ANALYSIS OF OPERATING CHARACTERISTICS
FOR THE HETEROGENEOUS BATCH ARRIVAL QUEUE
WITH SERVER STARTUP AND BREAKDOWNS

JAU-CHUAN KE$^1$ AND KUO-HSIUNG WANG$^2$

Communicated by Naoto Kaio

Abstract. In this paper we consider a like-queue production system in which server startup and breakdowns are possible. The server is turned on (i.e. begins startup) when $N$ units are accumulated in the system and off when the system is empty. We model this system by an $M^{[N]}/M/1$ queue with server breakdowns and startup time under the $N$ policy. The arrival rate varies according to the server’s status: off, startup, busy, or breakdown. While the server is working, he is subject to breakdowns according to a Poisson process. When the server breaks down, he requires repair at a repair facility, where the repair time follows the negative exponential distribution. We study the steady-state behaviour of the system size distribution at stationary point of time as well as the queue size distribution at departure point of time and obtain some useful results. The total expected cost function per unit time is developed to determine the optimal operating policy at a minimum cost. This paper provides the minimum expected cost and the optimal operating policy based on assumed numerical values of the system parameters. Sensitivity analysis is also provided.

Keywords. Batch arrivals, breakdowns, control, sensitivity analysis, startup, stochastic decomposition.
1. Introduction

In this paper we consider the modeling of a production system in which the production does not start until some specified number of units, say \( N \), are accumulated during the server off period. Units request for service usually arrive in batches with varying arrival rate. The server needs a startup time before providing the service; i.e., he must perform certain pre-service work to gear up the machinery for operation. When the server is working, he may meet unpredictable breakdowns but is immediately repaired. This production system can be modeled by an \( M^{[x]}/M/1 \) queue with server breakdowns and startup time under the \( N \) policy.

In this system, units arrive following a compound Poisson process where the arrival size is a random variable and the arrival rate varies according to the server’s status: off, startup, busy, or breakdown. The server is turned off as soon as the system becomes empty. When \( N \) units are accumulated in the system, the server is immediately turned on but is temporarily unavailable to the waiting units. He needs a startup time before starting his each service period. After the server finishes his startup, he starts to serve the waiting units until the system becomes empty. Whenever the server is working, it is assumed that the server can break down at any time. Whenever the server fails, it is immediately repaired at a repair facility.

The concept of \( N \) policy was first introduced by Yadin and Naor [23]. The so-called \( N \) policy means that the server does not start to provide service until there are \( N \) units waiting in the system. Batch arrival queues with \( N \) policy was first studied by Lee and Srinivasan [9]. Later, Lee et al. [10] concentrated on the interpretation of the system characteristics of the \( M^{[x]}/G/1 \) queueing system under the \( N \) policy.

Past work regarding queueing systems under \( N \) policy may be divided into two categories: (i) the case of server startup, and (ii) the case of server breakdowns. In the case of server startup, Baker [1] first proposed the \( N \) policy \( M/M/1 \) queueing system with exponential startup time. Borthakur et al. [2] extended Baker’s results to the general startup time. The \( N \) policy \( M/G/1 \) queueing system with startup time was first studied by Minh [15] and was investigated by several researchers such as Medhi and Templeton [14], Takagi [18], Lee and Park [13], Hur and Paik [8], and so on. In the case of server breakdowns, Wang [20] first proposed the \( N \) policy Markovian queueing system with server breakdowns. Wang [21] and Wang et al. [22] extended Wang’s model [20] to the \( N \) policy \( M/E_k/1 \) and \( M/H_2/1 \) queueing system cases, respectively. They developed the analytic closed-form solutions and provided a sensitivity analysis.

The purpose of this paper is threefold. Firstly, the state equations are established to get the steady-state probability distribution as well as the departure point queue size distribution and some probability interpretations of the system characteristics are made. Secondly, we formulate the system’s total expected cost in order to determine the optimal operating \( N \) policy numerically at the minimum cost for various values of system’s parameters. Thirdly, we perform a sensitivity analysis.
2. The system and assumptions

One busy cycle begins right when the system becomes empty and the server is off. The server remains off until there are \( N \) units in the system. We call this the off period. The startup period begins when the server performs startup as soon as the number of waiting units reach \( N \) and terminates when he starts providing the service. In the startup period, the units arriving while the server is performing startup as well as those arriving during the off period are not served yet. The busy period is initiated when the server completes his startup and starts serving the waiting units. During the busy period, the server may break down and starts his repair immediately. This is called the breakdown period. As soon as the server is repaired, he returns and provides service until there are no units in the system. Since the complete period starts when the startup period is over and terminates when there are no units in the system, the complete period is represented by the sum of the busy period and the breakdown period. We illustrate our model with a typical sample path shown in Figure 1. Furthermore, our model is considered under the following specifications:

1. The arrival process is a compound Poisson process with batch size \( X \) and various rates \( \lambda_i (i = 0, 1, 2, 3) \) where \( \lambda_0, \lambda_1, \lambda_2, \) and \( \lambda_3 \) denote the group arrival rates during the server off, startup, busy, and breakdown periods, respectively. Define that \( b_k = \Pr(X = k) \) is the probability that the arrival size \( X \) is \( k \) \((k \geq 1)\). Arriving units within batches at the server form a single waiting line and are served in the order of their arrivals. The units within a batch are served one at a time by a single server. The service time for an individual unit is exponentially distributed with mean \( 1/\mu \).

2. The server is turned off when the system becomes empty. As soon as the number of units in the system reaches \( N \), the server is immediately turned on but is temporarily unavailable to the waiting units. He needs to take
an exponential startup time with parameter \( r \). As soon as the startup period is over, he serves the waiting units immediately.

3. When the server is working, the server can break down at any time with a Poisson breakdown rate \( \alpha \).

4. When the server fails, it is immediately repaired at a repair rate \( \beta \), where the repair times are exponentially distributed.

5. If the server fails or one unit is in service, then the arriving units or waiting units have to wait in the queue until the server is free. The service is allowed to be interrupted if the server breaks down. Nevertheless, the server is immediately repaired. When the repair of a server is completed, the server immediately begins serving a unit.

2.1. Practical justifications of the model

A number of practical problems may be formulated as one in which the arrivals are in batches with arrival rate depending on system’s state or server’s state, and the server needs a startup time before providing the service and it may break down when working.

