Article

First Passage Analysis in a Queue with State Dependent Vacations

Jewgeni H. Dshalalow * and Ryan T. White

Department of Mathematical Sciences, College of Engineering and Science, Florida Institute of Technology, Melbourne, FL 32901, USA
* Correspondence: edshalalow@scienceatlantic.net

Abstract: This paper deals with a single-server queue where the server goes on maintenance when the queue is exhausted. Initially, the maintenance time is fixed by deterministic or random number \( T \). However, during server’s absence, customers are screened by a dispatcher who estimates his service times based on his needs. According to these estimates, the dispatcher shortens server’s maintenance time and as the result the server returns earlier than planned. Upon server’s return, if there are not enough customers waiting (under the \( N \)-Policy), the server rests and then resumes his service. At first, the input and service are general. We then prove a necessary and sufficient condition for a simple linear dependence between server’s absence time (including his rest) and the number of waiting customers. It turns out that the input must be (marked) Poisson. We use fluctuation and semi-regenerative analyses (previously established and embellished in our past work) to obtain explicit formulas for server’s return time and the queue length, both with discrete and continuous time parameter. We then dedicate an entire section to related control problems including the determination of the optimal \( T \)-value. We also support our tractable formulas with many numerical examples and validate our results by simulation.

Keywords: continuous time parameter queue; fluctuation theory; marked point process; random walk process; optimization; operational calculus; single vacation; semi-regenerative analysis; \( N \)-Policy; maintenance

MSC: 60G50; 60G55; 60K05; 60K25; 60K30; 90B22

1. Introduction

1.1. Description of the Model

This paper deals with a single server queueing exhaustive system under \( N \)-Policy and a single vacation. When the system is emptied, the server leaves for a routine maintenance during a period of time initially set to equal \( T (>0) \). During this time, the server is unavailable for service of regular customers. However, when a batch of customers arrives at the system, the dispatcher tentatively estimates their total service time according to their demands, say \( X \), and commands the server to shorten his vacation time by exactly \( X \) units of time. So, if at time \( t_1 \), a random batch of \( Y_1 \) customers comes in with its estimated service time of \( X_1 = x_1 + \ldots + x_{Y_1} \), the server remaining absence time at time \( t_1 \) becomes \((T - t_1 - X_1)^+\). If it is zero, the server immediately returns to the system and resumes his service. Otherwise, he is out of system until the next arrival at time \( t_2 \) of a second group of \( Y_2 \) customers with an estimated service time \( X_2 = x_{Y_1+1} + \ldots + x_{Y_1+Y_2} \) making the remaining absence time of the server \((T - X_1 - t_2 - X_2)^+\) and he returns to the system immediately if the latter quantity is zero. If during the entire absence of the server, there is no arrival, the server returns to the system and rests. In fact, he rests if there are less than \( N \) customers in the system.

Note that the estimated service time has no impact during the time when the server returns to the system, as it may be different from the real service time. In practice, it is
impossible to determine how much real service time customers end up requiring, but dependent on customers’ alleged needs it makes sense to estimate their service times to better organize server’s absence and return.

As mentioned, on his return, if there are less than \(N\) \((N \geq 1)\) customers waiting, the server rests until the queue replenishes to \(N\) or more customers. The latter is a more likely scenario, because the input is bulk. Only then is the service activated. We will return to further specifications.

Note that this is not a usual \(N\)-Policy/vacations-system in which the server goes on a single or multiple vacations. In this model, he is absent during a state dependent, periodically updated, random period of time. Namely, the service vacation time is gradually shortened by new arrivals under their estimated service times.

1.2. More Formal Description

Now we return to our more or less formal description of the system. Define an auxiliary continuous time parameter process \(S(t)\) as

\[
S(t) = t_1 + (t + X_1)1_{[t_1,t_2)}(t) + \ldots + (t + X_1 + \ldots + X_{k-1})1_{[t_{k-1},t_k)}(t) + \ldots
\]

where \(t_0 = 0\). This process determines server’s absence time at any time \(t \geq 0\) accelerated by increments \(X_i\) at epochs \(t_i, i = 1,2,\ldots\)

To simplify the formalism we first assume that \(N = 1\). When the server is done with their maintenance, he returns to the system, with possibly no customers waiting (it happens when \(t_1 > T\)). In this case, he waits until a next batch of customers arrives. The time of return and the number of customers present in the system resemble our approach in [1–4] now with one more “passive” component \(Y\), because on server’s return, we need to know how many customers have so far accumulated in the waiting room. In addition, the choice of the queue in the system is subject to two alternatives to be discussed in a moment.

Now with arrivals of customers at times \(t_1, t_2, \ldots \) and their respective batch sizes \(Y_1, Y_2, \ldots \), define \(B_n = Y_1 + \ldots + Y_n\) and \(A_n = A_{n-1} + \delta_n + X_n = t_n + X_1 + \ldots + X_n\), where \(\delta_n = t_n - t_{n-1}\).

From Figure 1, where \(S(t)\) defined in (1) is in green, we see an excerpt of the process between the two arrivals at \(t_{j-1}\) and \(t_j\). Here \(B_{j-1}\) is the total number of all customers that arrived by time \(t_{j-1}\), whereas \(E_{j-1} = X_1 + \ldots + X_{j-1}\) is the total estimated service time of these customers. \(S(t)\) is the cumulative time spent in maintenance by time \(t \in (t_{j-1}, t_j)\) appreciated by \(E_j\). Because of the latter, the actual maintenance time is shortened compared to the initially planned \(T\).

From Figure 2 below (where \(S(t)\) is in green), upon the server’s return, there will be \(B_{v-1}\) customers waiting, because the crossing of \(T\) by \(S(t)\) occurs at some instant \(\tau_v\) in interval \((t_{v-1}, t_v)\). Note that the next arrival at \(t_v\) will take place after server’s earlier return at \(\tau_v\).

Figure 3 below depicts the situation when the first crossing of \(T\) occurs at the arrival of the \(t_v\)th batch of customers \(Y_v\) and not earlier. It happens that the estimated service time \(X_v\) of that batch, along with \(\delta_v\), is the increment added to \(A_{v-1}\), which makes \(S(\tau_v) = S(t_v)\) cross \(T\) for the first time.
Figure 1. Excerpt of the process between arrivals at $t_{j-1}$ and $t_j$ with shortened maintenance time.

Figure 2. Crossing of $T$ prior to time $t_\nu$. 

\[ A_j = A_{j-1} + \delta_j + X_j \]

\[ A_{j-1} + \delta_j \]

\[ B_{j-1} \]

\[ S(t) \]

\[ \delta_j \]

\[ X_j \]

\[ t_{j-1} \]

\[ t_j \]

$A_i = \delta_i + X_i$

$B_i = X_i + Y_i$

$\delta_i$

$S(\tau_i) = T$

crossing of $T$ occurs at $\tau_{i,*} < t_\nu$

$T$

actual time spent in maintenance if crossing of $T$ by $S(t)$ occurs prior to $t_\nu$
As we see in Figures 2 and 3, the server’s return to the system takes place on the first crossing of $T$ by $S(t)$ with two incidents. If $\nu = \inf \{ n : A_n \geq T \}$, then $t_\nu$ is the nearest arrival epoch at which $A_\nu \geq T$. However, the crossing can take place earlier in interval $(t_{\nu-1}, t_\nu)$, because $S(t)$ also appreciates with the unit velocity. So if the latter happens, the number of customers in the system will be regarded as $B_{\nu-1}$. If $T$ will be crossed only at $t_\nu$, then the total number of customers on server’s return is $B_{\nu}$.

Figure 4 is yet another realization of the two variants of crossing depicting the events in a vicinity of the two crossing options dependent on the position of threshold $T$ within interval $(A_{\nu-1}, A_{\nu}]$, that is whether $T \in (A_{\nu-1}, A_{\nu-1} + \delta_{\nu})$ or $T \in (A_{\nu-1} + \delta_{\nu}, A_{\nu}]$. 

---

**Figure 3.** Crossing of $T$ at time $t_\nu$.

**Figure 4.** Two variants of crossings.
1.3. Our Results

Here we analyzed a single-server queue with unlimited buffer and single, state dependent, vacations. Initially, we make no assumptions on the nature of the input and service, that is, the system is of GI/G-NSV/1 type to obtain the return time and the number of customers $Q(\tau_n)$ (i.e., $B_{i-1}$ or $B_i$) on server’s return at $\tau_n$. Dependent on the situation, the total number of customers gathered at $\tau_n$ can be less than $N$, and for that matter, it can be even zero (if $t_1 > T$). In the latter case, the server rests until the queue crosses $N$ (which is equal to or greater than $N$), and only then does he resume his service. He processes customers singly under the FIFO discipline, and he goes on a single vacation when the queue is exhausted in accordance with the above specified state dependent vacation policy.

We use and embellish fluctuation theory to arrive at a closed form for joint and marginal distributions of $\tau_n$ and $Q(\tau_n)$. Further on, we obtain the distribution of the two moments $t_\mu$ and $W_\mu = Q(t_\mu)$, where $t_\mu$ is the beginning of the new busy period when $W_\mu \geq N$. The entire server’s endeavor consists of his maintenance time and a possible wait. A very interesting special relationships between $\tau_n$ and $Q(\tau_n)$ and $t_\mu$ and $W_\mu$ turn out to be linear with the same multiplier if and only if the input stream is Poisson. In general, we assume that the setup maintenance time $T$ (that is then randomly reduced) is a constant. However, in a variety of useful special cases, there is no analytical complexity to upgrade $T$ to a random variable, which we discuss.

We then continue with the rest of our queueing analysis under the assumption that the input is marked Poisson, herewith working on the $M^X/G-NSV/1/\infty$ type queue under our special vacation policy. Here we obtain a closed form for the invariant probability measure of the embedded process observed on departures. To make our study complete and attain to an optimal value of $T$ (initially set) we employ semi-regenerative analysis to obtain the steady state probability distribution of the continuous time parameter queueing process $Q(t)$. This is followed by a discussion of various performance measures and control problems related to the determination of the optimal initial value of $T$ to reduce the frequently unwanted numbers of activations and deactivations (switchovers) regarding busy and non-busy periods, the number of customers waiting during the maintenance time, and the number of overall waiting customers. We also provide numerous special cases, numerical examples, and simulated cases validating our results.

1.4. Topically Related Literature

There is a very extensive research done on queues with vacations and various associated policies. Monograph [5] by Tian and Zhang of 2006 is an excellent source of multitudes of papers on vacations that is well categorized and analyzed. A very useful recent paper of 2021 [6] by Panta et al. surveys over 150 articles. It seems like the pioneering work in the 70s has never faded, but in all truth it has become even more attractive for many, because this work is very practical and universally applicable. Perhaps most numerous are queues with multiple vacations, but they are not related to our work. In these settings, the server goes on multiple repeated vacation segments until a desired number of customers (such as $N$) accumulate. The server does not break any particular vacation segment in the middle, but he will not start a new one if the queue has attained to a desired minimal length. Many papers on $T$-Policy deal with multiple vacations except that every vacation segment there is assumed constant, equal to some real number $T$ and thus are special cases of multiple vacations. (Cf. Alfa [7]).

The work on single vacations is closest to ours. A standard setting such as in [8] by Liu et al., Gupur [9], Choudhury [10], Y. Tang and X. Tang [11], S. Jin and W. Yue [12], Ghosh et al. [13], and Lee et al. [14] treats an M/G/1-type exhaustive queue. The server goes on a single vacation trip when the system becomes empty. He resumes his service if there is at least one customer waiting or the server rests until one arrives. A slightly embellished variant the M/G/1-type (cf. Gupta and Sikdar [15]) is when a single server processes customers in batches sized between $a$ and $b$. If upon a service completion, the number of available units drops below $a$, the server goes on a single vacation, and on his
return, if there are still less than \( a \) units, he rests until queue replenishes to at least \( a \). A more advanced setting (such as in [16] by Lee et al.) takes on \( M^X/G/1 \)-type queue under \( N \)-Policy. That is, the server leaves the empty system on a single vacation and on his return, when he finds less than \( N \) customers waiting, he rests and then resumes his service until the queue crosses \( N \). Kempa [17] targets the virtual waiting time process in a single-vacation queue with a finite waiting room. A very interesting recent article [18] by Kazem and Al-Obaidi deals with a single vacation queue and \( N \)-Policy using fluctuation analysis.

Less standard settings pertain to \( GI/M/1 \) (cf. [19] by Chaea et al. and [20] by Gabryel et al.) and multiserver queues. In the former case they undergo a different analysis compared to \( M/G/1 \) counterpart. Multiserver queues assume synchronous leave of a group of servers. See [21] by Zhang and Tian (\( M/M/c \)), Wu and Ke [22] (\( M/M/c \)), Xu and Zhang [23] (\( M/M/c \)). Another paper [24] by Madan et al. deals with \( M/M_1M_2/2 \) queue with heterogeneous service with synchronous and asynchronous single vacations.

Other systems with single vacations are with additional features, such as retrials (Gao and Zhang [25] and Malik and Upadhya [26]), unreliable servers, or both (cf. Gao et al. [27], Ke and Lin [28]), and by Yang et al. [29] with retrials, server breakdowns and repair.

Another class of modeling close to ours is with state dependent vacations. In essence, all systems with \( N \)-Policy and multiple vacations have elements of queue length dependent vacations. Al-Matar and Dshalalow [30] modeled a system under \( N \)-Policy and multiple vacations. A vacation series terminates, when the total vacation time exceeds some random time \( T \). Then the server returns to the system, and if there are less than \( N \) customers in the queue, the server returns to the maintenance facility for the second phase and continues his maintenance until the queue crosses \( N \).

However, certain papers introduce special forms of state dependent vacations. Chao and Rahman [31,32] studied two systems with multiple server’s vacations, \( GI/M(n)/1/K \) and \( M(n)/G/1/K \). In the former, the vacations are distributed exponentially with state dependent rate \( v_n \), where \( n \) is the number of customers in the system. The vacation rate changes along with every new arrival, and the whole vacation lasts until there are \( N \) customers waiting. In the second model, it is the arrival rate that alters. The vacation times are independent and identically distributed (i.i.d.) and in accordance with the multiple vacations and \( N \)-Policy rule. The arrivals rates are \( \lambda_n \) and \( v_n \) dependent on whether the server is in his primary mode or on vacation, respectively, and either rate depends on the number of customers in the system.

A recent paper [33] of 2021, by Tamrakar, G.K. and Banerjee, deals with an interesting \( M/G/Y/1 \)-type queue with single and multiple vacations in which the service and vacation times depend on batch size and queue length (upon the vacation start), respectively. This system was inspired by a group testing method during a pandemic such as COVID-19. The batch size taken for testing lies between the two thresholds, \( a \) and \( b \) \((a \leq b)\). A new service consists of testing in a pool of \( r \) samples, provided \( a \leq r \leq b \). If the whole sample is tested negative, then the server is done with that group and takes a new one. If the pool of samples is tested positive, the server further tested, apparently breaking it in smaller groups. If \( r < a \), then the server (a test employee) goes on vacation. During the vacation, the server performs maintenance, such as restocking the health care inventory, increasing people’s awareness, and attending the quarantine room. The vacation time is picked from the set of \( a \) different distributions \((\mu_0, \ldots, \mu_{a-1})\) that are general in their nature and it is chosen \( \mu_r \) if the server leaves behind \( r \) units in the system.

Another recent paper [34] by Xie et al. studies an \( M/G/Y/1/K \) queue with queue-length dependent multiple vacations. The most interesting aspect of this work is an application to computer operating systems.

1.5. Technique Related Literature

Our techniques, at least in part, fall into the area of fluctuations of stochastic processes and first passage time problems. See Abolnikov et al. [35], Dshalalow [36], Dshalalow
et al. [37], Takács [38] applied to queueing, Dshalalow [2], Kyprianou and Pistorius [39], and Muzy et al. [40] applied to finance, Dshalalow and White [41,42] applied to reliability, Dshalalow [1], Takács [43], and White [44] applied to stochastic processes, and Redner [45] applied to physics.

The most basic view of fluctuations is that of random variables that can be measured by the variance. A more contemporary view of fluctuations pertains to the analytical behavior of stochastic processes about critical thresholds whose crossing at certain epoch of time (called the first passage time) is of interest. Fluctuations also cross with random walks. More specifically, a classical random walk of a particle or walker takes place on a multi-dimensional grid or lattice. The walker attempts to escape some bounded set $A$. In this case, not only the first passage time, but also walker’s location on its escape are common targets of an investigation, because not only does the walker crosses the boundary, but it may land at some remote location away from set $A$. It happens when the walker jumps from one node of the lattice to another rather than drifts one step at a time.

The most typical lattice on which the walk takes place is $\mathbb{Z}^d$. However, more recent work pertains to walks on graphs of arbitrary configurations, which in turn apply to the walks on networks. Another modification of a random walk is when the grid is not even deterministic, but it is generated randomly each time the walker lands at some node. Thus, the locations of respective nodes are in $\mathbb{R}^d$ and not equidistant. Under these more general conditions, the walker intends to escape a deterministic subset $A$ of $\mathbb{R}^d$ and the underlying studies target the escape time from $A$ as well as the location of the walker outside $A$. See for instance, a recent survey [4] by Dshalalow and White.

