbol{d}_{i-1}(\mathfrak{D}_1) + \boldsymbol{d}_i((\theta S \oplus D_0) - \tau I) + \boldsymbol{d}_{i+1}(\mathfrak{S}^0(1,\theta,\beta)) = 0, \qquad 2 \leqslant i \leqslant N-2,$$

$$\boldsymbol{d}_{N-1}(\tau I - (\theta S \oplus D_0)) - \boldsymbol{d}_{N-2}(\mathfrak{D}_1) = 0.$$

3.5. Special cases In this section we will look at some special cases. These cases are of interest on their own and they do not reduce to any of the known models. However, these cases are needed to compare the pure vacation as well as $MAP/PH/1$ with *crowdsouring* model studied in Chakravarthy and Ozkar [4].

3.5.1. Case: $\theta = 0.0$ In this case, a Type 1 arriving to the system when the server is on vacation will not have any impact unless the number of Type 1 customers hits the pre-determined threshold, N. A Type 2 arriving to the system when the server is on vacation has no impact. Note that this special case doesn't reduce to $MAP/PH/1$ model with *crowdsourcing* and pure vacation. This is because the server will break the vacation when the number of Type 1 arrivals hits N. However, the steady-state results will be close to the corresponding pure vacation model when L and N are very large.

3.5.2. Case: $\theta = 1.0$ In this case, a Type 1 arriving to the system when the server is on vacation will not see any difference in terms of getting a service at the regular rate. However, a Type 2 arrival has to wait either for the vacation to get over or for a Type 1 customer to offer the service. Even this case doesn't reduce to $MAP/PH/1$ model with *crowdsourcing*. However, from Type 1 arrivals' perspective, this case provides a somewhat priority service (when the server is on vacation) for them.

3.6. The system performance measures In this section we will list a number of system performance measures of interest along with their expressions. Since we are looking at two models, one with pure vacation and the other with working vacations, it should be pointed out that most measures are common and have similar expressions. However, the values of the measures will be different depending on the model. Where the expressions for the measures differ we will distinguish accordingly by using the symbols "v" or "wv".

1. Probability that the server is on vacation.

$$P_{vac}^{(wv)} = (\hat{\boldsymbol{a}}_v + \tilde{\boldsymbol{a}}_v)\boldsymbol{e}.$$

$$P_{vac}^{(v)} = (\hat{\boldsymbol{a}}_v + \tilde{\boldsymbol{a}}_v)\boldsymbol{e} + \sum_{j=1}^{L}(\hat{\boldsymbol{a}}_{v1,j} + \tilde{\boldsymbol{a}}_{v1,j})\boldsymbol{e}.$$

2. Probability that the server is on working vacation.

$$P_{workvac} = \sum_{j=1}^{N-1} (\hat{\boldsymbol{a}}_{wj} + \tilde{\boldsymbol{a}}_{wj})\boldsymbol{e}.$$

3. Probability that the server is on regular mode.

$$P_{busy}^{(reg)} = \sum_{j=1}^{L+1} (\hat{\boldsymbol{a}}_{1j} + \tilde{\boldsymbol{a}}_{1j})\boldsymbol{e} + \sum_{j=0}^{L} (\hat{\boldsymbol{a}}_{2j} + \tilde{\boldsymbol{a}}_{2j})\boldsymbol{e}.$$

4. Probability that the server is busy with a Type 1 customer.

$$P_{busyT_1}^{(wv)} = \sum_{j=1}^{N-1} (\hat{\boldsymbol{a}}_{wj} + \tilde{\boldsymbol{a}}_{wj})\boldsymbol{e} + \sum_{j=1}^{L+1} (\hat{\boldsymbol{a}}_{1j} + \tilde{\boldsymbol{a}}_{1j})\boldsymbol{e}.$$

$$P_{busyT_1}^{(v)} = \sum_{j=1}^{L+1} (\hat{\boldsymbol{a}}_{1j} + \tilde{\boldsymbol{a}}_{1j})\boldsymbol{e}.$$

5. Probability that the server is busy with a Type 2 customer.

$$P_{busy}^{(2)} = \sum_{j=0}^{L} (\hat{\boldsymbol{a}}_{2j} + \tilde{\boldsymbol{a}}_{2j})\boldsymbol{e}.$$

6. Probability an arriving Type 1 customer is lost.

$$P_{lost} = \frac{1}{\lambda_1}[(\hat{\boldsymbol{a}}_{1,L+1} + \tilde{\boldsymbol{a}}_{1,L+1}) + (\hat{\boldsymbol{a}}_{2,L} + \tilde{\boldsymbol{a}}_{2,L})](\boldsymbol{e} \otimes D_1\boldsymbol{e}).$$

7. Rate of a Type 2 customer leaving by getting service from the server.

$$R_{T2LS} = \sum_{j=0}^{L} (\hat{\boldsymbol{a}}_{2j} + \tilde{\boldsymbol{a}}_{2j})(S^0 \otimes \boldsymbol{e}).$$

8. Rate of a Type 2 customer leaving with a Type 1 customer.

$$R_{T2L1}^{(wv)} = \sum_{j=1}^{N-1} p(\tilde{\boldsymbol{x}}_{wj}(0) + \hat{\boldsymbol{b}}_{wj} + \tilde{\boldsymbol{b}}_{wj})(\theta S^0 \otimes \boldsymbol{e}) + \sum_{j=1}^{L+1} p(\tilde{\boldsymbol{x}}_{1j}(0) + \hat{\boldsymbol{b}}_{1j} + \tilde{\boldsymbol{b}}_{1j})(S^0 \otimes \boldsymbol{e}).$$

$$R_{T2L1}^{(v)} = \sum_{j=1}^{L+1} p(\tilde{\boldsymbol{x}}_{1j}(0) + \hat{\boldsymbol{b}}_{1j} + \tilde{\boldsymbol{b}}_{1j})(S^0 \otimes \boldsymbol{e}).$$

