RAIRO-Oper. Res. 60 (2026) 1439–1473
https://doi.org/10.1051/ro/2026040

RAIRO Operations Research
www.rairo-ro.org

PERFORMANCE ANALYSIS OF AN $M/G/1$ RETRIAL G-QUEUE WITH BALKING, OPTIONAL SERVICE AND GENERAL RETRIAL POLICY

KM Rashmi and Sujit Kumar Samanta*

Abstract. This study examines the steady-state dynamics of an $M/G/1$ retrial queue that incorporates features such as balking, optional service, negative customer, and general retrial time under a constant retrial policy. The probability generating functions for the orbit length with server status are obtained using the supplementary variable technique at various time epochs. We further extract the orbit size probabilities from these generating functions using the roots method. Additionally, the tail distribution of the orbit length is also presented for different time epochs. We conduct a comprehensive cost analysis to minimize the overall system operation cost. Advanced metaheuristic techniques such as particle swarm optimization and genetic algorithm are employed to determine the optimal system parameters that minimize the total cost. The impact of various system parameters on performance measures is demonstrated through detailed numerical examples.

Mathematics Subject Classification. 90B22, 60K25, 68M20.

Received March 11, 2025. Accepted April 7, 2026.

1. Introduction

A retrial queueing system does not allow customers to wait in front of the server. Instead, customers who find the server busy must leave the service area and return after a random amount of time. During this interval, these customers are considered to join a virtual waiting area known as the orbit. A practical example of a retrial queue can be observed in the landing process of airplanes at busy airports. When an airplane arrives at an airport, it sends a request for permission to land. If a runway is available, the airplane lands immediately. However, if all runways are occupied, the airplane cannot land and is instructed to enter a holding pattern in the sky (similar to the orbit in a retrial queueing system). The airplane remains in the holding pattern and retries landing by continuously contacting the air traffic control room for permission. The process persists until a runway becomes available, after which the airplane receives permission to land. This scenario illustrates the key characteristics of retrial queues, where customers (airplanes) that do not receive service on their first attempt may retry for service after some waiting time. Retrial queues are widely applicable in various other industrial scenarios such as telecommunications, call centers, inventory management, and cellular radio networks, among others. Retrial queueing systems have attracted significant attention from both academia and industry due to their practical relevance. For a comprehensive analysis of significant results and key methodologies related to

Keywords. Balking, negativecustomer, non-exponential retrial time, optional service, retrial queue.

Department of Mathematics, National Institute of Technology Raipur, Raipur 492010, Chhattisgarh, India.
*Corresponding author: sksamanta.maths@nitrr.ac.in

this topic, readers are encouraged to read the book by Falin [16], the survey paper by Kim and Kim [27], as well as the bibliography by Artalezo [3] and the articles by Gao and Wang [17], Liu *et al.* [30], Madheswari *et al.* [32], Rashmi and Samanta [42], Sanga and Antala [40], Sanga and Jain [41], Xu *et al.* [46] and additional references cited therein.

1.1. Literature review of *G*-queues

In recent years, researchers have studied an innovative type of queueing network that classifies customers into two distinct categories: positive customers and negative customers. A queueing system incorporating negative customers known as a G-queue was first introduced by Gelenbe [18] in the context of neural network research. Positive customers behave like traditional queueing customers; they arrive at the system, join the queue, and leave after receiving service. In contrast, negative customers do not require service, but it affects the system at the time of arrival as follows: If the server is idle, unavailable, or on vacation, the negative customer simply vanishes upon arrival. Otherwise, the negative customer impacts the queue differently depending on predefined rules. The three main rules are: (i) The negative customer removes all customers in the system. (ii) The negative customer eliminates only the customer at the front of the queue including one currently being served. (iii) The negative customer removes only the last customer in the queue. The scenario in which an arriving negative customer removes only the customer at the head of the system is particularly effective for modeling server breakdowns, where a negative arrival leads to the immediate loss of the customer currently in service. For instance, imagine a banking server processing online transactions, where positive customers represent genuine users performing tasks like money transfers or bill payments. Meanwhile, a negative customer could be any form of cyberattack. Such cyberattacks disrupt the system which lead to the server breakdown, and terminate ongoing transactions, which ultimately affects the overall service performance. Negative arrivals are widely utilized as control mechanisms in various telecommunication and computer networks, as discussed in the work of Morfopoulou [36]. These negative customers can represent phenomena such as breakdowns, call losses, killing signals, virus attacks in computers, and load balancing. For an in-depth analysis of queueing systems with negative arrivals, readers may go through the foundational studies of Gelenbe [19], Artalejo [2], Atencia and Moreno [5], and the references therein. For further insights into retrial queues involving negative customers, particularly systems that incorporate server breakdowns and repairs, refer to the research conducted by Liu *et al.* [29] and Wu and Yin [44].

1.2. Literature review of retrial queue with optional phase service

One of the key characteristics of a queueing system is the service process. The server may offer two types of services: the regular service, which is required for all customers, and the optional service, which customers may choose to take or forgo before leaving the system. For instance, consider an online food delivery service like Zomato or Swiggy. A customer places an order for food, which represents the regular service. Once the food is delivered, the app prompts the customer with an option to rate the delivery or provide feedback. This feedback option serves as an optional service, allowing the customer to either share their experience or skip the step without affecting the primary service. Another example of optional service can be observed at a mobile recharge center, where the server offers to activate a caller tune service after completing the recharge process. Here, the recharge represents the main service, while the caller tune activation remains an optional service that the customer can either accept or decline. Many scholars have explored this area during the past decades. For instance, Madan [31] examined the $M/G/1$ queue in which the server offers two distinct types of services. The initial regular service is required for every customer and follows a general distribution. The second service is available only upon request and has an exponential distribution. Building on Madan's work [31], Medhi [34] extended the model by assuming that both types of services follow general distributions. Further, Choudhary [10] undertook a comprehensive examination of this model. Kumar *et al.* [28] explored an $M/G/1$ retrial queue with an additional phase of service. Within this framework, the server may interrupt and displace a customer to commence serving another customer with greater priority during the initial phase of service. The interrupted

customers enter a retrial group, where they seek service according to the first-come-first-served (FCFS) principle. The customer at the front of the queue receives the first opportunity to reach the server during the initial phase of service. Additionally, Artalejo and Choudhury [4] investigated an $M/G/1$ retrial queue including optional service in which retrials are regulated by the traditional retrial policy. The motivation for examining these models arises from their applicability in computer and communication networks, where messages undergo processing in two phases by single server. Doshi [14] recognized the relevance of these models in distributed systems, where central server manages the control of two-phase execution. Further, Choudhury [11] studied an $M/G/1$ retrial queue with an additional phase of service, where the retrial times follow general distribution. Further, Arivudainambi and Godhandaraman [1] examined a retrial queueing system with optional service and vacations incorporating general retrial policy. They considered that an arriving customer may balk (decide not to join the system) if the server is either on vacation or busy.

1.3. Literature review of general retrial policy

Retrial queueing systems with non-exponential retrial times and general service times have been extensively studied over the years. The field gained significant traction after the pioneering work of Kapyrim [25], who first introduced the concept of general retrial times under a constant retrial policy. However, the methodology proposed by Kapyrim [25] was later critiqued by Falin [15], who identified a flaw that required correction. Building on this flaw, Yang *et al.* [47] proposed an approximation method to evaluate the steady-state performance of the model introduced by Kapyrim [25]. Subsequently, Gómez-Corral [21] conducted a detailed analysis of an $M/G/1$ retrial queue with the FCFS discipline and general retrial times. In recent years, numerous researchers have examined various retrial queueing models incorporating general retrial times. Readers interested in a broader perspective on these developments are encouraged to explore works such as Choudhury [11], GanaSekar and Kandiaiyan [20], Jain and Sanga [24], Meziani and Kernane [35], Dimitriou [13], Harini and Indhira [23], Mathavavisakan [33], Murugan and Pauline Veronica [6] and Xu *et al.* [45] as well as additional references cited therein.

1.4. Motivation and key contribution of the paper

Researchers have conducted numerous studies on retrial queueing systems. However, as far as the authors knowledge, there has been no significant research work on single server retrial queue that incorporates balking, second phase of service, negative customers, and general retrial distribution under constant retrial policy. To address this, we investigate the steady-state behavior of an $M/G/1$ retrial queue that incorporates these features. The key contributions of this study are as follows:

(i) We derive differential-difference equations at random epoch using the supplementary variables like the elapsed retrial time, the time spent in the first and second phases of service, and the elapsed repair time. These equations provide the foundation for determining the probability generating functions that allow us to extract the orbit length distributions by analyzing the roots of the corresponding characteristic equation.
(ii) We establish stability criteria for the system to ensure its proper functioning.
(iii) Applying the Poisson Arrivals See Time Averages (PASTA) principle, we relate post-departure and random epochs that enable us to determine the orbit length distribution at post-departure epoch.
(iv) We derive several performance measures that offer insights into the operational efficiency of the system.
(v) A detailed cost analysis is performed to assess the economic implications of the system's performance that provides valuable insights into its practical applications.
(vi) Advanced metaheuristic algorithms such as particle swarm optimization and the genetic algorithm are employed in this study for total cost optimization.
(vii) Through comprehensive numerical analyses, we evaluate the impact of varying retrial time, service times, and repair time distributions on various performance measures. These analyses validate the computational efficiency of our approach and the accuracy of our theoretical findings.

1.5. Organisation of the article

The structure of the paper is as follows. Section 2 provides details of the model. Sections 3 and 4 present the orbit length distributions at random and post-departure epochs, respectively. Section 5 derives the relevant performance measures along with a discussion of some important special cases of the model. Section 6 offers extensive numerical results with cost analysis. Finally, Section 7 concludes the paper with a summary and future scope.

2. Model description and its practical application

This article examines an $M/G/1$ retrial G-queue that incorporates general retrial policy, an optional service, and balking behavior of customers.