One particular problem where this model is applicable is in the study of a production line system. Consider a production line in manufacturing system of job-shop type, where the arrival stream of job orders follows a compound Poisson process with heterogeneous rates \( \lambda_i \) (\( i \) denotes the server’s status, \( i = 0, 1, 2, 3 \)) and each job order often requires the manufacture of a random number \( X \) (\( X \geq 1 \)); that is, each job has more than one item). For economic efficiency, it is required that the production does not start until a specified number of orders, say \( N \), is accumulated during an idle period. Each item within a job requires an exponentially distributed production time with mean \( 1/\mu \). As the size of job orders reaches \( N \), the operator needs a startup time to operate machine before starting production and after the production is started, the production may be interrupted because of emergent events. But the production must immediately resume whenever the emergency is solved. The emergent event occurs according to a Poisson process with rate \( \alpha \) and the time spent to recover an emergent event is exponentially distributed with mean \( 1/\beta \). We can interpret the emergent events as server breakdowns. The startup time is a random variable, which has an exponential distribution with mean \( 1/r \); it is corresponding to extra operations (for example, setup, warm up, etc.) before starting production.

In real-life situations it is not unusual to encounter that the arrivals join the queue in batches with different arrival rate, the server may perform some pre-service work to gear up the machinery, and the service may be interrupted. One may try to incorporate more realism in the model by considering that (i) batch arrivals occur with different rate depending on the arrival time, system’s state and server’s state; (ii) the server performs the preparatory work before starting each service period; and (iii) the server is subject to breakdowns due to unpredictable or uncontrollable factors. Therefore, one may consider it necessary to investigate the heterogeneous arrival queue with server startup and breakdowns.
3. System steady probability

Let the state $i = 0$ represent the server is idle, while the state $i = 1$ represents the server is turned on and is in operation, and the state $i = 2$ represents the server is in operation but found to be broken down. In steady-state, the following notations are used.

$P_0(n) \equiv$ the probability that there are $n$ units in the system when the server is idle. There are two situations when the server is idle: (i) the server is turned off if $n \leq N - 1$, and (ii) it is turned on and performing startup if $n \geq N$;

$P_1(n) \equiv$ the probability that there are $n$ units in the system when the server is turned on and is in operation, where $n = 1, 2, \ldots$; and

$P_2(n) \equiv$ the probability that there are $n$ units in the system when the server is in operation but found to be broken down, where $n = 1, 2, \ldots$

It is easy to set up the following steady-state system equations:

$$\lambda_0 P_0(0) = \mu P_1(1), \quad (1)$$

$$\lambda_0 P_0(n) = \lambda_0 \sum_{k=0}^{n-1} b_{n-k} P_0(k), \quad 1 \leq n \leq N - 1 \quad (2)$$

$$(\lambda_1 + r) P_0(n) = \lambda_0 \sum_{k=0}^{N-1} b_{n-k} P_0(k) + \lambda_1 \sum_{k=N}^{n-1} b_{n-k} P_0(k), \quad n \geq N \quad (3)$$

$$(\lambda_2 + \mu + \alpha) P_1(n) = \lambda_2 \sum_{k=1}^{n-1} b_{n-k} P_1(k) + \mu P_1(n+1) + \beta P_2(n), \quad 1 \leq n \leq N - 1 \quad (4)$$

$$(\lambda_2 + \mu + \alpha) P_1(n) = \lambda_2 \sum_{k=1}^{n-1} b_{n-k} P_1(k) + \mu P_1(n+1) + \beta P_2(n) + r P_0(n), \quad n \geq N \quad (5)$$

$$(\lambda_3 + \beta) P_2(n) = \lambda_3 \sum_{k=1}^{n-1} b_{n-k} P_2(k) + \alpha P_1(n), \quad n \geq 1 \quad (6)$$

where $a > b$ in the $\sum_{j=a}^{b}$ notation indicates that the term is zero.

The results for the $N$ policy $M/M/1$ queueing system with an unreliable server are obtained by setting $\lambda_0 = \lambda_1 = \lambda_2 = \lambda_3$, $b_1 = 1$, and $r = \infty$ in (1)–(6). Equations (1)–(6) for $P_i(n)$ then correspond to the existing results in the literature (Wang [19]). The results for the ordinary $M^{[x]}/M/1$ queueing system with a reliable server are obtained by setting $N = 1$, $\lambda_0 = \lambda_1 = \lambda_2 = \lambda_3$, $r = \infty$, $\alpha = 0$, and $\beta = \infty$ in (1)–(6). Equations (1)–(6) for $P_i(n)$ then correspond to the existing results in the literature (Gross and Harris [7], p. 157).
3.1. DERIVATIONS OF $P_0(n)$

Solving (2)–(3) recursively, we finally get

$$P_0(n) = \begin{cases} P_0(0) \Psi(n), & n = 1, 2, \ldots, N - 1 \\ P_0(0) \frac{\lambda_0}{\lambda_1 + r} \Psi(N), & n = N \\ P_0(0) \sum_{k=0}^{n-N-1} [\Lambda(n-k) \Theta(k)], & n = N + 1, N + 2, \ldots, \end{cases}$$

where

$$\Psi(n) = \begin{cases} 1, & n = 0 \\ \sum_{1 \leq k \leq n} \sum_{\ell_1, \ell_2, \ldots, \ell_k, \ell_1 + \ell_2 + \ldots + \ell_k = n} b_{\ell_1} b_{\ell_2} \ldots b_{\ell_k}, & n = 1, 2, \ldots \\ 0, & \text{otherwise}, \end{cases}$$

$$\Lambda(n) = \frac{\lambda_0}{\lambda_1 + r} \left[ \Psi(n) + \frac{\lambda_1 b_0 - \lambda \Psi(N)}{\lambda_1 + r} \sum_{k=1}^{n-N} b_k \Psi(n-k) \right],$$

and

$$\Theta(n) = \begin{cases} 1, & n = 0 \\ \sum_{1 \leq k \leq n} \sum_{\ell_1, \ell_2, \ldots, \ell_k, \ell_1 + \ell_2 + \ldots + \ell_k = n} h_{\ell_1} h_{\ell_2} \ldots h_{\ell_k}, & n = 1, 2, \ldots \\ 0, & \text{otherwise}, \end{cases}$$

with $h_j = \frac{\lambda_0 b_j}{\lambda_1 + r}$.