9. Mean number of Type 1 customers in the queue.

$$\mu_{T1Q}^{(wv)} = \sum_{j=1}^{N-1} (j-1)(\hat{\boldsymbol{a}}_{wj} + \tilde{\boldsymbol{a}}_{wj})\boldsymbol{e} + \sum_{j=1}^{L+1} (j-1)(\hat{\boldsymbol{a}}_{1j} + \tilde{\boldsymbol{a}}_{1j})\boldsymbol{e} + \sum_{j=0}^{L} j(\hat{\boldsymbol{a}}_{2j} + \tilde{\boldsymbol{a}}_{2j})\boldsymbol{e}.$$

$$\mu_{T1Q}^{(v)} = \sum_{j=1}^{L} j(\hat{\boldsymbol{a}}_{v1,j} + \tilde{\boldsymbol{a}}_{v1,j})\boldsymbol{e} + \sum_{j=1}^{L+1} (j-1)(\hat{\boldsymbol{a}}_{1j} + \tilde{\boldsymbol{a}}_{1j})\boldsymbol{e} + \sum_{j=0}^{L} j(\hat{\boldsymbol{a}}_{2j} + \tilde{\boldsymbol{a}}_{2j})\boldsymbol{e}.$$

10. Mean number of Type 2 customers in the queue.

$$\mu_{T2Q} = 2\tilde{\boldsymbol{x}}(0)(I - R_1)^{-2} R_2 \boldsymbol{e} + \tilde{\boldsymbol{x}}(0)(I - R_1)^{-2}(I + R_1)\boldsymbol{e}.$$

Using Little's law we can obtain the mean waiting times in the queue of an admitted Type 1 customer and a Type 2 customer as follows.

11. Mean waiting time of an admitted Type 1 customer in the queue.

$$\mu_{WTQ}^{(1)} = \frac{\mu_{T1Q}}{\lambda_1(1 - P_{lost})}.$$

12. Mean waiting time of a Type 2 customer in the queue.

$$\mu_{WTQ}^{(2)} = \frac{\mu_{T2Q}}{\lambda_2}.$$

4. Numerical Examples In this section we discuss the qualitative aspects of the *crowdsourcing* queueing model with working vacation under consideration through illustrative numerical examples. To verify the correctness and the accuracy of the code written to compute various system performance measures the results of Lemmas 3.1 and 3.3 were used as well as obtained the numerical solution for the Poisson arrivals in its simple form and using other forms with the help of eigenvector and eigenvalues (see Neuts [14]). For the arrival process, we consider the following five sets of values for D_0 and D_1 as follows.

1. Erlang ($ERLA$):

$$D_0 = \begin{pmatrix} -2 & 2 \\ 0 & -2 \end{pmatrix}, D_1 = \begin{pmatrix} 0 & 0 \\ 2 & 0 \end{pmatrix}$$

2. Exponential ($EXPA$):

$$D_0 = \begin{pmatrix} -1 \end{pmatrix}, D_1 = \begin{pmatrix} 1 \end{pmatrix}$$

3. Hyperexponential ($HEXA$):

$$D_0 = \begin{pmatrix} -1.90 & 0 \\ 0 & -0.19 \end{pmatrix}, D_1 = \begin{pmatrix} 1.71 & 0.19 \\ 0.171 & 0.019 \end{pmatrix}$$

4. MAP with negative correlation ($MNCA$):

$$D_0 = \begin{pmatrix} -1.00222 & 1.00222 & 0 \\ 0 & -1.00222 & 0 \\ 0 & 0 & -225.75 \end{pmatrix}, D_1 = \begin{pmatrix} 0 & 0 & 0 \\ 0.01002 & 0 & 0.9922 \\ 223.4925 & 0 & 2.2575 \end{pmatrix}$$

5. MAP **with positive correlation (**$MPCA$**):**

$$D_0 = \begin{pmatrix} -1.00222 & 1.00222 & 0 \\ 0 & -1.00222 & 0 \\ 0 & 0 & -225.75 \end{pmatrix}, D_1 = \begin{pmatrix} 0 & 0 & 0 \\ 0.9922 & 0 & 0.01002 \\ 2.2575 & 0 & 223.4925 \end{pmatrix}.$$

The above MAP processes will be normalized so as to have a specific arrival rate. However, these are qualitatively different in thaty they have different variance and correlation structure. The first three arrival processes, namely, $ERLA, EXPA$, and $HEXA$, have zero correlation for two successive inter-arrival times. The arrival processes labeled $MNCA$ and $MPCA$, respectively, have negative and positive correlation for two successive inter-arrival times with values -0.4889 and 0.4889. The ratio of the standard deviation of the inter-arrival times of these five arrival processes with respect to $ERLA$ are, respectively, 1, 1.41421, 3.17451, 1.99336, and 1.99336.
For the service times $(\boldsymbol{\beta}, S)$ we consider the following three $PH-$distributions. These distributions will be normalized so as to arrive at a desired value for μ.