Arrival process: We take into account two independent types of customer arrivals: positive customers and negative customers. Positive customers (or primary customers) arrive at the system following the Poisson process with rate λ_1, while negative customers follow separate Poisson process with rate λ_2.

Service mechanism: A single server offers the first phase of service (FPS) to all arriving customers. Following the completion of the FPS, customers have the option to proceed to a second phase of service (SPS) with probability p, or they may leave the system with probability $1-p$.

Balking behaviour: If a positive customer finds the server busy (engaged in FPS or SPS) or under repair, the customer may either balk with probability $1-b$ or join the orbit with probability b.

Retrial mechanism: Retrial customers are placed in the orbit on the first-come-first-served basis, meaning the oldest retrial customer is given the first opportunity to access the server for service. In particular, if the customer fails to access the server, its orbital position remains unchanged. A retrial customer can request service only when the server is idle.

Customer removal and server repair mechanism: If an arriving negative customer finds the server busy with a positive customer, it forces the currently served customer out of the system and causes the server to breakdown. In the event of a breakdown, the server is immediately sent for repair. Once repaired, the server is restored to its original state and is as good as new. After the repair is completed, the server becomes idle and is ready to begin the first phase service. If the server is idle or already under repair when a negative customer arrives, the negative customer leaves without affecting the system. The pictorial representation of the model is given in Figure 1.

The retrial time, first phase service time, second phase service time, and repair time all follow arbitrary distributions. All the stochastic processes used in this model are assumed to be independent of each other. Moreover, the service rates of the first and second phases are denoted by κ_1 and κ_2, the retrial rate by α, and the repair rate by δ. The probability density function (p.d.f.), cumulative distribution function (c.d.f.), Laplace-Stieltjes transform (L.-S.T.), as well as the mean of the first phase service time, second phase service time, retrial time, and repair time are given in Table 1.

Table 1. p.d.f., c.d.f., L.-S.T., and mean of service, retrial, and repair times.

Time	p.d.f.	c.d.f.	L.-S.T.	Mean
FPS	$b_1(x)$	$B_1(x)$	$\widetilde{B}_1(s)$	$E[B_1]=1/\kappa_1$
SPS	$b_2(x)$	$B_2(x)$	$\widetilde{B}_2(s)$	$E[B_2]=1/\kappa_2$
Retrial	$a(x)$	$A(x)$	$\widetilde{A}(s)$	$E[A]=1/\alpha$
Repair	$h(x)$	$H(x)$	$\widetilde{H}(s)$	$E[H]=1/\delta$

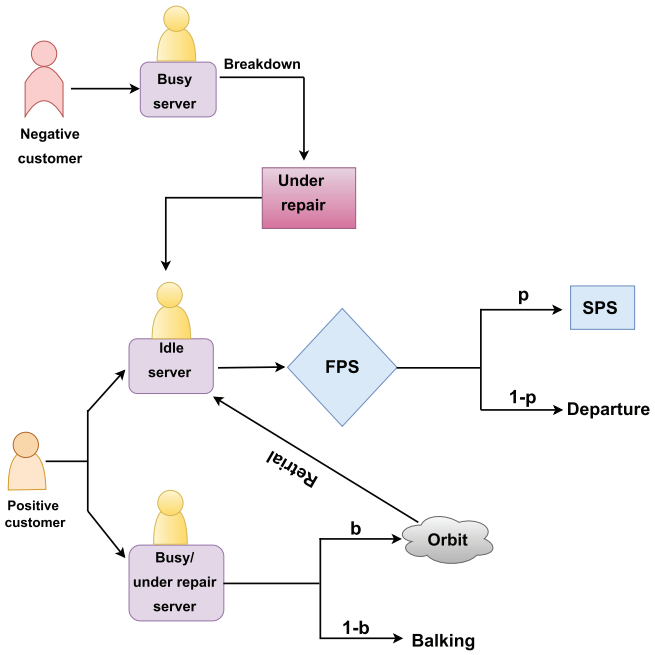

FIGURE 1. Pictorial representation of the model.

TABLE 2. Notations for the different parameters of the system.

Parameter	Description
λ_1	Arrival rate of positive customer
λ_2	Arrival rate of negative customer
p	Probability of opting SPS
b	Probability of joining the orbit

The functions $r(x)$, $\mu_1(x)$, $\mu_2(x)$ and $\eta(x)$ are the conditional completion rates for retrial, FPS, SPS and repair, respectively, at time x. They are expressed as

$$r(x) = \frac{a(x)}{1 - A(x)}, \quad \mu_1(x) = \frac{b_1(x)}{1 - B_1(x)},$$
$$\mu_2(x) = \frac{b_2(x)}{1 - B_2(x)}, \quad \eta(x) = \frac{h(x)}{1 - H(x)}.$$

The traffic load, denoted as ρ, is calculated as $\rho = \lambda_1(E[B_1] + pE[B_2])$. Moreover, the notations for the different parameters of the system are provided in Table 2.

2.1. Practical applications of the model

Aadhaar center

Our queueing model can be effectively useful for the functioning of an Aadhaar Center in India. Customers arrive at the center for either a new Aadhaar card or updating their Aadhaar details, which constitutes the essential (first-phase) service. Additionally, Aadhaar centers offer optional services such as Aadhaar-linked

money withdrawal or fund transfers through the Aadhaar-enabled Payment System (AePS). Customers may choose to avail themselves of these optional services after completing the essential one with probability p. If the center is overcrowded or the waiting time is too long, a customer may either leave and return later with probability b (joining the retrial queue) or decide not to return with probability $1 - b$ (balking). Each retrial customer enters the orbit queue, where the earliest request is given priority for the next retrial. The server, represented by the computer system processing Aadhaar applications, is susceptible to cyberattacks (negative customers). If such an attack occurs, it can corrupt the system, which leads to service disruptions and the loss of ongoing processing data. In such cases, the system requires repair and in the meantime, new arrivals may either enter the retrial queue or balk due to unavailability of service. This analogy closely aligns with our retrial queueing model that incorporates balking, retrial behavior, optional service, and system failures due to negative customer.

Post office service system

Consider a post office scenario where customers arrive to send letters or parcels. In addition to postal services, the post office offers optional banking services through the India Post Payments Bank (IPPB) and Post Office Savings Bank (POSB). After completing their primary service (sending letters or parcels), a customer may choose to avail themselves of banking services with probability p or leave the system without opting for any additional service with probability $1 - p$. The computer system acts as the server, processing both postal and banking transactions. However, the server is vulnerable to breakdowns due to virus attacks or technical failures, which results in the deletion of the currently processed file. Once the server breaks down, it is sent for repair, during which no service can be provided. If a customer arrives and finds the server busy or under repair, they may decide to retry for service after some time and join the orbit with probability b. Alternatively, the customer may choose to leave the system permanently without retrying with probability $1 - b$. Moreover, the orbit operates under the FCFS discipline and allow the retrial request that entered earliest to attempt service first. This example is a practical implementation of the theoretical model discussed in this paper since it closely aligns with the retrial queueing model with the features of balking, optional service, and negative customer.

3. Orbit length distribution at random epoch

This section focuses on analyzing the orbit length distribution at random epoch. To achieve this, we first derive the steady-state equations at random epoch by considering the elapsed retrial time, the elapsed service times in FPS and SPS, and the elapsed repair time as supplementary variables. These equations are used to determine the steady-state probability distribution at random epoch as well as various performance measures through the probability generating function approach. Let us define the state of the system using the following random variables:

- $N(t)$: The number of customers present in the retrial group at time t.
- $\xi(t)$: The state of the server at time t, where

$$\xi(t) = \begin{cases} 0, & \text{when the server is idle,} \\ 1, & \text{when the server is busy in FPS,} \\ 2, & \text{when the server is busy in SPS,} \\ 3, & \text{when the server is under repair.} \end{cases}$$

- If $\xi(t) = 0$ and $N(t) \geq 1$, then $\gamma_0(t)$ denotes the elapsed retrial time of a customer in the retrial group at time t.
- If $\xi(t) = 1$ and $N(t) \geq 0$, then $\gamma_1(t)$ represents the elapsed service time in FPS at time t.
- If $\xi(t) = 2$ and $N(t) \geq 0$, then $\gamma_2(t)$ signifies the elapsed service time in SPS at time t.
- If $\xi(t) = 3$ and $N(t) \geq 0$, then $\gamma_3(t)$ expresses the elapsed repair time of the server at time t.

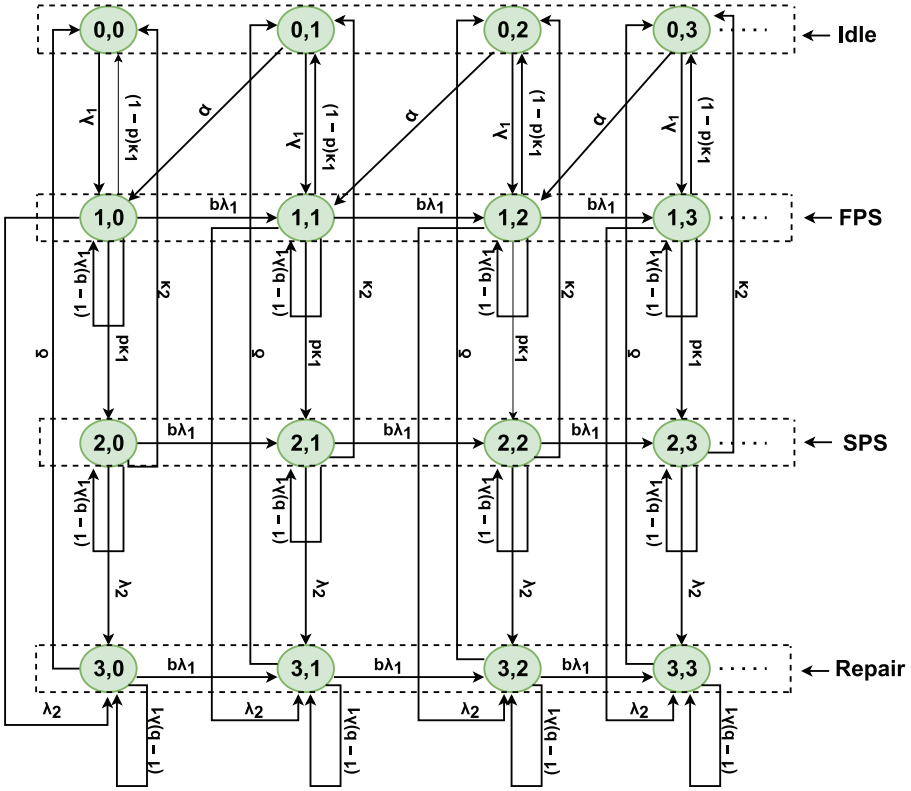

FIGURE 2. State transition diagram.