**Remark 1.** As $b_1 = 1$ (single unit arrival), we have $\Psi(n) = 1$, for $n = 0, 1, 2, \ldots, N - 1$.

**Remark 2.** The meaning of (8) is sum up all possible products of $k b_i$’s in which the total of subscript values of $b$ equals $n$. As

$$\Psi(4) = b_4 + b_3 b_1 + b_2 b_2 + b_1 b_3 + b_1 b_1 b_2 + b_1 b_2 b_1 + b_2 b_1 b_2 + b_1 b_1 b_1 = b_4 + 2b_3 b_1 + b_2^2 + 3b_1 b_2 + b_1^3.$$  

The meaning of (9) is the same as (8).

3.2. PROBABILITY GENERATING FUNCTION

Probability generating function (p.g.f.) technique may be used to obtain analytic solution $P_0(0)$ in neat closed-form expression as there is no way of solving (1)–(6) in a recursive manner. Define the respective probability generating
functions (p.g.f.) of $P_0(n)$, $P_1(n)$, and $P_2(n)$ as follows:

\[ H_0(z) = \sum_{n=0}^{N-1} z^n P_0(n), \]
\[ H_1(z) = \sum_{n=N}^{\infty} z^n P_0(n), \]
\[ Q(z) = \sum_{n=1}^{\infty} z^n P_1(n), \]
\[ R(z) = \sum_{n=1}^{\infty} z^n P_2(n), \]

where $|z| \leq 1$.

Further, define the p.g.f. of the arrival size $X$ as $X(z) = \sum_{k=1}^{\infty} z^k b_k$. It is found that $E(X) = X'(1)$ and $E[X(X - 1)] = X''(1)$.

From (7), $H_0(z)$ is expressed in term of $P_0(0)$:

\[ H_0(z) = P_0(0) \sum_{n=0}^{N-1} z^n \Psi(n) = I(z)P_0(0), \] (10)

where $I(z) = \sum_{n=0}^{N-1} z^n \Psi(n)$ with $I(1) = \sum_{n=0}^{N-1} \Psi(n)$ and $I'(1) = \sum_{n=0}^{N-1} n\Psi(n)$.

In (2)–(3), (2)–(3) is multiplied by $z^n$ ($n = 1, 2, \ldots$) and then the equations are added terms by terms (see Appendix 1). We finally obtain

\[ H_1(z) = \frac{\lambda_0 P_0(0) + \lambda_0 [X(z) - 1]H_0(z)}{\lambda_1 + r - \lambda_1 X(z)}. \] (11)

In (1), and (4)–(5), (1) is multiplied by $z$, (4)–(5) ($n = 1, 2, \ldots$) by $z^{n+1}$. Similarly, we get

\[ \left[ \lambda_2 z X(z) - (\lambda_2 + \alpha + \mu)z + \mu \right] Q(z) + \beta z R(z) = \lambda_0 z P_0(0) - rz H_1(z). \] (12)

From (6), we use the same procedure as above to obtain

\[ \alpha Q(z) + \left[ \lambda_3 X(z) - \lambda_3 - \beta \right] R(z) = 0. \] (13)
We solve $Q(z)$ and $R(z)$ from (12)–(13) and use (10)–(11) yielding

\[
Q(z) = z\lambda_0 \left( rI(z) + \lambda_1 \right) \left( X(z) - 1 \right) \left( \lambda_3 + \beta - \lambda_3 X(z) \right)
\times P_0(0) \left\{ \left\{ \lambda_1 + r - \lambda_1 X(z) \right\} \left\{ \left[ 1 - X(z) \right] \left[ z\lambda_2 \left( \lambda_3 (1 - X(z)) + \beta \right) \right]
+ \mu \lambda_3 (z - 1) + z\alpha \lambda_3 \right\} + \mu \beta (z - 1) \right\},
\] (14)

\[
R(z) = z\alpha \lambda_0 \left( rI(z) + \lambda_1 \right) \left( X(z) - 1 \right)
\times P_0(0) \left\{ \left\{ \lambda_1 + r - \lambda_1 X(z) \right\} \left\{ \left[ 1 - X(z) \right] \left[ z\lambda_2 \left( \lambda_3 (1 - X(z)) + \beta \right) \right]
+ \mu \lambda_3 (z - 1) + z\alpha \lambda_3 \right\} + \mu \beta (z - 1) \right\}.
\] (15)

Let $G(z)$ represent the p.g.f. of the number of units in the system; thus

\[
G(z) = H_0(z) + H_1(z) + Q(z) + R(z).
\] (16)

Evaluating $H_0(1), H_1(1), Q(1)$, and $R(1)$ in (10)–(11) and (14)–(15), the numerator and denominator are both 0 in (14)–(15). We apply L’Hopital’s rule once and finally obtain

\[
H_0(1) = I(1) P_0(0),
\] (17)

\[
H_1(1) = \frac{\lambda_0}{r} P_0(0),
\] (18)

\[
Q(1) = \frac{\lambda_0 \beta \left[ rI(1) + \lambda_1 \right] E(X)}{r \mu \beta - r \lambda_2 \beta + \alpha \lambda_3 E(X)} P_0(0),
\] (19)

\[
R(1) = \frac{\lambda_0 \alpha \left[ rI(1) + \lambda_1 \right] E(X)}{r \mu \beta - r \lambda_2 \beta + \alpha \lambda_3 E(X)} P_0(0).
\] (20)

To determine $P_0(0)$, using the normalizing condition finally yields

\[
P_0(0) = \left[ I(1) + \frac{\lambda_0}{r} + \frac{\lambda_0 (\alpha + \beta) \left[ rI(1) + \lambda_1 \right] E(X)}{r \mu \beta - r \lambda_2 \beta + \alpha \lambda_3 E(X)} \right]^{-1},
\] (21)

with $0 < P_0(0) < 1$ is sufficient for stationary.