Fluctuations of random walks are subject to widespread studies in stochastic analysis, cf. Barral and Loiseau [46] and an excellent survey by Bingham [47], then random walk on graphs by Confortia and Leonard [48], Telcs [49], and Volchenkov [50], random walks on trees by Andreolettia and Dielb [51] and Takács [52], and random walks on random lattices by White and Dshalalow [3].

Random walks find applications in various areas of science and technology, such as physics, in particular, nuclear physics, surveyed in [53] by Durkee in 2021 and astrophysics [54] by Uchaikin and Gusarov, epidemiology in Pu et al. [55], genetics by Murase et al. [56], finance in Jarner and Kronborg [57] and Scalas [58], medicine (Alzheimer) in Rahimiasi et al. [59] and (cancer research) in Seah et al. [60]), strategic networks in Dshalalow and White [41], electrical networks in [49] by Telcs, social networks by Li et al. [61] and Sarkar and Moore [62]) Additionally, a variant of random walks known as quantum random walks or just quantum walks in quantum mechanics and quantum computing, Kiumi et al. [63] and Sajida et al. [64]. Note that quantum walks are quantum analogs of classical random walks which are dynamical tools used to control the motion of a quantum particle in space and time.

Random walks are often used in stochastic models, in particular, queueing (cf., Abolnikov, et al. [65] and Lemoine [66]), stochastic games (e.g., Dshalalow [67]), and reliability analysis (cf. Dshalalow and White [42,68]).

1.6. Layout of the Paper

After an informal description in Section 1, we rigorously formalize our GI$^X$/G-NSV/1 model and obtain joint and marginal distributions for server’s return time $\tau_v$ and the number of waiting customers $Q(\tau_v)$ in the form of transforms. We use operational calculus to establish analytically tractable formulas, all in Section 2. In Section 3, we deal with special cases. First, we prove that the functionals of $\tau_v$ and $Q(\tau_v)$ are linearly related if and only if the input process is Poisson. It is allowed to be marked though. From that on, at least some of our further efforts pertain to the system GI$^X$/G-NSV/1. We also support our claims in Section 3 (and also some other sections) that the results are tractable. Section 4 is fully dedicated to numerical results that amplify our claims that the results are tame. We also provide simulated cases to validate the results. Sections 5–8 deal with queueing processes, which we analyze combining fluctuations and semi-regenerative analysis, again.
arriving at formulas for the steady state distribution in the form of analytically tractable and computationally tame functionals. Section 8 discusses various control problems, in particular, determination of the optimal value of $T$ that the system needs to set initially. Overall, we offer a stand-alone analysis different from the known to us literature on queueing, in particular, queueing under N-Policy and with single vacations or maintenance.

2. Fluctuation Analysis of the System

From Section 1, we can conveniently model the server’s absence and system’s status by the following scheme. In the absence of any arrivals in the system, the server will spend exactly $T$ units of time outside the system and then returns. If there is at least one arrival in interval $[0, T]$ of a random batch $Y_1$ of customers with a total estimated process time $X_1 = x_1 + \ldots + x_{Y_1}$ of $Y_1$ customers, then the server shortens his vacation by $X_1$ units of time. It means then that the server who vacationed $\delta_1 = t_1$ time on the first arrival, will be counted $A_1 = \delta_1 + X_1$ vacation time instead of $t_1$ which further shortens his vacation from $T$ down to $(T - \delta_1 - X_1)^+$. If $X_1 + \delta_1$ has not crossed $T$, the server continues his maintenance until $t_2$, which is the arrival time of the second batch $Y_2$ of customers with total estimated service time $X_2$. Thus, the server at $t_2$ spends a total of $X_1 + \delta_1 + X_2 + \delta_2 = A_2$ time outside the system (not $\delta_1 + \delta_2 = t_2$). The server continues this pattern until some $A_n$ or $A_n + \delta_{n+1}$ reaches or crosses $T$, when he returns to the system.

Consider the random measure

$$\mathcal{A} \otimes T \otimes B = \sum_{k=1}^{\infty} (X_k + \delta_k, \delta_k, Y_k) \epsilon_{t_k}$$

(2)

that describes the transition of the system during server’s maintenance with no regard of his return to the system. For example, $\mathcal{A} \otimes T \otimes B_{\lfloor 0, t \rfloor}$ gives the position of the server with respect to his time spent in maintenance at $t_m = \max\{t_k \leq t : k = 1, 2, \ldots\}$ that is accelerated through respective increments $X_1, \ldots, X_m$, the $t_m$, and the total number of customers, $Y_1 + \ldots + Y_m$ arrived in interval $[0, t]$.

Let

$$A_k = \sum_{i=1}^{k} (X_i + \delta_i) \quad \text{and} \quad B_k = \sum_{i=1}^{k} Y_i, \quad \text{while} \quad t_k = \sum_{i=1}^{k} \delta_i$$

(3)

Accordingly,

$$\gamma(a, \theta, z) = E e^{-a A_k - \theta \delta_k} z^{Y_k} = E e^{-a (X_k + \delta_k)} \theta \delta_k z^{Y_k}$$

(4)

with the marginal transform of the $k$th batch size of entering units

$$a(z) = Ez^{Y_k}$$

(5)

If $\mathcal{A} \otimes T \otimes B$ is with position independent marking, then

$$\gamma(a, \theta, z) = E e^{-a X_k} z^{Y_k} e^{-\theta \delta_k}$$

(6)

Further, define

$$\nu = \inf\{n : A_n \geq T\}$$

to be the minimum number $\nu$ of arrivals when $A_\nu$ crosses $T$, while $A_{\nu-1}$ did not. Obviously, $t_\nu$ is the first passage time when $A_\nu$ crosses $T$. Here we see that of the three components in the process $\mathcal{A} \otimes T \otimes B$, only $\mathcal{A}$ (namely $A_k$) is active, while the rest are passive. The reason why we call the components this way is because $\mathcal{A}$ fluctuates about threshold $T$, while the other two are not related to any threshold. Thus, the notion of active or passive applies to the relation of a component toward a threshold if it exists. In this case, $A_k$ has
a threshold that it crosses at some point of time, while the other two components do not have their thresholds and assume their values appreciated by the first passage time \( t_\nu \).

\[ \mathcal{A} \otimes \mathcal{T} \otimes \mathcal{B} \] gives a comprehensive descriptive measure of the status of the system during server’s maintenance, but it overlooks a fine crossing of \( S(t) \) and thus the server’s return between two arrivals. This is because \( t_\nu \) is not necessarily the return time of the server, since the threshold \( T \) is not only crossed by component \( \mathcal{A} \), but also by \( S(t) \) from within interval \( (t_\nu - 1, t_\nu) \). If \( S(t) \) crosses \( T \) earlier, the server will be immediately called off. Consequently, there may be an earlier return of the server at some point of time \( t \in (t_\nu - 1, t_\nu) \), if \( S(t) \) hits \( T \).

The return time is a r.v. denoted by \( \tau_\nu \) and it can assume one of the two values. It either equals \( t_\nu \) if there is no crossing of \( T \) by \( S(t) \) in interval \( (t_\nu - 1, t_\nu) \) (but exactly at \( t_\nu \)) or it equals \( t_\nu - 1 + T - A_{\nu - 1} \) and thus the return takes place in \( (t_\nu - 1, t_\nu) \). As per Figure 4 (shown in Section 1), it is easy to conclude that the return time of the server depends on the position of \( T \) towards the two subintervals within \( (A_{\nu - 1}, A_{\nu}) \). Namely, on the location of \( T \) in \( (A_{\nu - 1}, A_{\nu} + \delta_\nu) \) or \( T \in (A_{\nu - 1} + \delta_\nu, A_{\nu}) \). Correspondingly, \( \tau_\nu \) is specified as

\[
\tau_\nu = \begin{cases} 
    t_\nu, & T \in (A_{\nu - 1} + \delta_\nu, A_{\nu}) \\
    t_\nu - 1 + T - A_{\nu - 1}, & T \in (A_{\nu - 1}, A_{\nu} + \delta_\nu).
\end{cases}
\]

In Figure 4, we have two variants of crossing, where \( \tau_\nu \) takes one of the two values relative to the interval \( (t_\nu - 1, t_\nu) \).

As we see it, the first passage time \( \tau_\nu \) of \( S(t) \) can come earlier than the first passage time \( t_\nu \) of \( \mathcal{A} \otimes \mathcal{T} \otimes \mathcal{B} \).

Denote by \( S_\nu = S(\tau_\nu) \) specified as

\[
S(\tau_\nu) = \begin{cases} 
    A_{\nu}, & T \in (A_{\nu - 1} + \delta_\nu, A_{\nu}) \\
    T, & T \in (A_{\nu - 1}, A_{\nu} + \delta_\nu).
\end{cases}
\]

The number of customers in the system waiting upon server’s return is due to the following pick.

\[
Q_\nu = Q(\tau_\nu) = \begin{cases} 
    B_{\nu}, & T \in (A_{\nu - 1} + \delta_\nu, A_{\nu}) \\
    B_{\nu - 1}, & T \in (A_{\nu - 1}, A_{\nu - 1} + \delta_\nu)
\end{cases}
\]

Here, the first case indicates there is no crossing of \( T \) in \((t_\nu - 1, t_\nu)\) by \( S(t) \). The latter case indicates there is an actual crossing of \( T \) by \( S(t) \) at \( \tau_\nu \in (t_\nu - 1, t_\nu) \).

We target the LST \( \Phi_\nu \) of the joint distribution of the server’s return time \( \tau_\nu \), the crossing level \( S_\nu \) of \( S(t) \) at \( \tau_\nu \), \( T \) or \( A_\nu \), the pre-return time \( t_{\nu - 1} \) (the time of the \((v - 1)\)st batch’s arrival), the total estimated time \( A_{\nu - 1} \) of \( v - 1 \) arrived batches of customers, and the number of customers \( Q(\tau_\nu) \) waiting upon server’s return. Thus,

\[
\Phi_\nu(\alpha, \beta, \theta, \rho, z) = E e^{-\alpha A_{\nu - 1} - \beta S(\tau_\nu) - \theta t_{\nu - 1} - \theta \tau_\nu z Q(\tau_\nu)}.
\]

**Theorem 1.** The joint LST \( \Phi_\nu(\alpha, \beta, \theta, \rho, z) \) of the system upon server’s return satisfies

\[
\Phi_\nu(\alpha, \beta, \theta, \rho, z) = \Phi^{d}_\nu + \Phi^{s}_\nu = L^{-1}_{\alpha, \beta, \theta, \rho, z} \left( \frac{1}{x + \beta + \theta} \Phi^{d}_\nu(x) + \frac{1}{x} \Phi^{s}_\nu(x) \right)(T)
\]

where

\[
\Phi^{d}_\nu(x) = \frac{1 - \gamma(0, x + \beta + \theta, 1)}{1 - \gamma(\alpha + \beta + x, \theta + \theta, z)}
\]

(7)

\[
\Phi^{s}_\nu(x) = \frac{\gamma(\beta, \theta + x, z) - \gamma(\beta + x, \theta, z)}{1 - \gamma(\alpha + \beta + x, \theta + \theta, z)}
\]

(8)

\[
\gamma(\alpha, \theta, z) = E e^{-\alpha (X_{\nu} + \delta_\nu) - \theta \delta_\nu z Y_1}
\]
and $\mathcal{L}^{-1}$ is the inverse Laplace transform with respect to variable $x$.

**Proof.** First, we introduce the set of exit indices $\{v(p) = \inf\{n : A_n > p\} : p > 0\}$ replacing a single $T$, so that $\nu = \nu(T^-)$, with the associated set of functionals $\{\Phi_{\nu(p)} : p > 0\}$. We will derive an expression for $\Phi_{\nu(p)}$ and then use operational calculus to find a formula for $\Phi_{\nu}$.

The functional $\Phi_{\nu(p)}$ can be computed as a sum of functionals over a convenient partition of the sample space $\Omega$. The partition includes two subsets, such that, given $\nu(p) = j$, $p \in I^d_j = (A_{j-1}, A_j - \delta_j]$ or $p \in I^s_j = (A_{j-1} + \delta_j, A_j]$. Indeed, $\{\nu(p) = j\} = \{p \in (A_{j-1}, A_j]\}$ and we break the latter interval into two subintervals, $I^d_j \cup I^s_j$, where

$$
I^d_j (d—departure on level crossing) \quad I^s_j (s—return upon a surge)
$$

So that, if $p \in I^d_j$, crossing of $p$ by $S(t)$ occurs at time $t = \tau_j = t_{j-1} + p - A_{j-1}$, and in fact, $S(\tau_j) = p$ sharp. However, if $p \in I^s_j$, then the crossing of $p$ by $S(t)$ occurs exactly at $t_j$, but in this case, $S(t_j)$ almost surely (a.s.) exceeds $p$ and $S(t_j) = A_j = A_{j-1} + \delta_j + X_j$. Thus, the crossing of $p$ by $S(t)$ occurs at time $\tau_j$ specified as

$$
\tau_j = \begin{cases} 
        t_{j-1} + p - A_{j-1}, & \text{if } p \in I^d_j, \quad \text{while } Q(\tau_j) = B_{j-1} \\
        t_j, & \text{if } p \in I^s_j, \quad \text{while } Q(\tau_j) = B_j.
\end{cases}
$$

See Figure 5, with two alternative locations of threshold $p$ from within interval $(A_{j-1}, A_j]$.

![Figure 5. Alternate locations of $p$ within interval $(A_{j-1}, A_j]$](image-url)
Therefore, we have
\[ \Phi^d_v(p) = \sum_{j=1}^{\infty} e^{-\alpha A_j - \beta S_j - \delta t_j - \theta} \mathbf{1}_{p \in I_j^p} \mathcal{Z}(\gamma) \]

\[ = \sum_{j=1}^{\infty} e^{-\alpha A_j - \beta p - \delta t_j - \theta(1 + \lambda p - A_j - 1)} \mathbf{1}_{p \in I_j^p} \mathcal{Z}(B_{j-1}) \]

\[ = \sum_{j=1}^{\infty} e^{-(\beta + \beta)p} \mathbf{1}_{p \in I_j^p} e^{-\alpha(\theta - \beta)t_j - \theta} \mathcal{Z}(B_{j-1}) \text{ (with } p \text{-dependent part in red)} \]

Next, we apply operator \( \mathcal{L}_p^d \) to \( \Phi^d_v(p) \) defined as follows.
\[ \mathcal{L}_p^d = (x + \beta + \theta) \mathcal{L}_p(\cdot)(x) \]
where \( \mathcal{L}_p = \int_0^\infty e^{-xp}(\cdot)dp \) is the Laplace transform. Then,
\[ \Phi^{d*}_v(x) = \mathcal{L}_p^d \Phi^d_v(p)(x) = \sum_{j=1}^{\infty} e^{-(\alpha + \beta + x)A_j - \beta + \theta) t_j - \theta} \mathcal{Z}(B_{j-1}) \mathcal{E}(1 - e^{-(x + \beta + \theta)\delta}) \]

after
\[ (x + \beta + \theta) \mathcal{L}_p^d \left( e^{-(\beta + \beta)p} \mathbf{1}_{p \in I_j^p} \right)(x) = (x + \beta + \theta) \int_{p=A_j - 1}^{A_j - 1 + \delta} e^{-(\beta + \beta)p} e^{-xp} dp \]
\[ = e^{-(x + \beta + \theta)A_j - 1} - e^{-(x + \beta + \theta)(A_j - 1 + \delta)} \]
\[ = e^{-(x + \beta + \theta)A_j - 1} \left( 1 - e^{-(x + \beta + \theta)\delta} \right) \]

Then,
\[ \Phi^{d*}_v(x) = \frac{1 - \gamma(0, x + \beta + \theta, 1)}{1 - \gamma(\alpha + \beta + x, \theta + \beta, z)} \]
so that
\[ \Phi^d_v = \mathcal{L}^{-1}_x \left( \frac{1}{x + \beta + \theta} \Phi^{d*}_v(x) \right)(T) \]

Now we turn to the second summand.
\[ \Phi^s_v(p) = \sum_{j=1}^{\infty} e^{-\alpha A_j - \beta A_j - \theta t_j - \theta} \mathbf{1}_{p \in I_j^p} \mathcal{Z}(B_{j-1}) \]

There is only one term \( \mathbf{1}_{p \in I_j^p} \) dependent on \( p \). Thus, the pertinent operator is Laplace-Carson, \( \mathcal{L}_p^s = x \int_0^\infty e^{-px}(\cdot)dp \), so
\[ \Phi^{s*}_v(x) = \mathcal{L}_p^s \Phi^s_v(p)(x) \]
\[ = \sum_{j=1}^{\infty} e^{-(\alpha + \beta + x)A_j - \beta A_j - \theta t_j - \theta} \mathcal{Z}(B_{j-1}) \mathcal{E}(X_j + \delta_j) \left( 1 - e^{-xX_j} \right) \mathcal{Y}_j \]
\[ = \sum_{j=1}^{\infty} e^{-(\alpha + \beta + x)A_j - \beta A_j - \theta t_j - \theta} \mathcal{Z}(B_{j-1}) \left[ e^{-\beta(X_j + \delta_j)} - e^{-(\beta + x)(X_j + \delta_j)} - e^{-(\beta + x)(X_j + \delta_j)} - \delta_j \mathcal{Y}_j \right] \]
The joint LST

Corollary 1. so that

\[ \tau \]

upon variants of level crossings by \( S \)

maintenance time dependent queue located in the system, along with the choice of the queue lengths

in these works, the aging process accelerated by random shocks, was the main process, while in

this paper, it plays an auxiliary role. Furthermore, in the present construction, we include the

server’s maintenance

Remark 1. Note that there are some similarities in the construction of the server’s maintenance
time process \( S(t) \) with a reliability aging process \( S(t) \) studied in our recent papers [42,68]. However,
in these works, the aging process accelerated by random shocks, was the main process, while in

this paper, it plays an auxiliary role. Furthermore, in the present construction, we include the

maintenance time dependent queue located in the system, along with the choice of the queue lengths

upon variants of level crossings by \( S(t) \). Here only the queue length \( Q(\tau_v) \), the exit time (first

passage time) \( \tau_v \), and their relationship are of interest. Yet even the associated results on \( Q(\tau_v) \) and

\( \tau_v \) are integrated in a comprehensive analysis of our system throughout the article.