A. Erlang (ERLS) :

$$\boldsymbol{\beta} = (1,0), \quad S = \begin{pmatrix} -2 & 2 \\ 0 & -2 \end{pmatrix}.$$

B. Exponential (EXPS) :

$$\boldsymbol{\beta} = (1), \quad S = \begin{pmatrix} -1 \end{pmatrix}.$$

C. Hyperexponential (HEXS) :

$$\boldsymbol{\beta} = (0.9, 0.1), \quad S = \begin{pmatrix} -10 & 0 \\ 0 & -1 \end{pmatrix}.$$

In Chakravarthy and Ozkar [4] the effect of *crowdsourcing* in the context of $MAP/PH/1$ was studied analytically and significant improvements of such a system were presented. The purpose of our next example is to investigate such a significance for our model under study here. Similar to Chakravarthy and Ozkar [4] we will look at the ratio of the mean waiting time of Type 2 customers in the queue measure under study here. The ratio, $\frac{\mu^{(2)}_{WTQ}(p>0)}{\mu^{(2)}_{WTQ}(p=0)}$, will be of interest for the following example. In the sequel we will use the notation $CSQ-WV$ and $CSQ-PV$, respectively, for crowdsource with working vacation and crowdsource for pure vacation models.

Example 4.1 The effect of *crowdsourcing* in the presence of pure vacation and working vacation is studied in this example. Towards this we look at the two scenarios: (a) $p = 0$ which corresponds to having two independent arrival processes and (b) $p > 0$ which corresponds to the *crowdsourcing* model. Note that in the case of $CSQ - WV$, the server offers services (to only Type 1 customers) at a reduced rate during vacation in both the scenarios; however, for $CSQ - PV$ case the server does not offer services. First note that the mean waiting time of Type 2 customers in the queue, $\mu_{WTQ}^{(2,p=0)}$, when $p = 0$ is obviously larger than that of $\mu_{WTQ}^{(2,p>0)}$, for the case $p > 0$. Thus, the ratio $\frac{\mu_{WTQ}^{(2,p>0)}}{\mu_{WTQ}^{(2,p=0)}}$, will be in the interval (0,1]. We fix $\lambda_1 = 1, \mu = 1.1, \tau = 0.05, 0.1$ and $0.5, L = 10$ and vary p on the interval $(0, 1]$. Additionally, for $CSQ - WV$ case we fix $\theta = 0.2$ and vary N from 1 to 11. The values of λ_2 are chosen based on ρ and we look at two values: $\rho = 0.8$ and $\rho = 0.99$. The ratio is plotted in Figures 2 and 3 under different combinations of arrival and service distributions. These figures reveal the following observations.

- As the vacation rate, τ, increases, we see the ratios for $CSQ - WV$ and $CSQ - PV$ approach those of CSQ model. This is to be expected as the vacation will get over faster. This is even more pronounced when $\rho = 0.99$.

- We notice the ratios are lower when services are of Erlang type compared to hyperexponential ones and shows the significant role played by the variability in the service times.

- As expected, the $CSQ - PV$ model appears to have a higher ratio when the traffic intensity is not overly saturated.

- In the case of $MPCA$ arrivals we notice that one needs a larger value for the vacation rate so that the ratio values will be closer to those of the CSQ model. This is especially when the traffic intensity is moderate and when services are of Erlang type.

- The effect of N (for the $CSQ - WV$ case) for the various scenarios considered is seen when the inter-arrival times have more variability or are (positively) correlated.

In Chakravarthy and Ozkar [4] it was shown that even in the case of small p there is a significant advantage in considering *crowdsourcing* by offering more traffic load through increasing the rate of Type 2 customers into the system. Here, we will investigate a similar advantage for the current model under study by considering the cases when $L = 0$ and $L = 1$. In the former case, Type 1 customer is allowed only when the server is on vacation. Thus, the maximum number of Type 1 customers that can be present at any time in the system is 1 and 2, respectively.