Thus, $X(t) = \{\xi(t), N(t), \gamma_{\xi(t)}(t)\}_{t\geq 0}$ is a Markov process. We neglect $\gamma_{\xi(t)}(t)$ and consider the pair $Y(t) = \{\xi(t), N(t)\}$ to represent the transition diagram of $Y(t)$ which is provided in Figure 2. Note that the detailed service, retrial, and repair mechanisms are illustrated in Figure 1. Figure 2 provides the corresponding state transition diagram, where only the transition rates between states are shown. Each transition corresponds to a mechanism described in Figure 1. Here, (i) λ_1 denotes the arrival of a positive customer who proceeds to service; (ii) $b\lambda_1$ indicates that, upon arrival of a positive customer, the customer moves to the orbit, whereas $(1-b)\lambda_1$ indicates that the customer balks and leaves the system if the server is busy; (iii) a retrial from the orbit occurs at rate α; (iv) service completions take place at rates κ_1 (FPS) and κ_2 (SPS); (v) $p\kappa_1$ indicates that, upon completion of FPS service, the customer moves to SPS, whereas $(1-p)\kappa_1$ indicates that the customer departs the system after FPS service; (vi) negative arrivals (λ_2) trigger both customer removal and server breakdown; and (vii) the server is restored at rate δ.

Further, let us define their joint probabilities in the steady-state as

$$
\begin{aligned}
I_0 &= \lim_{t\to\infty} Pr[\xi(t) = 0,\ N(t) = 0],\\
I_n(x)\,\mathrm{d}x &= \lim_{t\to\infty} Pr[\xi(t) = 0,\ N(t) = n,\ x < \gamma_0(t) \leq x + \mathrm{d}x], \quad n \geq 1,\\
P_n(x)\,\mathrm{d}x &= \lim_{t\to\infty} Pr[\xi(t) = 1,\ N(t) = n,\ x < \gamma_1(t) \leq x + \mathrm{d}x], \quad n \geq 0,\\
Q_n(x)\,\mathrm{d}x &= \lim_{t\to\infty} Pr[\xi(t) = 2,\ N(t) = n,\ x < \gamma_2(t) \leq x + \mathrm{d}x], \quad n \geq 0,\\
R_n(x)\,\mathrm{d}x &= \lim_{t\to\infty} Pr[\xi(t) = 3,\ N(t) = n,\ x < \gamma_3(t) \leq x + \mathrm{d}x], \quad n \geq 0.
\end{aligned}
$$

3.1. Governing equations

We derive the following steady-state differential-difference equations to control the behaviour of the system using the supplementary variable technique and they are given by

$$\lambda_1 I_0 = (1-p)\int_0^\infty P_0(x)\mu_1(x)\,\mathrm{d}x + \int_0^\infty Q_0(x)\mu_2(x)\,\mathrm{d}x + \int_0^\infty R_0(x)\eta(x)\,\mathrm{d}x, \tag{1}$$

$$\frac{\mathrm{d}}{\mathrm{d}x}I_n(x) = -[\lambda_1 + r(x)]I_n(x), \qquad n \geq 1, \tag{2}$$

$$\frac{\mathrm{d}}{\mathrm{d}x}P_0(x) = -(\lambda_1+\lambda_2+\mu_1(x))P_0(x) + \lambda_1(1-b)P_0(x), \tag{3}$$

$$\frac{\mathrm{d}}{\mathrm{d}x}P_n(x) = -(\lambda_1+\lambda_2+\mu_1(x))P_n(x) + \lambda_1(1-b)P_n(x) + \lambda_1 b P_{n-1}(x), \qquad n \geq 1, \tag{4}$$

$$\frac{\mathrm{d}}{\mathrm{d}x}Q_0(x) = -(\lambda_1+\lambda_2+\mu_2(x))Q_0(x) + \lambda_1(1-b)Q_0(x), \tag{5}$$

$$\frac{\mathrm{d}}{\mathrm{d}x}Q_n(x) = -(\lambda_1+\lambda_2+\mu_2(x))Q_n(x) + \lambda_1(1-b)Q_n(x) + \lambda_1 b Q_{n-1}(x), \qquad n \geq 1, \tag{6}$$

$$\frac{\mathrm{d}}{\mathrm{d}x}R_0(x) = -(\lambda_1+\eta(x))R_0(x) + \lambda_1(1-b)R_0(x), \tag{7}$$

$$\frac{\mathrm{d}}{\mathrm{d}x}R_n(x) = -(\lambda_1+\eta(x))R_n(x) + \lambda_1(1-b)R_n(x) + \lambda_1 b R_{n-1}(x), \qquad n \geq 1. \tag{8}$$

The boundary conditions are given by

$$I_n(0) = (1-p)\int_0^\infty P_n(x)\mu_1(x)\,\mathrm{d}x + \int_0^\infty Q_n(x)\mu_2(x)\,\mathrm{d}x + \int_0^\infty R_n(x)\eta(x)\,\mathrm{d}x, \qquad n \geq 1, \tag{9}$$

$$P_0(0) = \lambda_1 I_0 + \int_{`0}^\infty I_1(x)r(x)\,\mathrm{d}x, \tag{10}$$

$$P_n(0) = \lambda_1\int_0^\infty I_n(x)\,\mathrm{d}x + \int_0^\infty I_{n+1}(x)r(x)\,\mathrm{d}x, \qquad n \geq 1, \tag{11}$$

$$Q_n(0) = p\int_0^\infty P_n(x)\mu_1(x)\,\mathrm{d}x, \qquad n \geq 0, \tag{12}$$

$$R_n(0) = \lambda_2\int_0^\infty P_n(x)\,\mathrm{d}x + \lambda_2\int_0^\infty Q_n(x)\,\mathrm{d}x, \qquad n \geq 0. \tag{13}$$

The normalizing equation is

$$I_0 + \sum_{n=1}^\infty \int_0^\infty I_n(x)\,\mathrm{d}x + \sum_{n=0}^\infty \int_0^\infty [P_n(x) + Q_n(x) + R_n(x)]\,\mathrm{d}x = 1.$$

3.2. Probability generating functions

To solve the above set of differential-difference equations, we need to define the following probability generating functions (p.g.f.) as

$$I(x,z) = \sum_{n=1}^\infty I_n(x)z^n, \quad P(x,z) = \sum_{n=0}^\infty P_n(x)z^n, \quad Q(x,z) = \sum_{n=0}^\infty Q_n(x)z^n, \quad R(x,z) = \sum_{n=0}^\infty R_n(x)z^n,$$

$$I(0,z) = \sum_{n=1}^\infty I_n(0)z^n, \quad P(0,z) = \sum_{n=0}^\infty P_n(0)z^n \quad Q(0,z) = \sum_{n=0}^\infty Q_n(0)z^n, \quad R(0,z) = \sum_{n=0}^\infty R_n(0)z^n,$$

$$I(z)=\int_0^\infty I(x,z)\,\mathrm{d}x,\quad P(z)=\int_0^\infty P(x,z)\,\mathrm{d}x,\quad Q(z)=\int_0^\infty Q(x,z)\,\mathrm{d}x,\quad R(z)=\int_0^\infty R(x,z)\,\mathrm{d}x,$$
$$O(z)=I_0+I(z)+P(z)+Q(z)+R(z).$$

Theorem 1. *The mathematical forms of $I(z)$, $P(z)$, $Q(z)$ and $R(z)$ are provided below:*

$$I(z)=\left[\frac{I_0z\Big[1-\widetilde{A}(\lambda_1)\Big][M(z)-\{b\lambda_1(1-z)+\lambda_2\}]}{z\{b\lambda_1(1-z)+\lambda_2\}-M(z)\Big\{z+(1-z)\widetilde{A}(\lambda_1)\Big\}}\right], \tag{14}$$

$$P(z)=\left[\frac{I_0\lambda_1(z-1)\widetilde{A}(\lambda_1)\Big[1-\widetilde{B}_1(\lambda_1b(1-z)+\lambda_2)\Big]}{z\{b\lambda_1(1-z)+\lambda_2\}-M(z)\Big\{z+(1-z)\widetilde{A}(\lambda_1)\Big\}}\right], \tag{15}$$

$$Q(z)=\left[\frac{I_0\lambda_1p(z-1)\widetilde{A}(\lambda_1)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2)\Big\{1-\widetilde{B}_2(\lambda_1b(1-z)+\lambda_2)\Big\}}{z\{b\lambda_1(1-z)+\lambda_2\}-M(z)\Big\{z+(1-z)\widetilde{A}(\lambda_1)\Big\}}\right], \tag{16}$$

$$R(z)=\left[\frac{I_0\lambda_1\lambda_2(z-1)\widetilde{A}(\lambda_1)}{z\{b\lambda_1(1-z)+\lambda_2\}-M(z)\Big\{z+(1-z)\widetilde{A}(\lambda_1)\Big\}}\right]\left[\frac{1-\widetilde{H}(\lambda_1b(1-z))}{\lambda_1b(1-z)}\right]$$
$$\times\Big\{1-\Big((1-p)+p\widetilde{B}_2(b\lambda_1(1-z)+\lambda_2)\Big)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2)\Big\}, \tag{17}$$