As already mentioned, the random quantities \( A_{\nu-1}, S(\tau_v), \) and \( I_{\nu-1} \) are auxiliary. We

need \( \tau_v \) (the exit from vacation) and \( Q(\tau_v) \)—the number of customers waiting in the system on server’s return.

\[ \phi(\theta, z) = \Phi_v(0, 0, 0, \theta, z) = E e^{-\nu v z Q(\tau_v)} \]

Corollary 1. The joint LST \( \phi(\theta, z) \) of the system’s status upon server’s return giving the number of
customers waiting and the total vacation time is

\[ \phi(\theta, z) = E e^{-\nu v z Q_v} = 1 + L_x^{-1} \left( \frac{1}{x + \theta} \Phi_v^d(x, \theta, z) + \frac{1}{x} \Phi_v^s(x, \theta, z) \right)(T) \]

Proof. By Theorem 1,

\[ \phi(\theta, z) = \Phi_v(0, 0, 0, \theta, z) = L_x^{-1} \left( \frac{1}{x + \theta} \Phi_v^d(x, \theta, z) + \frac{1}{x} \Phi_v^s(x, \theta, z) \right)(T) \]
The two terms simplify to
\[
\Phi_v^d(x, \theta, z) = \frac{1 - \gamma(0, x + \theta, 1)}{1 - \gamma(x, \theta, z)}
\]
\[
\Phi_v^s(x, \theta, z) = \frac{\gamma(0, x + \theta, z) - \gamma(x, \theta, z)}{1 - \gamma(x, \theta, z)} = 1 - \frac{1 - \gamma(0, x + \theta, z)}{1 - \gamma(x, \theta, z)}
\]

Plugging these into the Laplace inverse and minor simplification yields the statement of this corollary.

**Corollary 2.** The marginal distributions of the number of customers \(\mathbb{E}zQ^v\) and the total vacation time \(\mathbb{E}e^{-v}\) satisfy the following formulas.

\[
\Delta_0(\theta) = \mathbb{E}e^{-\theta v} = 1 - \theta L_x^{-1}
\left(\frac{1}{x} \frac{1 - \gamma(0, x + \theta, 1)}{1 - \gamma(x, \theta, 1)}\right) (T)
\]
\[
\alpha_0(z) = \mathbb{E}zQ^v = 1 - L_x^{-1}
\left(\frac{1}{x} \frac{1 - \gamma(0, x, 1) - \gamma(0, x, z)}{1 - \gamma(x, 0, z)}\right) (T)
\]

**Proof.** With \(z = 1\),

\[
\Delta_0(\theta) = \mathbb{E}e^{-\theta v} = \phi(\theta, 1) = 1 + L_x^{-1}
\left(\frac{1}{x + \theta} \frac{1 - \gamma(0, x + \theta, 1)}{1 - \gamma(x, \theta, 1)} - \frac{1}{x} \frac{1 - \gamma(0, x, 1)}{1 - \gamma(x, \theta, 1)}\right) (T)
\]
\[
= 1 + L_x^{-1}
\left(\frac{1}{x + \theta} \frac{1 - \gamma(0, x + \theta, 1)}{1 - \gamma(x, \theta, 1)}\right) (T)
\]
\[
= 1 - \theta L_x^{-1}
\left(\frac{1}{x} \frac{1 - \gamma(0, x + \theta, 1)}{1 - \gamma(x, \theta, 1)}\right) (T),
\]
confirming the first claim of the corollary. With \(\theta = 0\),

\[
\alpha_0(z) = \mathbb{E}zQ^v = \phi(0, z) = 1 + L_x^{-1}
\left(\frac{1}{x} \frac{1 - \gamma(0, x, 1) - \gamma(0, x, z)}{1 - \gamma(x, 0, z)}\right) (T)
\]
\[
= 1 - L_x^{-1}
\left(\frac{1}{x} \frac{1 - \gamma(0, x, 1) - \gamma(0, x, z)}{1 - \gamma(x, 0, z)}\right) (T)
\]

**Corollary 3.** The system’s status upon server’s return, under the special assumption that the input is with position independent marking, that is,

\[
\gamma(a, \theta, z) = \mathbb{E}e^{-aXzYz}e^{-(a + \theta)\delta},
\]
is characterized through the following functional

\[
\phi(\theta, z) = \mathbb{E}e^{-\theta v}zQ^v
\]
\[
= 1 + L_x^{-1}
\left(\frac{1}{x + \theta} \frac{1 - \varphi(x + \theta)}{1 - a[\xi(x)] \varphi(x + \theta)} - \frac{1}{x} \frac{1 - a(z) \varphi(x + \theta)}{1 - a[\xi(x)] \varphi(x + \theta)}\right) (T).
\]

**Proof.** By double expectation,

\[
\mathbb{E}e^{-aXzYz} = \mathbb{E}[zY] \mathbb{E}
\left[e^{-a(x_1 + \ldots + x_Y)}\right] = \mathbb{E}zY^a(a) = a[\xi(a)],
\]
where $\xi(\alpha) = \mathbb{E}e^{-\alpha \xi_1}$ is the LST of the estimated service time of one customer $x_1$ and $a(\gamma) = \mathbb{E}e^\gamma$ is the PGF of the size of each batch of arriving customers. Assuming further $\varphi(\theta) = \mathbb{E}e^{-\theta \tau}$ is the LST of the interarrival times,

$$\gamma(\alpha, \theta, \nu) = \mathbb{E}e^{-\alpha \xi}e^{-\theta \nu} = a[\xi(\alpha)]\varphi(\alpha + \theta)$$

Under this condition, Corollary 1 simplifies to

$$\phi(\theta, z) = \mathbb{E}e^{-\theta \tau}z^{Q_v} = 1 + \sum_{k=0}^{\infty} \theta^{k} \frac{1}{k!} \left[ \frac{1}{x(1 + \theta)} \frac{1}{1 - a(z)} \varphi(x + \theta) \right]^k \left( \frac{1}{x(1 + \theta)} - \frac{1}{x} \right)^k (T)$$

This is a logical consequence that the jointly mean vacation time (constant $T$) along with that no customer has arrived during the whole vacation time equals $T$.

From this, the marginal transforms of $\tau_v$ and $Q_v$ follow trivially.

**Corollary 4.** Under the conditions of Corollary 3, namely, position independent marking, the following formulas for the marginal distributions hold.

$$\Delta_0(\theta) = \phi(\theta, 1) = \mathbb{E}e^{-\theta \tau_v} = 1 - \theta \sum_{k=0}^{\infty} \frac{1}{k!} \left[ \frac{1}{x(1 + \theta)} \frac{1}{1 - a(z)} \varphi(x + \theta) \right]^k \left( \frac{1}{x(1 + \theta)} - \frac{1}{x} \right)^k (T)$$

$$a_0(z) = \phi(0, z) = \mathbb{E}z^{Q_v} = 1 - (1 - a(z)) \sum_{k=0}^{\infty} \frac{1}{k!} \left[ \frac{1}{x(1 + \theta)} \frac{1}{1 - a(z)} \varphi(x + \theta) \right]^k \left( \frac{1}{x(1 + \theta)} - \frac{1}{x} \right)^k (T)$$

**Corollary 5.** Under the conditions of Corollary 3, $a_0(0) = \mathbb{P}\{\delta > T\}$, where $\delta$ is the interarrival time of the input.

**Proof.** From Corollary 4 and a property of LSTs,

$$a_0(0) = 1 - \sum_{k=0}^{\infty} \frac{1}{k!} \left[ \frac{1}{x(1 + \theta)} \frac{1}{1 - a(z)} \varphi(x + \theta) \right]^k \left( \frac{1}{x(1 + \theta)} - \frac{1}{x} \right)^k (T) = 1 - \mathbb{P}\{\delta \leq T\} = \mathbb{P}\{\delta > T\}.$$

Further, from Corollary 4, we find formulas for the means of $\tau_v$ and $Q_v$.

**Corollary 6.** Under the conditions of Corollary 3, the means of $\tau_v$ and $Q_v$ satisfy the following formulas.

$$\Delta_0 = \mathbb{E}\tau_v = (-1) \frac{\partial}{\partial \theta} \Delta_0(\theta) \bigg|_{\theta=0} = \sum_{k=0}^{\infty} \frac{1}{k!} \left[ \frac{1}{x(1 + \theta)} \frac{1}{1 - a(z)} \varphi(x + \theta) \right]^k \left( \frac{1}{x(1 + \theta)} - \frac{1}{x} \right)^k (T),$$

$$a_0 = \mathbb{E}Q_v = \frac{\partial}{\partial \nu} a_0(z) \bigg|_{z=1} = \sum_{k=0}^{\infty} \frac{1}{k!} \left[ \frac{1}{x(1 + \theta)} \frac{1}{1 - a(z)} \varphi(x + \theta) \right]^k \left( \frac{1}{x(1 + \theta)} - \frac{1}{x} \right)^k (T),$$

where $a'(1) = a = \mathbb{E}Y$ is the mean batch size.
3. Special Cases and Examples

Example 1. If the interarrival times $\delta$ of batches of customers have an Erlang distribution with parameters $\lambda$ and $r$, then their LST is $E^{-x\delta} = \varphi(x) = \left(\frac{\lambda}{\lambda + x}\right)^r$ implying that

$$1 - \varphi(x) = \frac{x}{\lambda + x} \sum_{i=0}^{r-1} \left(\frac{\lambda}{\lambda + x}\right)^i$$

and then

$$E\tau_v = L_x^{-1}\left(\sum_{i=0}^{r-1} \left(\frac{\lambda}{\lambda + x}\right)^i \frac{1}{1 - a[\xi(x)]\left(\frac{\lambda}{\lambda + x}\right)^i}\right) = \sum_{i=0}^{r-1} \lambda^i L_x^{-1}\left(\frac{1}{x(\lambda + x)^i} - \lambda a[\xi(x)]\right),$$

while

$$EQ_v = aL_x^{-1}\left(\frac{\lambda^r}{x} \frac{1}{1 - a[\xi(x)]\left(\frac{\lambda}{\lambda + x}\right)^r}\right)(T) = a\lambda L_x^{-1}\left(\frac{1}{x(\lambda + x)^r} - \lambda a[\xi(x)]\right),$$

with $a = a'\left(\frac{1}{1}\right)$, which are very different. However, when the interarrival times $\delta$ are exponential, i.e., $r = 1$, some interesting structure emerges. As we see,

$$E\tau_v = L_x^{-1}\left(\frac{1}{x(\lambda + x - a[\xi(x)]\lambda)}\right)(T)$$

while

$$EQ(\tau_v) = \lambda EY L_x^{-1}\left(\frac{1}{x(\lambda + x - a[\xi(x)]\lambda)}\right)(T) = a\lambda E\tau_v$$

In other words, the mean queue length upon the return time is simply the mean number of arriving customers per unit time multiplied by the mean return time, seems a very intuitive result to be expected with marked renewal processes. However, when one observes the process backwards beginning from the first passage time $\tau_v$, the process is no longer renewal, nor even Markov. There are discussions in our past work [1,69] somewhat similar to our process, but not that complex. This is why this result is surprising.

Now we prove an even stronger result.

Corollary 7. If and only if $\varphi(x) = \frac{\lambda}{\lambda + x}$ (input is marked Poisson) are the marginal means related as follows.

$$a_0 = a_0'(1) = EQ_v = \lambda aE\tau_v = \lambda aL_x^{-1}\left(\frac{1}{x(\lambda + x - a[\xi(x)]\lambda)}\right)(T).$$

Proof. From Corollary 6, by the comparison of $\Delta_0$ and $a_0$, we easily conclude that $a_0 = c\Delta_0$ if and only if $\frac{\varphi(x)}{x} \frac{1}{1 - a[\xi(x)]\varphi(x)} = \frac{1}{x} \frac{1}{1 - a[\xi(x)]\varphi(x)}$. The latter reduces to the equation

$$\varphi(x) = \frac{c}{x} \frac{1 - \varphi(x)}{x} \text{ or } \varphi(x) = \frac{x}{1 + cx},$$

where $c = \frac{1}{\lambda}$. The rest is a straightforward exercise. \(\square\)
Example 2. From Corollary 4 (position independent marking), now under $\varphi(x) = \frac{\lambda}{x+\xi}$ with a marked Poisson input in mind,

$$\alpha_0(z) = 1 - [1 - a(z)]L_z^{-1}\left(\frac{1}{x} \frac{\lambda}{\lambda + x - a[z\xi(x)]} \right)(T) = 1 - \lambda[1 - a(z)]L(z), \quad (10)$$

where

$$L(z) = L_z^{-1}\left(\frac{1}{x} \frac{1}{\lambda + x - \lambda a[z\xi(x)]} \right)(T). \quad (11)$$

Then, its derivative is

$$\alpha_0'(z) = \lambda a'(z)L(z) - \lambda[1 - a(z)]L'(z), \quad (12)$$

where

$$L'(z) = \lambda L_z^{-1}\left(\frac{1}{x} \frac{\xi(x)a'[z\xi(x)]}{\lambda + x - \lambda a[z\xi(x)]} \right)(T). \quad (13)$$

In particular,

$$\alpha_0 = \alpha_0'(1) = \lambda aL(1) = \lambda a L_z^{-1}\left(\frac{1}{x} \frac{1}{\lambda + x - \lambda a[z\xi(x)]} \right)(T) \quad (14)$$

The second derivative is

$$\alpha_0''(z) = \lambda a''(z)L(z) + 2\lambda a'(z)L'(z) - \lambda[1 - a(z)]L''(z)$$

or

$$\alpha_0''(z) = \lambda a''(z)L_z^{-1}\left(\frac{1}{x} \frac{\xi(x)a'[z\xi(x)]}{\lambda + x - \lambda a[z\xi(x)]} \right)(T) + 2\lambda^2 a'(z)L_z^{-1}\left(\frac{1}{x} \frac{\xi(x)a'[z\xi(x)]}{\lambda + x - \lambda a[z\xi(x)]} \right)(T) + \lambda^2[1 - a(z)]L_z^{-1}\left[\frac{1}{x} \frac{\xi(x)a'[z\xi(x)]}{\lambda + x - \lambda a[z\xi(x)]} \right]^2(T),$$

implying

$$\alpha_0'' = \alpha_0''(1) = \lambda a''L(1) + 2\lambda aL'(1)$$

$$\alpha_0'' = \lambda a''(1)L_z^{-1}\left(\frac{1}{x} \frac{\xi(x)a'[z\xi(x)]}{\lambda + x - \lambda a[z\xi(x)]} \right)(T) + 2\lambda^2 aL_z^{-1}\left(\frac{1}{x} \frac{\xi(x)a'[z\xi(x)]}{\lambda + x - \lambda a[z\xi(x)]} \right)(T) \quad (14)$$

Example 3. This example will demonstrate that the results above are analytically tractable under a special case. All results herein were found via the theorems above and symbolic computation in Wolfram Mathematica. With the results of Example 2, make the following two further assumptions.