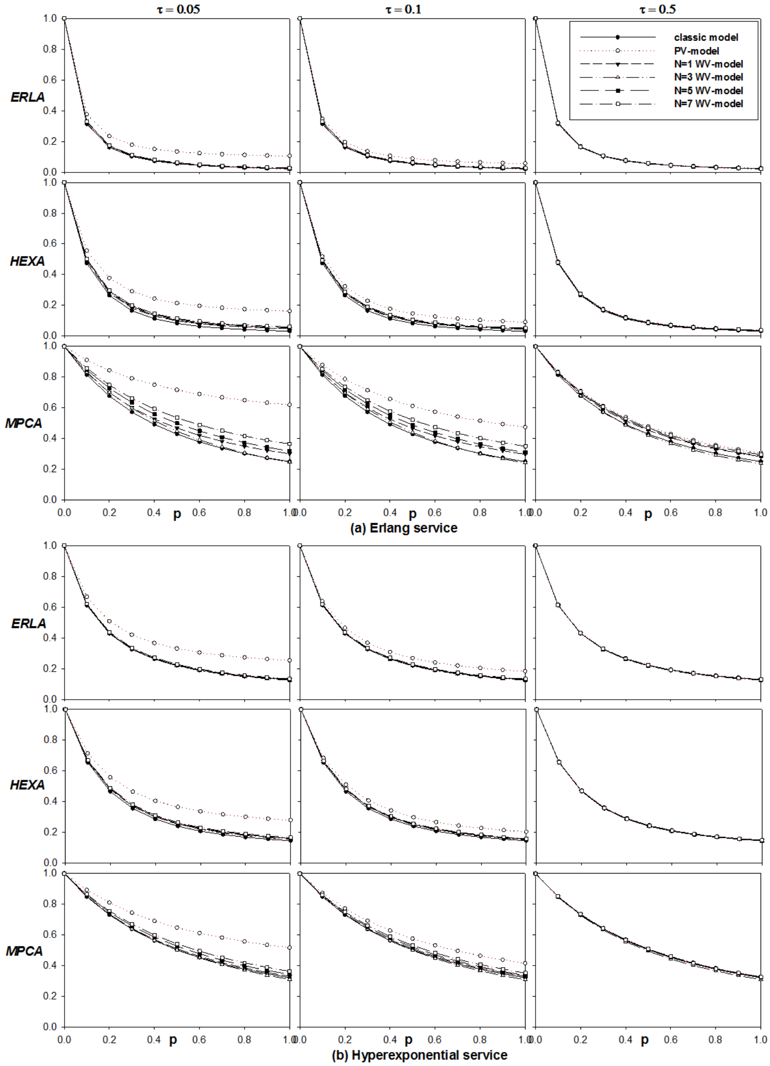

Figure 2: Ratios for mean waiting time of Type 2 customers for Erlang and Hyperexponential services for $\rho = 0.8$

Example 4.2 This example is very similar to Example 4.1 except that we look at the cases when (i) $L = 0, N = 1$, (ii) $L = 1, N = 1$ and (iii) $L = 1, N = 2$ for a few measures. Note that in case (i), Type 1 customer enters the system only when the server is on vacation and when that happens the customer is immediately served at normal rate. Also, there can be at most only one Type 1 customer in the system at any time. On the other hand, for case (ii) the system can have up to two Type 1 customers at any time. The server here only works at normal rate as in case (i). In contrast to two cases,

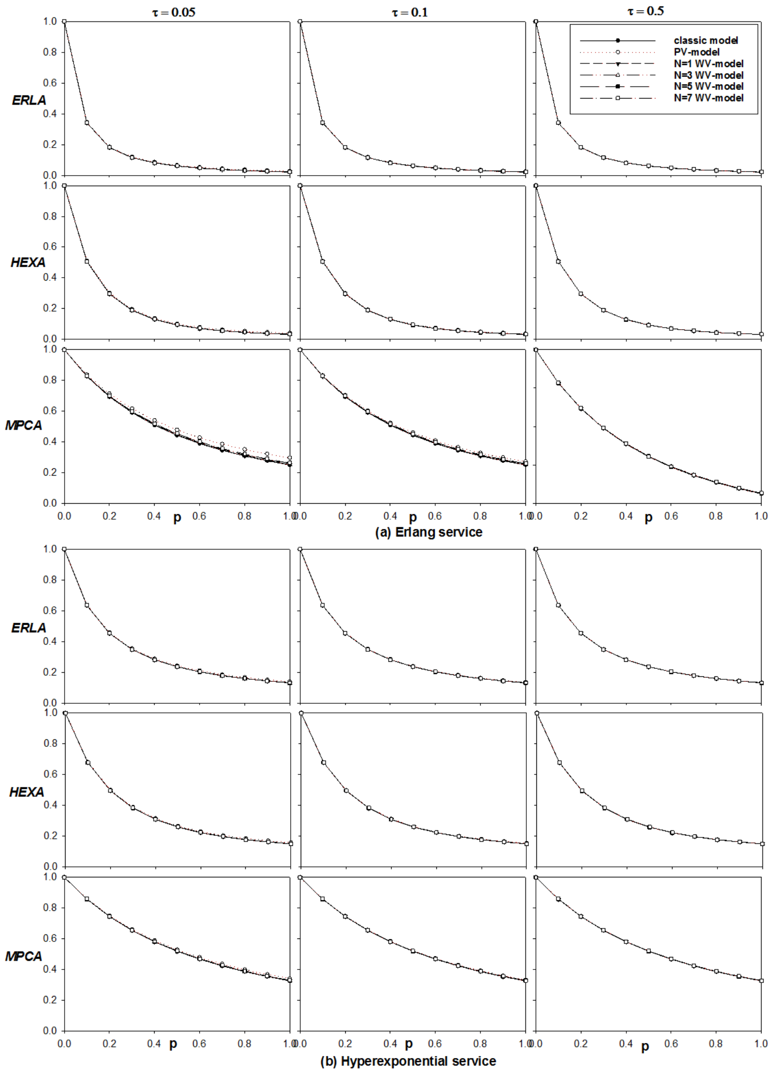

Figure 3: Ratios for mean waiting time of Type 2 customers for Erlang and Hyperexponential services for $\rho = 0.99$

(i) and (ii), the system can have the server work at a lower rate under case (iii).

Towards this end, we will fix $\lambda_1 = 1, \mu = 1.1, \rho = 0.8, 0.99, \theta = 0.2$ and vary p on the interval (0, 1] under different combinations of arrival and service distributions, and for $\tau = 0.1, 1.0$. In order to properly compare, we now look at the ratios $R_1 = \frac{\zeta(L=1,N=1,p)}{\zeta(L=0,N=1,p)}$ and $R_2 = \frac{\zeta(L=1,N=2,p)}{\zeta(L=0,N=1,p)}$, for some measure ζ. For example, if we are comparing the ratio R_1 for the measure μ_{T2Q} when $p = 0$, then we will look at the value of this measure for the combinations,

$\{L = 0, N = 1, p = 0\}$ and $\{L = 1, N = 1, p = 0\}$, and compute the appropriate ratio.