From equations (10)–(11) and (14)–(16), we can see that the stochastic decomposition property by Fuhrmann and Cooper [6] doesn’t holds for the heterogeneous
It is clear that the p.g.f. of the number of units in the $M^{[x]}/M/1$ queueing system with server breakdowns and startup can be decomposed into two independent terms as

\[ G(z) = \zeta(z) \times G_o(z; M^{[x]}/M/1), \]

where

\[ \zeta(z) = \frac{[\lambda + rI(z)]P_0(0)}{(1 - \rho_1)\lambda + r - \lambda X(z)} = \frac{r[\lambda + rI(z)]}{(\lambda + rI(1))\lambda + r - \lambda X(z)}. \]  

and

\[ G_o(z; M^{[x]}/M/1) = \frac{(1 - \rho_1)(1 - z)\lambda(X(z) - 1) - \beta}{\rho_0 [1 - X(z)] \left[ z\left(\lambda(1 - X(z)) + \beta\right) + \mu(z - 1) + z\alpha\right] + \beta(z - 1)}, \]

with $\rho_0 = \lambda/\mu$, $\rho = \rho_0 E[X]$ and $\rho_1 = \rho(1 + \alpha/\beta)$.

From equation (22), we observe the p.g.f. of the number of units in the $M^{[x]}/M/1$ queueing system with server breakdowns and startup is the convolution of the p.g.f. of two independent random variables one of which is the number of units in the system corresponding to an ordinary $M^{[x]}/M/1$ queueing system with an unreliable server (second term) and the other is the number of arrivals during the residual life of the startup period (first term).

For vacation queues, Lee et al. [11,12] have shown that the stochastic decomposition property holds for the homogeneous arrival in the $N$ policy $M/G/1$ queueing systems with a reliable server. Based on the earlier discussion, it is easily seen that the stochastic decomposition property holds for the homogeneous arrival in the $N$ policy $M^{[x]}/M/1$ queueing system with an unreliable server and startup, too.

### 4. Departure point queue size distribution

In this section we derive the p.g.f. of the limiting queue size distribution at departure point of time. Following the argument of PASTA (see Chaudhry and Templeton [3]) we state that a departing customer will see $'i'$ units in the queue just after his departure if and only if there were $'i' + 1'$ units in the system just
before the departure. Thus we may write
\[ \pi_i = \theta P_i(i + 1), \quad i = 0, 1, 2, \ldots, \]
where \( \pi_i = \Pr[ 'i' \text{ units in the queue just after a departure}] \), the definition of \( P_i(i) \) was given in the Section 3, and \( \theta \) is a constant to be evaluated.

Let \( \Pi_q(z) \) be the p.g.f. of \( \{ \pi_i; i = 0, 1, 2, \ldots \} \), then
\[ \Pi_q(z) = \theta \frac{Q(z)}{z}, \]
where \( Q(z) \) is given in (14).

Now using the normalizing condition; i.e., limit of \( \Pi_q(z) \) as \( z \to 1 \) is unity, we get
\[ \theta = \left\{ \frac{\lambda_0 \beta r I(1) + \lambda_1 E(X)}{r \mu \beta - r [\lambda_2 \beta + \alpha \lambda_3] E(X) P_0(0)} \right\}^{-1}. \]
Thus the p.g.f. of the departure point queue size distribution is given by
\[ \Pi_q(z) = \theta \lambda_0 \left( r I(z) + \lambda_1 \right) \left( X(z) - 1 \right) \left( \lambda_3 + \beta - \lambda_3 X(z) \right) \]
\[ \times P_0(0) / \left\{ \{ \lambda_1 + r - \lambda_1 X(z) \} \left\{ [1 - X(z)] \left[ z \lambda_2 \lambda_3 (1 - X(z)) + \beta \right] + \mu \lambda_3 (z - 1) + z \alpha \lambda_3 \right\} + \mu \beta (z - 1) \right\}. \] (25)

4.1. Some remarks

In particular, if we take \( \lambda_0 = \lambda_1 = \lambda_2 = \lambda_3 = \lambda, \theta = 1/\rho \) and \( \Pi_q(z) \) can be simplified as
\[ \Pi_q(z) = \rho_0 \left( r I(z) + \lambda \right) \left( X(z) - 1 \right) \left( \lambda + \beta - \lambda X(z) \right) P_0(0) / \left\{ \rho \left\{ \lambda + r - \lambda X(z) \right\} \right\} \]
\[ \times \left\{ \rho_0 \left[ 1 - X(z) \right] \left[ z \left[ \lambda (1 - X(z)) + \beta \right] + \mu (z - 1) + z \alpha \right] + \beta (z - 1) \right\}. \] (26)

Now from equation (26) we observe that \( \Pi_q(z) \) decomposes into three independent terms:
\[ \Pi_q(z) = \zeta(z) T(z) \Pi_q(z; M^{[\infty]}/M/1), \] (27)
where \( \zeta(z) \) is given by (23), and

\[
T(z) = \frac{1 - X(z)}{E(X)(1 - z)},
\]

and

\[
\Pi_q(z; M[x]/M/1) = \frac{(1 - \rho_1)(z - 1)[\lambda(1 - X(z)) + \beta]}{\rho_0 \left[ 1 - X(z) \right] \left[ z(\lambda(1 - X(z)) + \beta) + \mu(z - 1) + z\alpha \right] + \beta(z - 1)}.
\]

It is important to be noted that the departure point queue size distribution given by (27) decomposes into three independent random variables: one (the first term) is that the number of units arrive during the residual life of the startup period. Particularly, we may call it queue size distribution due to residual startup period. Another (the second term) is the number of units placed before an arbitrary test unit (tagged unit) in a batch in which the tagged unit arrives (see Takagi [17], p. 46), and the last one (the third term) is the departure point queue size of the ordinary M[x]/M/1 queueing system with an unreliable server.

The above discussions tell us that, the stochastic decomposition property of the departure point queue size holds for the homogeneous arrival in the \( N \) policy M[x]/M/1 queueing system with an unreliable server and startup. If \( \alpha = 0 \) and \( \beta = \infty \), equation (26) can be simplified as

\[
\Pi_q(z) = \frac{\lambda(\lambda + rI(z))(X(z) - 1)P_0(0)}{\rho(\lambda + r - \lambda X(z)) \left[ z\lambda(1 - X(z)) + \mu(z - 1) \right]},
\]

which verifies the p.g.f. of the departure point queue size distribution for the \( N \) policy M[x]/M/1 queueing system with a reliable server and startup.

