Improving Efficiency in Queueing Systems by Utilizing the Server's Idle Time

Gabi Hanukov
Department of Management, Bar-Ilan University, Ramat Gan, 5290002, Israel, german.khanukov@live.biu.ac.il

Tal Avinadav
Department of Management, Bar-Ilan University, Ramat Gan, 5290002, Israel, tal.avinadav@biu.ac.il

Tatyana Chernonog
Department of Management, Bar-Ilan University, Ramat Gan, 5290002, Israel, tatyana.chernonog@biu.ac.il

Uriel Spiegel
Department of Management, Bar-Ilan University, Ramat Gan, 5290002, Israel, uriel.spiegel@biu.ac.il
Visiting Professor, Department of Economics, University of Pennsylvania, Philadelphia, USA

Uri Yechiali
Department of Statistics and Operations Research, School of Mathematical Sciences, Tel Aviv University, Tel Aviv, 6997801, Israel, uriy@post.tau.ac.il

Abstract

Two sources of idleness exist in queueing systems: either customers wait unattended to be served, or servers wait idle for incoming customers. Vacation models, in which servers perform ancillary duties instead of being idle, have been studied as a potential means of better utilizing servers' available time. We propose a different approach to increase the system's efficiency: namely, letting the 'idle' server produce an inventory of 'preliminary services' in anticipation of incoming customers. A two-dimensional stochastic process is formulated, in which the number of customers in the system and the inventory level of preliminary services are considered. We obtain closed-form expressions of the steady-state probabilities and calculate system performance measures by applying a blend of analytical methods: sequential calculation, probability-generating functions, and a matrix geometric approach. When using the latter, we obtain closed-form expressions for all entries of the rate matrix *R*, and show their relationship to Catalan numbers. We further establish a condition under which a server that utilizes some of its idle time to produce preliminary services actually ends up being idle for a greater proportion of the time compared with a server that does not do so. Finally, an economic model is considered by which the optimal capacity (maximal allowed inventory level) of preliminary services is determined.

Subject classifications: Markovian; server's idle time; vacation models; preliminary services; probability generating functions; matrix geometric.

Area of review: Queues; Applications.

1. Introduction

Queueing systems are characterized by two sources of idleness: either customers wait in line to be served, or servers stay idle while waiting for arrival of customers. Due to the stochastic nature of queues, neither of the two sources of idleness can be entirely eliminated. Waiting times of customers can be reduced by increasing the number of servers; however, doing so leads to an increase in the servers' idle time. To address this dichotomy, means of utilizing servers' idle time in queueing systems have been investigated from various points of view. Specifically, vacation models have been proposed and studied, in which the idle time of the server is used for ancillary duties that are not directly related to the server's main task, thus improving the overall efficiency of the system. For extensive discussion of vacation models, see, for example, Levy and Yechiali (1975, 1976), Takagi (1991), Doshi (1986), Kella and Yechiali (1988), Rosenberg and Yechiali (1993), Boxma et al. (2002), Servi and Finn (2002), Yechiali (2004), Baba (2005), Tian and Zhang (2006), Ke et al. (2013) and Mytalas and Zazanis (2015). For cases in which the system's operating costs are high, Yadin and Naor (1963) suggested an N-policy vacation-type model: when the server becomes idle, it stays dormant until N customers are accumulated. This queue-dependent policy has been further analyzed in Kella (1989), Moreno (2007), Lee and Yang (2013), Lim et al. (2013), Wei et al. (2013), Yang and Wu (2015), and Haridass and Arumuganathan (2015). Other works (e.g., Armony 2005; Cachon and Zhang 2007; Armony and Ward 2010, 2013; Mandelbaum et al. 2012) have investigated fair routing of customers to idle servers in large-scale systems with heterogeneous customers.

Our innovative approach in the current paper is that, instead of being diverted to ancillary duties, the idle server should be utilized to perform part of the service required by potential customers before their arrival. We focus on services that can be decomposed into two stages: (i) a preliminary preparation stage, which can be performed without the presence of the customer and whose output can be preserved until an actual service is requested; and (ii) a complementary stage that requires the presence of the customer in order to be completed. By doing so, the server can utilize its idle time in order to produce an 'inventory' of preliminary services and use this inventory to reduce customers' mean waiting time. We assume that

the quality of service is not affected by decomposing the service into two separate stages with an intermission between them. A representative example is fast food restaurants, in which food, e.g., hamburger patties, can be prepared before demand occurs, and only upon the arrival of a customer is a hamburger patty heated up, inserted into a bun and served to the customer. Another example is a bicycle shop, which can assemble part of a bicycle before a purchase occurs, and subsequently assemble the remaining parts in accordance with the customer's specific requirements and preferences. Hypothetically, the server can produce preliminary services during its entire idle time in order to minimize the customers' sojourn time. However, there may be cases in which it is economically beneficial to keep the server idle instead of occupying it with production of additional preliminary services, e.g., because of space constraints or cost considerations. Thus, we assume that the number of PSs that the server can produce is constrained by a maximal inventory level, or 'capacity', and that when the number of PSs reaches this capacity the server remains idle when there are no customers in the system.

Herein, we analyze this queueing-inventory system embedded in an M/M/1 queue. To this end, we formulate a two-dimensional stochastic process in which the variables are, respectively, the number of actual customers in the system, and the inventory level of preliminary services. A blend of analysis methods is used in the investigation: (i) sequential calculation of steady-state probabilities; (ii) probability generating functions (PGFs); and (iii) a matrix geometric approach. Although the matrix geometric method (Neuts 1981) usually requires substantial numerical calculations of the so-called rate matrix R (see, e.g., Latouche and Ramaswami 1999; N. Perel and Yechiali 2014a,b), in our model, we find closed-form expressions for all the entries of R, and show their relation to Catalan numbers. This finding significantly reduces the computational effort of obtaining a full analysis of the queueing-inventory system. By using the matrix geometric approach, we obtain the condition for system stability. Furthermore, we establish a condition under which, seemingly paradoxically, a server that utilizes some of its idle time to perform preliminary services actually ends up being idle for a greater proportion of time than it would in a standard M/M/1 queue. We also perform an economic analysis in order to determine the

optimal capacity of preliminary services, and compare the performance of the proposed model with that of a standard M/M/1 queue, in which the server stays idle when no customers are present.

The remainder of this paper is organized as follows. In Section 2, the model is formulated and its parameters are defined. In Section 3, we present methods to calculate analytically the steady-state probabilities of the queueing-inventory system, and we subsequently use these probabilities to calculate various performance measures. Section 4 provides examples of analytical solutions for certain values of the inventory capacity constraint. Section 5 presents economic analysis of the queueing-inventory system, in which the expected cost is minimized by controlling the inventory capacity, and a sensitivity analysis is conducted with respect to arrival and service rates and to cost parameters. It is observed, among other results, that the total expected cost function is convex over the inventory capacity. Section 6 concludes with directions for further research.

2. Model Formulation

In this work we study an M/M/1-type queueing system with a Poisson arrival rate λ and exponentially-distributed service time with mean $1/\mu$. We assume that the service can be split into two parts: one part (denoted PS for 'preliminary service') may be performed when there are no customers present, whereas the other part (denoted CS for 'complementary service') can be performed only after a customer arrives. When the system is empty, the server, instead of being idle, produces PSs at a Poisson rate α. The PSs are stored until the arrival of customers, and can be considered as work-in-process inventory whose aim is to reduce the sojourn time of customers in the system. The capacity of PSs is limited to n, and when the inventoried PSs reach the capacity, the server stops producing PSs and becomes idle. When a customer arrives at the front of the queue and a PS is available, the server immediately starts rendering a CS for that customer. The CS time is assumed to be exponentially distributed with mean $1/\beta$, where $\beta > \mu$. When there are no inventoried PSs available and there are customers in the system, the full service is given as indicated above, according to rate μ.

The process can be formulated as a two-dimensional continuous-time Markov chain with a state space $\{L_t, S_t\}$, where L_t denotes the number of customers in the system at time *t*, and S_t denotes the number of PSs at time *t*. Let $L \equiv \lim_{t\to\infty} L_t$ and $S \equiv \lim_{t\to\infty} S_t$. Define the joint probability of the two-dimensional process as $p_{i,j} = \Pr(L = i, S = j)$ $i = 0,1,2,...,\infty$, $j = 0,1,2,...,n$. The transition rate diagram is depicted in Figure 1, and the steady-state equations of the process are given in Table 1.

Two main methods used to solve such two-dimensional Markov processes are PGFs (e.g., E. Perel and Yechiali, 2008) and matrix geometric procedures (Neuts 1981). Use of the PGF method entails calculation of the roots of a polynomial, obtained from the determinant of a matrix denoted $A(z)$, and analysis of the roots' properties. The matrix geometric method entails calculation of a rate matrix *R*, from which all the system's steady-state probabilities and performance measures are derived. There are connections between the roots of $|A(z)|$ and the entries of matrix *R* (see Corollary 1(iv) in Section 3.3). In some cases, it is possible to obtain the steady-state probabilities in a direct manner, by combining the balance equations with the so-called vertical and horizontal 'cuts' of the transition rate diagram in Figure 1. We will first show that the latter method can be applied to our service system; notably, this is rarely possible in the solution procedure of such two-dimensional systems. Next, we present the PGF method, by which we calculate mean values of the number of customers in the system and the inventory level of PSs, as well as mean waiting times. Finally, we employ the matrix geometric method, and derive the stability condition.