In Figures 4 and 5 below we display the ratios R_1 and R_2, respectively, for selected measures and for representative scenarios. A quick look at these figures reveal the following observations.

- The impact of having a buffer for Type 1 customer (as opposed to not having) is clearly seen with regard to the two measures: μ_{T2Q} and $P_{T2LS} = \frac{R_{T2LS}}{\lambda_2}$, and as p increases to 1, the significance of the impact becomes less. This is the case for both moderate ($\rho = 0.8$) and heavy ($\rho = 0.99$) traffic intensities.

- Similarly the impact of server serving at a lower rate (this happens when $N = 2$) is clearly seen for the two measures.

- When comparing the impact of Erlang and hyperexponential services, we see some interesting trend. When the service times are of Erlang type, the ratio appears to decrease as p increases, whereas for hyper-exponential type services, the ratio appears to increase as p increases. This is the case for all types of arrival processes.

- The ratios for the measures, P_{lost} and $P_{busy}^{(2)}$, are insignificant to p but are significant to the types of arrival and service times. This implies that having a buffer space, that is when $L = 1$, for Type 1 customer with ($N = 2$) or without ($N = 1$) the server having to serve at a lower rate is not affected by the fraction of Type 1 customers opting to serve Type 2 customers. This, initially might look somewhat surprising, but noticing that going from ($L = 1, N = 1$) to ($L = 1, N = 2$) the server serves at a lower rate only about 5% of the times (this is the maximum percentage when looking at all the combinations considered and the values for all the combinations are not displayed here).

Table 1: Optimum values of (L^*, p^*, N^*) for WV-model, (L^*, p^*) for PV-model

PH	*Cost*	*ERLA*	*EXPA*	*HEXA*	*MNCA*	*MPCA*
ERLS	A	(5,1,1)-(5,1)	(5,1,1)-(5,1)	(6,1,1)-(6,1)	(6,1,1)-(6,1)	(10,1,1)-(10,1)
	B	(4,1,1)-(4,1)	(5,1,1)-(5,1)	(5,1,1)-(5,1)	(5,1,1)-(5,1)	(9,1,1)-(9,1)
	C	(3,1,1)-(3,1)	(3,1,1)-(3,1)	(3,1,1)-(3,1)	(4,1,1)-(4,1)	(5,1,1)-(5,1)
	D	(3,1,1)-(3,1)	(3,1,1)-(3,1)	(3,1,1)-(3,1)	(4,1,1)-(4,1)	(5,1,1)-(4,1)
HEXS	A	(6,1,1)-(6,1)	(7,1,1)-(6,1)	(7,1,1)-(7,1)	(7,1,1)-(7,1)	(10,1,1)-(9,1)
	B	(6,1,1)-(6,1)	(6,1,1)-(6,1)	(6,1,1)-(6,1)	(6,1,1)-(6,1)	(9,1,1)-(8,1)
	C	(3,1,1)-(3,1)	(3,1,1)-(3,1)	(3,1,1)-(3,1)	(4,1,1)-(4,1)	(4,1,1)-(4,1)
	D	(3,1,1)-(3,1)	(3,1,1)-(3,1)	(3,1,1)-(3,1)	(3,1,1)-(3,1)	(3,1,1)-(3,1)

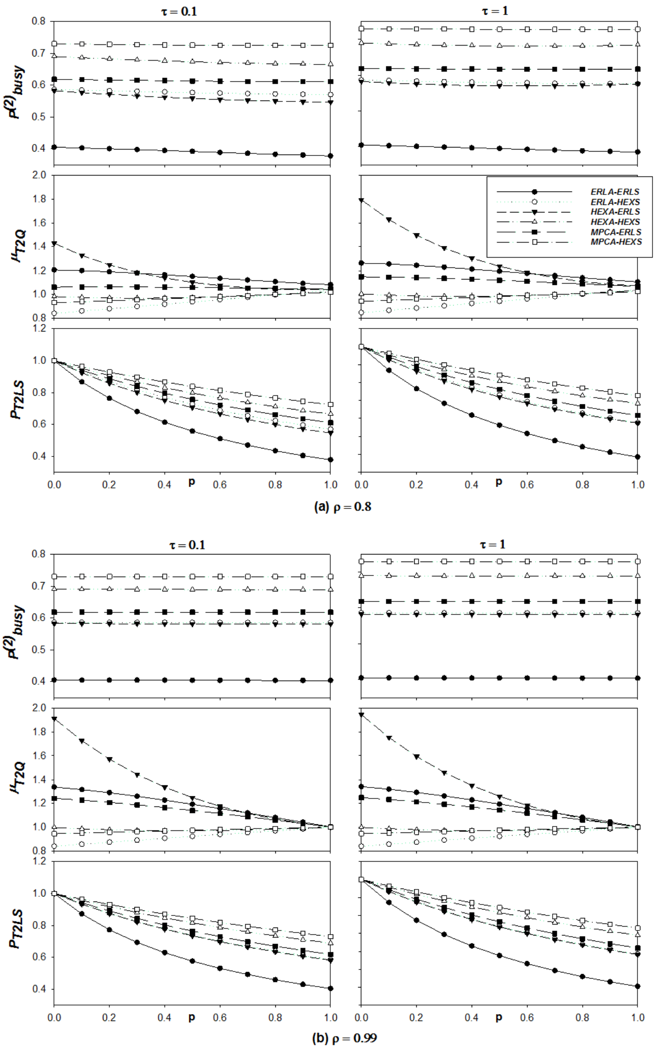

Figure 4: Comparison of the R_1 under different scenarios for $\rho = 0.8$ and $\rho = 0.99$