Suppose that we have \( \alpha = 0 \) and \( \beta = \infty \); then if we put \( N = 1 \) and \( \Pr[X = 1] = 1 \), \( X(z) = z \) and \( E(X) = 1 \) and therefore \( T(z) = 1 \). Hence (26) becomes

\[
\Pi_q(z) = \frac{(\lambda + r)P_0(0)}{(1 - \rho_0 z)(\lambda + r - \lambda z)},
\]

which furnishes the p.g.f. of the departure point queue size distribution for an ordinary M/M/1 queueing system with a reliable server and startup (see Choudhury [4]).

From earlier inferences we have

\[
G_0(z; M[x]/M/1) = \Pi_q(z; M[x]/M/1),
\]

which verifies the results by Cooper [5].
Further, the result by Choudhury [4] as
\[ G_q(z;M^{[s]}/M/1) = \Pi_q(z;M^{[s]}/M/1)[1 + \rho_0(1 - X(z))], \]
where \( G_q(z;M^{[s]}/M/1) \) is the p.g.f. of the number of units in the queue at stationary point in time equilibrium state for the ordinary M\([x]\)/M/1 queueing system with an unreliable server.

5. Expected number of arrivals in the system

Using (10)–(11), and (14)–(16), we compute the mean queue length
\[
L_N = \left\{ I'(1) + \frac{\lambda_0 E(X)}{r^2} \left[ I(1) + \lambda_1 \frac{[\alpha + \beta][rI(1) + \lambda_1][E(X)]}{\mu - (\lambda_2 + \alpha \lambda_3)E(X)} \right] \right. \\
+ 2r \left[ \frac{\lambda_0}{\mu - (\lambda_2 + \alpha \lambda_3)E(X)} \left[ rI(1) + \lambda_1 \right] [\alpha + \beta] E[X(X + 1)] \right] \\
+ 2r I'(1)(\alpha + \beta) E(X) - 2 \lambda_3 [rI(1) + \lambda_1][E(X)]^2 \right. \\
+ \left. \left. 2r I'(1)(\alpha + \beta) E(X) - 2 \lambda_3 [rI(1) + \lambda_1][E(X)]^2 \right] \right) P_0(0), \tag{28}
\]
where \( P_0(0) \) is given is (21).

5.1. Special cases

In this section, we present some existing results in the literature which are special cases of our model.

Case A: If \( N = 1, \lambda_0 = \lambda_1 = \lambda_2 = \lambda_3, b_1 = 1, r = \infty \), and \( \beta = \infty \), the ordinary M/M/1 queueing system with a reliable server case. When \( N = 1, \lambda_0 = \lambda_1 = \lambda_2 = \lambda_3, b_1 = 1, r = \infty \), and \( \beta = \infty \), expression (14) for \( Q(z) \) reduces to a special case of expression (2.14) of Gross and Harris ([7], p. 67).

Case B: If \( N = 1, \lambda_0 = \lambda_1 = \lambda_2 = \lambda_3, r = \infty \), and \( \beta = \infty \), the ordinary M\([x]\)/M/1 queueing system with a reliable server case. When \( N = 1, \lambda_0 = \lambda_1 = \lambda_2 = \lambda_3, b_1 = 1, r = \infty \), and \( \beta = \infty \), expression (14) for \( Q(z) \) reduces to a special case of expression (3.3) of Gross and Harris [7].

Case C: If \( N = 1, \lambda_0 = \lambda_1 = \lambda_2 = \lambda_3, b_1 = 1, \) and \( r = \infty \), the ordinary M/M/1 queueing system with an unreliable server case. When \( N = 1, \lambda_0 = \lambda_1 = \lambda_2 = \lambda_3, b_1 = 1, \) and \( r = \infty \), expressions (14) for \( Q(z) \) and (15) for \( R(z) \) reduce to a special case of expressions (16) and (17) of Wang [19], respectively.
Case D: If $\lambda_0 = \lambda_1 = \lambda_2 = \lambda_3$, $b_1 = 1$, $r = \infty$, $\alpha = 0$, and $\beta = \infty$, the $N$ policy M/M/1 queueing system with a reliable server case. When $\lambda_0 = \lambda_1 = \lambda_2 = \lambda_3$, $b_1 = 1$, $r = \infty$, $\alpha = 0$, and $\beta = \infty$, expression (14) for $Q(z)$ reduces to a special case of expression (4.62) of Sivazlian and Stanfel ([16], p. 255).

Case E: If $\lambda_0 = \lambda_1 = \lambda_2 = \lambda_3$, $b_1 = 1$, and $r = \infty$, the $N$ policy M/M/1 queueing system with an unreliable server case. When $\lambda_0 = \lambda_1 = \lambda_2 = \lambda_3$, $b_1 = 1$, and $r = \infty$, expressions (14) for $Q(z)$ and (15) for $R(z)$ reduce to a special case of expressions (18) and (19) of Wang [20], respectively.

Case F: If $\lambda_0 = \lambda_1 = \lambda_2 = \lambda_3$, $b_1 = 1$, $\alpha = 0$, and $\beta = \infty$, the $N$ policy M/M/1 queueing system with a reliable server case. When $\lambda_0 = \lambda_1 = \lambda_2 = \lambda_3$, $b_1 = 1$, $\alpha = 0$, and $\beta = \infty$, expression (28) for $L_N$ reduces to a special case of expression $L(n)$ of Baker ([1], p. 72).