Table 1. The steady-state equations

	$j=0$	$1\le j\le n-1$	$j=n$
$i=0$	$(\alpha+\lambda)p_{0,0}=\beta p_{1,1}+\mu p_{1,0}$	$(\alpha+\lambda)p_{0,j}=\alpha p_{0,j-1}+\beta p_{1,j+1}$	$\lambda p_{0,n}=\alpha p_{0,n-1}$
$i\ge 1$	$(\mu+\lambda)p_{i,0}=$ $\lambda p_{i-1,0}+\mu p_{i+1,0}+\beta p_{i+1,1}$	$(\beta+\lambda)p_{i,j}=\lambda p_{i-1,j}+\beta p_{i+1,j+1}$	$(\beta+\lambda)p_{i,n}=\lambda p_{i-1,n}$

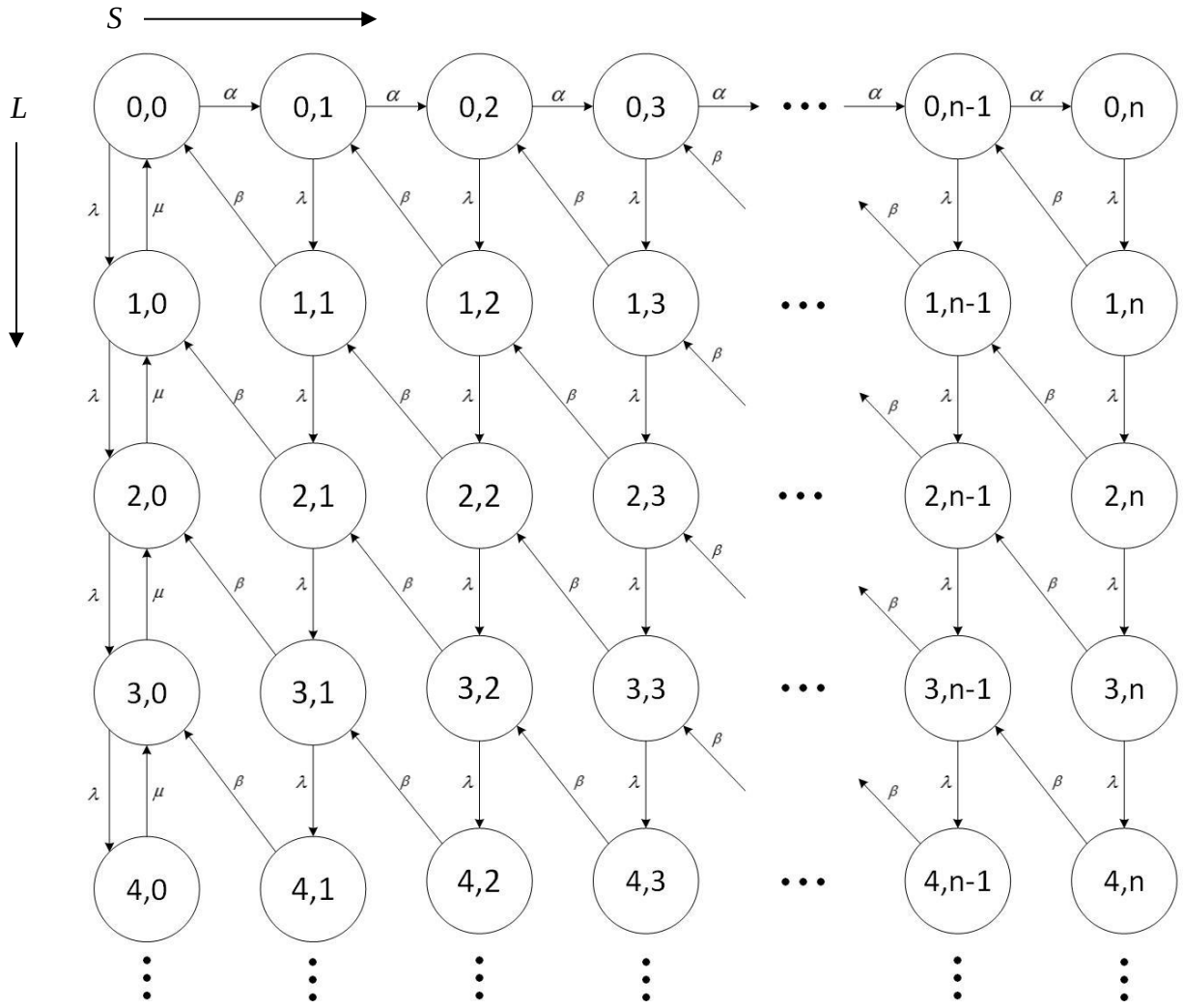


Figure 1. System states and transition-rate diagram

3. Steady-State Analysis

3.1. Sequential Calculation

Herein, we provide a set of $n(n+1)/2$ flow balance equations and an additional equation, which is based on horizontal and vertical cuts, from which it is possible to extract the $[n(n+1)/2+1]$ probabilities $p_{0,0}$ and $p_{i,j}$, $i=0,1,...,n-1$, $j=i+1,i+2,...,n$. Then, any steady-state probability can be calculated sequentially, as we show in what follows. The $n(n+1)/2$ balance equations are:

$$(\beta+\lambda)p_{i,n}=\lambda p_{i-1,n}, \qquad i=1,2,...,n-1 \tag{1}$$

$$(\beta+\lambda)p_{i,j}=\lambda p_{i-1,j}+\beta p_{i+1,j+1},\ i=1,2,...,n-2,\ j=i+1,i+2,..,n-1 \tag{2}$$

$$(\alpha+\lambda)p_{0,j}=\alpha p_{0,j-1}+\beta p_{1,j+1}, \quad j=1,2,...,n-1 \tag{3}$$

$$\lambda p_{0,n}=\alpha p_{0,n-1} \tag{4}$$

In order to obtain the final equation required to complete the set, let $p_{\bullet,j}\equiv\sum_{i=0}^{\infty}p_{i,j}$, $j=0,...,n$, and let $p_{i,\bullet}\equiv\sum_{j=0}^{n}p_{i,j}$, $i=0,1,2,...,\infty$. Then, for all horizontal cuts between row i and row $i+1$, the equilibrium is obtained by:

$$\lambda p_{i,\bullet}=\mu p_{i+1,0}+\beta(p_{i+1,\bullet}-p_{i+1,0}),\ i=0,1,2,...,\infty, \tag{5}$$

and for all vertical cuts between column j and column $j+1$, the equilibrium is obtained by:

$$\alpha p_{0,j}=\beta(p_{\bullet,j+1}-p_{0,j+1}),\ j=0,...,n-1. \tag{6}$$

Summing equation (5) over $i=0,1,2,...,\infty$ yields

$$\lambda\sum_{i=0}^{\infty}p_{i,\bullet}=\mu(p_{\bullet,0}-p_{0,0})+\beta\left(\sum_{i=0}^{\infty}p_{i,\bullet}-p_{0,\bullet}-(p_{\bullet,0}-p_{0,0})\right), \tag{7}$$

and summing equation (6) over $j=0,...,n-1$ yields

$$\alpha(p_{0,\bullet}-p_{0,n})=\beta\left(\sum_{j=0}^{n-1}p_{\bullet,j+1}-(p_{0,\bullet}-p_{0,0})\right). \tag{8}$$

By the relation $\sum_{i=0}^{\infty} p_{i,\bullet} = \sum_{j=0}^{n} p_{\bullet,j} = 1$, equation (7) can be written as

$$\lambda = (\mu - \beta)(p_{\bullet,0} - p_{0,0}) + \beta(1 - p_{0,\bullet}), \tag{9}$$

and equation (8) as

$$\alpha(p_{0,\bullet} - p_{0,n}) = \beta\big(1 - p_{\bullet,0} - (p_{0,\bullet} - p_{0,0})\big). \tag{10}$$

By extracting $p_{\bullet,0}$ from equation (10) and substituting it in equation (9), we obtain

$$(\alpha(\beta - \mu) - \beta\mu)\, p_{0,\bullet} = \alpha(\beta - \mu)\, p_{0,n} - \beta(\mu - \lambda), \tag{11}$$

which includes only the $n+1$ boundary probabilities $p_{0,j}$, $j = 0,1,2,...,n$. Therefore, the set of equations (1)–(4) and (11) can be solved, since they construct a set of $[n(n+1)/2+1]$ independent linear equations in the probabilities $p_{0,0}$ and $p_{i,j}$, $i = 0,1,...,n-1$, $j = i+1, i+2,...,n$.