Example 4.3 In this example we will discuss an optimization problem of interest. Suppose that r, c_1, c_2 and c_3 denote, respectively, the revenue per processed customer, the costs of holding a Type 1 customer and a Type 2 customer in the queue, and the cost using a Type 1 customer to serve a Type 2 customer, on a per unit of time basis. Without loss of generality we will fix $r = 1$. The goal of the optimization problem is to search for optimum values of p, L and N for the working vacation model and of p and L for the pure vacation model. The objective function, ETP, giving the expected total

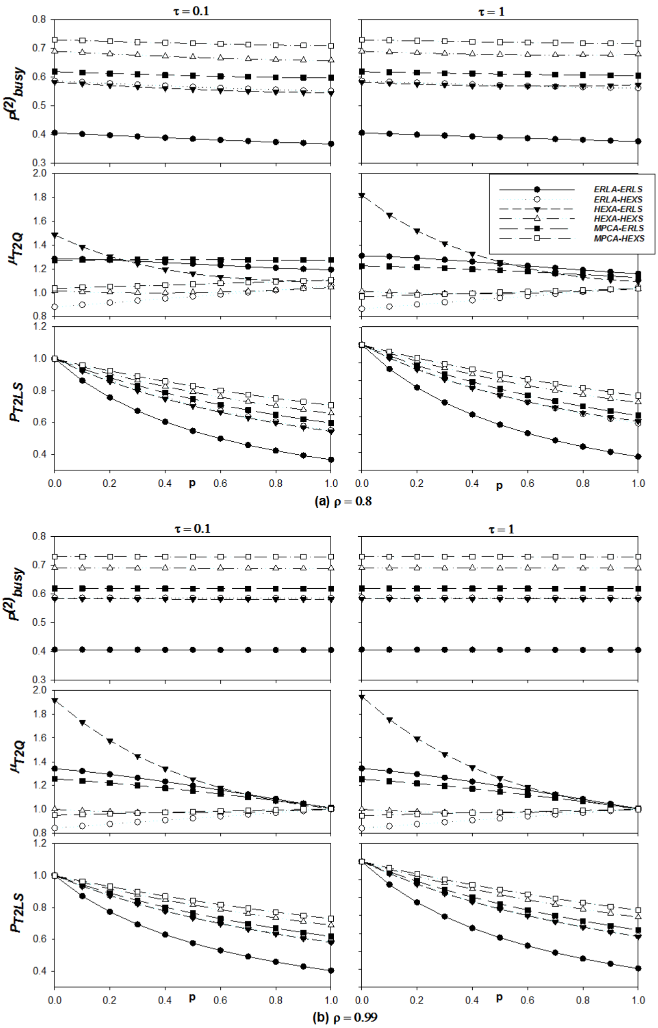

Figure 5: Comparison of the R_2 under different scenarios for $\rho = 0.8$ and $\rho = 0.99$

profit per unit of time, is to be maximized and the expression for ETP is as follows.

$$ETP = \lambda_1(1 - P_{lost}) + \lambda_2 - c_1\mu_{T1Q} - c_2\mu_{T2Q} - c_3\lambda_2 P_{T2L1}.$$

Towards finding the optimum values, we fix $\lambda_1 = 1, \mu = 1.1$, $\theta = 0.2$ (for WV-model), $\tau = 0.1$ and $\rho = 0.99$. In Table 1 we display the optimum, (L^*, p^*, N^*) (for WV-model) and (L^*, p^*) (for V-model) under various scenarios. In the following we use the following notation for the cost vectors: A $= (0.05, 0.001, 0.1)$; B $= (0.05, 0.001, 0.2)$; C $= (0.10, 0.001, 0.1)$; D $= (0.10, 0.001, 0.2)$.

Table 2: Percentages of reduction in ETP at optimum values

PH	$Cost$	$CSQ-WV$			$CSQ-PV$		
		$ERLA$	$HEXA$	$MPCA$	$ERLA$	$HEXA$	$MPCA$
$ERLS$	A	24.65	23.01	17.55	24.50	22.88	17.40
	B	23.07	21.27	16.20	22.93	21.15	16.02
	C	24.09	21.12	17.25	23.95	21.01	17.06
	D	22.45	19.53	15.91	22.29	19.41	15.73
$HEXS$	A	23.54	23.09	19.75	23.31	22.92	19.51
	B	21.22	21.01	17.70	20.96	20.79	17.46
	C	21.15	19.98	15.91	20.97	19.84	15.63
	D	18.88	18.03	13.82	18.63	17.80	13.59

Table 3: Percentages of reduction in ETP at optimum values for various N

$Cost$	N	$ERLS$			$HEXS$		
		$ERLA$	$HEXA$	$MPCA$	$ERLA$	$HEXA$	$MPCA$
A	2	24.54	22.90	17.26	23.30	22.94	19.42
	3	24.55	22.91	17.35	23.33	22.95	19.56
	4	24.56	22.92	17.43	23.46	22.96	20.26
	5	24.94	23.28	17.47	24.26	23.51	21.14
B	2	22.99	21.19	15.90	20.97	20.81	17.38
	3	23.01	21.19	15.99	21.00	20.82	17.52
	4	23.02	21.20	16.06	21.41	20.90	18.44
	5	23.51	21.74	16.14	22.44	21.69	19.49
C	2	24.01	21.07	17.09	21.06	19.92	15.74
	3	24.02	21.07	17.14	21.09	19.93	17.05
	4	24.96	22.30	17.25	23.19	21.74	19.10
	5	26.17	24.03	17.42	25.64	24.32	21.15
D	2	22.35	19.46	15.74	18.65	17.85	13.71
	3	22.37	19.47	15.81	19.14	18.07	15.41
	4	23.57	21.01	15.98	21.67	20.32	17.74
	5	24.78	22.73	16.17	24.14	22.90	19.88

All optimum occurs when $p = 1$ and $N = 1$ for WV-model; and for the PV-model, all optimum occurs when $p = 1$. These are as expected. The values of L^* depend on the type of arrival and service processes. Also, note that the values of L^* in Table 1 are identical to those of $MAP/PH/1-CSQ$ model Chakravarthy and Ozkar [4].