where

$$M(z)=\Big[\Big((1-p)+p\widetilde{B}_2(b\lambda_1(1-z)+\lambda_2)\Big)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2)\Big]\Big[b\lambda_1(1-z)+\lambda_2\Big\{1-\widetilde{H}(b\lambda_1(1-z))\Big\}\Big]$$
$$+\lambda_2\widetilde{H}(b\lambda_1(1-z)),$$

with

$$\widetilde{A}(\lambda_1)=\int_0^\infty e^{-\lambda_1x-\int_0^x r(u)\,\mathrm{d}u}r(x)\,\mathrm{d}x,$$
$$\widetilde{H}(b\lambda_1(1-z))=\int_0^\infty e^{-b\lambda_1(1-z)x-\int_0^x \eta(u)\,\mathrm{d}u}\eta(x)\,\mathrm{d}x,$$
$$\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2)=\int_0^\infty e^{-[b\lambda_1(1-z)+\lambda_2]x-\int_0^x \mu_1(u)\,\mathrm{d}u}\mu_1(x)\,\mathrm{d}x,$$
$$\widetilde{B}_2(b\lambda_1(1-z)+\lambda_2)=\int_0^\infty e^{-[b\lambda_1(1-z)+\lambda_2]x-\int_0^x \mu_2(u)\,\mathrm{d}u}\mu_2(x)\,\mathrm{d}x.$$

Proof. Multiplying (2)–(13) by appropriate power of z and taking the summation of them, we get

$$\frac{\partial I(x,z)}{\partial x}=-[\lambda_1+r(x)]I(x,z), \tag{18}$$
$$\frac{\partial P(x,z)}{\partial x}=-[b\lambda_1(1-z)+\lambda_2+\mu_1(x)]P(x,z), \tag{19}$$
$$\frac{\partial Q(x,z)}{\partial x}=-[b\lambda_1(1-z)+\lambda_2+\mu_2(x)]Q(x,z), \tag{20}$$
$$\frac{\partial R(x,z)}{\partial x}=-[b\lambda_1(1-z)+\eta(x)]R(x,z), \tag{21}$$
$$I(0,z)=(1-p)\int_0^\infty P(x,z)\mu_1(x)\,\mathrm{d}x+\int_0^\infty Q(x,z)\mu_2(x)\,\mathrm{d}x+\int_0^\infty R(x,z)\eta(x)\,\mathrm{d}x-\lambda_1I_0, \tag{22}$$

$$P(0,z) = \lambda_1 I_0 + \lambda_1 \int_0^\infty I(x,z)\,\mathrm{d}x + \frac{1}{z}\int_0^\infty I(x,z)r(x)\,\mathrm{d}x, \tag{23}$$

$$Q(0,z) = p\int_0^\infty P(x,z)\mu_1(x)\,\mathrm{d}x, \tag{24}$$

$$R(0,z) = \lambda_2 \int_0^\infty P(x,z)\,\mathrm{d}x + \lambda_2 \int_0^\infty Q(x,z)\,\mathrm{d}x. \tag{25}$$

Solve the partial differential equations (18)–(21), we obtain

$$I(x,z) = I(0,z)e^{-\lambda_1 x - \int_0^x r(u)\,\mathrm{d}u}, \tag{26}$$

$$P(x,z) = P(0,z)e^{-[b\lambda_1(1-z)+\lambda_2]x - \int_0^x \mu_1(u)\,\mathrm{d}u}, \tag{27}$$

$$Q(x,z) = Q(0,z)e^{-[b\lambda_1(1-z)+\lambda_2]x - \int_0^x \mu_2(u)\,\mathrm{d}u}, \tag{28}$$

$$R(x,z) = R(0,z)e^{-b\lambda_1(1-z)x - \int_0^x \eta(u)\,\mathrm{d}u}. \tag{29}$$

Using (26)–(29) in (22), we get

$$\begin{aligned} I(0,z) = {} & (1-p)P(0,z)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2) + Q(0,z)\widetilde{B}_2(b\lambda_1(1-z)+\lambda_2) + R(0,z) \\ & \times \widetilde{H}(b\lambda_1(1-z)) - \lambda_1 I_0. \end{aligned} \tag{30}$$

Using (26) in (23), we obtain

$$P(0,z) = \frac{1}{z}\Big[\lambda_1 z I_0 + I(0,z)\Big\{z - z\widetilde{A}(\lambda_1) + \widetilde{A}(\lambda_1)\Big\}\Big]. \tag{31}$$

Using (27) in (24), we get

$$Q(0,z) = pP(0,z)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2). \tag{32}$$

Using (27) and (28) in (25), we obtain

$$R(0,z) = \left[\frac{\lambda_2 P(0,z)\Big\{1 - \Big((1-p) + p\widetilde{B}_2(b\lambda_1(1-z)+\lambda_2)\Big)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2)\Big\}}{b\lambda_1(1-z)+\lambda_2}\right]. \tag{33}$$

Using (31)–(33) in (30) and solving for $I(0,z)$, we get

$$I(0,z) = \left[\frac{I_0 z\lambda_1[M(z) - \{b\lambda_1(1-z)+\lambda_2\}]}{z\{b\lambda_1(1-z)+\lambda_2\} - M(z)\Big\{z + (1-z)\widetilde{A}(\lambda_1)\Big\}}\right]. \tag{34}$$

Using (34) in (31) and solving for $P(0,z)$, we get

$$P(0,z) = \left[\frac{I_0\lambda_1[b\lambda_1(1-z)+\lambda_2](z-1)\widetilde{A}(\lambda_1)}{z\{b\lambda_1(1-z)+\lambda_2\} - M(z)\Big\{z + (1-z)\widetilde{A}(\lambda_1)\Big\}}\right]. \tag{35}$$

Now, using (35) in (32), (33) and solving for $Q(0,z)$ and $R(0,z)$, respectively, we get

$$Q(0,z) = \left[\frac{I_0\lambda_1 p[b\lambda_1(1-z)+\lambda_2](z-1)\widetilde{A}(\lambda_1)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2)}{z\{b\lambda_1(1-z)+\lambda_2\} - M(z)\Big\{z + (1-z)\widetilde{A}(\lambda_1)\Big\}}\right], \tag{36}$$

$$R(0,z) = \left[\frac{I_0\lambda_1\lambda_2(z-1)\widetilde{A}(\lambda_1)}{z\{b\lambda_1(1-z)+\lambda_2\} - M(z)\Big\{z+(1-z)\widetilde{A}(\lambda_1)\Big\}}\right] \times\Big\{1-\Big(q+p\widetilde{B}_2(b\lambda_1(1-z)+\lambda_2)\Big)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2)\Big\}. \tag{37}$$

Using (34), (35), (36), (37) in (26), (27), (28), (29), respectively, we get

$$I(x,z) = \left[\frac{I_0 z\lambda_1[M(z)-\{b\lambda_1(1-z)+\lambda_2\}]}{z\{b\lambda_1(1-z)+\lambda_2\} - M(z)\Big\{z+(1-z)\widetilde{A}(\lambda_1)\Big\}}\right] e^{-\lambda_1 x-\int_0^x r(u)\,du}, \tag{38}$$

$$P(x,z) = \left[\frac{I_0\lambda_1[b\lambda_1(1-z)+\lambda_2](z-1)\widetilde{A}(\lambda_1)}{z\{b\lambda_1(1-z)+\lambda_2\} - M(z)\Big\{z+(1-z)\widetilde{A}(\lambda_1)\Big\}}\right] e^{-[b\lambda_1(1-z)+\lambda_2]x-\int_0^x \mu_1(u)\,du}, \tag{39}$$

$$Q(x,z) = \left[\frac{I_0\lambda_1 p[b\lambda_1(1-z)+\lambda_2](z-1)\widetilde{A}(\lambda_1)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2)}{z\{b\lambda_1(1-z)+\lambda_2\} - M(z)\Big\{z+(1-z)\widetilde{A}(\lambda_1)\Big\}}\right] e^{-[b\lambda_1(1-z)+\lambda_2]x-\int_0^x \mu_2(u)\,du}, \tag{40}$$

$$R(x,z) = \left[\frac{I_0\lambda_1\lambda_2(z-1)\widetilde{A}(\lambda_1)}{z\{b\lambda_1(1-z)+\lambda_2\} - M(z)\Big\{z+(1-z)\widetilde{A}(\lambda_1)\Big\}}\right] \times\Big\{1-\Big((1-p)+p\widetilde{B}_2(b\lambda_1(1-z)+\lambda_2)\Big)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2)\Big\}e^{-b\lambda_1(1-z)x-\int_0^x \eta(u)\,du}. \tag{41}$$

Integrate (38), (39), (40), (41) with respect to x from 0 to ∞, we obtain the desired results (14), (15), (16), (17), respectively. □

Corollary 1. *The orbit length $O(z)$ is given by*

$$O(z) = \left[\frac{I_0\widetilde{A}(\lambda_1)}{b}\right]\left[\begin{array}{l}(1-b)\big((1-p)+p\widetilde{B}_2(b\lambda_1(1-z)+\lambda_2)\big)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2)\\ \times\{\lambda_1 b(1-z)+\lambda_2(1-\widetilde{H}(b\lambda_1(1-z)))\}\\ +(1-b)\lambda_2\widetilde{H}(b\lambda_1(1-z))-(1-bz)\{\lambda_1 b(1-z)+\lambda_2\}\\ \hline z\{b\lambda_1(1-z)+\lambda_2\}-M(z)\{z+(1-z)\widetilde{A}(\lambda_1)\}\end{array}\right], \tag{42}$$

where

$$I_0 = \frac{\lambda_2\widetilde{A}(\lambda_1)-b\lambda_1\{1+\lambda_2 E[H]\}\Big\{1-\Big((1-p)+p\widetilde{B}(\lambda_2)\Big)\widetilde{B}_1(\lambda_2)\Big\}}{\lambda_2\widetilde{A}(\lambda_1)+(1-b)\lambda_1\widetilde{A}(\lambda_1)\{1+\lambda_2 E[H]\}\Big\{1-\Big((1-p)+p\widetilde{B}(\lambda_2)\Big)\widetilde{B}_1(\lambda_2)\Big\}}. \tag{43}$$