Now we show how to calculate the entire set of probabilities $p_{i,j}$, $i = 1,2,...,\infty$, $j = 0,1,...,\min\{i,n\}$ in diagonal sequential order (from right to left in Figure 1), on the basis of the already-calculated $[n(n+1)/2+1]$ probabilities: $p_{0,0}$ and $p_{i,j}$, $i = 0,1,...,n-1$, $j = i+1, i+2,...,n$. We start by calculating $p_{n,n}$ from $(\beta + \lambda)\, p_{n,n} = \lambda p_{n-1,n}$. Then, we calculate, one by one, $p_{n-i,n-i}$, $i = 1,...,n-1$, from $(\beta + \lambda)\, p_{n-i,n-i} = \lambda p_{n-i-1,n-i} + \beta p_{n-i+1,n-i+1}$. Next, we calculate $p_{n+1,n}$ from $(\beta + \lambda)\, p_{n+1,n} = \lambda p_{n,n}$. Then we calculate, one by one, $p_{n+1-i,n-i}$, $i = 1,...,n-1$, from $(\beta + \lambda)\, p_{n-i+1,n-i} = \lambda p_{n-i,n-i} + \beta p_{n-i+2,n-i+1}$, and finally, we obtain $p_{1,0}$ from $(\alpha + \lambda)\, p_{0,0} = \mu p_{1,0} + \beta p_{1,1}$. The process is repeated starting with $p_{n+2,n}$, via $p_{n+2-i,n-i}$, $i = 1,...,n-1$, and ends with calculating $p_{2,0}$ from $(\mu + \lambda)\, p_{1,0} = \lambda p_{0,0} + \mu p_{2,0} + \beta p_{2,1}$. In a similar manner, we calculate $p_{n+k-i,n-i}$, $i = 0,...,n$, for $k = 3,4,5,...,\infty$.

By extending equation (1) for $i \geq n$, we obtain $p_{i,n} = \big(\lambda/(\beta+\lambda)\big)^{i} p_{0,n}$, $i = 0,1,2,...,\infty$. Therefore, the probability $p_{\bullet,n}$ can be expressed as a function of the already-calculated $p_{0,n}$:

$$p_{\bullet,n} = \sum_{i=0}^{\infty} p_{i,n} = (1 + \lambda/\beta)\, p_{0,n}. \tag{12}$$

We now compare the fraction of the server's idle time in our model with that in a stable M/M/1 queue, namely, $1-\lambda/\mu>0$. In our model, the server is idle when there are no customers in the system and the server has already prepared *n* PSs. The probability of this event is $p_{0,n}$.

Proposition 1. $\operatorname{sgn}\left(p_{0,n}-\left(1-\frac{\lambda}{\mu}\right)\right)=\operatorname{sgn}\left(\frac{1}{\mu}-\left(\frac{1}{\alpha}+\frac{1}{\beta}\right)\right)$.

Proof. Dividing equation (11) by $\alpha\beta\mu$ yields

$$\left(\frac{1}{\mu}-\frac{1}{\beta}-\frac{1}{\alpha}\right)p_{0,\bullet}=\left(\frac{1}{\mu}-\frac{1}{\beta}\right)p_{0,n}-\frac{1}{\alpha}\left(1-\frac{\lambda}{\mu}\right),$$

which can be written as

$$\left(\frac{1}{\mu}-\frac{1}{\beta}-\frac{1}{\alpha}\right)\sum_{j=0}^{n-1}p_{0,j}=\frac{1}{\alpha}\left(p_{0,n}-1+\frac{\lambda}{\mu}\right).$$

The claim is proved since both $\alpha>0$ and $\sum_{j=0}^{n-1}p_{0,j}>0$. □

Thus, by Proposition 1, in our model, the probability that the server will be idle is greater than (equal to) that in an M/M/1 queue *if and only if* the mean duration of full service is greater than (equal to) the sum of the mean durations of PS and CS. It is conceivable that the total mean duration of the decomposed service, $1/\alpha+1/\beta$, is longer than that of the full (non-decomposed) service, $1/\mu$. However, there may be cases in which $1/\alpha+1/\beta<1/\mu$, due to a negative effect of the customer's presence during the entire service process. The latter case leads to what may be looked at as a paradox: utilizing some of the server's idle time to increase productivity results in the server being idle for a greater proportion of time.

Although we can calculate any probability $p_{i,j}$, expressions for large values of *n* become extremely cumbersome, thereby limiting the capability to compute performance measures of the process. In the following section, we present an alternative method, which is more applicable to such calculations.

3.2. The PGF Method

An alternative method to obtain the steady-state probabilities is based on calculating $n+1$ partial PGFs of the random variable *L*, each one for a given value of the number of PSs $(j = 0,1,...,n)$, as follows:

$$G_j(z) = \sum_{i=0}^{\infty} p_{i,j} z^i, \quad j = 0,1,...,n, \ |z| \le 1. \tag{13}$$

By multiplying each equation in Table 1 by z^i, summing the equations over all values of *i* for each value of *j* separately, and rearranging terms, we get a system of $n+1$ independent linear equations for the $n+1$ (yet unknown) PGFs, $G_j(z)$, $j = 0,1,...,n$:

$$j = 0: \qquad \big((1-z)(\lambda z - \mu)\big)G_0(z) - \beta G_1(z) = \big(\mu(z-1) - \alpha z\big)p_{0,0} - \beta p_{0,1} \tag{14}$$

$$1 \le j \le n-1: \quad z\big(\lambda(1-z) + \beta\big)G_j(z) - \beta G_{j+1}(z) = \alpha z p_{0,j-1} - z(\alpha - \beta)\, p_{0,j} - \beta p_{0,j+1} \tag{15}$$

$$j = n: \qquad \big(\lambda(1-z) + \beta\big)G_n(z) = \alpha p_{0,n-1} + \beta p_{0,n}. \tag{16}$$

Define $d(z) \equiv (1-z)(\lambda z - \mu)$ and $a(z) \equiv \lambda(1-z) + \beta$, then the set of linear equations (14)–(16) can be expressed in a matrix form as $A(z)\vec{g}(z) = \vec{b}(z)$, where

$$A_{(n+1)\times(n+1)}(z) = \begin{pmatrix} d(z) & -\beta & 0 & 0 & \dots & 0 & 0 \\ 0 & za(z) & -\beta & 0 & \dots & 0 & 0 \\ 0 & 0 & za(z) & -\beta & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \dots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \dots & za(z) & -\beta \\ 0 & 0 & 0 & 0 & \dots & 0 & a(z) \end{pmatrix},$$

$$\vec{g}(z) = \begin{pmatrix} G_0(z) \\ G_1(z) \\ \vdots \\ G_j(z) \\ \vdots \\ G_{n-1}(z) \\ G_n(z) \end{pmatrix} \quad \text{and} \quad \vec{b}(z) = \begin{pmatrix} \big(\mu(z-1) - \alpha z\big)p_{0,0} - \beta p_{0,1} \\ \alpha z p_{0,0} - (\alpha z - \beta z)\, p_{0,1} - \beta p_{0,2} \\ \vdots \\ \alpha z p_{0,j-1} - (\alpha z - \beta z)\, p_{0,j} - \beta p_{0,j+1} \\ \vdots \\ \alpha z p_{0,n-2} - (\alpha z - \beta z)\, p_{0,n-1} - \beta p_{0,n} \\ \alpha p_{0,n-1} + \beta p_{0,n} \end{pmatrix}.$$

To obtain $G_j(z)$, we use Cramer's rule. That is, for every $0 \le j \le n$,

$$G_j(z) = \frac{|A_j(z)|}{|A(z)|},$$

where the determinant of $A(z)$ is

$$|A(z)| = d(z)z^{n-1}a^n(z) = (1-z)(\lambda z - \mu)(\lambda(1-z)+\beta)^n z^{n-1}, \tag{17}$$

and $A_j(z)$ is a matrix obtained from $A(z)$ by replacing the *j*-th column with the right-hand side vector $\vec{b}(z)$. This leads to an expression of $G_j(z)$ in terms of the $n+1$ unknown boundary probabilities, $\{p_{0,j}, 0 \le j \le n\}$, appearing in $\vec{b}(z)$. In order to get $\vec{b}(z)$, we need to find $n+1$ equations relating the above $n+1$ boundary probabilities. The traditional method of doing so (e.g., E. Perel and Yechiali, 2008) is by characterizing and using the roots of $|A(z)|$. Since $G_j(z)$ is a PGF defined for all $|z| \le 1$, each root of $|A(z)|$ in that interval is also a root of $|A_j(z)|$ for every $0 \le j \le n$. Solving $|A(z)| = 0$ results in the following roots: $z = 1$, $z = \mu/\lambda$, n multiple roots of $z = 1 + \beta/\lambda > 1$, and $n-1$ multiple roots of $z = 0$. Calculating $|A_j(z)|$ for $j = 0,1,...,n$ yields

$$|A_0(z)| = \sum_{k=0}^{n-1} \beta^k b_k(z) z^{n-k-1} a^{n-k}(z) + \beta^n b_n(z), \tag{18}$$

$$|A_j(z)| = d(z)(za(z))^{j-1}\left[\sum_{k=j}^{n-1} \beta^{k-j} b_k(z) z^{n-k-1} a^{n-k}(z) + \beta^{n-j} b_n(z)\right], \; j = 1,2,...,n. \tag{19}$$

However, except for the equations $|A_0(0)| = 0$ and $|A_1(0)| = 0$ (each yielding equation (4)), substituting the roots $z = 0$ or $z = 1$ in $|A_j(z)|$ results in determinants equal to zero with no elements in the polynomial. Thus, in our case, the use of Cramer's rule is inapplicable to extract the needed equations to calculate the boundary probabilities. Instead, the probabilities $\{p_{0,j}, 0 \le j \le n\}$ can be obtained by solving equations (1)–(4) and (11), as we showed in the previous section. Thus, with known $\vec{b}(z)$, all $G_j(z)$ are derived.