6. Optimal design of the $N$ policy

Let $O$, $S$, $B$, and $D$ denote the lengths of the server off, startup, busy, and breakdown periods, respectively. Applying the memoryless property of the Poisson process, we find that the mean length of the off period is

$$E[O] = N/\lambda_0.$$  (29)

The expected length of the off period, the startup period, the busy period, and the breakdown period, are denoted by $E[O]$, $E[S]$, $E[B]$ and $E[D]$, respectively. The expected length of a busy cycle is given by


From (17)–(20), we obtain the following long-run fraction of time the server is off, startup, busy, and broken down, respectively:

$$\frac{E[O]}{E[C]} = H_0(1) = I(1)P_0(0),$$  (30)

$$\frac{E[S]}{E[C]} = H_1(1) = \frac{\lambda_0}{r}P_0(0),$$  (31)

$$\frac{E[B]}{E[C]} = Q(1) = \frac{\lambda_0 \beta [rI(1) + \lambda_1]E(X)}{r\mu \beta - r[\lambda_2 \beta + \alpha \lambda_3]E(X)}P_0(0),$$  (32)

$$\frac{E[D]}{E[C]} = R(1) = \frac{\lambda_0 \alpha [rI(1) + \lambda_1]E(X)}{r\mu \beta - r[\lambda_2 \beta + \alpha \lambda_3]E(X)}P_0(0).$$  (33)

Thus, we have the expected number of busy cycles per unit time

$$\frac{1}{E[C]} = \frac{\lambda_0 I(1)P_0(0)}{N}.\quad (34)$$
6.1. Determining the optimal policy

We develop the total expected cost function per unit time for the $M^k/[M]/1$ queue under the $N$ policy with server breakdowns and startup time, in which $N$ is a decision variable. Following the cost structure is constructed, our objective is to determine the optimal operating $N$ policy so as to minimize this function. Let

- $C_h \equiv$ holding cost per unit time for each unit present in the system;
- $C_f \equiv$ setup cost per busy cycle;
- $C_o \equiv$ startup cost per unit time for keeping the server off;
- $C_s \equiv$ startup cost per unit time for the preparatory work of the server before starting the service;
- $C_b \equiv$ cost per unit time for keeping the server on and in operation;
- $C_d \equiv$ breakdown cost per unit time for a broken server.

Using the definitions of each cost element and its corresponding system characteristics, the total expected cost function per unit time is given by

$$T_{\text{cost}}(N) = C_h L_N + \frac{C_f}{E[C]} + C_o \frac{E[O]}{E[C]} + C_s \frac{E[S]}{E[C]} + C_b \frac{E[B]}{E[C]} + C_d \frac{E[D]}{E[C]}$$  \hspace{1cm} (35)

Using (28), (30)–(34), the results of (35) can be explicitly expressed which is a very long and complex formula for $T_{\text{cost}}(N)$. This is due to the fact that there are many parameters (e.g., $X$, $\lambda_0$, $\lambda_1$, $\lambda_3$, $r$, $\mu$, $\alpha$, and $\beta$) involved in our model. We obtain the optimal value $N$ which minimizes the cost function, $T_{\text{cost}}(N)$, by differentiating it with respect to $N$ and setting the result to be zero, i.e.,

$$\frac{\partial T_{\text{cost}}(N)}{\partial N} = 0.$$  \hspace{1cm} (36)

The solution $N$ to (36) may not be integer, and the optimal positive integer value of $N$ is one of the integers surrounding $N^*$ which gives a smaller cost $T_{\text{cost}}$. Here, it should be pointed out explicitly that the solution really gives the minimum value, and the $\frac{\partial^2 T_{\text{cost}}(N)}{\partial N^2}$ is greater than 0 when the values of system parameters satisfy suitable conditions. However, it is quite tedious to present the explicitly expression. Therefore, we will perform the numerical experiments to demonstrate that the function is really convex and the solution gives a minimum.

6.2. Numerical studies

We now perform a sensitivity analysis on the optimum value $N^*$ based on changes in specific values of the system parameters. Let the batch size $X$ be a geometric distribution with parameter $p$, and employ the following cost elements:

- Case 1: $C_h = 5$, $C_o = 10$, $C_b = 100$, $C_d = 200$, $C_s = 125$, $C_f = 500$.
- Case 2: $C_h = 5$, $C_o = 20$, $C_b = 200$, $C_d = 400$, $C_s = 250$, $C_f = 500$.
- Case 3: $C_h = 5$, $C_o = 20$, $C_b = 200$, $C_d = 400$, $C_s = 500$, $C_f = 500$.
- Case 4: $C_h = 5$, $C_o = 20$, $C_b = 200$, $C_d = 400$, $C_s = 500$, $C_f = 1000$.
- Case 5: $C_h = 10$, $C_o = 20$, $C_b = 200$, $C_d = 400$, $C_s = 500$, $C_f = 1000$. 
Table 1. The optimal value of \( N \) and its minimum expected cost for geometric batch size \((X \sim \text{Geo}(p))\) and \((p, \lambda_0, \lambda_1, \lambda_2, \lambda_3, \mu, \beta) = (0.55, 0.3, 0.4, 0.5, 0.2, 1.2, 3.0)\).

<table>
<thead>
<tr>
<th>Case</th>
<th>( N^* )</th>
<th>( T_{\text{cost}}(N^*) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>137.064</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
<td>216.592</td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>230.151</td>
</tr>
<tr>
<td>4</td>
<td>9</td>
<td>234.343</td>
</tr>
<tr>
<td>5</td>
<td>6</td>
<td>290.849</td>
</tr>
</tbody>
</table>

Table 2. The optimal value of \( N \) and its minimum expected cost for geometric batch size \((X \sim \text{Geo}(p))\) and \((p, \lambda_0, \lambda_1, \lambda_2, \lambda_3, r, \alpha) = (0.55, 0.3, 0.4, 0.5, 0.2, 0.2, 0.05)\) and for various values of \((\mu, \beta)\). From Table 2, we observe that (i) \( T_{\text{cost}}(N^*) \) increases as \( \mu \) and \( \beta \) decrease for any cases; (ii) \( N^* \) increases as \( \mu \) increases for any cases; and (iii) \( N^* \) does not change at all when \( \beta \) changes from 2.0 to 5.0 for any cases. Intuitively, \( N^* \) is insensitive to changes in \( \beta \).

The optimal value of \( N, N^* \), and its minimum expected cost \( T_{\text{cost}}(N^*) \) for the above five cases are shown in Table 1 for \((p, \lambda_0, \lambda_1, \lambda_2, \lambda_3, \mu, \beta) = (0.55, 0.3, 0.4, 0.5, 0.2, 1.2, 3.0)\) and for various values of \((r, \alpha)\). One observes from Table 1 that (i) \( T_{\text{cost}}(N^*) \) increases as \( r \) decreases or \( \alpha \) increases for any cases; (ii) for a large value \( r \), \( N^* \) does not change at all when \( r \) changes from 0.5 to 0.7 for any cases; and (iii) \( N^* \) does not change at all when \( \alpha \) changes from 0.1 to 0.3 for any cases. Intuitively, \( N^* \) is insensitive to changes in \( \alpha \).