1. The batch sizes $Y$ are geometric with parameter $p$ and PGF $a(z) = \frac{pz}{1-qz}$ where $q = 1 - p$.
2. The estimated service times $x_1$ are exponential with parameter $\xi$ and LST $\xi(x) = \frac{x}{\xi}$. 
Then, Theorem 1 implies

\[ \Phi_v(\alpha, \beta, \theta, \theta, z) = \mathcal{L}_x^{-1} \left( \frac{1}{x + \beta + \theta} \frac{1 - \gamma(0,x + \beta + \theta,1)}{1 - \gamma(\alpha + \beta + x, \theta + \theta, z)} + \frac{1}{x - 1} \frac{\gamma(\beta, x, z) - \gamma(\beta + x, \theta, z)}{1 - \gamma(\alpha + \beta + x, \theta + \theta, z)} \right)(T), \]

where Corollary 3 and the assumptions above give

\[ \gamma(\alpha, \theta, z) = a[z\xi(\alpha)] \phi(\alpha + \theta) = \frac{\lambda p^2 \phi z}{(\alpha + \lambda + \theta)(\alpha + \xi - (1 - p)\xi z)} \]

First, we will find the probability the server initiates his return due to a smooth level (M) crossing by \( S(t) \). This simply requires an inverse Laplace transform of a rational expression,

\[ \Phi_v^0(0,0,0,0,1) = \mathbb{P}\{T \in I^d\} = \mathcal{L}_x^{-1} \left( \frac{1}{x} \frac{1 - \gamma(0,x,1)}{1 - \gamma(x,0,1)} \right)(T) = \frac{\lambda - \lambda e^{-bT}}{\lambda + p^2} \] (15)

and, therefore, the probability the dispatcher initiates the server’s return due to \( S(t) \) being crossed by a jump, i.e., an arriving batch whose estimated service time exceeds the remaining vacation time of the server, is

\[ \mathbb{P}\{T \in I^s\} = 1 - \frac{\lambda}{\lambda + p^2} (1 - e^{-bT}) = \frac{p^2 + \lambda e^{-bT}}{\lambda + p^2} \] (16)

Next, let us pursue the mean and variance of the return time \( \tau_v \). Here, we find

\[ \Delta_0 = \mathbb{E}\tau_v = -\lim_{\theta \to 0} \frac{\partial}{\partial \theta} \mathcal{L}_x^{-1} \left( \frac{1}{x + \theta} \frac{1 - \gamma(0,x + \beta + \theta,1)}{1 - \gamma(x,\theta,1)} + \frac{1}{x} \frac{\gamma(0,\theta + x,1) - \gamma(x,\theta,1)}{1 - \gamma(x,\theta,1)} \right)(T) \]

\[ = \frac{\lambda}{b^2} \left(1 - e^{-bT}\right) + \frac{p^2}{bT} \] (17)

The second moment is

\[ \mathbb{E}\tau_v^2 = -\lim_{\theta \to 0} \frac{\partial^2}{\partial \theta^2} \mathcal{L}_x^{-1} \left( \frac{1}{x + \theta} \frac{1 - \gamma(0,x + \beta + \theta,1)}{1 - \gamma(x,\theta,1)} + \frac{1}{x} \frac{\gamma(0,\theta + x,1) - \gamma(x,\theta,1)}{1 - \gamma(x,\theta,1)} \right)(T) \]

This formula was computed with software explicitly, but in the interest of space, we present only the formula for variance derived as \( \mathbb{E}\tau_v^2 - (\mathbb{E}\tau_v)^2 \) as follows.

\[ Var(\tau_v) = \frac{\lambda}{b^4} \left[ \lambda - 4p^2 + 2p\xi bT + \left(4p^2 + 2p^2 + 2\lambda^2 T - 2\lambda^2 T\right) e^{-bT} - \lambda e^{-2bT} \right] \]

By Example 1, under the Poisson input we have

\[ \alpha_0 = \mathbb{E}(\tau_v) = a\lambda \mathbb{E}\tau_v = \frac{\lambda}{p} \mathbb{E}\tau_v = \frac{\lambda^2}{p b^2} \left(1 - e^{-bT}\right) + \frac{\lambda \xi}{b} \]

(18)

Plotting \( \alpha_0 = \alpha_0(T) \) with \( \lambda = 1 \) and \( x = p = 0.1 \), we have in Figure 6 the mean queue length grown during server’s maintenance.
Figure 6. Mean number of customers amalgamated during server’s maintenance.

As expected, $\alpha_0$ is monotone increasing in $T$.

In Figure 7, we plot the mean maintenance time as a function of $T$ under the same assumptions as those for $\alpha_0$.

Figure 7. Mean maintenance time $\Delta_0 = E(\tau_\nu)$ for $0 \leq T \leq 30$.

As we see, the mean maintenance time is largely shortened (at $T = 30$ down to only 1.3) by incoming customers due to their estimated service times.

Next, we are interested to see the dependence of $\Delta_0$ from $\xi$ (the reciprocal of the mean estimated service time) rather than from $T$. For this reason, we rewrite Formula (17) in the form of

$$\Delta_0 = pT \frac{1}{\lambda c + p} + \frac{\lambda c^2}{(\lambda c + p)^2} \left(1 - e^{-(\lambda + p^T)T}\right),$$

where $c = \frac{1}{\xi}$ is the mean estimated service time of one customer. Figure 8 below depicts $\Delta_0(c)$ under $\lambda = 1$ (same as the mean interarrival), $T = 30$ (fixed), and $p = 0.1$ (mean arriving batch size of 10).
As we said, the mean maintenance time $\Delta_0$ in Figure 8 is now a function of $c = \frac{1}{\xi}$. We see that the $\Delta_0$ equals 30 when $c = 0$. Then $\Delta_0(c)$ sharply falls with $c$ increasing from 0 to 10 and then $\Delta_0$ slows down from $c = 2$ ($\Delta_0(2) \approx 2.33$), equals 1.278 at $c = 10$, then asymptotically approaches 1.

In Figure 9, below, we took $\lambda = 0.1$ (mean interarrival time of 10), $p = 0.1$ (mean batch size of 10), and $T = 30$.

This time, the behavior of $\Delta_0$ is smoother compared to Figure 8, and at $c = 30$ (same value as the originally planned maintenance time), $\Delta_0(30) \approx 9.9$, close to the mean arrival time of customers $\frac{1}{\lambda} = 10$. To interpret this result, let us look at $\Delta_0(c)$ again, now in the form of

$$
\Delta_0(c) = \frac{1}{\lambda} \left[ \frac{pT}{c + p/\lambda} + \frac{1}{(1 + p/c\lambda)^2} \left( 1 - e^{-\left(\frac{\lambda + p/c}{\lambda}\right) T} \right) \right]
$$

Here we see that $\Delta_0(c) \to \frac{1}{\lambda} \left[ 1 - e^{-\lambda T} \right]$ (still $< \frac{1}{\lambda}$) asymptotically in $c$ and it is smaller than $\frac{1}{\lambda}$. However, as other “experiments” show, $\Delta_0(c)$ may be larger than $\frac{1}{\lambda}$ even for fairly large $c$. This is because $\Delta_0$ can be impacted by the first summand containing $T$ and it offsets factor $1 - e^{-\lambda T}$. 
For example, in Figure 10, we see a very sharp decline of $\Delta_0$ just around 0, with $T = 30$ and $\lambda = 10$. Now at $c = 100$ (large), $\Delta_0(100) \approx 0.103 > \frac{1}{\lambda}$. It takes a very large $c$ to have $\Delta_0(c) = \frac{1}{\lambda} = 0.1$ and it is not smaller than 0.1. This is because $T$ is relatively large and its impact is still seen for very large $c$’s. It takes $T = 0.5$ and $c = 1000$ to see $\Delta_0(c)$ drop below 0.1, such as $\Delta_0(1000) = 0.09932925$ or even further, with $T = 0.1$ to get $\Delta_0(1000) = 0.06321216$.

![Figure 10. Mean maintenance time as a function of mean estimated service time $c = \frac{1}{p}$ with $\lambda = 10$ and $T = 30$.](image)

For the variance, we exploit the property of the PGF $a_0(z)$ of $Q(\tau_\nu)$ where

$$
\Sigma_0 = \text{Var}(Q(\tau_\nu)) = a_0''(1-) + a_0'(1-) - \left[a_0'(1-)^2\right] = a_0''(1-) + EQ(\tau_\nu) - (EQ(\tau_\nu))^2 = a_0'' + a_0 - a_0^2
$$

(19)

Conversely,

$$
a_0'' = a_0''(1-) = \Sigma_0 + a_0^2 - a_0.
$$

(20)

Computing the second derivative as $z$ approaches 1 from inside the unit ball, and combining with the $a_0$ terms, we find

$$
\Sigma_0 = \frac{\lambda}{p^2b^4}\left[c_0 + c_1e^{-bT} + c_2e^{-2bT}\right] = \frac{\lambda}{p^2b^4}\left[c_{01} + c_{02}T + c_{11}e^{-bT} + c_{12}Te^{-bT} + c_{2}e^{-2bT}\right],
$$

(21)

where

$$
c_0 = \lambda^3q + 4\lambda^2p\xi^2 - 6\lambda^2p^2\xi + 8\lambda p^2\xi^2 - 5\lambda p^3\xi^2 + (\lambda^3 p^2\xi + \lambda^2 p^3\xi^2 + 2\lambda p^3\xi^3 - \lambda p^4\xi^3 + 2p^4\xi^4 - p^5\xi^4)T = c_{01} + c_{02}T
$$

(22)

$$
c_1 = \lambda^3p - 4\lambda^2p\xi^2 + 6\lambda^2p^2\xi - 8\lambda p^2\xi^2 + 5\lambda p^3\xi^2 + (2\lambda^3 p\xi^2 + 2\lambda^3 p^2\xi + 4\lambda^2 p^3\xi^2 - 2\lambda p^3\xi^3 q)T = c_{11} + c_{12}T
$$

(23)

$$
c_2 = -\lambda^3
$$

(24)

In Figure 11, we plot $\Sigma_0(T)$ of Formula (21) with $\lambda = 1$ and $\xi = p = 0.1$. 
Figure 11. \( \Sigma_0(T) \) with \( \lambda = 1 \) and \( \xi = p = 0.1 \).

Figure 12 depicts \( a''_0 \) of Formulas (18)–(24) with \( \lambda = 1 \) and \( \xi = p = 0.1 \).

Figure 12. \( a''_0 \) with \( \lambda = 1 \) and \( \xi = p = 0.1 \).

Below in Figure 13 is another variant of \( a''_0 \) with \( \lambda = 2, \xi = 3, \) and \( p = 0.1 \).

Figure 13. \( a''_0 \) with \( \lambda = 2, \xi = 3, \) and \( p = 0.1 \).
4. Numerical Examples and Simulation

Next, Formulas (16)–(20) for probabilities, means, and variances derived for the case of Poisson arrivals, geometric batch sizes, and exponential estimated service times will be shown to agree with Monte Carlo simulations of the process under various numerical assumptions on the parameters: the arrival rate \( \lambda \), the parameter of the batch size \( p \), and the parameter of the estimated service time for each customer \( \xi \), and the vacation time \( T \). In each case, all 5 formulas are validated for a variety of parameter values, each based on 100,000 simulations of vacations of the server.

Code for simulating the vacations of the present queueing system, implementations of Formulas (16)–(20), and numerous experiments generating all diagrams in this section is available publicly at the GitHub repository linked at the end of the article.

Figures 14–16 show the probability of the server initiating his own return, i.e., event \( I^d \) where a smooth level crossing occurs, from (16) under various sets of parameters, each for several \( T \) values.

**Figure 14.** The probability of a server-initiated return compared to the arrival rate \( \lambda \).

**Figure 15.** The probability of a server-initiated return compared to the batch size parameter \( p \).
Figure 16. The probability of a server-initiated return compared to the estimated service time parameter $\xi$.

Figure 14 has predicted and empirical probabilities for $p = 0.5, \xi = 5, \lambda \in \{0.5, 1, \ldots, 10\}$, and $T \in \{0.1, 0.25, 0.5, 1\}$. It shows that, as $\lambda$ grows, i.e., arrivals occur more frequently, probabilities that the server returns on his own shrink because surges are more frequent.

Figure 15 has predicted and empirical probabilities for $\lambda = 1, \xi = 5, p \in \{0.05, 0.1, \ldots, 1\}$, and $T \in \{0.1, 0.5, 1, 2\}$. It shows that, as $p$ grows, i.e., the batch sizes tend to be smaller, so the probabilities that the server returns on his own grow.

Figure 16 has predicted and empirical probabilities for $\lambda = 1, \xi \in \{0.5, 1, \ldots, 10\}$, $p = 0.5$, and $T \in \{0.1, 0.5, 1, 2\}$. It shows that, as $\xi$ grows, i.e., the estimated service times tend to be smaller, so the probabilities that the server returns on his own grow.

In all cases, a larger $T$ implies higher probabilities since, with all else equal, this means threshold is further away and, hence the surges are less likely to cause the dispatcher to call the server back.

Figures 17–19 show the empirical means and variances of the return time compared with graphs of Formulas (18) and (19) under various sets of parameters, each for several $T$ values.

Figure 17. Mean and variance of return time compared to the arrival rate $\lambda$. 
Figure 18. Mean and variance of return time compared to the batch size parameter $p$.

Figure 18 includes values for $\lambda \in \{0.5, 1, \ldots, 10\}$, $p = 0.5$, $\xi = 5$, and $T \in \{0.1, 0.25, 0.5, 1\}$. The mean return time of course shrinks as $\lambda$ grows and waiting times become longer. Variances seem to peak and then decrease as $\lambda$ grows.

Figure 19. Mean and variance of return time compared to the estimated service time parameter $\xi$.

Figure 19 includes values for $\lambda = 1$, $\xi = 5$, $p \in \{0.05, 0.1, \ldots, 1\}$, and $T \in \{0.1, 0.5, 1, 2\}$. The mean return time of course grows as $p$ grows and batch sizes become smaller. Variance shrinks as $p$ grows because the variance in the batch sizes shrinks.

The mean return time of course grows as $\xi$ grows and service times shrink, and the dispatcher subtracts less from the server’s vacation. Shorter service times also cause the return time more predictable and have less variance.

Figures 20–22 show the empirical means and variances of the return queue length compared with graphs of Formulas (20) and (21) under various sets of parameters, each for several $T$ values. The property $EQ_0 = \frac{1}{T}E\tau_0$ is clearly seen as the plots show the same trend as the means in Figures 17–19, just with different scaling.

Figure 20 includes values for $p = 0.5$, $\xi = 5$, $\lambda \in \{0.5, 1, \ldots, 10\}$, and $T \in \{0.1, 0.25, 0.5, 1\}$. The variances grows as $\lambda$ grows since shorter waiting times cause higher variance in the arrivals per unit time, and hence the return queue length.

Figure 21 includes values for $\lambda = 1$, $p = 0.5$, $\xi \in \{0.5, 1, \ldots, 10\}$, and $T \in \{0.1, 0.5, 1, 2\}$. Larger $p$ reduces the variance precipitously as smaller batches make for lower variance in queue length upon the server’s return.

For small $T$, $\xi$ has little effect on the variance of the return queue length, presumably because the server will quickly return from his vacation regardless of customer activity. However, larger $\xi$ and the resulting smaller service times increase variance for larger $T$ because, with dispatcher-initiated returns more likely, these cases will typically involve unpredictable large batches and frequent arrivals to overcome the short service times.
Figure 20. Mean and variance of return queue length compared to the arrival $\lambda$.

Figure 21. Mean and variance of return queue length compared to the batch size parameter $p$.

Figure 22. Mean and variance of return queue length compared to the service time parameter $\xi$.

5. System Policy on Server Return

When the server returns to the system, he need not expect customers waiting, even a single one. For example, if $t_1 > T$, this will be exactly the case. We go one step further. Suppose the server does not resume his service unless there are at least $N$ customers waiting. Otherwise, the server waits for the buffer to be replenished accordingly. It corresponds to the $N$-Policy queue or rather single-vacation-$N$-Policy combined (NSV-Policy for short).

Notation 1. We use the original formula due to Dshalalow [2]. See also Dshalalow and White [4,41]. Under the following notation,

(a) $\phi(\theta, z) = \Phi_\nu(0, 0, 0, \theta, z)$ is the distribution of waiting room load on server’s return along with the total vacation time

(b) $W_n = Q(\tau_\nu) + Y_1 + \ldots + Y_n$ (we can regard $Q(\tau_\nu)$ as $Y_0$) - $W_n$ is the active component

(c) $\Delta_n = \tau_\nu + t_1 + \ldots + t_n$ (we can regard $\tau_\nu$ as $I_0$)
(d) \( \mu = \inf \{ n : W_n \geq N \} \).