The advantage of using the PGF method over the sequential method proposed in Section 2.1 is the ability to calculate directly (rather than sequentially) any steady-state probability by

$$p_{i,j} = \frac{d^i}{dz^i} G_j(z)\Big|_{z=0},\ i = 0,1,..,\infty \text{ and } j = 0,1,..,n. \tag{20}$$

Moreover, the main advantage of the PGF method is that it provides the ability to directly calculate performance measures of the queueing system. For a system with a capacity of *n* PSs, let $L(n)$ and $L_q(n)$ be the mean number of customers in the system and in queue, respectively. Similarly, let $W(n)$ and $W_q(n)$ be the mean time a customer sojourns in the system and in queue; $S(n)$ and $S_q(n)$ the mean number of PSs in the system and in inventory; and $T(n)$ and $T_q(n)$ the mean time a PS resides in the system and in inventory.

Using the results above we have:

$$L(n) = \sum_{i=1}^{\infty} i p_{i,\bullet} = \sum_{j=0}^{n} \frac{d}{dz} G_j(z)\Big|_{z=1} \tag{21}$$

$$L_q(n) = \sum_{i=1}^{\infty} (i-1) p_{i,\bullet} = \sum_{i=1}^{\infty} i p_{i,\bullet} - \sum_{i=1}^{\infty} p_{i,\bullet} = L(n) - (1 - p_{0,\bullet}). \tag{22}$$

By using Little's law,

$$W(n) = L(n)/\lambda \tag{23}$$

$$W_q(n) = L_q(n)/\lambda. \tag{24}$$

As for the PSs,

$$S(n) = \sum_{j=1}^{n} j p_{\bullet,j} = \sum_{j=1}^{n} j G_j(1) \tag{25}$$

$$\begin{aligned} S_q(n) &= \sum_{j=1}^{n} \left[j p_{0,j} + (j-1)(p_{\bullet,j} - p_{0,j}) \right] = \sum_{j=1}^{n} \left[p_{0,j} + (j-1) G_j(1) \right] \\ &= S(n) + p_{0,\bullet} + p_{\bullet,0} - p_{0,0} - 1 = S(n) - \sum_{i \cdot j > 0} p_{i,j}. \end{aligned} \tag{26}$$

In the summation operator in equation (26) we separate the calculation for $i=0$ (the element $jp_{0,j}$) from the calculation for $i \geq 1$ (the element $(j-1)(p_{\bullet,j} - p_{0,j})$), since when the system is not empty, one of the PSs becomes a CS. The last term of equation (26) can be explained as follows: we subtract one unit from $S(n)$ when both the number of customers and number of CSs is positive, since this unit is converted from a PS to a CS.

Although the server's PS production rate is α, the server generates PSs only when it is idle and the inventory level is less than *n*. Therefore, the effective production rate of PSs is

$$\alpha_{eff}(n) = \alpha(p_{0,\bullet} - p_{0,n}) \tag{27}$$

Using Little's law, we obtain:

$$T(n) = S(n) / \alpha_{eff}(n), \tag{28}$$

$$T_q(n) = S_q(n) / \alpha_{eff}(n). \tag{29}$$

3.3. The Matrix Geometric Method

An alternative to the combined approach of sequential calculations and the PGF method is the use of matrix geometric analysis (Neuts 1981). In this method, given the lexicographic order of the system's states, $\{(0,0),(0,1),...,(0,n);(1,0),(1,1),...,(1,n);...;(i,0),(i,1),...,(i,n);...\}$, we construct an infinitesimal generator matrix, denoted *Q*:

$$Q = \begin{pmatrix} B & A_0 & 0 & 0 & \cdots \\ A_2 & A_1 & A_0 & 0 & \cdots \\ 0 & A_2 & A_1 & A_0 & \\ \vdots & & \ddots & \ddots & \ddots \end{pmatrix}, \tag{30}$$

where the matrices *B*, A_0, A_1 and A_2, each of order $(n+1)\times(n+1)$, are given by

$$B = \begin{pmatrix} -(\alpha+\lambda) & \alpha & 0 & \cdots & 0 & 0 \\ 0 & -(\alpha+\lambda) & \alpha & & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & 0 & 0 & & -(\alpha+\lambda) & \alpha \\ 0 & 0 & 0 & \cdots & 0 & -\lambda \end{pmatrix} \tag{31}$$

$$A_0 = \begin{pmatrix} \lambda & 0 & \cdots & 0 & 0 \\ 0 & \lambda & & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & & \lambda & 0 \\ 0 & 0 & \cdots & 0 & \lambda \end{pmatrix} \tag{32}$$

$$A_1 = \begin{pmatrix} -(\mu+\lambda) & 0 & \cdots & 0 & 0 \\ 0 & -(\beta+\lambda) & & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & & -(\beta+\lambda) & 0 \\ 0 & 0 & \cdots & 0 & -(\beta+\lambda) \end{pmatrix} \tag{33}$$

and

$$A_2 = \begin{pmatrix} \mu & 0 & \cdots & 0 & 0 \\ \beta & 0 & \cdots & 0 & 0 \\ 0 & \beta & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & \beta & 0 \end{pmatrix}. \tag{34}$$

Let $\vec{e} \equiv (1,1,...,1)^T$ and let

$$A \equiv A_0 + A_1 + A_2 = \begin{pmatrix} 0 & 0 & 0 & \cdots & 0 & 0 \\ \beta & -\beta & 0 & \cdots & 0 & 0 \\ 0 & \beta & -\beta & \ddots & 0 & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots \\ 0 & 0 & 0 & \cdots & \beta & -\beta \end{pmatrix}. \tag{35}$$

Theorem 1. The stability condition of the queuing-inventory system is $\lambda < \mu$.

Proof. The condition for stability (Neuts 1981, p. 83) is

$$\vec{\pi} A_0 \vec{e} < \vec{\pi} A_2 \vec{e}, \tag{36}$$

where $\vec{\pi} = (\pi_0, \pi_1, ..., \pi_n)$ is the unique solution of the linear system

$$\begin{aligned} \vec{\pi} A &= \vec{0} \\ \vec{\pi} \vec{e} &= 1. \end{aligned} \tag{37}$$

In our case, $\vec{\pi} = (1,0,...,0)$ due to the structure of *A*, and the stability condition in equation (36) translates into $\lambda < \mu$. □

This result can be intuitively explained as follows: the CS production rate β is exercised only when there are PSs available in the system. At the moment the PSs are exhausted, the service returns to its regular rate, μ. Thus, when the number of customers in the system becomes sufficiently large, all PSs will be used, and the system will imitate a standard M/M/1 queue, resulting in the stability condition of the latter queue, which is not influenced either by β or by the production rate of PSs, α.

Let $\vec{p}_i \equiv (p_{i,0}, p_{i,1},..., p_{i,n-1}, p_{i,n})$ and let R be a matrix of size $(n+1)\times(n+1)$ that satisfies

$$A_0 + RA_1 + R^2 A_2 = 0_{n+1,n+1}. \tag{38}$$

The explicit entries of the left-hand side of equation (38) are given in the supplementary file. Since there may be several values for each entry in R, only the smallest positive value should be taken (Neuts 1981, p. 82). In most cases, the entries $r_{i,j}$ of the matrix R can be found only by numerical calculations (see Chapter 8 in Latouche and Ramaswami 1999). A common method for computing the matrix R is by *successive substitutions* (Neuts 1981, p. 37). For our problem, however, we have succeeded in obtaining closed-form expressions for *all* $r_{i,j}$, as given in Theorem 2 below. Such an explicit *complete* solution for the matrix R is rare in the literature.

Theorem 2.

$$r_{i,0} = \begin{cases} \lambda/\mu & i = 0 \\ \dfrac{C_i \beta^{i-1}\lambda^{i+1}}{\mu(\beta+\lambda)^{2i-1}} + \displaystyle\sum_{k=1}^{i-1} \frac{C_{i-k}\beta^{i-k-1}\lambda^{i-k+1}}{(\beta+\lambda)^{2(i-k)}} r_{k,0} & 1 \le i \le n \end{cases}, \tag{39}$$

$$r_{i,j} = \begin{cases} 0 & 0 \le i < j \le n \\ \dfrac{C_{i-j}\beta^{i-j}\lambda^{i-j+1}}{(\beta+\lambda)^{2(i-j)+1}} & 0 < j \le i \le n \end{cases}, \tag{40}$$

where

$$C_m = \frac{(2m)!}{(m+1)!m!} \text{ is the } m\text{-th Catalan number (Koshy 2008), } m = 0,1,2,... . \tag{41}$$

Proof. See supplementary file.

As a consequence of Theorem 2 and its proof, we state:

Corollary 1.

(i) All the entries of R above the main diagonal are zero.

(ii) Except for the first column, all the entries in any diagonal, whether the main diagonal or any diagonal below it, are equal to each other.

(iii) The numerical coefficients in the explicit expressions of each entry are related to Catalan numbers.

(iv) The entries on the main diagonal of R are the reciprocals of the roots, which are greater than 1, of $|A(z)|$, namely, $r_{0,0} = \lambda/\mu$ and $r_{i,i} = \lambda/(\beta+\lambda)$, $i = 1,2,...,n$.