The optimal value of \( N, N^* \), and its minimum expected cost \( T_{\text{cost}}(N^*) \) for five cost cases are shown in Table 2 for \((p, \lambda_0, \lambda_1, \lambda_2, \lambda_3, r, \alpha) = (0.55, 0.3, 0.4, 0.5, 0.2, 0.2, 0.05)\) and for different values of \((\mu, \beta)\). From Table 2, we observe that (i) \( T_{\text{cost}}(N^*) \) increases as \( \mu \) and \( \beta \) decrease for any cases; (ii) \( N^* \) increases as \( \mu \) increases for any cases; and (iii) \( N^* \) does not change at all when \( \beta \) changes from 2.0 to 5.0 for any cases. Intuitively, \( N^* \) is insensitive to changes in \( \beta \).
Table 3. The optimal value of \( N \) and its minimum expected cost for geometric batch size \((X \sim \text{Geo}(p))\) and \((p, \lambda_2, \lambda_3, r, \mu, \alpha, \beta) = (0.45, 0.6, 0.2, 0.2, 2.0, 0.05, 3.0)\).

<table>
<thead>
<tr>
<th>Case</th>
<th>( N^* )</th>
<th>( T_{\text{cost}}(N^*) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>7</td>
<td>136.445</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>200.510</td>
</tr>
<tr>
<td>3</td>
<td>14</td>
<td>219.954</td>
</tr>
<tr>
<td>4</td>
<td>15</td>
<td>223.307</td>
</tr>
<tr>
<td>5</td>
<td>9</td>
<td>298.267</td>
</tr>
</tbody>
</table>

Table 4. The optimal value of \( N \) and its minimum expected cost for geometric batch size \((X \sim \text{Geo}(p))\) and \((p, \lambda_0, \lambda_1, r, \mu, \alpha, \beta) = (0.45, 0.3, 0.4, 0.2, 2.0, 0.05, 3.0)\).

<table>
<thead>
<tr>
<th>Case</th>
<th>( N^* )</th>
<th>( T_{\text{cost}}(N^*) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>4</td>
<td>140.846</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>205.135</td>
</tr>
<tr>
<td>3</td>
<td>16</td>
<td>233.780</td>
</tr>
<tr>
<td>4</td>
<td>17</td>
<td>238.461</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>320.243</td>
</tr>
</tbody>
</table>

The optimal value of \( N, N^* \), and its minimum expected cost \( T_{\text{cost}}(N^*) \) for five cases are shown in Table 3 for \((p, \lambda_2, \lambda_3, r, \mu, \alpha, \beta) = (0.45, 0.6, 0.2, 0.2, 2.0, 0.05, 3.0)\) and for various values of \((\lambda_0, \lambda_1)\). One observes from Table 3 that (i) \( T_{\text{cost}}(N^*) \) increases as \(\lambda_0\) and \(\lambda_1\) increase for any cases; (ii) \( N^* \) slightly changes when \(\lambda_0\) changes from 0.2 to 0.5 for any cases; and (iii) \( N^* \) does not change at all when \(\lambda_1\) changes from 0.3 to 0.6 for any cases. Intuitively, \( N^* \) is insensitive to changes in \(\lambda_1\).

The optimal value of \( N, N^* \), and its minimum expected cost \( T_{\text{cost}}(N^*) \) are shown in Table 4 for \((p, \lambda_0, \lambda_1, r, \mu, \alpha, \beta) = (0.45, 0.3, 0.4, 0.2, 2.0, 0.05, 3.0)\) and for different values of \((\lambda_2, \lambda_3)\). From Table 4, we find that (i) \( T_{\text{cost}}(N^*) \) increases as \(\lambda_2\) and \(\lambda_3\) increase for any cases; (ii) \( N^* \) increases as \(\lambda_2\) decreases for any cases; and (iii) \( N^* \) and \( T_{\text{cost}}(N^*) \) do not change at all when \(\lambda_3\) changes from 0.1 to 0.6 for any cases. Intuitively, \( N^* \) is insensitive to changes in \(\lambda_3\).
The optimal value of $N$, $N^*$, and its minimum expected cost $T_{\text{cost}}(N^*)$ are shown in Table 5 for $(p, \lambda_0, \lambda_3, r, \mu, \alpha, \beta) = (0.35, 0.3, 0.2, 0.2, 2.0, 0.05, 3.0)$ and for different values of $(\lambda_1, \lambda_2)$. Table 5 depicts that (i) $T_{\text{cost}}(N^*)$ increases as $\lambda_1$ and $\lambda_2$ increase for any cases; (ii) $N^*$ increases as $\lambda_2$ decreases for any cases; and (iii) $N^*$ slightly changes when $\lambda_1$ changes from 0.3 to 0.6 for any cases.

Furthermore, we choose the uniform batch size (set $X \equiv U(1, 4)$). The numerical results for the optimal value $N$, $N^*$, and its minimum expected cost $T_{\text{cost}}(N^*)$ are shown in Table 6 for $(\lambda_0, \lambda_3, r, \mu, \alpha, \beta) = (0.3, 0.2, 0.2, 2.0, 0.05, 3.0)$ and for different values of $(\lambda_1, \lambda_2)$. Table 6 shows that (i) $T_{\text{cost}}(N^*)$ increases as $\lambda_1$ and $\lambda_2$ increase for any cases; (ii) $N^*$ increases as $\lambda_2$ decreases for any cases; and (iii) $N^*$ slightly changes when $\lambda_1$ changes from 0.3 to 0.6 for any cases.

It can be easily see from Table 1 through 6 that (i) $N^*$ increases as $C_s$ increases or $C_h$ decreases (see Case 2-3 and Case 4-5); and (ii) $C_h$ and $C_s$ have a larger effect on $N^*$ than $C_\ell$ (see Case 3-4). Tables 5 and 6 indicate that $X$ affects $N^*$.
From our numerical investigations, it appears that (i) $\alpha$, $\beta$, and $\lambda_3$ do not affect $N^*$; (ii) $r$, $\lambda_0$ and $\lambda_1$ rarely affect $N^*$; and (iii) $\lambda_2$ and $\mu$ affect $N^*$ significantly. It is interesting that $C_h$ and $C_s$ have much stronger effect on $N^*$ than $X$, $\lambda_0$, $\lambda_1$, $\lambda_2$, $\lambda_3$, $r$, $\mu$, $\alpha$, and $\beta$.