Then, if the input is marked Poisson with position independent marking, meaning the joint transform of the inter-arrival time and the arriving batch size is \( \mathbb{E} e^{-\theta z^y} = a(z) \frac{\lambda}{\lambda + \theta} \), then from Dshalalow [2],

\[
\mathcal{F}_\mu(\theta, z) = \mathbb{E} z^W e^{-\theta \Delta_\mu} \\
= D^{N-1}_y \left[ \phi(\theta, z) - \phi(\theta, yz) + \frac{\phi(\theta, yz)}{1 - a(yz) \varphi(\theta)} \varphi(\theta \{ a(z) - a(yz) \} \right] \tag{25}
\]

With the input general renewal, still with position independent marking, \( \gamma(\theta, z) = \mathbb{E} e^{-\theta z^y} = a(z) \varphi(\theta) \) where \( \varphi(\theta) = \mathbb{E} e^{-\theta z} \), we have from Dshalalow [2],

\[
\mathcal{F}_\mu(\theta, z) = \mathbb{E} z^W e^{-\theta \Delta_\mu} \\
= D^{N-1}_y \left[ \phi(\theta, z) - \phi(\theta, yz) + \frac{\phi(\theta, yz)}{1 - a(yz) \varphi(\theta)} \varphi(\theta \{ a(z) - a(yz) \} \right] \\
= \phi(\theta, z) - [1 - a(z) \varphi(\theta)] D^{N-1}_y \left[ \frac{\phi(\theta, yz)}{1 - a(yz) \varphi(\theta)} \right]. \tag{26}
\]

**Theorem 2.** In the queuing system with NSV-Policy and a marked renewal process with position independent marking as input, with the LST of the interarrival time \( \varphi(\theta) \) and mean \( \varphi \), the mean length of the service cycle \( \Delta = \Delta_\mu \) that starts with zero customers, followed by state dependent single vacation time and wait specified by the N-Policy and the mean number of customers \( \alpha = \mathbb{E} W_\mu \) accumulated in the system upon service resumption are related as

\[
\alpha = \frac{1}{\varphi} a \Delta \tag{27}
\]

if and only if the input to the system is marked Poisson.

**Proof.** From (26), we get the marginals

\[
\Delta(\theta) = \mathbb{E} e^{-\theta \Delta_\mu} = \mathcal{F}_\mu(\theta, 1) = \Delta_\mu(\theta) - [1 - \varphi(\theta)] D^{N-1}_y \left[ \frac{\phi(\theta, y)}{1 - \varphi(\theta) a(y)} \right]
\]

\[
a(z) = \mathbb{E} z^W = \mathcal{F}_\mu(0, z) = a_0(z) - [1 - a(z)] D^{N-1}_y \left[ \frac{a_0(y)}{1 - a(y)} \right]
\]

Then,

\[
\Delta = \mathbb{E} \Delta_\mu = \Delta_0 + \varphi D^{N-1}_y \left[ \frac{a_0(y)}{1 - a(y)} \right]
\]

\[
\alpha = \mathbb{E} W_\mu = a_0 + a D^{N-1}_y \left[ \frac{a_0(y)}{1 - a(y)} \right]
\]

From Corollary 7, \( a_0 = \lambda a \Delta_0 \) if and only if the input to the system is Poisson, in particular, \( \varphi = \frac{1}{\lambda} \) implying that

\[
\alpha = \lambda a \Delta = \lambda a \Delta_0 + \lambda a \frac{1}{\lambda} D^{N-1}_y \left[ \frac{a_0(y)}{1 - a(y)} \right] = a_0 + \lambda a D^{N-1}_y \left[ \frac{a_0(y)}{1 - a(y)} \right] \tag{28}
\]

\( \square \)

**Remark 2.** From Corollary 7, \( \Delta_0 = \mathbb{E} \tau_0 = \zeta^{-1}(\frac{1}{e^\lambda - 1 + \frac{1}{\lambda a_0(x)}}) (T) \) which is a part of Formula (28) and it includes the specification \( a(z) \) on the marks \( Y \)'s and \( \zeta(\theta) \) on the marks \( X \)'s.
Corollary 8. The queueing system with SV-Policy under the conditions of Theorem 2, with marked Poisson input and no N-Policy \((N = 1)\), has the following characteristics of the first service cycle starting with \(Q_0 = 0\),

\[
E\Delta_\mu = \Delta = \Delta_0 + \frac{1}{\lambda}e^{-\lambda T} \quad \text{and} \quad EW_\mu = \alpha = a_0 + ae^{-\lambda T}.
\]

**Proof.** From Corollary 5, \(a_0(0) = \Phi(0) = P\{t_1 > T\} = P\{\delta > T\} = 1 - F_\delta(T) = e^{-\lambda T}\). So, with \(N = 1\),

\[
\Delta = E\Delta_\mu = \Delta_0 + \frac{1}{\lambda}e^{-\lambda T} \\
\alpha = \lambda a\Delta = a_0 + ae^{-\lambda T}
\]

\(\blacksquare\)

**Example 4.** Under the assumptions of Example 3 about \(a(z) = \frac{pz_2}{1 - pq}, \xi(\theta) = \frac{r}{s + \sigma}, \text{and} \ N = 1\)

\[
\Delta = E\Delta_\mu = \frac{p\xi}{\lambda + p\xi}T + \frac{1}{\lambda} \left( \frac{\lambda}{\lambda + p \xi} \right)^2 \left( 1 - e^{-(\lambda + p \xi)T} \right) + \frac{1}{\lambda}e^{-\lambda T} \tag{30}
\]

\[
\alpha = EW_\mu = \frac{\lambda \xi}{\lambda + p \xi}T + \frac{1}{\lambda} \left( \frac{\lambda}{\lambda + p \xi} \right)^2 \left( 1 - e^{-(\lambda + p \xi)T} \right) + \frac{1}{\lambda}e^{-\lambda T} \tag{31}
\]

If \(T\) is random with LST \(\eta(\theta) = Ee^{-\theta T}\),

\[
\Delta = \frac{p\xi}{\lambda + p\xi}ET + \frac{1}{\lambda} \left( \frac{\lambda}{\lambda + p \xi} \right)^2 \left( 1 - \eta(\lambda + p \xi) \right) + \frac{1}{\lambda} \eta(\lambda) \tag{32}
\]

\[
\alpha = \frac{\lambda \xi}{\lambda + p \xi}ET + \frac{1}{\lambda} \left( \frac{\lambda}{\lambda + p \xi} \right)^2 \left( 1 - \eta(\lambda + p \xi) \right) + \frac{1}{\lambda} \eta(\lambda) \tag{33}
\]

6. Queueing

6.1. Service Cycle

Recall that when the server returns to the system and he finds the waiting room (buffer) with less than \(N\) customers wafting, he rests until there are at least a total of \(N\) customers available and then immediately resumes his service, which he also does if on his return, the buffer has at least \(N\) waiting customers. Thus, a new service begins and so does a new busy period. It continues with another service of a customer and so on. When the queue becomes exhausted, an underlying busy period ends. Furthermore, it prompts the server to leave the system and go on one, state dependent, vacation trip, and then to return with a possible wait and rest. This ends one of his service cycles.

A service cycle is thus a random period of time that includes a service time from its beginning to the end unless the queue gets empty. In the latter case, the server leaves on vacation, then returns to the system, and possibly rests for a while. When at least one customer becomes available, the server ends the cycle and resumes his service. Thus, a service cycle includes a mere service time, or a service time and vacation and wait altogether. If the system is under the N-Policy, then service is resumed only after at least \(N\) customers are amalgamated.

Speaking of a service cycle, it makes sense to distinguish one, regular, that includes only service of one customer and a long one that contains vacation and a possible rest. Now if the system is being observed beginning with \(T_0 = 0\), the time \(T_1\) denotes the end of the first service cycle, whether regular or long. Then, \(\{T_n : n = 0, 1, \ldots\}\) is the sequence of the ends of successive service cycles \(\{(T_n, T_{n+1}]\}\), also known as departure epochs of serviced customers.
The end of a long service cycle is followed by the beginning of a new busy period that begins with a regular service cycle. A new busy period begins with a number of customers specified by \( F_\mu(0, z) = E z^{\beta \theta} \) in one of the forms discussed in Section 5, in notation \( a(z) \).

6.2. M/G/1-Type Queue

If the queue in mind is of an M/G/1-type, then the queueing process \( Q(t) \) is with right-continuous paths and defined on a filtered space \( (\Omega, \mathcal{F}_t, \{\mathcal{F}_i\}, \mathbb{P}) \). It is semi-regenerative with respect to the sequence \( \{T_n\} \) of the ends of service cycles (stopping times relative to \( \{\mathcal{F}_i\} \)), so that \( Q_n = Q(T_n) \) forms a homogeneous Markov chain with the TPM (transition probability matrix) \( P = (p_{ij} : i, j \in \mathbb{N}_0) \). The TPM \( P \) is a \( \Delta_2 \)-matrix defined so in Abolnikov-Dukhovny [35] as it reads

\[
P = \begin{pmatrix}
p_{00} & p_{01} & p_{02} & p_{03} & \cdots \\
p_0 & q_1 & q_2 & q_3 & \cdots \\
0 & q_0 & q_1 & q_2 & \cdots \\
0 & 0 & q_0 & q_1 & \cdots \\
& & & & \ddots
\end{pmatrix}
\]

(34)

Abolnikov-Dukhovny also provided a necessary and sufficient ergodicity condition for \( \{Q_t\} \) that we mention in a moment. Given the ergodicity condition is met, the invariant formula (35) for

\[
\begin{align*}
\beta(\theta) &= E e^{-\theta \sigma}, \\
p_0 &= \frac{1 - \rho}{a}, \\
\rho &= \lambda b_1, \\
a &= E W_\mu.
\end{align*}
\]

\( \rho < 1 \) is the sufficient and necessary condition for \( p \) to exist (established in [35]) and for Formula (35) for \( P(z) \) to hold. The formula for \( P(z) \) is similar to that for the M\( ^X \)/G/1/\( \infty \) system

\[
P(z) = \frac{1 - \rho}{a} \beta(\lambda - \lambda a(z))[a(z) - 1]/[z - \beta(\lambda - \lambda a(z))],
\]

(36)

where in (36), \( a(z) \) is replaced with \( a(z) \) and \( a \) is replaced with \( a = E W_\mu \) as per (35).

6.3. Mean Queue Length

Let \( B(z) = \beta(\lambda - \lambda a(z)) \). Then, \( \frac{d}{dz} B(z) = -\lambda a'(z) \beta'(\theta) \big|_{\theta=\lambda-\lambda a(z)} \) and

\[
\begin{align*}
\lim_{z \to 1^-} \frac{d}{dz} B(z) &= \rho = \lambda b_1, \\
\lim_{z \to 1^-} \frac{d^2}{dz^2} B(z) &=-\lambda \left[ a''(z)\beta'(\theta) \big|_{\theta=\lambda-\lambda a(z)} + a'(z)\beta''(\theta) \big|_{\theta=\lambda-\lambda a(z)} (-\lambda a'(z)) \right], \\
\lim_{z \to 1^-} \frac{d^2}{dz^2} B(z) &= \lambda b_2''(1) + (\lambda a')^2 b_2, \text{ where } b_2 = E \sigma^2.
\end{align*}
\]

(37)
If \( A(z) = \frac{a(z) - 1}{z - \beta(\lambda - \lambda a(z))} \), then using (37) and (38),
\[
\lim_{z \to 1^-} A(z) = \lim_{z \to 1^-} \frac{a'(z)}{1 - \beta'(\lambda - \lambda a(z))(-\lambda a'(z))} = \frac{a}{1 - \rho} = \frac{1}{p_0}.
\]

Next,
\[
A'(z) = \frac{a'(z)[z - B(z)] - [a(z) - 1][1 - B'(z)]}{(z - B(z))^2}
\]
and thus,
\[
\lim_{z \to 1^-} A'(z) = \frac{a''(1)}{2(1 - \rho)} + \frac{\lambda b_1 a''(1) + (\lambda a)^2 b_2}{2(1 - \rho)^2}.
\]

Now, from \( P'(1) = p_0[B'(1)A(1) + B(1)A'(1)] \) and (39) and (40), we finally have
\[
P'(1^-) = \mathbb{E}Q_{\infty} = \rho + \frac{a''(1)}{2a} + \frac{\lambda b_1 a''(1) + (\lambda a)^2 b_2}{2(1 - \rho)}
\]
or from (19) and (20) in the form of
\[
P'(1^-) = \rho + \frac{1}{2} \left( \frac{\text{Var}Q(\tau_1)}{\rho} + a_0 - 1 \right) + \frac{\lambda b_1 a''(1) + (\lambda a)^2 b_2}{2(1 - \rho)}
\]

**Remark 3.** From Theorem 2,
\[
a'(z) = a'_0(z) + a'(z)D_y^{N-1} \left\{ \frac{a_0(yz)}{1 - a(yz)} \right\} - [1 - a(z)]D_y^{N-1} \left\{ \frac{\partial}{\partial z} \frac{a_0(yz)}{1 - a(yz)} \right\}
\]
implying that
\[
a'(z) = a'_0(z) + a'(z)D_y^{N-1} \left\{ \frac{a_0(yz)}{1 - a(yz)} \right\} - [1 - a(z)]D_y^{N-1} \left\{ \frac{\partial}{\partial z} \frac{a_0(yz)}{1 - a(yz)} \right\}
\]
and
\[
a'(1) = a'_0(1) + aD_y^{N-1} \left\{ \frac{a_0(y)}{1 - a(y)} \right\}
\]
\[
a''(z) = a''_0(z) + a''(z)D_y^{N-1} \left\{ \frac{a_0(yz)}{1 - a(yz)} \right\} + 2a'(z)D_y^{N-1} \left\{ \frac{\partial}{\partial z} \frac{a_0(yz)}{1 - a(yz)} \right\} - [1 - a(z)]D_y^{N-1} \left\{ \frac{\partial}{\partial z} \frac{a_0(yz)}{1 - a(yz)} \right\}
\]
\[
a''(1) = a''_0(1) + a''(1)D_y^{N-1} \left\{ \frac{a_0(y)}{1 - a(y)} \right\} + 2aD_y^{N-1} \left\{ \frac{\partial}{\partial z} \frac{a_0(yz)}{1 - a(yz)} \right\}_{z=1}
\]
\[
= a''_0(1) + a''(1)D_y^{N-1} \left\{ \frac{a_0(y)}{1 - a(y)} \right\} + 2aD_y^{N-1} \left\{ \frac{a_0'(y)[1 - a(y)] + a_0(y)a'(y)}{[1 - a(y)]^2} \right\}
\]
\[
= a''_0(1) + a''(1)D_y^{N-1} \left\{ \frac{a_0(y)}{1 - a(y)} \right\} + 2aD_y^{N-2} \left\{ \frac{a_0'(y)[1 - a(y)] + a_0(y)a'(y)}{[1 - a(y)]^2} \right\}
\]

**Remark 4.** From (42), denote \( M_N(x, z) = D_y^{N-1} \left\{ \frac{a_0(yz)}{1 - a(yz)} \right\} \) and further,
\[
a_0(y) = 1 - \lambda [1 - a(y)]L(y),
\]
where under the assumption that \( a(y) = \frac{pq}{1-qy} \) and \( \zeta(x) = \frac{\zeta}{\zeta + x} \),

\[
L(y) = L_x^{-1} \left( \frac{1}{x} \frac{x + p\zeta}{x^2 + bx + \lambda \zeta p\zeta (1-y)} \right)(T), \quad b = \lambda + p\zeta
\]  

(see Formulas (10) and (11) under no assumptions on \( a(y) \) and \( \zeta(x) \).) Then,

\[
\frac{a_0(y)}{1 - a(y)} = \frac{1}{1 - \lambda L(y)} = \frac{1 - qy}{1 - y} - \lambda L_x^{-1} \left( \frac{x + p\zeta}{x} \frac{1}{x^2 + bx + \lambda \zeta p\zeta 1 - \frac{1 - \lambda p\zeta y}{x^2 + bx + \lambda \zeta p\zeta y}} \right)(T).
\]

Assuming \( b(x) = \frac{\lambda p\zeta}{x^2 + bx + \lambda \zeta p\zeta} \), this implies that

\[
M_N(x, 1) = D_y^{N-1} - \frac{1}{1-y} - qD_y^{N-2} \frac{1}{1-y} - \lambda L_x^{-1} \left( \frac{x + p\zeta}{x} \frac{1}{x^2 + bx + \lambda \zeta p\zeta} \frac{1 - b^N(x)}{1 - b(x)} \right)(T).
\]

\[
= N - q(N - 1) - \lambda L_x^{-1} \left( \frac{x + p\zeta}{x} \frac{1 - b^N(x)}{x + b} \right)(T).
\]

\[
= Np + q - \lambda L_x^{-1} \left( \frac{x + p\zeta}{x^2} \frac{1}{x + b} \right)(T).
\]

\[
+ \lambda(\lambda p\zeta)^N L_x^{-1} \left( \frac{x + p\zeta}{x^2} \frac{1}{x + b} \frac{x^2 + bx + \lambda \zeta p\zeta} \right)(T) \quad (49)
\]

From (44),

\[
a'(1) = a_0'(1) + aD_y^{N-1} \left\{ \frac{a_0(y)}{1 - a(y)} \right\} = a_0'(1) + \frac{1}{p} M_N(x, 1) \quad (50)
\]

and from (46),

\[
a''(1) = a_0''(1) + \frac{2q}{p} M_N(x, 1) + \frac{2}{p} D_y^{N-2} \left\{ \frac{a_0(y) [1 - a(y)] + a_0(0)a'(y)}{[1 - a(y)]^2} \right\} \quad (51)
\]