In order to calculate $p_{i,j}$, $i = 0,1,...,\infty$, $j = 0,1,...,n$, we first have to obtain the vector of boundary probabilities $\vec{p}_0$. This is accomplished (Latouche and Ramaswami 1999, p. 144) by solving the following system:

$$\begin{matrix} \vec{p}_0[B + RA_2] = \vec{0} \\ \vec{p}_0[I - R]^{-1}\vec{e} = 1 \end{matrix}. \tag{42}$$

Finally, the rest of the steady-state probabilities are calculated by

$$\vec{p}_i = \vec{p}_0 R^i,\ i = 1,2,...,\infty. \tag{43}$$

Using the results above we state

$$L(n) = \sum_{i=1}^{\infty} i \sum_{j=0}^{n} p_{i,j} = \sum_{i=1}^{\infty} i\vec{p}_i\vec{e} = \sum_{i=1}^{\infty} i\vec{p}_1 R^{i-1}\vec{e} = \vec{p}_1\left(\sum_{i=1}^{\infty} iR^{i-1}\right)\vec{e}.$$

Since $\sum_{i=1}^{\infty} iR^{i-1} = \left([I-R]^{-1}\right)^2 = [I-R]^{-2}$, then

$$L(n) = \vec{p}_1[I-R]^{-2}\vec{e}. \tag{44}$$

Similarly to equation (22),

$$L_q(n) = L(n) - (1 - \vec{p}_0\vec{e}). \tag{45}$$

In order to obtain $S(n)$, we define the column vector $\vec{v} \equiv (0,1,2,...,n)^T$, so

$$S(n) = \sum_{i=0}^{\infty}\sum_{j=0}^{n} jp_{i,j} = \sum_{i=0}^{\infty} \vec{p}_i \vec{v} = \left(\sum_{i=0}^{\infty} \vec{p}_i\right)\vec{v} = \vec{p}_0\left(\sum_{i=0}^{\infty} R^i\right)\vec{v} = \vec{p}_0[I-R]^{-1}\vec{v}\,. \tag{46}$$

Similarly, in order to get $S_q(n)$, we define $\vec{u} \equiv (0,0,1,2,...,n-1)^T$ for $n \geq 1$, so

$$\begin{aligned} S_q(n) &= \sum_{j=1}^{n} jp_{0,j} + \sum_{i=1}^{\infty}\sum_{j=1}^{n}(j-1)p_{i,j} = \sum_{j=1}^{n} p_{0,j} + \sum_{i=0}^{\infty}\sum_{j=1}^{n}(j-1)p_{i,j} \\ &= \vec{p}_0\vec{e} - p_{0,0} + \sum_{i=0}^{\infty}\vec{p}_i\vec{u} = \vec{p}_0\vec{e} - p_{0,0} + \left(\sum_{i=0}^{\infty}\vec{p}_i\right)\vec{u} = \vec{p}_0\vec{e} - p_{0,0} + \vec{p}_0\left(\sum_{i=0}^{\infty} R^i\right)\vec{u} \\ &= \vec{p}_0\vec{e} - p_{0,0} + \vec{p}_0[I-R]^{-1}\vec{u}. \end{aligned} \tag{47}$$

The expressions of W, W_q, T and T_q are obtained by using Little's law, as presented in equations (23), (24), and (27)–(29).

4. Examples

Herein, we apply the three proposed solution methods presented in Section 3. We use Maple 2015 software to extract closed-form expressions of the steady-state probabilities and various performance measures (e.g., $L(n), L_q(n), S(n), S_q(n), T(n), T_q(n)$). The sequential calculation method and the PGF method enable us to solve problems up to $n = 16$ due to computational limitations of the software. In contrast, when using the matrix geometric method, we are able to solve problems even beyond $n = 200$ in a few minutes. This computational-effort advantage is achieved due to Theorem 2, in which we analytically obtain all the entries of the matrix *R*.

4.1. Combining the Sequential Calculation Method with the PGF Method

The Case of n = 1

In order to calculate the boundary probabilities, $p_{0,0}$ and $p_{0,1}$, we solve equations (1)–(4) and (11) with $n=1$. However, for $n=1$, equations (1)–(3) do not exist, and we remain with equations (4) and (11), which can be explicitly written as

$$\lambda p_{0,1} = \alpha p_{0,0}, \tag{48}$$

$$(\alpha(\beta-\mu)-\beta\mu)(p_{0,0}+p_{0,1}) = \alpha(\beta-\mu)p_{0,1} - \beta(\mu-\lambda). \tag{49}$$

Solving equations (48) and (49) results in

$$p_{0,0} = \frac{\lambda\beta(\mu-\lambda)}{\alpha\beta(\mu-\lambda)+\lambda\mu(\alpha+\beta)} = \frac{1}{\frac{\alpha}{\lambda}+\frac{1+\alpha/\beta}{1-\lambda/\mu}}, \tag{50}$$

$$p_{0,1} = \frac{\alpha\beta(\mu-\lambda)}{\alpha\beta(\mu-\lambda)+\lambda\mu(\alpha+\beta)} = \frac{1}{1+\frac{1/\alpha+1/\beta}{1/\lambda-1/\mu}}. \tag{51}$$

In order to find the PGFs $G_0(z)$ and $G_1(z)$, we solve equations (14)–(16) with $n=1$ while substituting equations (50) and (51). However, for $n=1$, equation (15) does not exist, and we remain with equations (14) and (16), which result in

$$G_0(z) = \frac{\lambda\beta(\mu-\lambda)\big(\mu(\lambda+\beta)-\lambda z(\mu-\alpha)\big)}{\big(\alpha\beta(\mu-\lambda)+\lambda\mu(\alpha+\beta)\big)\big(\lambda(1-z)+\beta\big)(\mu-\lambda z)}, \tag{52}$$

$$G_1(z) = \frac{\alpha\beta(\mu-\lambda)(\lambda+\beta)}{\big(\alpha\beta(\mu-\lambda)+\lambda\mu(\alpha+\beta)\big)\big(\lambda(1-z)+\beta\big)}. \tag{53}$$

Then, using equations (21)–(22), we obtain the following performance measures:

$$L(1) = \frac{\lambda\big(\beta\mu(\alpha\mu+\beta\lambda)-\alpha\lambda(\mu-\lambda)(\beta-\mu)\big)}{\beta(\mu-\lambda)\big(\lambda\mu(\alpha+\beta)+\alpha\beta(\mu-\lambda)\big)}, \tag{54}$$

$$L_q(1) = \frac{\lambda^2\big(\beta\lambda(\alpha+\beta)+\alpha\mu(\mu-\lambda)\big)}{\beta(\mu-\lambda)\big(\lambda\mu(\alpha+\beta)+\alpha\beta(\mu-\lambda)\big)}, \tag{55}$$

and by using equations (25)–(26), we obtain

$$S(1) = \frac{\beta + \lambda}{\beta + \lambda \dfrac{\mu(\alpha + \beta)}{\alpha(\mu - \lambda)}}, \tag{56}$$

$$S_q(1) = \frac{1}{1 + \dfrac{\lambda\mu(\alpha + \beta)}{\alpha\beta(\mu - \lambda)}} = \frac{\beta}{\beta + \lambda} S(1) \cdot \tag{57}$$

Using Little's law, the expressions of $W(1)$ and $W_q(1)$ are readily obtained by using equations (23)–(24). In order to obtain the expression of $T(1)$ and $T_q(1)$, as presented in equations (28)–(29), we first have to calculate the effective production rate of PSs, which is

$$\alpha_{eff}(1) = \alpha p_{0,0} = \frac{1}{\dfrac{1}{\lambda} + \dfrac{1/\alpha + 1/\beta}{1 - \lambda/\mu}} \cdot \tag{58}$$

Substituting equations (56)–(58) in equations (28) and (29) results in

$$T(1) = 1/\beta + 1/\lambda, \tag{59}$$

$$T_q(1) = 1/\lambda. \tag{60}$$

Equations (59) and (60) can be interpreted as follows: the mean time of one unit in inventory, $T_q(1)$, equals the mean time between consecutive customer arrivals, whereas the mean time of one unit in the system , $T(1)$, includes the mean CS production time, $1/\beta$, in addition to $T_q(1)$.

Note that when α approaches zero, the server does not succeed in building any PS when it is idle, so the system becomes a standard M/M/1 queue. This result coincides with the expressions in equations (50)–(51) and (54)–(55), which reduce to the corresponding equations of the M/M/1 queue when $\alpha \to 0$.