Proof. By adding I_0, (14), (15), (16), and (17) and performing some mathematical calculations, we obtain the required expression for (42). Furthermore, using the normalizing condition $O(1) = 1$ we derive the required expression (43). □

Corollary 2. *The probability generating function $S(z)$ for the system length is given by*

$$S(z) = \left[\frac{I_0\widetilde{A}(\lambda_1)}{b}\right]\left[\frac{bz\{b\lambda_1(1-z)+\lambda_2\} - bM(z) + \Big[b\lambda_1 z(1-z) + \lambda_2\Big(1-\widetilde{H}(b\lambda_1(1-z))\Big)\Big]\Big\{\Big((1-p)+p\widetilde{B}_2(b\lambda_1(1-z)+\lambda_2)\Big)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2)-1\Big\}}{z\{b\lambda_1(1-z)+\lambda_2\} - M(z)\Big\{z+(1-z)\widetilde{A}(\lambda_1)\Big\}}\right]. \quad (44)$$

Proof. The expression of $S(z)$ can be written as $S(z) = I_0 + I(z) + z(P(z) + Q(z)) + R(z)$. By utilizing the expressions of $I(z), P(z), Q(z), R(z)$ and simplifying, we obtain the required expression given in (44). □

3.3. Stability criteria

It is now possible to characterize the stability of the system by considering the value of I_0 that has been obtained in (43). In particular, to ensure the stability of the system, I_0 needs to be greater than zero as per (42) with $O(1) = 1$. Thus, we have

$$\lambda_2\widetilde{A}(\lambda_1) > b\lambda_1\{1+\lambda_2 E[H]\}\Big\{1-\Big((1-p)+p\widetilde{B}(\lambda_2)\Big)\widetilde{B}_1(\lambda_2)\Big\}. \quad (45)$$

3.4. Extract the remaining probabilities by roots method

This subsection is devoted to extract the remaining probabilities $I_n, n \geq 1$, $P_n, n \geq 0$, $Q_n, n \geq 0$, and $R_n, n \geq 0$ from the known functions $I(z)$, $P(z)$, $Q(z)$, and $R(z)$, respectively. For this purpose, we utilize the partial fraction approach which necessitates to identify the zeros of the denominators of respective functions. To avoid the repeated analysis and representation for the rational functions $I(z)$, $P(z)$, $Q(z)$, and $R(z)$ as given in (14) to (17), we express them as follows:

$$K(z) = \frac{N_K(z)}{D(z)}, \quad K = I, P, Q, R,$$

where $N_K(z)$ represents the numerator of the functions $I(z)$, $P(z)$, $Q(z)$ and $R(z)$, when K corresponds to I, P, Q and R, respectively, and given by

$$\begin{aligned}
N_I(z) &= I_0 z\Big[1-\widetilde{A}(\lambda_1)\Big][M(z) - \{b\lambda_1(1-z)+\lambda_2\}]\\
N_P(z) &= I_0\lambda_1(z-1)\widetilde{A}(\lambda_1)\Big[1-\widetilde{B}_1(\lambda_1 b(1-z)+\lambda_2)\Big]\\
N_Q(z) &= I_0\lambda_1 p(z-1)\widetilde{A}(\lambda_1)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2)\Big\{1-\widetilde{B}_2(\lambda_1 b(1-z)+\lambda_2)\Big\},\\
N_R(z) &= \left[\frac{\Big[I_0\lambda_1\lambda_2(z-1)\widetilde{A}(\lambda_1)\Big]\Big[1-\widetilde{H}(\lambda_1 b(1-z))\Big]}{\lambda_1 b(1-z)}\right]\\
&\quad\times\Big\{1-\Big((1-p)+p\widetilde{B}_2(b\lambda_1(1-z)+\lambda_2)\Big)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2)\Big\},\\
D(z) &= z\{b\lambda_1(1-z)+\lambda_2\} - M(z)\Big\{z+(1-z)\widetilde{A}(\lambda_1)\Big\}.
\end{aligned}$$

To perform the partial fraction decomposition of $K(z)$, we begin by identifying the zeros of $D(z)$. Notably, the zeros of $D(z)$ that have absolute values less than or equal to one also serve as zeros of $N_K(z)$ and do not contribute to the partial fraction decomposition. When the stability condition (45) is satisfied, the function $D(z)$ has one zero on the unit circle and none inside the unit circle. Therefore, we concentrate on the zeros of $D(z)$

with absolute values greater than one. The number of such zeros is dependent on the distributions of retrial, service and repair times. Let m be the total number of distinct zeros of $D(z)$ with absolute values greater than one, which we denote as $\beta_1, \beta_2, \ldots, \beta_m$. By applying the partial fraction decomposition to $K(z)$, we obtain

$$K(z) = \sum_{i=1}^{m} \frac{C_{K,i}}{\beta_i - z}, \qquad K = I, P, Q, R, \tag{46}$$

where

$$C_{K,i} = -\frac{N_K(\beta_i)}{D^{(1)}(\beta_i)}, \quad i = 1, \ldots, m.$$

Now, collecting the coefficients of z^n from both sides of (46), we obtain

$$K_0 = \sum_{i=1}^{m} \frac{C_{K,i}}{\beta_i}, \qquad K = P, Q, R,$$
$$K_n = \sum_{i=1}^{m} \frac{C_{K,i}}{\beta_i^{n+1}} \quad n \geq 1, \quad K = I, P, Q, R.$$

Remark 1. If some of the zeros of $D(z)$ with absolute values greater than one are repeated, minor modifications to the partial fraction process are necessary. For further information, readers can refer to Samanta and Rashmi [39] and Pradhan *et al.* [37].

3.5. Tail distribution of orbit length at random epoch

Here, our goal is to derive the tail distribution of the orbit length at random epoch by examining a single zero of $D(z)$, rather than all of its zeros. Specifically, we focus on the zero of $D(z)$ that exceeds one in absolute value and has the smallest modulus. We assume that all poles are simple (multiplicity 1). It is essential to highlight that this particular zero is both real and strictly positive. However, if $D(z)$ contains two complex conjugate zeros or a negative real zero with the smallest modulus, it could lead to negative probabilities for large n, as demonstrated in Bruneel *et al.* [9] and Desmet *et al.* [12]. Assuming β is the positive real zero with the smallest modulus and multiplicity one. Hence, we express $M(z)$ *via* the partial fraction method as follows:

$$K(z) \simeq \frac{C_K}{\beta - z}, \quad K = I, P, Q, R, \tag{47}$$

where the constant coefficients C_K are given by

$$C_K = -\frac{N_K(\beta)}{D^{(1)}(\beta)}.$$

Thus, by collecting the coefficients of z^n, $n \geq 0$ from (47), the tail distribution of orbit length is given by

$$K_0 \simeq \frac{C_K}{\beta}, \qquad K = P, Q, R,$$
$$K_n \simeq \frac{C_K}{\beta^{n+1}} \quad n \geq 1, \quad K = I, P, Q, R.$$

4. Orbit length distribution at post-departure epoch

This section aims to derive the steady-state probability distribution from the p.g.f. of the orbit length distribution at post-departure epoch. We employ the PASTA concept to attain the objective. The PASTA principle asserts that the count of customers in the retrial group right after a customer's exit is directly proportional to the count of customers in the orbit right before the customer's exit. Let P_n^+ and Q_n^+, for $n \geq 0$, represent the probability of having n customers in the orbit right after the departure of a customer from FPS and SPS, respectively. Therefore, for $n \geq 0$, we have

$$P_n^+ = J \int_0^\infty P_n(x)\mu_1(x)\,\mathrm{d}x, \tag{48}$$

$$Q_n^+ = J \int_0^\infty Q_n(x)\mu_2(x)\,\mathrm{d}x, \tag{49}$$

where J is a normalizing constant.

Let Ω_n^+, $n \geq 0$, represent the steady state probability of having n customers left in the orbit after the departure of a customer either from FPS and SPS. Hence, we have

$$\Omega_n^+ = J[P_n^+ + Q_n^+]. \tag{50}$$

In order to determine Ω_n^+, let us define $P^+(z) = \sum_{n=0}^\infty P_n^+ z^n$, $Q^+(z) = \sum_{n=0}^\infty Q_n^+ z^n$ and $\Omega^+(z) = \sum_{n=0}^\infty \Omega_n^+ z^n$.