For \( N = 2 \),

\[
a''(1) = a_0''(1) + \frac{2q}{p} M_2(x, 1) + \frac{2}{p} (a_0(0) + a_0(0)a'(0)) \quad (52)
\]

In terms of our previous abbreviation \( b = \lambda + p\zeta \), \( M_2(x, 1) \) turns to
\[ M_2(x, 1) \]
\[ = 2p + q - \lambda L_x^{-1} \left( \frac{x + p\xi}{x^2} \frac{1}{x + b} \right)(T) \]
\[ + \lambda(\lambda p\xi)^2 L_x^{-1} \left( \frac{x + p\xi}{x^2} \frac{1}{x + b} \frac{1}{\lambda^2 + bx + \lambda p\xi} \right)(T) \]
\[ = 2p + q - \frac{\lambda \left[ \lambda \left( 1 - e^{-bT} \right) + bp\xi T \right]}{b^2} + \lambda(\lambda p\xi)^2 \]
\[ \times \left[ \frac{p\xi T}{bx_1^2 x_2^2} - \frac{\lambda e^{-bT}}{b^2(b + x_1)^2(b + x_2)^2} + \frac{(x_1 + p\xi)Te_{x_1 T}}{x_1^2(b + x_1)(x_1 - x_2)^2} \right. \]
\[ + \frac{\lambda x_1 x_2 + 2bp\xi(x_1 + x_2)}{b^2(x_1 x_2)^2} + \frac{(x_2 + p\xi)Te_{x_2 T}}{x_2^2(b + x_2)(x_1 - x_2)^2} \]
\[ + \left. \frac{(-3bx_1^2 - 4x_1^2 + b x_1 x_2 + 2x_1^2 x_2 - 4bp\xi x_1 - 5p\xi x_1^2 + 2bp\xi x_1 + 3p\xi x_1 x_2) e_{x_1 T}}{x_1^2(b + x_1)^2(x_1 - x_2)^2} \right] \]
\[ + \frac{(-3bx_2^2 - 4x_2^2 + b x_1 x_2 + 2x_1 x_2^2 - 4bp\xi x_2 - 5p\xi x_2^2 + 2bp\xi x_2 + 3p\xi x_1 x_2) e_{x_2 T}}{x_2^2(b + x_1)^2(x_2 - x_1)^2} \] \[ \right], \quad (53) \]

where

\[ x_1 = \frac{-b - \sqrt{b^2 - 4\lambda p\xi}}{2} \quad \text{and} \quad x_2 = \frac{-b + \sqrt{b^2 - 4\lambda p\xi}}{2}, \quad (54) \]

assuming \( b^2 - 4\lambda p\xi > 0 \), demonstrating the formula can be reduced to simple algebraic expressions under \( N \) policy. Note that we can proceed in the same manner when the roots are complex numbers. We avoid cases with multiple roots though, considering them probabilistically impossible and thus impractical in a real-world situation.

**Example 5.** From (41), under the assumptions in Example 3, that is, \( a(z) = \frac{p^2}{1 - qz} \) and \( a''(1) = \frac{2q}{p} \) we have

\[ \mathbb{E}Q_{\infty} = P'(1-) = \rho + \frac{a''(1)}{2\alpha} + \left[ \lambda b_1 \frac{2q}{p^2} + \left( \frac{\lambda}{p} \right)^2 \right] \frac{1}{2(1 - \rho)}. \quad (55) \]

In particular, for \( N = 1 \),

\[ \mathbb{D} \left\{ \frac{a_0(yz)}{1 - a(yz)} \right\} = a_0(0) = 1 - P \{ \delta > T \} = e^{-\lambda T} \quad (56) \]

(see Corollary 5) Then,

\[ a(z) = a_0(z) - [1 - a(z)]e^{-\lambda T} \quad (57) \]
\[ a'(z) = a'_0(z) + a'(z)e^{-\lambda T} \Rightarrow a = a_0 + ae^{-\lambda T} = a_0 + \frac{1}{p} e^{-\lambda T} \quad (58) \]

and

\[ a''(z) = a''_0(z) + a''(z)e^{-\lambda T} \Rightarrow a''(1) = a''_0(1) + \frac{2q}{p^2} e^{-\lambda T} \quad (59) \]
Then, from (41) and (58) and (59),
\[ E_Q = \rho + \frac{a''_0 + e^{-\lambda T} \frac{2q}{p^2}}{2 \left( a_0 + \frac{1}{p} e^{-\lambda T} \right)} + \left[ \frac{2 b_1 \frac{2q}{p^2} + \left( \frac{\lambda}{p} \right)^2 b_2}{2(1 - \rho)} \right] \frac{1}{2} \]

(60)

In Figure 23, we see $E_Q(T)$ depicted for $\lambda = 0.01$, $\xi = 0.4$, $p = 0.1$, $b_1 = 0.1$, $b_2 = 4$. It is monotone increasing in $T$.

Figure 23. $\lambda = 0.01$ (slow traffic) also equal to server load $\rho = \lambda b_1 = 0.01$ and it yields $E_Q(50) \approx 9.32$.

In Figure 24, $E_Q$ picks up much quicker with $\lambda = 0.8$, $\xi = 0.4$, $p = 0.1$, $b_1 = 0.1$, and $b_2 = 4$.

Figure 24. Here the input stream is far more intense, with $\lambda = 0.8$ producing $E_Q(50) \approx 962$.

By Formula (19) we have another version of $E_Q$ where $a''_0$ is replaced with variance $\Sigma_0 = E_Q(\tau_{\nu})$.

\[ E_Q = \rho + \frac{\Sigma_0 + a^2_0 - a_0 + e^{-\lambda T} \frac{2q}{p^2}}{2 \left( a_0 + \frac{1}{p} e^{-\lambda T} \right)} + \left[ \frac{2 \left( \frac{\lambda}{p} \right)^2 b_2}{2(1 - \rho)} \right] \frac{1}{2} \]

(61)
6.4. Mean Stationary Service Cycle

Recall that all underlying processes are considered on a filtered probability space \((\Omega, \mathcal{F}, (\mathcal{F}_t : t \geq 0), P, (\mathbb{P}^x, x = 0, 1, \ldots))\). Because the service cycles represent important time epochs of decision makings over which the queueing process conditionally regenerates, we would like to discuss the computational aspect of ends of service cycles \(\{T_n\}\) in some form and lay foundation for the forthcoming analysis of the continuous time parameter process \(Q(t)\) rendered in Section 7. Let \(C_n = ||(T_n, T_{n+1})||\) (the length of the \(n\)th cycle). Then, with \(C_n = \lim_{n \to \infty} C_n\), a.s. (62)

We call \(E_C\) the mean value of the stationary service cycle. We will show that \(E_C\) exists and equals \(\frac{1}{\lambda a}\) under the condition that \(\rho < 1\).

Now,

\[
E_C = E[T_{n+1} - T_n] = \sum_{i=0}^{\infty} E[T_{n+1} - T_n Q_n = i] P\{Q_n = i\} = \sum_{i=0}^{\infty} \mathbb{E}[T_1 - T_0 | \mathbb{P}\{Q_n = i\}] = \sum_{i=0}^{\infty} c_i \mathbb{P}\{Q_n = i\},
\]

where

\[
c_i := \mathbb{E}[T_1 - T_0] = \begin{cases} \mathbb{E}\Delta + b_1, & i = 0 \\ b, & i > 0 \end{cases}
\]

implying that

\[
E_C = \mathbb{P}\{Q_n = 0\} \mathbb{E}\Delta + b = \mathbb{P}\{Q_n = 0\} \Delta + b_1, \tag{65}
\]

where \(\Delta = \mathbb{E}\Delta\mu\).

The limit on the right of (65) exists if and only if \(\rho = \lambda ab_1 < 1\), so does the limit on the left, which by the Lebesgue dominated convergence theorem (applied to \(E_C\)) gives \(E_C\) introduced in (62). Hence,

\[
E_C = p_0 \Delta + b_1 = \frac{1 - \rho}{\lambda a} \Delta + b_1,
\]

and by Formula (27) of Theorem 2 we have from the last equation that

\[
E_C = (1 - \rho) \Delta a + b_1 = (1 - \rho) \frac{1}{\lambda a} + b_1 = \frac{1}{\lambda a}
\]

Thus, we proved that the limit below exists and equals

\[
E_C = \lim_{n \to \infty} E_C = \frac{1}{\lambda a}. \tag{66}
\]

This is an interesting result showing that in spite of all complex settings of the first service cycle that includes a state dependent vacation time and a random wait for \(N\) customers, the mean service cycle is identical to that of the regular \(M^X/G/1/\infty\) queue with no vacation and no \(N\)-Policy. We need to be reminded that, in the context of a \(G^X/G-NSV/1\) system, (66) holds under two necessary and sufficient conditions.

1. The arrival process is marked Poisson.
2. The offered load \(\rho = \lambda ab_1 < 1\).

In conclusion, we derive the following proposition.
Proposition 1. For a \( G^X/G – NSV/1 \) type queue (with state dependent vacation and N-Policy), the mean stationary service cycle \( \mathbb{E}C \) defined in (62) equals \( (\lambda a)^{-1} \) if and only if the arrival process is marked Poisson and the offered load \( \rho < 1 \).

Combining Proposition 1 and Theorem 2, we have the following corollary.

Corollary 9. In the queueing system with NSV-Policy and a marked renewal process with position independent marking as input, with the LST of the interarrival time \( \varphi(\theta) \) and mean \( \varphi \), the mean length of the service cycle \( \Delta = \mathbb{E} \Delta \mu \) that starts with zero customers, followed by state dependent single vacation time and service time specified by the N-Policy and the mean number of customers \( \alpha = \mathbb{E}W_\mu \) accumulated in the system upon service resumption are related as

\[
\Delta = \alpha \mathbb{E}C
\]

if and only if the input to the system is marked Poisson with the rate \( \lambda \) for its support counting measure and for \( \rho < 1 \).

Thus, \( \mathbb{E}C \) acts as a multiplier in Theorem 2 between \( \alpha = \mathbb{E}W_\mu \) and \( \Delta = \mathbb{E} \Delta \mu \).

Remark 5. Because by Proposition 1, \( \mathbb{E}C = \lim_{n \to \infty} \mathbb{E}C_n \) exists (and equals \( \frac{1}{\lambda a} \)) under the necessary and sufficient condition that \( \rho < 1 \), we conclude from (63) that

\[
\frac{1}{\lambda a} = \mathbb{E}C = \langle p, c \rangle = \sum_{i=0}^{\infty} p_i c_i
\]

where

\[
c = \{c_i = \mathbb{E}[T_1], i = 0, 1, \ldots\}.
\]

7. Semi-Regenerative Analysis

Let \( Z = (Z(t)) \) be an \( F_T \)-adapted process on a filtered space denoted by \( (\Omega, \mathcal{F}(\Omega), (\mathcal{F}_t), (\mathcal{P}^x)_{x=0,1,\ldots}) \) and \( \tau \) be a stopping time relative to \( (\mathcal{F}_t) \). \( Z \) is said to have a locally strong Markov property at \( \tau \) if for each \( t \geq 0 \),

\[
\mathbb{E}^x[g(Z(t+\tau)) \mid \mathcal{F}_t] = \mathbb{E}^Z[g(Z(\tau))] \text{ a.s.}
\]

holding true for any integrable Borel measurable function \( g \).

An integer-valued \( F_T \)-adapted process \( Z \geq 0 \) is called semi-regenerative if

(i) There is a point process \( \{T_n; n = 0, 1, \ldots\} \) such that for each \( n \), \( T_n \) is a stopping time relative to \( (\mathcal{F}_t) \).

(ii) \( Z \) has a locally strong Markov property at \( T_n, n = 0, 1, \ldots \)

(iii) \( Z \) is a.s. right-continuous.

(iv) \( Z(T_n, T_n) := (X_n, T_n) \) is a time-homogeneous Markov renewal process embedded in \( Z \) over \( (T_n) \).

Let

\[
K_{ij}(t) := \mathbb{P}^i \{Z(t) = j; T_1 > t\}.
\]

Then, matrix \( K(t) = \{K_{ij}(t)\} \) is called the semi-regenerative kernel. In the context of our queueing system \( M^X/G_\mu/1 \), the queueing process \( \{Q(t)\} \) is semi-regenerative relative to the sequence \( \{T_n\} \) of departures. So we will use notation \( Q = \{Q(t)\} \) in place of \( Z \).
Then from the key renewal theorem and (68) and (69) of Remark 5,
\[ \pi_k := \lim_{t \to \infty} \mathbb{P}^i\{Q(t) = k\} = \frac{1}{\mathcal{E}} \sum_{j=0}^{\infty} p_j \int_{s=0}^{\infty} K_{jk}(u) du \]  
(70)

Denote matrix
\[ \mathcal{H} = \int_{u=0}^{\infty} K(u) du \]  
(71)
and call it the integrated semi-regenerative kernel. We assume that the semi-regenerative kernel \( K \) is integrable over \( \mathbb{R}_+ \). If \( h_j(z) \) is the PGF of the \( j \)th row of \( \mathcal{H} \), also expressed as
\[ h_j(z) = \int_{t=0}^{\infty} \mathbb{E}^j\left[z^{Q(t)}1_{[t<T_j]}\right] dt, \]  
(72)
then Equation (70) can be rewritten as
\[ \pi(z) = \sum_{k=0}^{\infty} \pi_k z^k = \frac{1}{\mathcal{E}} \sum_{j=0}^{\infty} h_j(z) p_j. \]  
(73)

With the notation
\[ h(z) = (h_0(z), h_1(z), \ldots)^T \]  
(74)
and Equation (68), (73) reads as
\[ \pi(z) = \frac{1}{\mathcal{E}} \langle \mathbf{p}, h(z) \rangle = \frac{1}{\lambda a} \langle \mathbf{p}, h(z) \rangle \]  
(75)
with no restrictions on \( a(z) \) and \( \xi(x) \).

**Theorem 3.** In a single-server queue \( a \) with marked Poisson input stream and general service, suppose the first service cycle \([T_0, T_1]\) consists of the first service period preceded by serve’s vacation time that lasts from \( T_0 = 0 \) to \( T_1 \). Upon server’s return at time \( T_1 \), he may or may not wait for at least \( N \) customers until time \( \Delta_N \). The queue length \( Q(\Delta_N) \) becomes \( W_N \) and the first service begins, while new customers enter the system. The Laplace functional of the continuous time parameter queueing process \( Q(t) \) observed over the first service cycle, jointly with \( \Delta_N \) satisfies the formula
\[ \int_{t=0}^{\infty} e^{-\theta t} \mathbb{E}^{\mathbb{Q}_0}\left[z^{Q(t)}1_{[t<T_1=\Delta_N+\xi_1]}\right] dt 
= \frac{\mathbb{E}_{\mathbb{Q}_0} z^{W_N e^{-\theta \Delta_N}}}{\theta + \lambda - \lambda a(z)} + \frac{1 - \beta(\theta + \lambda - \lambda a(z))}{\theta + \lambda - \lambda a(z)} \mathbb{E}_{\mathbb{Q}_0}\left[z^{W_N e^{-\theta \Delta_N}}\right] \]  
(76)
\[ = \frac{\mathbb{E}_{\mathbb{Q}_0} z^{W_N e^{-\theta \Delta_N}}}{\theta + \lambda - \lambda a(z)} + \frac{\beta(\theta + \lambda - \lambda a(z))}{\theta + \lambda - \lambda a(z)} \mathbb{E}_{\mathbb{Q}_0}\left[z^{W_N e^{-\theta \Delta_N}}\right], \text{ a.s.} \]  
(77)

If \( Q_0 = 0 \) a.s.,
\[ \int_{t=0}^{\infty} e^{-\theta t} \mathbb{E}^{\mathbb{Q}_0}\left[z^{Q(t)}1_{[t<T_1=\Delta_N+\xi_1]}\right] dt 
= \frac{1}{\theta + \lambda - \lambda a(z)} \left[1 - \beta(\theta + \lambda - \lambda a(z)) \mathbb{E}^{\mathbb{Q}_0}\left[z^{W_N e^{-\theta \Delta_N}}\right]\right], \]  
(78)
where
\[ \beta(\theta) = \mathbb{E}e^{-\theta \xi_1}. \]  
(79)
Proof. (i) We show that
\[
\int_{t=0}^{\infty} e^{-\theta t} \mathbb{E} Q_0 \left[ z Q(t) 1_{[\Delta_1 \leq t < T_1 = \Delta_2 + \tau_1]} \right] dt = \frac{1 - \beta(\theta + \lambda - \lambda a(z))}{\theta + \lambda - \lambda a(z)} \mathbb{E} Q_0 \left[ z W_\mu e^{-\delta \mu} \right]
\]