The Case of n = 2

Applying the sequential calculation method (Section 3.1) in combination with the PGF method (Section 3.2), we obtain the following boundary probabilities and performance measures:

$$p_{0,0} = \frac{(\mu - \lambda)\beta(\beta + \alpha + \lambda)\lambda^2}{((\mu - \alpha)\beta + \alpha\mu)\lambda^3 + ((\mu - \alpha)\beta + \alpha\mu)(\beta + 2\alpha)\lambda^2 + ((\mu - \alpha)\beta + 2\alpha\mu)\beta\alpha\lambda + \alpha^2\beta^2\mu}$$

$$p_{0,1} = \frac{\lambda\alpha\beta(\beta+\lambda)(\mu-\lambda)}{((\mu-\alpha)\beta+\alpha\mu)\lambda^3+((\mu-\alpha)\beta+\alpha\mu)(\beta+2\alpha)\lambda^2+((\mu-\alpha)\beta+2\alpha\mu)\beta\alpha\lambda+\alpha^2\beta^2\mu}$$

$$p_{0,2} = \frac{\alpha^2\beta(\beta+\lambda)(\mu-\lambda)}{((\mu-\alpha)\beta+\alpha\mu)\lambda^3+((\mu-\alpha)\beta+\alpha\mu)(\beta+2\alpha)\lambda^2+((\mu-\alpha)\beta+2\alpha\mu)\beta\alpha\lambda+\alpha^2\beta^2\mu}$$

$$L(2) = \frac{\left(\begin{array}{l}(\lambda^3(\mu\beta^2+\alpha(\beta-\mu)(\lambda+3\alpha-\beta+\mu))+((\beta^2-4\beta\mu+3\mu^2)\alpha^2 \\ \qquad +\mu\beta(2\mu\alpha+\beta^2))\lambda^2+\mu\alpha((3\mu-2\beta)\alpha+\beta\mu)\beta\lambda+\mu^2\beta^2\alpha^2)\lambda\end{array}\right)}{(\mu-\lambda)\beta(\lambda^2(\alpha(\mu-\beta)+\beta\mu)(\lambda+\beta+2\alpha)+\alpha\beta((2\mu-\beta)+\beta\mu)\lambda+\mu\beta^2\alpha^2)}$$

$$S(2) = \frac{2\alpha(\mu-\lambda)(0.5\lambda^3+(\beta+1.5\alpha)\lambda^2+0.5\beta\lambda(\beta+4\alpha)+\alpha\beta^2)}{\lambda^2(\alpha(\mu-\beta)+\beta\mu)(\lambda+\beta+2\alpha)+\beta((2\mu-\beta)\alpha+\beta\mu)\alpha\lambda+\alpha^2\beta^2\mu}$$

Cases of $n \geq 3$

Explicit values of the performance measures for cases of $n = 3,4,...,16,$ were obtained using Maple 2015 software. Since the expressions are very long and cumbersome, they are not presented here.

4.2. The Matrix Geometric Method

Calculating the matrix R for n = 7

The main source of complexity in the matrix geometric method is the calculation of the matrix R by solving equation (38), given the matrices in equations (32)–(34). The explicit representation of R for $n = 7$ is given below:

$$R = \begin{pmatrix}
\frac{\lambda}{\mu} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{\lambda^2}{\mu(\beta+\lambda)} & \frac{\lambda}{\beta+\lambda} & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{\lambda^3(2\beta+\lambda)}{\mu(\beta+\lambda)^3} & \frac{\beta\lambda^2}{(\beta+\lambda)^3} & \frac{\lambda}{\beta+\lambda} & 0 & 0 & 0 & 0 & 0 \\
\frac{\lambda^4(5\beta^2+4\beta\lambda+\lambda^2)}{\mu(\beta+\lambda)^5} & \frac{2\beta^2\lambda^3}{(\beta+\lambda)^5} & \frac{\beta\lambda^2}{(\beta+\lambda)^3} & \frac{\lambda}{\beta+\lambda} & 0 & 0 & 0 & 0 \\
\frac{\lambda^5(14\beta^3+14\beta^2\lambda+6\beta\lambda^2+\lambda^3)}{\mu(\beta+\lambda)^7} & \frac{5\beta^3\lambda^4}{(\beta+\lambda)^7} & \frac{2\beta^2\lambda^3}{(\beta+\lambda)^5} & \frac{\beta\lambda^2}{(\beta+\lambda)^3} & \frac{\lambda}{\beta+\lambda} & 0 & 0 & 0 \\
\frac{\lambda^6(42\beta^4+48\beta^3\lambda+27\beta^2\lambda^2+8\beta\lambda^3+\lambda^4)}{\mu(\beta+\lambda)^9} & \frac{14\beta^4\lambda^5}{(\beta+\lambda)^9} & \frac{5\beta^3\lambda^4}{(\beta+\lambda)^7} & \frac{2\beta^2\lambda^3}{(\beta+\lambda)^5} & \frac{\beta\lambda^2}{(\beta+\lambda)^3} & \frac{\lambda}{\beta+\lambda} & 0 & 0 \\
\frac{\lambda^7(132\beta^5+165\beta^4\lambda+110\beta^3\lambda^2+44\beta^2\lambda^3+10\beta\lambda^4+\lambda^5)}{\mu(\beta+\lambda)^{11}} & \frac{42\beta^5\lambda^6}{(\beta+\lambda)^{11}} & \frac{14\beta^4\lambda^5}{(\beta+\lambda)^9} & \frac{5\beta^3\lambda^4}{(\beta+\lambda)^7} & \frac{2\beta^2\lambda^3}{(\beta+\lambda)^5} & \frac{\beta\lambda^2}{(\beta+\lambda)^3} & \frac{\lambda}{\beta+\lambda} & 0 \\
\frac{\lambda^8(429\beta^6+572\beta^5\lambda+429\beta^4\lambda^2+208\beta^3\lambda^3+65\beta^2\lambda^4+12\beta\lambda^5+\lambda^6)}{\mu(\beta+\lambda)^{13}} & \frac{132\beta^6\lambda^7}{(\beta+\lambda)^{13}} & \frac{42\beta^5\lambda^6}{(\beta+\lambda)^{11}} & \frac{14\beta^4\lambda^5}{(\beta+\lambda)^9} & \frac{5\beta^3\lambda^4}{(\beta+\lambda)^7} & \frac{2\beta^2\lambda^3}{(\beta+\lambda)^5} & \frac{\beta\lambda^2}{(\beta+\lambda)^3} & \frac{\lambda}{\beta+\lambda}
\end{pmatrix}$$

5. Economic Analysis: Finding the Optimal *n*

In this section we provide an economic analysis of our queueing-inventory model. We consider two types of cost rates: one is proportional to the number of customers in the system; the other is proportional to the number of PSs held in inventory. Let *c* be the cost per unit of time per customer in the system, and let *h* be the holding cost per unit of time per inventoried PS (assuming no holding cost for a PS that moves on to the CS phase). The objective is to minimize the total expected cost per time unit by controlling the PS capacity, *n*, i.e.,

$$\min_{n\in\{0,1,2,\ldots\}} \left\{ Z(n) = cL(n) + hS_q(n) \right\} . \tag{61}$$

In Section 3.3, we obtained closed-form expressions for $L(n)$ (equation (44)) and $S_q(n)$ (equation (47)), so a line search can be readily applied to find the optimal value of *n* over a closed interval. However, it is difficult to derive the properties of $Z(n)$ analytically, especially with regard to convexity. Therefore, using an efficient line-search method, such as the golden section search (see Bazaraa et al. 2006), does not guarantee finding the global minimum of $Z(n)$. Since $L(n)$ decreases in *n* and $S_q(n)$ increases in *n*, we conjecture that $Z(n)$ is convex in *n*. In this section, we examine our conjecture using numerical examples and conduct a sensitivity analysis. Indeed, the numerical analysis for values of *n* up to 100 supports our conjecture.

We use the following parameter values as a base-example: $\lambda = 8$, $\mu = 10$, $\alpha = 20$, $\beta = 18$, $c = 1$ and $h = 0.2$. These values were chosen such that (i) $\lambda < \mu$, (ii) $1/\alpha + 1/\beta > 1/\mu$, and (iii) h is considerably smaller than *c*. We calculate $Z(n)$ for $n = \{0,1,2,...,100\}$ and repeat this process for other parameter values as follows: we keep the values of μ and *c* fixed, and use four additional values (two above and two below the base value) for each parameter, where the other parameter values of the base example are held constant. Specifically, we use the following additional parameter values: $\lambda = \{5, 7, 9, 9.5\}$, $\alpha = \{15, 17.5, 22.5, 25\}$, $\beta = \{14, 16, 20, 22\}$ and $h = \{0.1, 0.15, 0.25, 0.3\}$. Since in the numerical examples the difference series $\{Z(n+1) - Z(n)\}$ for each value of λ, α, β and h are all

monotonic increasing over the domain $n = \{0,1,2,...,99\}$, the objective function $Z(n)$ is convex on the integers over this domain. In order to emphasize the difference between the plots for different parameter values, we limit the *n* axis in Figures 2(b), 2(c) and 2(d) to $n = 20$. In what follows we define n^* as the optimal PS capacity in each numerical example, and the optimal point $\left(n^*, Z(n^*)\right)$ is depicted as a triangle on the corresponding curve in Figure 2.

Figure 2(a) presents $Z(n)$ for five different values of λ, and shows that n^* and $Z(n^*)$ increase in λ, and that when λ gets closer to μ, $Z(n)$ becomes only slightly sensitive to changes in *n* in the vicinity of n^*. Figure 2(b) presents $Z(n)$ for five different values of α, and shows that n^* does not increase in α, whereas $Z(n^*)$ decreases in α. This result can be explained as follows: when α is large, the server does not have to prepare many PSs in advance, since, when it becomes idle, it can quickly prepare new PSs; thus, it is preferable to maintain a low inventory and thereby reduce holding costs. Figure 2(c) presents $Z(n)$ for five different values of β, and shows, interestingly, that n^*, as a function of β, first increases from 7 to 8 and then decreases from 8 to 7, whereas $Z(n^*)$ decreases in β. We investigated the effect of β for additional parameter values: $\alpha = \{15,25\}$ and $\mu = \{9,12,14\}$, which are not presented in the figure, and obtained the same qualitative result. Figure 2(d) presents $Z(n)$ for five different values of *h*, and shows that n^* decreases in *h*, whereas $Z(n^*)$ increases in *h*.