Theorem 2. *The expression for the $\Omega^+(z)$ is given by*

$$\Omega^+(z) = JI_0\lambda_1\widetilde{A}(\lambda_1)\left[\frac{(z-1)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2)\Big(1+p\widetilde{B}_2(b\lambda_1(1-z)+\lambda_2)\Big)\{b\lambda_1(1-z)+\lambda_2\}}{z\{b\lambda_1(1-z)+\lambda_2\} - K(z)\Big\{z+(1-z)\widetilde{A}(\lambda_1)\Big\}}\right], \tag{51}$$

with

$$J = \frac{\lambda_2\widetilde{A}(\lambda_1) - b\lambda_1\{1+\lambda_2 E[H]\}\Big(1-\Big\{(1-p)+p\widetilde{B}(\lambda_2)\Big\}\widetilde{B}_1(\lambda_2)\Big)}{I_0\lambda_1\lambda_2\widetilde{B}_1(\lambda_2)\Big[1+p\widetilde{B}_2(\lambda_2)\Big]}. \tag{52}$$

Proof. Multiply (48) and (49) by z^n, add them over n and after performing some mathematical calculations, we have

$$P^+(z) = JP(0,z)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2), \tag{53}$$

$$Q^+(z) = JQ(0,z)\widetilde{B}_2(b\lambda_1(1-z)+\lambda_2). \tag{54}$$

Multiplying (50) by z^n and taking summation on $n \geq 0$, we obtain

$$\Omega^+(z) = J\big[P^+(z) + Q^+(z)\big]. \tag{55}$$

Using (53) and (54) in (55), we get

$$\Omega^+(z) = J\Big[P(0,z)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2) + Q(0,z)\widetilde{B}_2(b\lambda_1(1-z)+\lambda_2)\Big]. \tag{56}$$

Using (35) and (36) in (56), and performing some mathematical calculation, we obtain the required result (51) and utilizing the normalizing condition $\Omega^+(1) = 1$, we get the result (52). □

4.1. Determination of Ω_n^+, $n \geq 0$

Now, we proceed to determine the probabilities Ω_n^+ from the known function $\Omega^+(z)$ obtained in (51). To facilitate this, we rewrite the function as follows:

$$\Omega^+(z) = \frac{N(z)}{D(z)},$$

where

$$N(z) = JI_0\lambda_1\widetilde{A}(\lambda_1)\Big[(z-1)\widetilde{B}_1(b\lambda_1(1-z)+\lambda_2)\Big(1+p\widetilde{B}_2(b\lambda_1(1-z)+\lambda_2)\Big)\{b\lambda_1(1-z)+\lambda_2\}\Big].$$

By following the same methodology outlined above for the orbit length distribution at random epoch, we determine the orbit length distribution at post-departure epoch using (51). Applying the partial fraction method, $\Omega^+(z)$ can be uniquely decomposed as:

$$\Omega^+(z) = \sum_{k=1}^{m} \frac{F_k}{\beta_k - z}, \tag{57}$$

where

$$F_k = -\frac{N(\beta_k)}{D^{(1)}(\beta_k)}, \quad k = 1, 2, \ldots, m.$$

Collect the coefficients of z^n from both sides of (57), we obtain

$$\Omega_n^+ = \sum_{k=1}^{m} \frac{F_k}{\beta_k^{n+1}} \quad n \geq 0.$$

Remark 2. It is noted that P_n^+ and Q_n^+ can be derived from (53) and (54), respectively, in a similar manner as we discussed for Ω_n^+.

4.2. Tail distribution of orbit length at post-departure epoch

Here we aim to obtain the tail distribution of the orbit length at post-departure epoch in a similar way as we described in Subsection 3.4. It is given by

$$\Omega^+(z) \simeq \frac{F}{\beta - z}, \tag{58}$$

where the constant coefficient F is given by

$$F = -\frac{N(\beta)}{D^{(1)}(\beta)}.$$

Thus, collect the coefficients of z^n, $n \geq 0$ from (58), the tail distribution of orbit length is given by

$$\Omega_n^+ \simeq \frac{F}{\beta^{n+1}} \quad n \geq 0.$$

5. Performance measures

This section aims to assess the performance measures of the system. The key performance metrics such as the average orbit length L_o and the mean waiting time $W_o = L_o/\lambda_1$ can be easily obtained as the orbit length distribution at random epoch is already known to us.

5.1. Mean performance measures

(i) When the server is idle, the mean orbit length is given by

$$L_o^I = \sum_{n=1}^{\infty} nI_n = \sum_{i=1}^{m} \frac{C_{I,i}}{(\beta_i - 1)^2}.$$

(ii) If the server is busy in FPS, the corresponding mean orbit length is

$$L_o^P = \sum_{n=1}^{\infty} nP_n = \sum_{i=1}^{m} \frac{C_{P,i}}{(\beta_i - 1)^2}.$$

(iii) When the server is busy in SPS, the corresponding mean orbit length is

$$L_o^Q = \sum_{n=1}^{\infty} nQ_n = \sum_{i=1}^{m} \frac{C_{Q,i}}{(\beta_i - 1)^2}.$$

(iv) During the server under repair, the mean orbit length is

$$L_o^R = \sum_{n=1}^{\infty} nR_n = \sum_{i=1}^{m} \frac{C_{R,i}}{(\beta_i - 1)^2}.$$

Overall the mean number of customers in the orbit is given by

$$\begin{aligned} L_o &= \sum_{n=1}^{\infty} n(I_n + P_n + Q_n + R_n), \\ &= \sum_{i=1}^{m} \left[\frac{C_{I,i}}{(\beta_i - 1)^2} + \frac{C_{P,i}}{(\beta_i - 1)^2} + \frac{C_{Q,i}}{(\beta_i - 1)^2} + \frac{C_{R,i}}{(\beta_i - 1)^2} \right]. \end{aligned}$$

5.2. Probabilistic performance measures

(i) Probability that the server is idle during retrial time is given by

$$P_I = \frac{b\lambda_1 \left[1 - \widetilde{A}(\lambda_1)\right] \left[\left\{1 + \frac{\lambda_2}{\delta}\right\} \left(1 - \left\{(1-p) + p\widetilde{B}(\lambda_2)\right\} \widetilde{B}_1(\lambda_2)\right) \right]}{\lambda_2 \widetilde{A}(\lambda_1) + (1-b)\lambda_1 \widetilde{A}(\lambda_1) \left\{1 + \frac{\lambda_2}{\delta}\right\} \left\{1 - \left((1-p) + p\widetilde{B}(\lambda_2)\right) \widetilde{B}_1(\lambda_2)\right\}}. \tag{59}$$

(ii) Probability that the server is busy with FPS is given by

$$P_{\text{FPS}} = \frac{\lambda_1 \widetilde{A}(\lambda_1) \left[1 - \widetilde{B}_1(\lambda_2)\right]}{\lambda_2 \widetilde{A}(\lambda_1) + (1-b)\lambda_1 \widetilde{A}(\lambda_1) \left\{1 + \frac{\lambda_2}{\delta}\right\} \left\{1 - \left((1-p) + p\widetilde{B}(\lambda_2)\right) \widetilde{B}_1(\lambda_2)\right\}}. \tag{60}$$

(iii) Probability that the server is busy with SPS is given by

$$P_{\text{SPS}} = \frac{p\lambda_1 \widetilde{A}(\lambda_1) \widetilde{B}_1(\lambda_2) \left[1 - \widetilde{B}_2(\lambda_2)\right]}{\lambda_2 \widetilde{A}(\lambda_1) + (1-b)\lambda_1 \widetilde{A}(\lambda_1) \left\{1 + \frac{\lambda_2}{\delta}\right\} \left\{1 - \left((1-p) + p\widetilde{B}(\lambda_2)\right) \widetilde{B}_1(\lambda_2)\right\}}. \tag{61}$$

(iv) Probability that the server is under repair is given by

$$P_R = \frac{\left[\frac{\lambda_1\lambda_2\widetilde{A}(\lambda_1)}{\delta}\right]\left[1-\left\{(1-p)+p\widetilde{B}(\lambda_2)\right\}\widetilde{B}_1(\lambda_2)\right]}{\lambda_2\widetilde{A}(\lambda_1)+(1-b)\lambda_1\widetilde{A}(\lambda_1)\left\{1+\frac{\lambda_2}{\delta}\right\}\left\{1-\left((1-p)+p\widetilde{B}(\lambda_2)\right)\widetilde{B}_1(\lambda_2)\right\}}. \tag{62}$$

It is noteworthy by substituting $z=1$ in (14)–(17) and performing the necessary mathematical calculations, we derive the desired result (59)–(62).

5.3. Special cases

In this subsection, we generate a number of previous research findings for well-known queueing models as special cases of our model by specifying certain parameters.

(i) When we consider $\lambda_2 = 0$ (no negative customers which means there is no server breakdown) and $b=1$ (no balking), our model reduces to a retrial queueing model with optional service and general retrial times. In this case, (14), (15), and (16), respectively, reduce to the following

$$I(z) = \left[\frac{I_0 z\left[1-\widetilde{A}(\lambda_1)\right]\left[1-\left((1-p)+p\widetilde{B}_2(\lambda_1(1-z))\right)\widetilde{B}_1(\lambda_1(1-z))\right]}{\widetilde{A}(\lambda_1)(1-z)\left((1-p)+p\widetilde{B}_2(\lambda_1(1-z))\right)\widetilde{B}_1(\lambda_1(1-z)) - z\left(1-\left((1-p)+p\widetilde{B}_2(\lambda_1(1-z))\right)\widetilde{B}_1(\lambda_1(1-z))\right)}\right], \tag{63}$$

$$P(z) = \left[\frac{I_0\widetilde{A}(\lambda_1)\left[1-\widetilde{B}_1(\lambda_1(1-z))\right]}{\widetilde{A}(\lambda_1)(1-z)\left((1-p)+p\widetilde{B}_2(\lambda_1(1-z))\right)\widetilde{B}_1(\lambda_1(1-z)) - z\left(1-\left((1-p)+p\widetilde{B}_2(\lambda_1(1-z))\right)\widetilde{B}_1(\lambda_1(1-z))\right)}\right], \tag{64}$$

$$Q(z) = \left[\frac{I_0 p\widetilde{A}(\lambda_1)\widetilde{B}_1(\lambda_1(1-z))\left\{1-\widetilde{B}_2(\lambda_1(1-z))\right\}}{\widetilde{A}(\lambda_1)(1-z)\left((1-p)+p\widetilde{B}_2(\lambda_1(1-z))\right)\widetilde{B}_1(\lambda_1(1-z)) - z\left(1-\left((1-p)+p\widetilde{B}_2(\lambda_1(1-z))\right)\widetilde{B}_1(\lambda_1(1-z))\right)}\right], \tag{65}$$

where

$$I_0 = \frac{\widetilde{A}(\lambda_1) - \lambda_1\{E[B_1]+pE[B_2]\}}{\widetilde{A}(\lambda_1)}.$$

The outcomes (63), (64), (65) correspond with Theorem 3.1(ii) of Choudhury [11].