On \([\Delta_\mu, \Delta_\mu + \sigma_1] \), \(Q(t) = W_\mu + \Pi(\Delta_\mu, t)\). Further, because \(\sigma_1 \Pi \Delta_\mu \otimes W_\mu\), (II means independent),
\[
\int_{t=0}^{\infty} e^{-\theta t} \mathbb{E} Q_0 \left[ z Q(t) 1_{[\Delta_1 \leq t < T_1 = \Delta_2 + \tau_1]} \right] dt = \int_{t=0}^{\infty} e^{-\theta t} \mathbb{E} Q_0 \left[ z W_\mu + \Pi(\Delta_\mu, t) \right] \left[ z Q(t) 1_{[\Delta_1 \leq t < T_1 = \Delta_2 + \tau_1]} \right] dt
\]
\[
= \sum_{k=0}^{\infty} \int_{t=0}^{\infty} \int_{\Delta_k < t < \Delta_{k+1}} e^{-\theta t} \mathbb{E} Q_0 \left[ z Q(t) 1_{[\Delta_1 \leq t < T_1 = \Delta_2 + \tau_1]} \right] dt
\]
\[
= \sum_{k=0}^{\infty} \int_{t=0}^{\infty} e^{-\theta t} \mathbb{E} Q_0 \left[ z Q(t) 1_{[\Delta_1 \leq t < T_1 = \Delta_2 + \tau_1]} \right] dt
\]
\[
= \frac{1}{\theta + \lambda - \lambda a(z)} \left[ \sum_{k=0}^{\infty} \int_{t=0}^{\infty} e^{-\theta t} dP_{Q_0}(\delta, k) \right] - \sum_{k=0}^{\infty} \int_{t=0}^{\infty} e^{-\theta t} \mathbb{E} e^{-\gamma t [\theta + \lambda - \lambda a(z)]} dP_{Q_0}(\delta, k)
\]
\[
= 1 - \frac{\beta(\theta + \lambda - \lambda a(z))}{\theta + \lambda - \lambda a(z)} \mathbb{E} Q_0 \left[ z W_\mu e^{-\delta \mu} \right]
\]

(ii) Next, we have
\[
\int_{t=0}^{\infty} e^{-\theta t} \mathbb{E} Q_0 \left[ z Q(t) 1_{[0 \leq t < \Delta_\mu]} \right] dt = \int_{t=0}^{\infty} e^{-\theta t} \mathbb{E} Q_0 \left[ z Q_0 + \Pi(0, t) 1_{[0 \leq t < \Delta_\mu]} \right] dt
\]
\[
= \frac{\mathbb{E} Q_0}{\theta + \lambda - \lambda a(z)} - \lim_{\sigma_1 \to \infty} \int_{t=0}^{\infty} e^{-\theta t} \mathbb{E} Q_0 \left[ z Q(t) 1_{[\Delta_1 \leq t < \Delta_2 + \tau_1]} \right] dt
\]
From (i), \(\lim_{\sigma_1 \to \infty} \mathbb{E} e^{-\gamma t [\theta + \lambda - \lambda a(z)]} = 0\) and thus
\[
\int_{t=0}^{\infty} e^{-\theta t} \mathbb{E} Q_0 \left[ z Q(t) 1_{[0 \leq t < \Delta_\mu]} \right] dt = \frac{\mathbb{E} Q_0}{\theta + \lambda - \lambda a(z)} - \frac{1}{\theta + \lambda - \lambda a(z)} \left[ \sum_{k=0}^{\infty} \int_{t=0}^{\infty} e^{-\theta t} dP_{Q_0}(\delta, k) \right] - \sum_{k=0}^{\infty} \int_{t=0}^{\infty} e^{-\theta t} dP_{Q_0}(\delta, k)
\]
\[
= \frac{\mathbb{E} Q_0}{\theta + \lambda - \lambda a(z)} - \frac{1}{\theta + \lambda - \lambda a(z)} \mathbb{E} Q_0 \left[ z W_\mu e^{-\delta \mu} \right]
\]
which completes the proof of the theorem, because (78) is obvious. \(\Box\)

**Theorem 4.** In a single-server queue with marked Poisson input stream, general service, suppose the first service cycle \([T_0, T_1]\) consists of the first service period preceded by serve’s vacation time that lasts from \(T_0 = 0\) to \(\tau_1\). Upon server’s return at time \(\tau_1\), he may or may not wait for at least
$N$ customers until time $\Delta_\mu$. Then the PGF of the stationary distribution of the continuous time parameter queueing process $Q(1)$ satisfies the formula

$$\pi(z) = P(z)M(z),$$  \hfill (80)

where

$$P(z) = p_0 \frac{\beta(\lambda - \lambda a(z)) \lambda [a(z) - 1]}{z - \beta(\lambda - \lambda a(z))}, p_0 = 1 - \rho, \text{ and } M(z) = a \frac{1 - z}{1 - a(z)}. \hfill (81)$$

**Proof.** From (77), because the system is exhaustive

$$h_0(\theta, z) = \frac{1 - a(\theta, z)}{\theta + \lambda - \lambda a(z)} + \frac{1 - \beta(\theta + \lambda - \lambda a(z))}{\theta + \lambda - \lambda a(z)} a(\theta, z)$$

$$= \frac{1}{\theta + \lambda - \lambda a(z)} - \frac{\beta(\lambda - \lambda a(z))}{\lambda - \lambda a(z)} a(\theta, z)$$

$$h_0(z) = h_0(0, z) = \lambda - \lambda a(z) - \frac{\beta(\lambda - \lambda a(z))}{\lambda - \lambda a(z)} a(z) \hfill (82)$$

For $Q_0 = j > 0$, the server does not leave the system, and to utilize Theorem 70, we can set

$$T_1 = \sigma_1, \Delta_\mu = 0, W_\mu = j, \Delta_\mu = 0, a(\theta, z) = E z^j e^{-\theta z} = z^j \hfill (83)$$

From (76),

$$h_j(\theta, z) = \frac{z^j - z^j}{\theta + \lambda - \lambda a(z)} + \frac{1 - \beta(\theta + \lambda - \lambda a(z))}{\theta + \lambda - \lambda a(z)} z^j, j > 0, \hfill (84)$$

implying that

$$h_j(z) = D(z) z^j, D(z) = \frac{1 - \beta(\lambda - \lambda a(z))}{\lambda - \lambda a(z)} \hfill (85)$$

and

$$\langle p, h(z) \rangle = p_0 h_0(z) + \sum_{i=1}^{\infty} p_i h_i(z)$$

$$= \left[ \frac{1}{\lambda - \lambda a(z)} - \frac{\beta(\lambda - \lambda a(z))}{\lambda - \lambda a(z)} a(z) \right] p_0 + D(z) P(z) - D(z) p_0$$

$$= \left[ \frac{1}{\lambda - \lambda a(z)} - \frac{\beta(\lambda - \lambda a(z))}{\lambda - \lambda a(z)} a(z) \frac{1 - \beta(\lambda - \lambda a(z))}{\lambda - \lambda a(z)} \right] p_0 + D(z) P(z)$$

$$= \beta(\lambda - \lambda a(z)) \left[ 1 - a(z) \right] p_0 + D(z) P(z). \hfill (86)$$

Combining (35) with (74) and (75) and (86) we get

$$E \xi \pi(z) = \frac{\beta(\lambda - \lambda a(z))}{\lambda - \lambda a(z)} \left[ 1 - a(z) \right] p_0 + D(z) P(z),$$

where

$$D(z) = \frac{1 - \beta(\lambda - \lambda a(z))}{\lambda - \lambda a(z)}$$
and 
\[ P(z) = p_0 \beta(\lambda - \lambda a(z))[a(z) - 1] / [z - \beta(\lambda - \lambda a(z))] \]
\[ p_0 = \frac{1 - \rho}{\alpha} \]

Thus, 
\[ \mathbb{E}C \cdot \pi(z) = \rho_0 \beta(\lambda - \lambda a(z)) [1 - a(z)] \left[ 1 - \frac{1 - \beta(\lambda - \lambda a(z))}{z - \beta(\lambda - \lambda a(z))} \right] = P(z) \frac{1 - z}{\lambda - \lambda a(z)}. \]

With \( \mathbb{E}C = \frac{1}{\alpha} \), we finally have that
\[ \pi(z) = P(z) a \frac{1 - z}{1 - a(z)}. \]

\[ \square \]

**Remark 6.** In particular, for \( a(z) = z, \pi(z) = P(z) \) which is the same result as for a regular \( M/G/1 \) queue but now with a complex state dependent NPSV system.

**Example 6.** From Theorem 4 and Example 5,
\[ \mathbb{E}Q(\infty) = \pi'(1-) = \lim_{z \to 1-} \frac{d}{dz} P(z) M(z) = P'(1-) M(1-) + M'(1-) \] (87)
\[ \lim_{z \to 1-} M(z) = \lim_{z \to 1-} a \cdot \frac{1 - z}{1 - a(z)} = 1, \quad M'(1-) = a \frac{1 - a(z) - a'(z)(1 - z)}{[1 - a(z)]^2} \bigg|_{z = 1-} = a \frac{-a'(z) - a''(z)(1 - z) + a'(z)}{2[1 - a(z)](-a'(z))} \bigg|_{z = 1-} = a \frac{1}{2a'(z)} \frac{1 - z}{1 - a(z)} \bigg|_{z = 1-} = \frac{1}{2} a''(1-) . \] (88)

Therefore, 
\[ \mathbb{E}Q(\infty) = \pi'(1-) = P'(1-) + \frac{1}{2} a''(1-) = \mathbb{E}Q_{\infty} + \frac{1}{2} a''(1-) , \] (89)

where \( \mathbb{E}Q_{\infty} \) was treated in Section 6. See Formula (41) and follow-up discussions.

Under the assumptions of Example 5, with \( a(z) = \frac{p_0}{1 - p_0}, \) we have \( a''(1) = \frac{2q}{p^2} \) and from (89) and (55),
\[ \mathbb{E}Q(\infty) = \pi'(1-) = \rho + \frac{a''(1)}{2} + \left[ \frac{\lambda b 2q}{p^2} + \left( \frac{\lambda}{p} \right)^2 b_2 \right] \frac{1}{2(1 - \rho)} + \frac{q}{p^2} \] (90)

The version of (90) under \( N = 1 \), according to (60), reads
\[ \mathbb{E}Q(\infty) = \pi'(1-) = \rho + \frac{a''(1) + e^{-\lambda T} 2q}{2(a_0 + \frac{1}{p} e^{-\lambda T})} + \left[ \frac{\lambda b 2q}{p^2} + \left( \frac{\lambda}{p} \right)^2 b_2 \right] \frac{1}{2(1 - \rho)} + \frac{q}{p^2} \] (91)

In Figure 25, below we see \( \mathbb{E}Q(\infty) \) behaves analogous to \( \mathbb{E}Q_{\infty} \) (on Figure 24) because it differs in the additive term \( q / p^2 \). However, with \( p \) small such as \( p = 0.1 \), \( \mathbb{E}Q(\infty) \) moves up by as much as 90 compared to \( \mathbb{E}Q_{\infty} \).
8. Performance Measures

8.1. Switchovers

A switchover is manifested by server’s exiting a busy period, after the queue becomes exhausted, and starting his work at the maintenance facility. When the server resumes his service of customers, the associated previous busy cycle ends and the new one begins. A busy cycle consists of a busy period and maintenance period with a possible wait. Each busy cycle thus includes exactly one switchover or equivalently one attendance of maintenance.

While maintenance is a mandatory work, in some applications, a large number of back and forth movements in and out of the system (switchovers) is undesirable and it may indicate that the input traffic is sparse. If so, the server may be better off to stay longer at the maintenance (that is increasing \( T \)) to ensure that enough customers at the main facility will accumulate. This can also be maintained by increasing the threshold \( N \) thereby letting the server wait longer for the input to deliver \( N \) or more units before a new cycle begins. It seems plausible though that working at the maintenance facility is economically more efficient than staying idle and waiting for units to replenish the buffer to \( N \) or more.

As most performance measures require, the number of switchovers should be counted within a fixed time interval, say \([0, t]\), thus making the time sensitive analysis necessary.

Suppose \( T^j_k := \{ T^j_n : k = 0, 1, \ldots \} \) are the successive moments of those \( T_n \)'s at which \( Q_n \) enters state \( \{j\} \), if \( T_n \) is the \( n \)th departure epoch. \((T^j_i)\) is a delayed renewal process embedded in \((T)\). For any fixed \( t \geq 0 \), the random variable

\[
V^j(t) = \sum_{n=0}^{\infty} \mathbf{1}_{[0,t]}(T^j_n) = \sum_{n=0}^{\infty} \mathbf{1}_{[0,t]}(Q_n)\mathbf{1}_{[0,t]}(T_n)
\]

(92)

gives the total number of visits of state \( j \) by the Markov renewal process \((Q_n, T_n)\) in time interval \([0, t]\). In particular, for \( j = 0 \), \( V^0(t) \) is the total number of visits of state 0 by \( Q_n \) over \([0, t]\) or equivalently, the total number of server’s idle periods (maintenance and wait) in interval \([0, t]\).

Next, we define

\[
R(i,j,t) = \mathbb{E}^i V^j(t).
\]

(93)

The functional matrix \(\{R(i,j,t) : i, j = 0, 1, \ldots ; t \geq 0\}\) is the Markov Renewal Function (MRF). The MRF gives the expected number of entrances of the queue (upon departures) in every state \( j \) during interval \([0, t]\), given that the queue started from a state \( i \).
From Çinlar [70] and equations (68) and (69),
\[
\lim_{t \to \infty} \frac{R(i,0,t)}{t} = \frac{p_0}{\alpha C} = \lambda a p_0
\] (94)

The latter has the following interpretation. The mean number of entrances of the queueing process in state 0 upon departure epochs per unit of time observed over a fairly long period of time is proportional to the steady state probability \( p_0 \) of the queue length upon a departure. For most models it is the mean number of switchovers (or more precisely, the switchover rate) between busy and idle or vacation periods. That being said we focus on the quantity
\[
\lim_{t \to \infty} \frac{R(i,0,t)}{t} = \frac{p_0}{\alpha C} = \lambda a \frac{1 - \rho}{\alpha}
\] (95)

that gives us the mean number of switchovers from busy to idle periods per unit time, or equivalently, the number of maintenance periods per unit time. With \( s \) being the penalty for one switchover, we have
\[
S = S(T) = s \cdot \lambda a \frac{1 - \rho}{\alpha}
\] (96)
as the penalty for all switchovers (maintenance periods) per unit time.

**Example 7.** Under the assumptions of Example 3 about \( a(z) = \frac{pz}{1-pz} \) and \( \xi(\theta) = \frac{\xi}{\xi + \theta} \) and setting \( N = 1 \),
\[
\alpha_0 = \frac{\lambda}{\rho} L(1) = \frac{\lambda \xi}{\beta} T + \frac{1}{\rho} \left( \frac{\lambda}{\beta} \right)^2 \left( 1 - e^{-\beta T} \right), \quad b = \lambda + p \xi,
\]
and from Formula (29),
\[
\mathbb{E}W_{\mu} = \alpha = \alpha_0 + ae^{-\lambda T} = \alpha_0 + \frac{1}{\rho} e^{-\lambda T}
\]
\[
= \frac{\lambda \xi}{\lambda + p \xi} T + \frac{1}{\rho} \left( \frac{\lambda}{\lambda + p \xi} \right)^2 \left( 1 - e^{-(\lambda + p \xi) T} \right) + \frac{1}{\rho} e^{-\lambda T}.
\]

That, with (96), reads
\[
S(T) = s \cdot \left( 1 - \frac{\lambda b}{\rho} \right) \left[ \frac{\lambda \xi}{\lambda + p \xi} T + \frac{1}{\rho} \left( \frac{\lambda}{\lambda + p \xi} \right)^2 \left( 1 - e^{-(\lambda + p \xi) T} \right) + \frac{1}{\rho} e^{-\lambda T} \right]^{-1}. \quad (97)
\]

In this particular case, under \( N = 1 \), only \( T \) is subject to adjustments. For example, system’s management might be interested in maintenance time lasting as long as possible to ensure a thorough work. Obviously, it does not agree with the opposite objective to make \( S(T) \) as low as possible. So we are looking for
\[
S_0 = \min\{S(T) : T_0 \leq T\} \quad (98)
\]

In Figure 26, below, we depicted the curve of \( S \), which as suspected is monotone decreasing in \( T \), meaning that with \( T \) increasing from 0 to 50, the number of switchovers per unit time decreases. Here \( s = 1, \lambda = 2, p = 0.99, b_1 = \mathbb{E}c = 0.491 \), and \( \xi = 0.4 \).
Figure 26. Switchover penalty per unit time $S$ with $s = 1$, $\lambda = 2$, $p = 0.99$, $b_1 = \mathbb{E}\sigma = 0.491$, and $\xi = 0.4$.