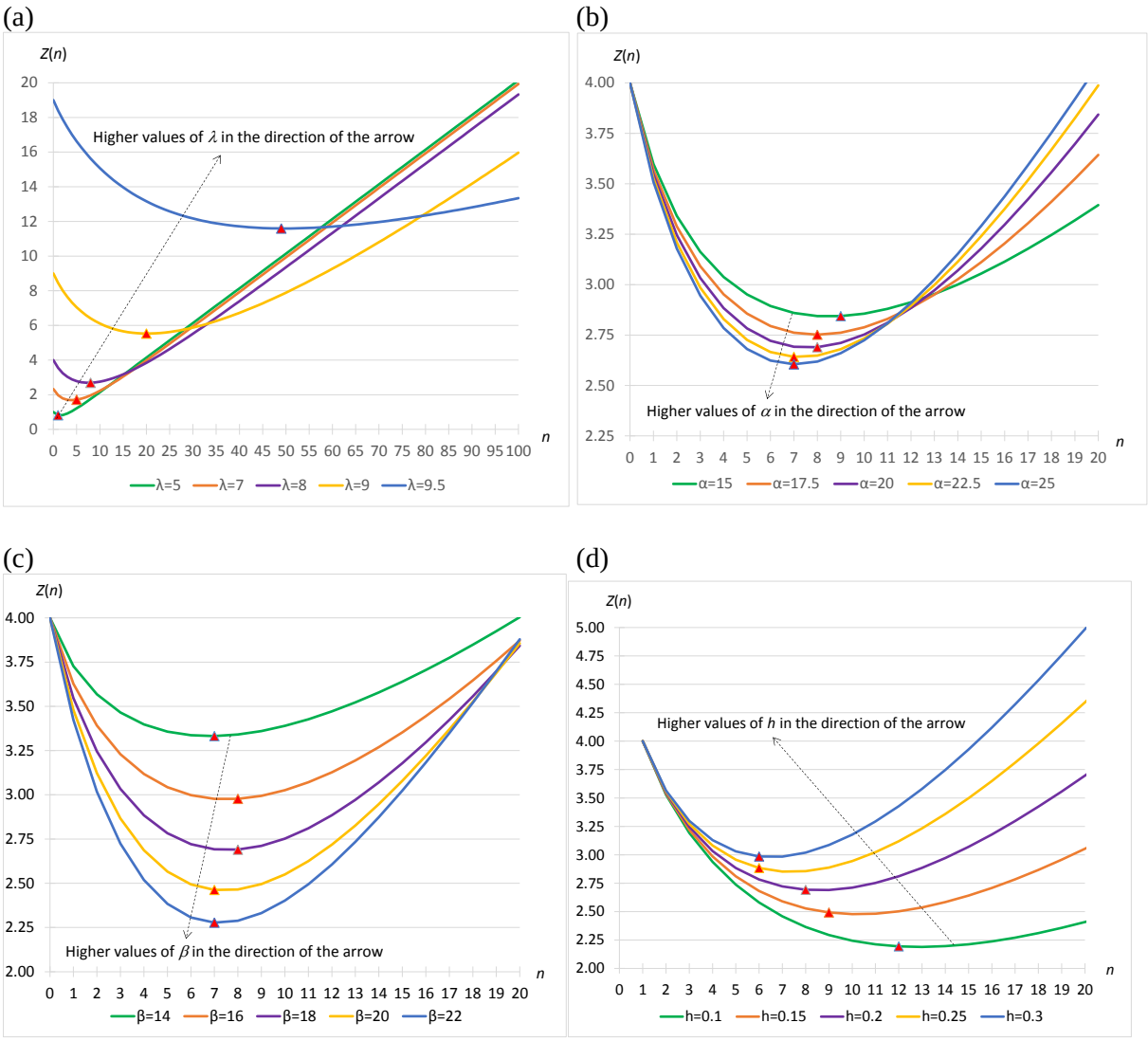
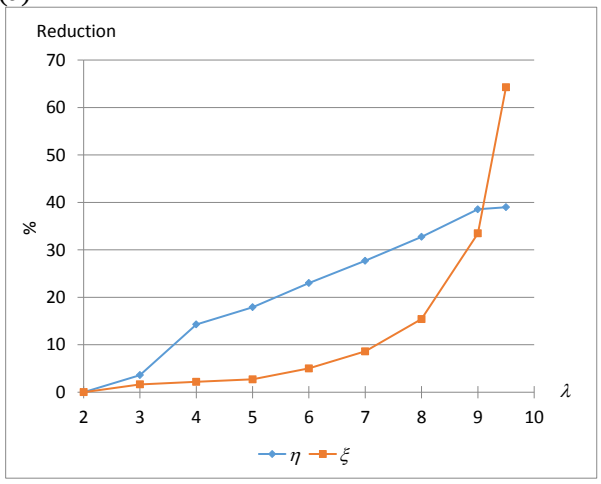


Figure 2. Total expected cost per time unit, $Z(n)$, as a function of n when $\mu = 10$ and $c = 1$ for (a) $\lambda = \{5, 7, 8, 9, 9.5\}$, $\alpha = 20$, $\beta = 18$ and $h = 0.2$; (b) $\alpha = \{15, 17.5, 20, 22.5, 25\}$, $\lambda = 8$, $\beta = 18$ and $h = 0.2$; (c) $\beta = \{14, 16, 18, 20, 22\}$, $\lambda = 8$, $\alpha = 20$ and $h = 0.2$; (d) $h = \{0.1, 0.15, 0.2, 0.25, 0.3\}$, $\lambda = 8$, $\alpha = 20$ and $\beta = 18$.

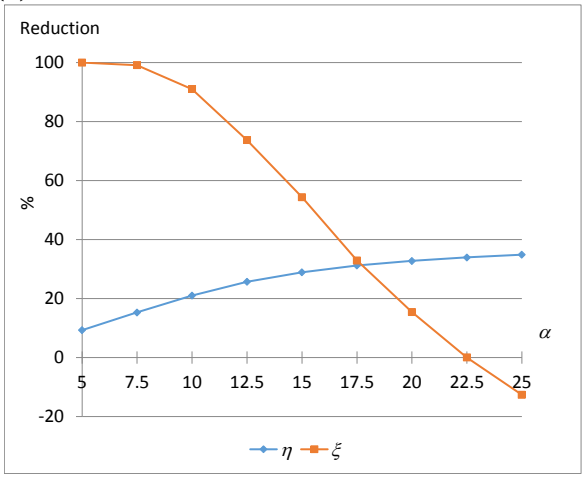
We compare the performance of the optimal solution under our queueing-inventory model with that of the corresponding M/M/1 model (obtained by substituting $n = 0$) in terms of percentage reduction in the total expected cost and in the idle time of the server, i.e., $\eta = \dfrac{Z(0) - Z(n^*)}{Z(0)} \times 100\%$ and

$\xi = \dfrac{(1-\lambda/\mu) - p_{0,n^*}}{1-\lambda/\mu} \times 100\%$, respectively. Figures 3(a)–3(d) present η and ξ for different parameter values of λ, α, β and h, where the other parameter values are held constant as in the base example.

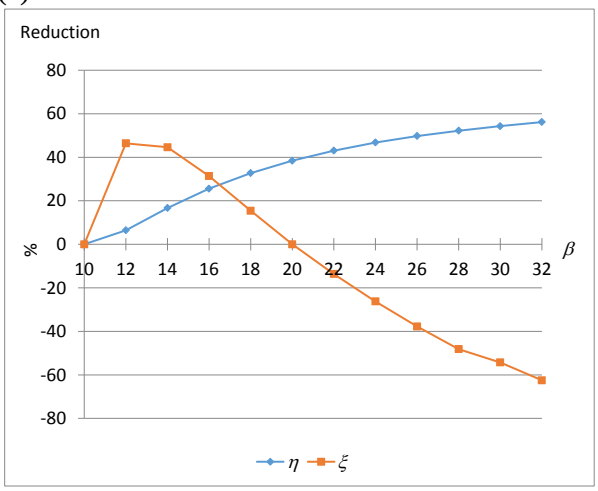
(a)



(b)



(c)



(d)

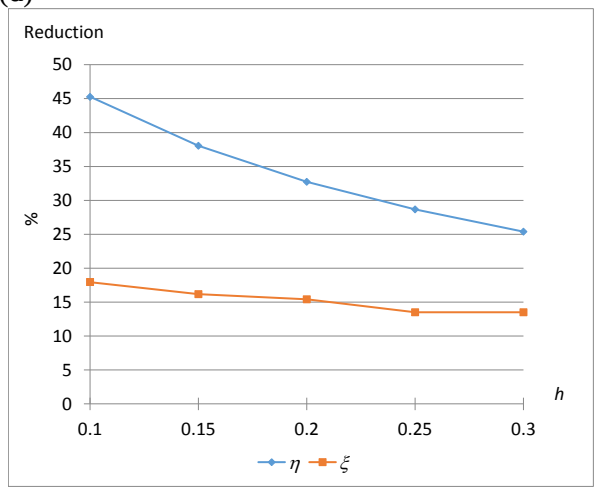


Figure 3. (a) η and ξ for $\lambda = \{2, 3, 4, 5, 6, 7, 8, 9, 9.5\}$ when $\mu = 10$, $\alpha = 20$, $\beta = 18$, $c = 1$ and $h = 0.2$; (b) η and ξ for $\alpha = \{5, 7.5, 10, 12.5, 15, 17.5, 20, 22.5, 25\}$ when $\lambda = 8$, $\mu = 10$, $\beta = 18$, $c = 1$ and $h = 0.2$; (c) η and ξ for $\beta = \{10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32\}$ when $\lambda = 8$, $\mu = 10$, $\alpha = 20$, $c = 1$ and $h = 0.2$; (d) η and ξ for $h = \{0.1, 0.15, 0.2, 0.25, 0.3\}$ when $\lambda = 8$, $\mu = 10$, $\alpha = 20$, $\beta = 18$ and $c = 1$.