(ii) When we consider $\lambda_2 = 0$ (no negative customers which means there is no server breakdown), $b=1$ (no balking), and $p=0$ (no optional service), our model reduces to a retrial queue with general retrial times. Consequently, result (44) simplifies to

$$S(z) = \left[\frac{I_0\widetilde{A}(\lambda_1)(1-z)\widetilde{B}_1(\lambda_1(1-z))}{\widetilde{A}(\lambda_1)(1-z)\widetilde{B}_1(\lambda_1(1-z)) - z\left\{1-\widetilde{B}_1(\lambda_1(1-z))\right\}}\right], \tag{66}$$

where

$$I_0 = \frac{\widetilde{A}(\lambda_1) - \lambda_1 E[B_1]}{\widetilde{A}(\lambda_1)}.$$

The outcome (66) corresponds with equation (16) of Gómez-Corral [21].

(iii) When we consider $\lambda_2 = 0$ (no negative customers which means there is no server breakdown), $b = 1$ (no balking), $p = 0$ (no optional service) and $\widetilde{A}(\lambda_1) \to 1$, our model reduces to a classical $M/G/1$ queueing system. Consequently, result (44) simplifies as

$$S(z) = \left[\frac{I_0(1-z)\widetilde{B}_1(\lambda_1(1-z))}{\widetilde{B}_1(\lambda_1(1-z)) - z}\right]. \tag{67}$$

These expressions are consistent with the well-known P-K formula, see Gross *et al.* [22].

6. Numerical results

This section provides a comprehensive validation of the analytical results discussed in earlier sections by presenting a range of numerical examples. To achieve this, we generate numerical outcomes under different distributions for service times, retrial time, and repair time, as well as varying model parameters. These results confirm the accuracy of the proposed methodology and provide a foundation for researchers exploring alternative approaches. To deepen the analysis, we develop a cost function and examine the impact of variations in the model parameters on the overall system cost. For Table 3, the retrial time, FPS time, SPS time, and repair time are considered as Hyper-Exponential, Inverse Gaussian (IG), Erlang (Er) distribution, and Deterministic (D) distribution, respectively. Further, for Table 3, we set the values of the parameters as $\lambda_1 = 7.0$, $\lambda_2 = 2.0$, $p = 0.7$ and $b = 0.4$. On the other hand, Table 4 employs Phase-type (PH) distribution for retrial time, Gamma (Ga) distribution for FPS time, Pareto (Pr) distribution for SFS time, and Hyper-Erlang (HEr) distribution for repair time. Moreover, for Table 4, we set the values of the parameters as $\lambda_1 = 7.5$, $\lambda_2 = 3.0$, $p = 0.55$ and $b = 0.4$. These varied distributions allow us to conduct a comprehensive numerical analysis of the orbit length distribution across different time epochs. Tables 3, 4, and 5 present the exact and tail distributions of orbit length corresponding to various server states at different time epochs. The steady-state probabilities are shown for selected values of n ($n \geq 0$) across different columns in the tables. From Tables 3, 4, and 5, it can be observed that the exact and tail distribution values converge beyond a certain range of n. In the random epoch case, Table 3 presents results for the $M/IG/1$ retrial G-queue with $\rho = 0.5$, while Table 4 corresponds to the $M/Ga/1$ retrial G-queue with $\rho = 0.7$. For the post-departure case (Tab. 5), we consider both systems: the $M/IG/1$ retrial G-queue with $\rho = 0.5$ and the $M/Ga/1$ retrial G-queue with $\rho = 0.7$. Moreover, the tables include key performance measures, such as L_o^I, L_o^P, L_o^Q, L_o^R and L_o which are summarized at the bottom. Tables 3 and 4 confirm the relation $\sum_{n=1}^{\infty} I_n = P_I$, $\sum_{n=0}^{\infty} P_n = P_{\text{FPS}}$, $\sum_{n=0}^{\infty} Q_n = P_{\text{SPS}}$ and $\sum_{n=0}^{\infty} R_n = P_R$ which demonstrate the consistency in the results. Additionally, the average waiting time W_o in the orbit for an arbitrary customer is also provided in Tables 3 and 4. It is to be noted that when the server is idle, the incoming customer gets FPS service immediately and then either proceeds to SPS or leaves the system. In addition, a customer from the orbit can also retry when the server is idle. Thus, in both scenarios, the number of customers in the system does not increase while the server is idle.

Example 1. The $M/IG/1$ retrial G-queue.

Hyper-exponential distribution: The p.d.f. and the L.-S.T. of hyper-exponential distribution are given by

$$a(x) = \sum_{i=1}^{2} g_i G_i e^{-G_i x}, \quad \text{and} \quad \widetilde{A}(s) = \sum_{i=1}^{2} \frac{g_i G_i}{G_i + s},$$

respectively, where g_i are the mixing probabilities such that $\sum_{i=1}^{2} g_i = 1$ and G_i are the rate parameters of the exponential components. For computational purposes, we assign the following values: $g_1 = 0.6$, $g_2 = 0.4$, $G_1 = 12$, and $G_2 = 9$. The mean is expressed as $\sum_{i=1}^{2} \frac{g_i}{G_i}$, resulting $E[A] = 0.09444445$ which yields retrial rate $\alpha = 10.58823529$.

Inverse Gaussian distribution: The p.d.f. and the L.-S.T. of IG distribution are expressed as

$$b_1(x) = \sqrt{\frac{\sigma}{2\pi x^3}} e^{\frac{-\sigma(x-\tau)^2}{2\tau^2 x}},$$
$$\widetilde{B}_1(s) = e^{\frac{\sigma}{\tau} - \frac{\sqrt{\frac{\sigma}{\tau^2}+2s}}{\sqrt{\frac{1}{\sigma}}}}, \tag{68}$$

where the sharpening parameters are $\sigma = 0.9$ and $\tau = 0.1$, while $erf[x] = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\, \mathrm{d}t$ specified the error function. We use the Padé approximation method to convert the transcendental function $\widetilde{B}_1(s)$ given in (68) to a rational function. In order to do this, we use Padé(3,4) of (68), and it is provided by

$$\widetilde{B}_1(s) = \frac{1.00000000 - 0.02610667s + 0.00029138s^2 - 0.00000143s^3}{1.000000000 + 0.07389332s + 0.00212515s^2 + 0.00002895s^3 + 0.00000001s^4}.$$

Further, $E[B_1] = 0.1$ is the mean service time and it gives $\kappa_1 = 10.0$.

Erlang distribution (Er): The p.d.f. and the L.-S.T. of the Erlang service time distribution are given as

$$b_2(x) = \frac{\varpi^k x^{k-1} e^{-\varpi x}}{(k-1)!}, \quad \text{and} \quad \widetilde{B_2}(s) = \left(\frac{\varpi}{\varpi + s}\right)^k,$$

respectively, where we set $k = 2$ and $\varpi = 10$. This leads to a mean service time of $E[B_2] \equiv \frac{k}{\varpi} = \frac{1}{5}$ and $\kappa_2 = 5$.

Deterministic distribution (D): The p.d.f. of Deterministic distribution for repair time is given by

$$h(x) = \begin{cases} 1, & \text{when } \ x = E[H], \\ 0, & \text{otherwise.} \end{cases}$$

The L.-S.T. of the Deterministic distribution is $\widetilde{H}(s) = e^{-E[H]s}$. Here, we assume $E[H] = 1/12$ which gives $\delta = 12$. To convert the transcendental function $\widetilde{H}(s)$ into a rational function, we utilize the Padé approximation method. In particular, we use the Padé(2,3) approximation of $\widetilde{H}(s)$ and it is given by

$$\widetilde{H}(s) = \frac{1.0000000 - 0.03571428s + 0.00049603s^2}{1.0000000 + 0.047619047s + 0.000992063s^2 + 0.00001102s^3}.$$

Example 2. The $M/Ga/1$ retrial G-queue.

Phase-type distribution: The p.d.f. and L.-S.T. of PH distribution are given by

$$a(x) = \boldsymbol{\psi}\ e^{(\mathbf{J}x)}\mathbf{J}^0, x \geq 0, \quad \text{and} \quad \widetilde{A}(s) = \boldsymbol{\psi}\ (s\mathbf{I} - \mathbf{J})^{-1}\mathbf{J}^0,$$

respectively. In these expressions, $\mathbf{J}^0 = -\mathbf{J}\mathbf{1}$, where $\mathbf{1}$ is a 2×1 column vector with all elements equal to 1. Here, $\boldsymbol{\psi} = (0.4, 0.6)$ is taken as the initial probability row vector, and $\mathbf{J}$ is a 2×2 subgenerator matrix. The identity matrix of order 2 is denoted by $\mathbf{I}$. The matrix $\mathbf{J}$ is specified as follows:

$$\mathbf{J} = \begin{pmatrix} -5.0 & 1.0 \\ 2.0 & -10.0 \end{pmatrix}.$$

With this setup, the L.-S.T. becomes $\widetilde{A}(s) = \frac{16(0.40s+3)}{s^2+15s+48}$, which leads to $\alpha = 5.58$.

Gamma distribution: The p.d.f. and L.-S.T. of Gamma distribution are given by

$$b_1(x) = \frac{\varphi^{\omega} x^{\omega-1} e^{-\varphi x}}{\Gamma(\omega)}, \quad \text{and} \quad \widetilde{B}_1(s) = \left(\frac{\varphi}{s+\varphi}\right)^{\omega},$$

respectively. Here, $\omega = 2.0$ is the shape parameter, and $\varphi = 30.0$ is the rate parameter. Substituting these values, we obtain $\widetilde{B}_1(s) = \frac{900}{(s+30)^2}$ that leads to $E[B_1] = 0.06666667$ with $\kappa_1 = 15$.