7. Conclusions

In this paper, we have developed the analytic closed-form solutions for the $M^{[x]}/M/1$ queueing system with server breakdowns and startup time under the $N$ policy. More especially, the stochastic decomposition property of state-steady probabilities and departure point queue size distribution has been investigated and some important remarks have been given. The model is very useful for real systems since the behavior of arriving units is considered. Usually, the problems related to the Markovian queueing systems are treated as special cases of this model. We also have performed a sensitivity analysis among the optimal value of $N$, specific values of system parameters, and the cost elements. Through the numerical results, we were able to analyze the complex but exact solutions for a practical and general queueing system, make an intelligent decision based on quantitative measures.

Appendix 1. The derivations of $H_1(z)$

Equation (2) is multiplied by $z^n (n = 1, 2, \ldots, N - 1)$ and are expanded as follows:

\[
\begin{align*}
    z\lambda_0 P_0(1) &= z\lambda_0 b_1 P_0(0), \\
    z^2\lambda_0 P_0(2) &= z^2\lambda_0 b_2 P_0(0) + z^2\lambda_0 b_1 P_0(1), \\
    z^3\lambda_0 P_0(3) &= z^3\lambda_0 b_3 P_0(0) + z^3\lambda_0 b_2 P_0(1) + z^3\lambda_0 b_1 P_0(2), \\
    & \vdots \\
    z^{N-1}\lambda_0 P_0(N-1) &= z^{N-1}\lambda_0 b_{N-1} P_0(0) + z^{N-1}\lambda_0 b_{N-2} P_0(2) + \ldots + z^{N-1}\lambda_0 b_1 P_0(N-2).
\end{align*}
\]

The equations listed above are added terms by terms. Thus we have

\[
\lambda_0 \sum_{n=1}^{N-1} z^n P_0(n) = \lambda_0 P_0(0) \sum_{n=1}^{N-1} z^n b_n + \lambda_0 z P_0(1) \sum_{n=1}^{N-2} z^n b_n + \lambda_0 z^2 P_0(2) \sum_{n=1}^{N-3} z^n b_n + \ldots + \lambda_0 z^{N-2} P_0(N-2) \sum_{n=1}^{N-1} z^n b_n,
\]
or equivalently

\[
\lambda_0 H_0(z) - \lambda_0 P_0(0) = \lambda_0 P_0(0) \sum_{n=1}^{N-1} z^n b_n + \lambda_0 z P_0(1) \sum_{n=1}^{N-2} z^n b_n + \lambda_0 z^2 P_0(2) \sum_{n=1}^{N-3} z^n b_n \\
+ \cdots + \lambda_0 z^{N-2} P_0(N-2) \sum_{n=1}^{1} z^n b_n. \tag{A.1}
\]

Similarly, equation (3) is multiplied by \(z^n(z = N, N + 1, \ldots)\) and are expanded as follows:

\[
z^N(\lambda_1 + r) P_0(N) = z^N \lambda_0 b_N P_0(0) + z^N \lambda_0 b_{N-1} P_0(1) + \cdots + z^N \lambda_0 b_1 P_0(N - 1),
\]

\[
z^{N+1}(\lambda_1 + r) P_0(N+1) = z^{N+1} \lambda_0 b_{N+1} P_0(0) + z^{N+1} \lambda_0 b_N P_0(1) + \cdots + z^{N+1} \lambda_0 b_2 P_0(N - 1) + z^{N+1} \lambda_1 P_0(N),
\]

\[
z^{N+2}(\lambda_1 + r) P_0(N+2) = z^{N+2} \lambda_0 b_{N+2} P_0(0) + z^{N+2} \lambda_0 b_{N+1} P_0(1) + \cdots + z^{N+2} \lambda_0 b_3 P_0(N - 1) + z^{N+2} \lambda_1 b_2 P_0(N) + z^{N+2} \lambda_1 P_0(N + 1),
\]

The equations listed above are added terms by terms. Thus we have

\[
(\lambda_1 + r) \sum_{n=N}^{\infty} z^n P_0(n) = \lambda_0 P_0(0) \sum_{n=N}^{\infty} z^n b_n + \lambda_0 z P_0(1) \sum_{n=N-1}^{\infty} z^n b_n \\
+ \lambda_0 z^2 P_0(2) \sum_{n=N-2}^{\infty} z^n b_n + \cdots + \lambda_0 z^{N-2} P_0(N-2) \sum_{n=2}^{\infty} z^n b_n \\
+ \lambda_0 z^{N-1} P_0(N-1) \sum_{n=1}^{\infty} z^n b_n + \lambda_1 z^{N} P_0(N) \sum_{n=1}^{\infty} z^n b_n \\
+ \lambda_1 z^{N+1} P_0(N+1) \sum_{n=1}^{\infty} z^n b_n + \cdots,
\]
or equivalently

\[(\lambda_1 + r)H_1(z) = \lambda_0 P_0(0) \sum_{n=N}^{\infty} z^n b_n + \lambda_0 z P_0(1) \sum_{n=N-1}^{\infty} z^n b_n\]

\[+ \lambda_0 z^2 P_0(2) \sum_{n=N-2}^{\infty} z^n b_n + \cdots + \lambda_0 z^{N-2} P_0(N-2) \sum_{n=2}^{\infty} z^n b_n\]

\[+ \lambda_0 z^{N-1} P_0(N-1) \sum_{n=1}^{\infty} z^n b_n + \lambda_1 z^N P_0(N) \sum_{n=1}^{\infty} z^n b_n\]

\[+ \lambda_1 z^{N+1} P_0(N+1) \sum_{n=1}^{\infty} z^n b_n + \ldots \quad (A.2)\]

Adding (A.1) to (A.2) and rearranging some terms, it finally yields

\[[(\lambda_1 + r) - \lambda_1 X(z)]H_1(z) = \lambda_0 [X(z) - 1]H_0(z) + \lambda_0 P_0(0),\]

which gets (11).

**References**


To access this journal online:
www.edpsciences.org