Figures 3(a)–3(d) show that our model is always economically more beneficial than the standard M/M/1 queue $(0 \le \eta \le 100\%)$, whereas the server's idle time may be smaller than $(0 < \xi \le 100\%)$, equal to $(\xi = 0)$ or larger $(\xi < 0)$ than that in an M/M/1 queue. These results are due to the objective of minimizing the total expected cost per time unit (equation (61)). Figure 3(a) shows that the percentage reduction, either in the total expected cost or in the fraction of the server's idle time, obtained by using our model is higher for higher values of the customer arrival rate λ. Figure 3(b) shows that a higher PS production rate α results in a higher percentage reduction in the total expected cost, whereas the percentage reduction in the fraction of the server's idle time decreases. Moreover, as is claimed in Proposition 1, we see that from $\alpha = 1/(1/\mu - 1/\beta) = 22.5$ the fraction of the server's idle time in our model is even larger than that in an M/M/1 queue, as reflected in negative values of ξ. Figure 3(c) shows that a higher CS production rate β results in a higher percentage reduction in the total expected cost. An interesting observation is that ξ is not monotonic in β: first it increases and then it decreases. Moreover, as is claimed in Proposition 1, we see that from $\beta = 1/(1/\mu - 1/\alpha) = 20$ the fraction of the server's idle time in the proposed model is even larger than that in an M/M/1 queue, as reflected in negative values of ξ. Figure 3(d) shows that the percentage reductions in both the total expected cost and the fraction of the server's idle time are lower for higher values of the holding cost *h*. This observation can be explained by the lower incentive to prepare PSs and hold them in inventory, as is shown in Figure 2(d).

6. Conclusions

In this study, analyzing an M/M/1-type system, we investigated the implications of attempting to increase the system's efficiency by utilizing the server's idle time to produce preliminary services for incoming customers. Although this procedure is often exercised in real-life queues, it has not been studied analytically in the literature. In order to investigate such a system, we constructed a two-dimensional space state that considers both queueing sizes and inventory levels. Combining a blend of analytical

methods, we obtained closed-form expressions for the steady-state probabilities of the system states, and calculated various performance measures. In particular, when using the matrix geometric approach, we were able to analytically derive closed-form expressions of all the entries of the rate matrix *R*, regardless of its size. This enabled us to solve problems even for $n \geq 200$. The explicit formulae of the entries $r_{i,j}$ of *R* are based on Catalan numbers. This result is notable in light of the fact that, in typical applications of the matrix geometric method, explicit calculation of the entries of *R* is rarely possible.

We have proven analytically that (i) the stability condition of our model is identical to that of a standard M/M/1 queue, and that (ii) there is a condition under which utilizing some of the server's idle time to produce preliminary services results in the server being idle for a greater proportion of time than it would be in a standard M/M/1 system. Numerical examples reveal several interesting properties of the problem: (i) the total expected cost function is convex over the inventory capacity; thus, efficient line-search methods can be used to find its optimal value; (ii) the optimal inventory capacity and the reduction in the fraction of the server's idle time (as compared with the standard M/M/1 system) are not monotonic in the CS production rate. Thus, we show that using the server's idle time to produce PSs can improve the system's overall performance.

There are various possible directions for further research in this domain. For example, it would be interesting to extend the model to multiple servers or to limited-capacity queueing systems. One might also augment the objective function $Z(n)$ with operating costs of the server as a function of the PS production rate α and/or the CS production rate β. Another potential direction for further research is analyzing a queue with two types of customers: one that agrees to use a preliminary service to reduce online service duration, and another that insists on obtaining an uninterrupted full service.

References

Armony, M. 2005. Dynamic routing in large-scale service systems with heterogeneous servers. *Queueing Systems* **51** 287-329.

Armony, M., A. R. Ward. 2010. Fair dynamic routing in large-scale heterogeneous-server systems. *Operations Research* **58** 624-637.

Armony, M., A. R. Ward. 2013. Blind fair routing in large-scale service systems with heterogeneous customers and servers. *Operations Research* **61** 228-243.

Baba, Y. 2005. Analysis of a GI/M/1 queue with multiple working vacations. *Operations Research Letters* **33** 201-209.

Bazaraa, M. S., H. D. Sherali, C. M. Shetty. 2006. *Nonlinear Programming, Theory and Algorithms*. John Wiley, Hoboken, NJ.

Boxma, O. J., S. Schlegel, U. Yechiali. 2002. A note on the M/G/1 queue with a waiting server, timer and vacations. *American Mathematical Society Translations Series* **2** 207 25-35.

Cachon, G. P., F. Zhang. 2007. Obtaining fast service in a queueing system via performance-based allocation of demand. *Management Science* **53** 408-420.

Doshi, B. T. 1986. Queueing systems with vacations—a survey. *Queueing Systems* **1** 29-66.

Haridass, M., R. Arumuganathan. 2015. Analysis of a single server batch arrival retrial queueing system with modified vacations and N-policy. *RAIRO-Operations Research* **49** 279-296.

Ke, J. C., C. H. Wu, Z. G. Zhang. 2013. A note on a multi-server queue with vacations of multiple groups of servers. *Quality Technology and Quantitative Management* **10** 513-525.

Kella, O., U. Yechiali. 1988. Priorities in M/G/1 queue with server vacations. *Naval Research Logistics* **35** 23-34.

Kella, O. 1989. The threshold policy in the M/G/1 queue with server vacations. *Naval Research Logistics* **36** 111-123.

Koshy, T. 2008. *Catalan Numbers with Applications*. Oxford University Press, Oxford.

Latouche, G., V. Ramaswami. 1999. *Introduction to Matrix Analytic Methods in Stochastic Modeling*. *ASA-SIAM Series on Statistics and Applied Probability*. SIAM, Philadelphia, PA .

Lee, D. H., W. S. Yang. 2013. The N-policy of a discrete time Geo/G/1 queue with disasters and its application to wireless sensor networks. *Applied Mathematical Modelling* **37** 9722-9731.

Levy, Y., U. Yechiali. 1975. Utilization of idle time in an M/G/1 queueing system. *Management Science* **22** 202-211.

Levy, Y., U. Yechiali. 1976. An M/M/s queue with servers' vacations. *Infor* **14** 153-163.

Lim, D. E., D. H. Lee, W. S. Yang, K. C. Chae. 2013. Analysis of the GI/Geo/1 queue with N-policy. *Applied Mathematical Modelling* **37** 4643-4652.

Mandelbaum, A., P. Momcilovic, Y. Tseytlin. 2012. On fair routing from emergency departments to hospital wards: QED queues with heterogeneous servers. *Management Science* **58** 1273-1291.

Moreno, P. 2007. A discrete-time single-server queue with a modified N-policy. *International Journal of Systems Science* **38** 483-492.

Mytalas, G. C., M. A. Zazanis. 2015. An MX/G/1 queueing system with disasters and repairs under a multiple adapted vacation policy. *Naval Research Logistics* **62** 171-189.

Neuts, M. F. 1981. *Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach*. Johns Hopkins University Press, Baltimore, MD.

Perel, E., U. Yechiali. 2008. Queues where customers of one queue act as servers of the other queue. *Queueing Systems* **60** 271-288.

Perel, N., U. Yechiali. 2014a. The Israeli queue with infinite number of groups. *Probability in the Engineering and Informational Sciences* **28** 1-19.

Perel, N., U. Yechiali. 2014b. The Israeli queue with retrials. *Queueing Systems* **78** 31-56.

Rosenberg, E., U. Yechiali. 1993. The $M^X/G/1$ queue with single and multiple vacations under the LIFO service regime. *Operations Research Letters* **14**(3) 171-179.

Servi, L. D., S. G. Finn. 2002. M/M/1 queues with working vacations (m/m/1/wv). *Performance Evaluation* **50** 41-52.

Takagi, H. 1991. *Queueing Analysis, Volume 1: Vacation and Priority Systems.* North-Holland, Amsterdam.

Tian, N., Z. G. Zhang. 2006. *Vacation Queueing Models: Theory and Applications*. Springer Science & Business Media, New York, NY.

Yadin, M., P. Naor. 1963. Queueing systems with a removable service station. *Operational Research Quarterly* **14** 393-405.

Yechiali, U. 2004. On the M^X/G/1 queue with a waiting server and vacations. *Sankhya* **66**(1) 159-174.

Yang, D. Y., C. H. Wu. 2015. Cost-minimization analysis of a working vacation queue with N-policy and server breakdowns. *Computers & Industrial Engineering* **82** 151-158.

Wei, Y., M. Yu, Y. Tang, J. Gu. 2013. Queue size distribution and capacity optimum design for N-policy Geo (λ1, λ2, λ3)/G/1 queue with setup time and variable input rate. *Mathematical and Computer Modelling* **57** 1559-1571.