Next, we fix $p = 0.5$. That is, now we look at the case where only 50% of Type 1 customers opt to serve. In Table 2 we display the reduction in the percentages of ETP when going from the optimal point when $p = 1$ to

$p = 0.5$ for the WV as well as PV models. While the reduction percentages for WV model are somewhat closer to those of the $MAP/PH/1 - CSQ$ model (see Chakravarthy and Ozkar [4]), the reduction percentages are significantly smaller for the PV model case as compared to WV model.

In Table 3, we also display the reduction in the percentages of ETP when going from the optimal point when $p = 1$ to $p = 0.5$ for various N, $2 \leqslant N \leqslant 5$, in the $CSQ - WV$ model. We see that the reduction (in the percentages) increases as value of N is increased.

5. Concluding Remarks. In this paper we considered a queueing system useful in *crowdsourcing* and with the server taking (multiple) vacations. Further, the server can offer services to a special group of customers during vacationing but at a lower rate. Using a versatile class of point process for arrivals and phase type services, we showed the significant benefits in introducing this type of variants to the classical queueing models through illustrative numerical examples .

References

[1] Y. Baba, Analysis of a GI/M/1 queue with multiple working vacations. Operations Research Letters 33:201-209, 2005. doi: 10.1016/j.orl.2004.05.006; Zbl 1099.90013 MR 2101056

[2] A.D. Banik, U.C. Gupta, S.S. Pathak On the GI/M/1/N queue with multiple working vacations - analytic analysis and computation. Applied Mathematical Modelling, 31: 1701-1710, 2007. doi: doi:10.1016/j.apm.2006.05.010

[3] S.R. Chakravarthy, A.N. Dudin, A Queueing Model for Crowdsourcing. Journal of Operational Research Society, 2016. doi: 10.1057/s41274-016-0099-x

[4] S.R. Chakravarthy, S. Ozkar, Analysis of a stochastic model for crowdsourcing using MAP arrivals and phase type services, 2016. Submitted for Publication

[5] S. Djelassi, I. Decoopman, Customers' participation in product development through crowdsourcing: Issues and implications. Industrial Marketing Management 42:683-692, 2013. doi: 10.1016/j.indmarman.2013.05.006

[6] A. Graham, *Kronecker Products and Matrix Calculus with Applications.* Ellis Horwood, Chichester, New York, 1981. Zbl 0497.26005 MR 0640865

[7] J. Howe, Crowdsourcing: A definition, 2006[1]. Accessed April 26, 2016.

[8] M. Hosseini, A. Shahri, K. Phalp, J. Taylor, R. Ali, Crowdsourcing: A taxonomy and systematic mapping study. Computer Science Review 17:43-69, 2015. doi: 10.1016/j.cosrev.2015.05.001 MR 3471691

[9] J. Kim, D. Choi, K. Chae, Analysis of queue-length distribution of the M/G/1 queue with working vacations. International Conference on Statistics and Related Fields, Hawaii, 2003.

[10] G. Latouche, V. Ramaswami, *Introduction to matrix analytic methods in stochastic modeling.* ASA-SIAM, 1999. Zbl 0922.60001 MR 1674122

[11] J. Li, N. Tian The $M/M/1$ queue with working vacations and vacation interuptions. Journal of Systems Science and Systems Engineering 16(1):121-127, 2007. doi:10.1007/s11518-006-5030-6 Zbl 1177.60071

[1] http://www.crowdsourcing.com/cs/

[12] M. Marcus, H. Minc, *A Survey of Matrix Theory and Matrix Inequalities*. Allyn and Bacon, Boston, MA, 1964. Zbl 0126.02404 MR 0162808

[13] M.F. Neuts, *Structured stochastic matrices of $M/G/1$ type and their applications*. Marcel Dekker, NY, 1989. MR 1010040

[14] M.F. Neuts, *Algorithmic Probability: A collection of problems*. Chapman and Hall, London, 1995. Zbl 0870.60002 MR 1316776

[15] Quora, 2015[2]. (accessed on June 30, 2015)

[16] L. Servi, S. Finn, $M/M/1$ queue with working vacations ($M/M/1/WV$). Performance Evaluation 50:41-52, 2002. doi: 10.1016/S0166-5316(02)00057-3

[17] C. Sreenivasan, S.R. Chakravarthy, A. Krishnamoorthy, $MAP/PH/1$ queue with working vacations, vacation interruptions and N Policy. Applied Mathematical Modelling 37:3879-3893, 2013. doi: 10.1016/j.apm.2012.07.054 MR 3020538

[18] W.H. Steeb, Y. Hardy, *Matrix Calculus and Kronecker Product. A practical approach to linear and multilinear algebra*. World Scientific Publishing, Singapore, 2011. Zbl 1239.15017 MR 2807612