In Figure 27, below, we depicted the curve of $S$, which is also monotone decreasing in $T$, but unlike the previous plot, $S(T)$ is concave down. Then concavity changes for larger values of $T$ beyond $T = 100$. Here $s = 1$, $\lambda = 0.01$, $p = 0.1$, $b_1 = \mathbb{E}\sigma = 0.1$, and $\xi = 0.4$.

Figure 27. Switchover penalty per unit time $S$ with $s = 1$, $\lambda = 0.01$, $p = 0.1$, $b_1 = \mathbb{E}\sigma = 0.1$, and $\xi = 0.4$.

8.2. Buffer Load

Let $q(k)$ be the expense function due to the presence of $k$ customers in the system per unit time. Then,

$$Q[0, t] = \mathbb{E} \int_{u=0}^{t} q(Q(u)) du$$

(99)

gives the mean expense due to all customers present in the system on interval $[0, t]$ as illustrated in Figure 28 below.
We can represent $Q[0,t]$ as follows by using the Fubini’s Theorem,

\[ Q[0,t] = \sum_{k \geq 1} \mathbb{E}^{i} \int_{u=0}^{t} 1_{\{k\}}(Q(u))q(Q(u))du \]

\[ = \sum_{k \geq 1} q(k)\mathbb{E}^{i} \int_{u=0}^{t} 1_{\{k\}}(Q(u))du \]

\[ = \sum_{k \geq 1} q(k) \left[ \int_{u=0}^{t} \mathbb{E}^{i} 1_{\{k\}}(Q(u))du \right] \]

\[ = \sum_{k \geq 1} q(k) \int_{u=0}^{t} \mathbb{P}^{i} \{ Q(u) = k \} du \] \hspace{1cm} (100)

Now from Çinlar [70] or Dshalalow [36] (p. 98), Theorem A.1,

\[ \lim_{t \to \infty} \frac{1}{t} \int_{u=0}^{t} \mathbb{P}^{i} \{ Q(u) = k \} du = \pi_{k}, k = 0, 1, \ldots \] \hspace{1cm} (101)

Applying (101) to (100) and by the monotone convergence theorem,

\[ Q = \lim_{t \to \infty} \frac{Q[0,t]}{t} = \sum_{k \geq 1} q(k)\pi_{k} \] \hspace{1cm} (102)

$Q$ is the penalty rate for all customers that occupy the system per unit time as observed over a long period time.

Assuming the expense function $q(k)$ to be linear $q(k) = qk$, we have

\[ Q = q\pi'(1-\pi) = q \cdot \mathbb{E}Q(\infty) \] \hspace{1cm} (103)

Another variant of the expense function $q(k) = q^{k}$, making

\[ Q = \pi(q), \] \hspace{1cm} (104)

which from (80) and (81) reads

\[ Q = \frac{p_{0}a\beta(\lambda - \lambda a(q))}{z - \frac{a(q)-1}{\beta(\lambda - \lambda a(q))}} \frac{1-q}{1-a(q)} \] \hspace{1cm} (105)

In either case, we look for

\[ Q_{0} = \min \{ Q(T) : T_{0} \leq T \} \] \hspace{1cm} (106)
Example 8. Combining the penalties for too many switchovers and the queue cost per unit time gives a simple objective function (if the cost is linear)

\[ Q(T) = s \cdot \lambda a \frac{1 - \rho}{\alpha} + q \cdot \mathbb{E}Q(\infty) \]

as a linear combination of switchovers \( \lambda a \frac{1 - \rho}{\alpha} \) and queue length \( \mathbb{E}Q(\infty) \) with respective penalties of \( s = 1000 \) and \( q = 2 \) (per switchover and per customers per unit time) produces Figure 29 as a function of \( T \), for \( 0 \leq T \leq 50 \). Here we use \( \lambda = 0.01, \xi = 0.4, p = 0.1, b_1 = 0.1, \) and \( b_2 = 4 \), on the left and with same parameters except for \( b_1 = 2.5 \) on the right.

Figure 29. \( Q(T) \) with \( b_1 = 0.1 \) and \( b_1 = 2.5 \), and \( \min Q(t) \) attained at 39 and 27, respectively.

8.3. Number of Customers during Server’s Idle Period

Recall that the mean number of customers gathered in the system during the pure maintenance period is \( \mathbb{E}Q(\tau_V) = \alpha_0 \). Again, the longer the maintenance period lasts, the more customers come during a maintenance who end up waiting. In some applications, those customers will be most impatient waiting in the system especially with no server present. This situation may become undesirable and it can mean an extra stiff penalty per waiting customer that the system will incur.

With a linear cost for \( k \) customers, we have \( U(k) = uk \) and thus,

\[ u \cdot \mathbb{E}Q(\tau_V) = u \cdot \alpha_0 \quad (107) \]

is the cost for all customers in the system waiting during a maintenance period which is a multiple of the expected queue length upon server’s return. If a total number of waiting customers gathered during server’s inactivity that needs to be taken into account whether or not during maintenance, then \( \alpha_0 \) needs to be replaced with \( \alpha \) yielding

\[ u \cdot \mathbb{E}W_\mu = u \cdot \alpha. \quad (108) \]

Remark 7. From Section 8.1, \( \lambda \alpha p_0 \) is the mean number of maintenances per unit time. We assume that

\[ M_c = \lambda \alpha p_0 a_0 = \lambda a \frac{1 - \rho}{\alpha} a_0 \quad (109) \]

can be regarded as the mean number of customers waiting during the maintenance periods per unit time. Note that the latter is very speculative without any rigorous justification. In fact, the law of double expectation cannot be used in this case, because the number of visits of state 0 and the length of one maintenance period (and thus the number of customers during the maintenance period) are
not independent. This, however, does not mean that the above assertion is wrong either, because a similarly paradoxical property shown in Corollary 7 holds true.

**Example 9.** From Example 3,

\[
\alpha_0 = \frac{\lambda_\xi}{\lambda + p_\xi T} + \frac{1}{p} \left( \frac{\lambda}{\lambda + p_\xi} \right)^2 \left( 1 - e^{-(\lambda + p_\xi)T} \right) \tag{110}
\]

Here \( u \cdot \alpha_0 \) is the penalty for the mean number of customers \( \mathbb{E}Q(\tau_\nu) \) present in the system during a maintenance period. Then, from (109) and (110),

\[
M_c := \alpha_0 \cdot \lambda \alpha (1 - \rho) / a \tag{111}
\]

is the mean number of customers during all maintenance instances per unit time. Under the assumption that \( N = 1 \), using Formula (58) gives

\[
M_c = \alpha_0 \lambda \alpha (1 - \rho) / \left( \alpha_0 + 1 - e^{-\lambda T} \right) \tag{112}
\]

**Example 10.** Let \( U(k) = u^k \) \((u < 1)\) be the penalty for \( k \) customers waiting by the end of time interval \([0, t]\) (if a maintenance period lasts \( t \)) regardless on when they came in. Then,

\[
u^\alpha_0 = \alpha_0(u) = \mathbb{E}U^Q(\tau_\nu), \quad (u < 1)
\]

is the mean penalty for \( Q(\tau_\nu) \) customers during one maintenance period.

From Example 2, we have

\[
\alpha_0(u) = 1 - \lambda [1 - a(u)] \mathcal{L}_x^{-1} \left( \frac{1}{x \lambda} \right) \left( \frac{1}{x \lambda + x - \lambda a[u_x(x)]} \right) (T).
\]

**Example 11.** If the penalty for \( k \) customers during a maintenance period is \( U(k) = uk^2 \), then

\[
\mathbb{E}U(Q(\tau_\nu)) = u \cdot \mathbb{E}Q^2(\tau_\nu) \tag{113}
\]

is the expected cost for all customers waiting during one maintenance period. From Example 3,

\[
\mathbb{E}Q^2(\tau_\nu) = \alpha_0''(1) + \alpha_0 = \alpha_0'' + \alpha_0. \tag{114}
\]

We can regard

\[
C_m = u \cdot \mathbb{E}Q^2(\tau_\nu) / \mathbb{E}\tau_\nu \tag{115}
\]

the expected cost for all customers waiting during a unit time of a maintenance period. By Corollary 7,

\[
C_m = u \lambda a \left( \alpha_0'' + \alpha_0^3 \right) / \alpha_0 = u \lambda a \left( \alpha_0 + \frac{1}{\alpha_0} \alpha_0'' \right). \tag{116}
\]

Note that \( \mathbb{E}Q^2(\tau_\nu) / \mathbb{E}\tau_\nu \neq \mathbb{E}[\alpha_0''(\tau_\nu)] \) (the latter would be a right choice for \( C_m \)), but it seems more plausible compared to all other choices of penalties for waiting customers during maintenance.

**Example 12.** In this example we consider a linear objective function

\[
O(T) = s \cdot \lambda a \frac{1 - \rho}{a} + q \cdot \mathbb{E}Q(\infty) + u \cdot \lambda a \left( \alpha_0 + \frac{1}{\alpha_0} \alpha_0'' \right) \tag{117}
\]
made of the switchovers’ cost \( S(T) = s \cdot \lambda a^{1-\rho}/\alpha \) of (97) per unit time, buffer load cost per unit time (Section 8.2) \( q \cdot E_Q(\infty) \) of (86), (102), and (103), \( \alpha = E W_\mu \) of (31), and cost \( C_m = u \cdot E Q^2(\tau_\nu)/E \tau = u \cdot \lambda a \left( \alpha_0 + \frac{1}{\alpha_0} \sum_0 \right) \) per all customers in the system waiting per unit maintenance time Example 10).

To draw Figure 30 below, we set \( s = 1200 \) cost per switchover, cost per customer in the system \( q = 1.5 \), and the cost per customer during a maintenance period as \( c = 8 \). We set \( \lambda = 0.01 \), \( \xi = 4 \), \( p = 0.99 \), \( b_1 = 10 \), and \( b_2 = 4 \). We considered \( O(T) \) for \( 0 \leq T \leq 1000 \) and found \( O(363.4775) = \min O(t) \) (see the snapshot on the right in a close vicinity of \( T = 363.4775 \).

**Figure 30.** \( O(T) \) reaches its minimum at \( T \approx 363.4775 \). Its concavity is down on approximately \((0, 100)\) and up thereafter.

**Example 13.** In this example we consider a linear objective function

\[
O(T) = s \cdot \lambda a^{1-\rho}/\alpha + q \cdot E_Q(\infty) + c \cdot \alpha(T)
\]

made of the switchovers’ cost \( S(T) = s \cdot \lambda a^{1-\rho}/\alpha \) of (97), \( E Q(\infty) \) of (86), (102), and (103), and \( \alpha = E W_\mu \) of (31) which represents the total number of customers accumulated during an entire maintenance period and wait. We note that this objective function is inconsistent, because the first two summands represent costs of switchovers and buffer load per unit time, while the third term does not.

**Example 14.** Another variant of the objective function would be \( c \cdot \alpha(T) \) in the objective function \( O(T) = s \cdot \lambda a^{1-\rho}/\alpha + q \cdot E_Q(\infty) + c \cdot \alpha(T) \) of (118) with \( M_c = a_0 \lambda a (1 - \rho)/\alpha \) of (112) bringing all terms to those in unit time, although this choice of the objective function has its shortcoming as pointed out in Remark 7.

Thus, the modified version of (118) reads

\[
O(T) = s \cdot \lambda a^{1-\rho}/\alpha + q \cdot E_Q(\infty) + c \cdot a_0 \lambda a (1 - \rho)/\alpha
\]

\[
= (s + ca_0) \lambda a^{1-\rho}/\alpha + q E_Q(\infty),
\]

where we set the same parameters as in Figure 30, namely, \( s = 1200 \) cost per switchover, cost per customer in the system \( q = 1.5 \), and the cost per customer during a mean maintenance period and wait combined is \( c = 8 \). We also set \( \lambda = 0.01 \), \( \xi = 4 \), \( p = 0.99 \), \( b_1 = 10 \), and \( b_2 = 4 \) in Figure 31. Here, \( \min \{O(T) : 0 < T \leq 1600\} \approx O(366) \).
Even though the choice $c \cdot a_0 \lambda(1 - \rho) / \alpha$ of penalty cost for waiting customers during maintenance is less credible than that in Example 12 (Figure 30), the minimum 366 of $O(T)$ comes pretty close to the minimum of $O(T)$ in Figure 30.

9. Conclusions

In this paper, we studied a single server queueing system of $\text{GI}^X/G/1$ type with $N$-Policy and single vacation, in notation, $\text{GI}^X/G-NSV/1$. Under our assumptions, the vacation length is state dependent. When the queue is exhausted, the server leaves the system for a routine maintenance initially constrained by a deterministic or random number $T$, which is the time the server is absent from the system. If during server’s maintenance, there are new customers entering the system, for each one of them, the dispatcher estimates their service times (in line with their needs) and shortens the server vacation time $T$ accordingly. (Note that the real service times of customers may turn out to be different from those estimated by the dispatcher.) If there are less than $N$ waiting customers upon server’s return to the system, he rests until the queue length reaches or exceeds $N$. The latter case is more likely, because the input is bulk.

There are two key factors in how the system is run. The times of customers’ departures determine when the server leaves, namely when the queue is exhausted. An embellished variant is when the queue drops below some specified threshold to order the server to leave, which is especially interesting if the service is batch. We do not study the system under this policy (that along with $N$-Policy is known as hysteretic control), but it is worth including in our future work. Thus, the departure times, define the beginning of the maintenance period, service cycles, and busy periods. A service cycle is the period of time between two successive departure epochs that include a pure service time or a service time together with a maintenance period followed by a possible wait of the server. The other key factor is customers’ arrivals times (that come in random batches) that, as mentioned, impact the server maintenance time.

We began our analysis of the system on a maintenance period whose policy is unordinary and it required a special treatment using first passage time principles of random walks on non-rectangular stochastic grid. We managed to obtain explicit formulas for joint and marginal distributions of the times of server’s return and the beginning of a new busy cycle, along with the associated contents of system’s buffer upon these times. We justified our claims through many explicit examples and special cases. In one of the main assertions, we proved that the ratio between the times of return and beginning of a busy period are linearly dependent upon the contents of the buffer at these times, respectively, if an only if the input to the system is marked Poisson of rate $L$ for the underlying support counting process and mean size $a$ of arriving batches of customers. This factor is $La$ which is also...
the reciprocal of the mean stationary service cycle. This would be a noteworthy result for queues even of a lesser complexity than ours.

To tame or next study, we continued our analysis under the assumption that the input is marked Poisson, first considering the embedded queueing process on departures, obtaining the invariant probability measure and the mean stationary service cycle \( EC \) which turned out to equal \( (\lambda a)^{-1} \). Furthermore, we went through special cases pertaining to the mean queue length, N-Policy, and the use of discrete operational calculus that we developed and embellished in our earlier papers. We provided discussion on numerical examples and graphics.

We then turned to semi-regenerative analysis of the system to obtain explicit results for the continuous time parameter queue. We found an explicit relation between the PGF (probability generating function) for the embedded and continuous time parameter queueing processes in equilibrium. In the final section, we discussed various performance measures of the system (such as the number of switchovers between busy and maintenance periods, the queue length, the number of impatiently waiting customers during server’s absence) all pertaining to control of the system through initially set up time \( T \). Because the control imposed was time sensitive, the prior semi-regenerative analysis of the system was imperative. We closed the last section with a number of numerical and graphical examples under detailed discussions.

Overall, we validated our main results through stochastic simulation and dedicated an entire Section 4 for this. We believe that our system can be further embellished and still remain analytically tractable. It is novel and it can be applied to various servicing systems. In particular, it can be used in computer operating systems that can be pre-programmed according to our results to optimize the performance of an underlying system through the choice of maintenance time \( T \) and of threshold \( N \) (especially if a large number of switchovers is undesirable).


Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.


Acknowledgments: The authors are very grateful to the anonymous referees for their constructive and useful suggestions that improved our work and presentation.

Conflicts of Interest: The authors declare no conflict of interest.

References


6. Tamrakar, G.K.; Banerjee, A. Study on Infinite Buffer Batch Size Dependent Bulk Service Queue with Queue Length Dependent Vacation.


42. Dshalalow, J.H.; White, R.T. Fluctuation Analysis of a Soft-Extreme Shock Reliability Model. Mathematics 2022, 10, 3312. [CrossRef]
44. White, R.T. On the Exiting Patterns of Multivariate Renewal-Reward Processes with an Application to Stochastic Networks. Symmetry 2022, 14, 1167. [CrossRef]
53. Durkee, J.W. Analytic and Monte Carlo random walk assessments of neutron fission chains and the probability of extinction. Prog. Nucl. Energy 2021, 142, 104008. [CrossRef]
64. Sajid, M.; ul Ain, Q.; Qureshi, H.; Tayyebe, T. One-dimensional quantum walks with a time and spin-dependent phase shift. Phys. Lett. A 2021, 416, 127674. [CrossRef]
68. Dshalalow, J.H.; White, R.T. Random Walk Analysis in a Reliability System under Constant Degradation and Random Shocks. Axioms 2021, 10, 199. [CrossRef]