[19] N.S. Tian, Z.G. Zhang, *Vacation Queueing Models: Theory and applications*. International Series in Operations Research & Management Science, 93. Springer, New York, 2006. Zbl 1104.60004 MR 2248264

[20] N.S. Tian, J.H. Li, Z.G. Zhang, Matrix Analytic Method and Working Vacation Queues - A Survey. International Journal of Information and Management Sciences 20:603-633, 2009. Zbl 1183.90128 MR 2656194

[21] D. Wu, H. Takagi, $M/G/1$ queue with multiple working vacations. Performance Evaluation 63(7):654-681, 2006. doi: 10.1016/j.peva.2005.05.005

[22] M. Zhang, Z. Hou, Performance analysis of MAP/G/1 queue with working vacations and vacation interruption. Applied Mathematical Modelling 35(4):1551-1560, 2011. doi: 10.1016/j.apm.2010.09.031 Zbl 1217.90078 MR 2763797

[2]`http://www.quora.com/What-are-the-best-examples-of-crowdsourcing`

Model kolejkowy $MAP/PH/1$ z planowymi przerwami i zastosowaniem uslug zewnetrznych okazjonalnych

Srinivas Chakravarthy and Serife Ozkar

Streszczenie Odwolywanie sie do madrosci tlumu (*crowdsourcing*-u, okazjonalnych serwisów zewnetrznych) jest wykorzystywane w róznych dziedzinach. Znane sa przyklady ze sluzby zdrowia, informatyki, nauk o srodowisku, z biznesu oraz marketingu. Jednakze dopiero od niedawna zastosowano modele teorii kolejek na uzytek modelowania tej metody powiezania zadan. Badania te obejmuja modele kolejek typu *M/M/c*, *MAP/PH/1* i *MAP/PH/c*. Motywacja dla tych modeli sa uslugi, których realizacje zlecamy do pewnej grupy klientów, a nastepnie ta grupa klientów decyduje sie swiadczyc podobne uslugi dla innych grup klientów. Przykladowo, jedna grupa klientów odwiedza sklepy w celu zakupu penych towarów, podczas gdy drugi typ klientów zleca zakup tych dóbr przez Internet czy telefon i oczekuje ich dostarczenia. Wówczas obsluga sklepu stacjonarnego wykorzystuje odwiedzajacych ich klientów jako kurierów do obslugi inny grupy klientów. Nie wszyscy klienci sklepie sa gotowi, a w niektórych przypadkach jest to niemozliwe, aby pelnic role posredników dzialajacych na rzecz sklepu stacjonarnego. Wprowadzamy zatem prawdopodobienstwo tego, ze klienci jest sklonny przyjac zlecenie obslugi innych klientów. Ninieszy artykul zajmuje sie obsluga z mozliwoscia wakacje i urlop w pracy przy modelu ovslugi $MAP/PH/1$ z wykorzystaniem *crowdsourcing*u. Zastosowano macierzowe metody analityczne do badania systemu w stanu ustalonym. Podano przyklady numeryczne wykazujace znaczace korzysci z wprowadzenia takich wariantów w klasycznych modelach kolejkowych.

2010 Klasyfikacja tematyczna AMS (2010): 90B22 (60K25; 90B05).

Słowa kluczowe: kolejkowanie • crowdsourcing(*madrość tłumu*) • macierzowa analiza stanów ustalonych • pseudo procesy narodzin i śmierci • strumien zgłoszeń Markowa.

Srinivas R. Chakravarthy is professor in the Department of Industrial and Manufacturing Engineering at Kettering University (formerly known as GMI Engineering & Management Institute), Flint, Michigan.
Srinivas Chakravarthy's research interests are in the areas of algorithmic probability, queuing, reliability, inventory, and simulation. He has published more than 100 papers in leading journals and made more than 85 presentations at national and international conferences. He co-organized the First International Conference on MAMs in Stochastic Models in 1995 held in Flint. His recognitions and awards include Rodes Professor, Kettering University, Kettering University Distinguished Research Award, Kettering University/GMI Alumni Outstanding Teaching Award, GMI Outstanding Research Award, and GMI Alumni Outstanding Teaching Award, and Educator of the Year Award by IEOM Society, 2016.
Srinivas Chakravarthy has significant industrial experience by consulting with GM, FORD, PCE, and UPS. His professional activities include serving as (a) Area Editor for the journal, Simulation Modelling Theory and Practice; (b) Associate Editor for the journal IAPQR TRANSACTIONS - Indian Association for Productivity, Quality & Reliability; (c) Advisory Board Member for several other journals and International Conferences; and (d) Reviewer for many professional journals.
References to her research papers are found in MathSciNet under ID:221968.

Serife Ozkar is a research assistant pursuing her doctoral studies in the Department of Statistics, Dokuz Eylul University. This paper was written while she was a visiting research scholar in the Department of Industrial and Manufacturing Engineering, Kettering University, Flint, Michigan.
Serife Ozkar's research interests are in the areas of queuing systems and stochastic processes. References to her research papers are found in MathSciNet under ID:1145982.

Srinivas R. Chakravarthy
Kettering University
Department of Industrial and Manufacturing Engineering
Flint, MI 48504, USA
E-mail: schakrav@kettering.edu

Serife Ozkar
Dokuz Eylul University
Faculty of Science
Department of Statistics, 35160, Izmir, Turkey
E-mail: serife.ozkar@deu.edu.tr

Communicated by: Lukasz Stettner

(Received: 23rd of June 2016; revised: 27th of September 2016)