Pareto distribution: The p.d.f. of PR service time distribution is taken as $b_2(x) = \frac{\zeta c^{\zeta}}{(\zeta+x)^{\zeta+1}}$, $x \geq 0$, under the assumption that $\zeta = 0.7$ is the shape parameter and $c = 0.1$ is the scale parameter. Since the PR distribution lacks L.-S.T., we apply the Geometric Transform Approximation Method (GTAM) developed by Shortle *et al.* [43] to generate an approximate L.-S.T. of $b_2(x)$. As the assumed value of the shape parameter ζ is less than one, the mean of PR distribution does not exist in this case. Therefore, we take into account the median of the distribution in order to calculate the approximate L.-S.T. of the PR distribution using the approach described by Shortle *et al.* [43]. However, when the value of shape parameter ζ is greater than one, then the mean of the PR distribution exists, and the mean can be used to determine the approximate L.-S.T. of the PR distribution. For further information on approximate L.-S.T. using mean and median, interested readers are directed to the works of Samanta and Bank [38] as well as Bank and Samanta [7], who used mean and median to derive approximate L.-S.T. based on Shortle *et al.* [43]. Consequently, the approximate L.-S.T. of the PR distribution is given by

$$\widetilde{B}_2(s) = \frac{1.00000000 - 0.01923735s + 0.00249276s^2 - 0.00001456s^3 + 0.000000003s^4}{1.00000000 + 0.1111236563s + 0.00555287s^2 + 0.00015837s^3 + 0.000002630s^4 + 0.00000001s^5}.$$

This gives $E[B_2] = 0.130361015$ and $\kappa_2 = 7.67100501$.

Hyper Erlang distribution: Let $h(x) = \sum_{i=1}^{N} \phi_i \frac{\nu_i^{\theta_i} x^{\theta_i-1} e^{-\nu_i x}}{(\theta_i-1)!}$ be the p.d.f. of HEr service time distribution, where $\nu_i, \phi_i \geq 0$, $\sum_{i=1}^{N} \phi_i = 1$ and $N, \theta_1, \theta_2, \ldots, \theta_N$ are positive integers. The L.-S.T. of HEr service time distribution is given by $\widetilde{H}(s) = \sum_{i=1}^{N} \phi_i \left(\frac{\nu_i}{\nu_i+s}\right)^{\theta_i}$. Here, we assume $N = 2, \phi_1 = 0.6, \phi_2 = 0.4, \theta_1 = 2, \theta_2 = 4$, $\nu_1 = 40$, and $\nu_2 = 20$. The mean repair time is presented by $E[H] = \sum_{i=1}^{N} \frac{\theta_i \phi_i}{\nu_i} = 0.11$ that yields $\delta = 9.09090909$.

6.1. Sensitivity analysis of the performance measures

In this subsection, we investigate the influence of various system parameters on the performance measures of the proposed queueing model. By examining factors such as arrival rates for positive and negative customers, retrial rate, repair rate and different values of p and b, we analyze their effects on key metrics like waiting time and I_0. These findings provide valuable insights for system managers looking to optimize performance while ensuring system stability. We explore the impact of different parameter values as shown in Figures 3–10 using the distributions from Example 1. These analyses help in understanding how adjustments are required to the system parameters which can influence its overall behavior. Figures 3 and 4 demonstrate the effect of λ_2 on I_0 and W_o for various values of δ, respectively. As observed in Figure 3, the probability I_0 increases with an increase in δ. This is because a higher repair rate (δ) allows the server to become available more quickly and customers in the orbit are able to retry faster. Hence, the number of customers in the orbit decreases which leads to a rise in I_0. Additionally, as the arrival rate of negative customers (λ_2) increases, I_0 also grows. This occurs because frequent interruptions by negative customers lead to the server becomes idle frequently after repairs, thereby increasing the probability I_0. Consequently, with the same arguments, the average waiting time W_o decreases as shown in Figure 4.

Customers in the orbit attempt to retry more frequently as the retrial rate (α) increases which leads to a noticeable reduction in the number of customers in the orbit. Consequently, customers experience shorter

TABLE 3. Exact and approximate (tail) orbit length distributions at random epochs in an $M/IG/1$ retrial G-queue for different server states.

n	I_n		P_n		Q_n		R_n	
	Exact distribution	Tail distribution	Exact distribution	Tail distribution	Exact distribution	Tail distribution	Exact distribution	Tail distribution
0	0.02137531	0.02137531	0.01974203	0.01631808	0.01258977	0.01634608	0.00480613	0.00547697
1	0.00974163	0.01000477	0.01589342	0.01551646	0.01510193	0.01554308	0.00514639	0.00520791
2	0.00948031	0.00951329	0.01477081	0.01475421	0.01475258	0.01477953	0.00494640	0.00495207
3	0.00904175	0.00904595	0.01402694	0.01402941	0.01405470	0.01405349	0.00470822	0.00470889
4	0.00860108	0.00860157	0.01333947	0.01334024	0.01336375	0.01336311	0.00447742	0.00447749
5	0.00817897	0.00817902	0.01268477	0.01268488	0.01270676	0.01270667	0.00425752	0.00425753
6	0.00777722	0.00777723	0.01206173	0.01206174	0.01510193	0.01208244	0.00404838	0.00404838
7	0.00739517	0.00739517	0.01146921	0.01146921	0.01475258	0.01148889	0.00384950	0.00384950
8	0.00703188	0.00703188	0.01090579	0.01090579	0.01405470	0.01092450	0.00366040	0.00366040
9	0.00668644	0.00668644	0.01037004	0.01037004	0.01336375	0.01038783	0.00348058	0.00348058
10	0.00635797	0.00635797	0.00986061	0.00986061	0.00987753	0.00987753	0.00330960	0.00330960
20	0.00384196	0.00384196	0.00595852	0.00595852	0.00596875	0.00596875	0.00199991	0.00199991
30	0.00232160	0.00232160	0.00360059	0.00360059	0.00360677	0.00360677	0.00120849	0.00120849
40	0.00140289	0.00140289	0.00217574	0.00217574	0.00217948	0.00217948	0.00073026	0.00073026
50	0.00084773	0.00084773	0.00131475	0.00131475	0.00131700	0.00131700	0.00044128	0.00044128
60	0.00051226	0.00051226	0.00079447	0.00079447	0.00079583	0.00079583	0.00026665	0.00026665
70	0.00030954	0.00030954	0.00048008	0.00048008	0.00048090	0.00048090	0.00016113	0.00016113
80	0.00018705	0.00018705	0.00029010	0.00029010	0.00029059	0.00029059	0.00009736	0.00009736
90	0.00011303	0.00011303	0.00017530	0.00017530	0.00017560	0.00017560	0.00005883	0.00005883
100	0.00006830	0.00006830	0.00010592	0.00010592	0.00010611	0.00010611	0.00003555	0.00003555
110	0.00004127	0.00004127	0.00006401	0.00006401	0.00006412	0.00006412	0.00002148	0.00002148
120	0.00002494	0.00002494	0.00003868	0.00003868	0.00003874	0.00003874	0.00001298	0.00001298
130	0.00001507	0.00001507	0.00002337	0.00002337	0.00002341	0.00002341	0.00000784	0.00000784
140	0.00000910	0.00000910	0.00001412	0.00001412	0.00001414	0.00001414	0.00000474	0.00000474
150	0.00000550	0.00000550	0.00000853	0.00000853	0.00000854	0.00000854	0.00000286	0.00000286
160	0.00000332	0.00000332	0.00000515	0.00000515	0.00000516	0.00000516	0.00000173	0.00000173
170	0.00000200	0.00000200	0.00000311	0.00000311	0.00000312	0.00000312	0.00000104	0.00000104
180	0.00000121	0.00000121	0.00000188	0.00000188	0.00000189	0.00000189	0.00000006	0.00000006
190	0.00000007	0.00000007	0.00000113	0.00000113	0.00000115	0.00000115	0.00000003	0.00000003
200	0.00000004	0.00000004	0.00000006	0.00000006	0.00000007	0.00000007	0.00000002	0.00000002
210	0.00000002	0.00000002	0.00000004	0.00000004	0.00000005	0.00000005	0.00000001	0.00000001
220	0.00000001	0.00000001	0.00000002	0.00000002	0.00000004	0.00000004	0.00000000	0.00000000
230	0.00000000	0.00000000	0.00000001	0.00000001	0.00000002	0.00000002	0.00000000	0.00000000
240	0.00000000	0.00000000	0.00000000	0.00000000	0.00000001	0.00000001	0.00000000	0.00000000
250	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000
260	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000
Sum	0.22473466	0.22503553	0.33598986	0.33217571	0.32852326	0.33274571	0.11075218	0.11149087
	$L_o^I = 4.14542320$,		$L_o^P = 6.43008948$,		$L_o^Q = 6.440235189$,		$L_o^R = 2.15797520$,	
	$L_o = 19.17372308$, $W_o = 2.739103297$							

waiting times which leads to a decrease in the average waiting time (W_o) as illustrated in Figure 5. Conversely, an increase in the arrival rate of positive customers (λ_1) causes more customers to join the orbit. This process leads to a longer average waiting time (W_o) as demonstrated in Figure 5. As the repair rate (δ) increases, the server becomes available more quickly and customers in the orbit are able to retry faster. Consequently, the number of customers in the orbit decreases. This reduction also leads to a decrease in the average waiting time (W_o) as illustrated in Figure 6. Similarly, when the retrial rate (α) increases, customers retry from the orbit more frequently leads to a further reduction of W_o as depicted in Figure 6.

TABLE 4. Exact and approximate (tail) orbit length distributions at random epochs in an $M/Ga/1$ retrial G-queue for different server states.

n	I_n		P_n		Q_n		R_n	
	Exact distribution	Tail distribution	Exact distribution	Tail distribution	Exact distribution	Tail distribution	Exact distribution	Tail distribution
0	0.03841330	0.03841330	0.02121440	0.01790917	0.01298788	0.01460988	0.00862236	0.01097341
1	0.02154615	0.02188214	0.01697886	0.01669307	0.01366910	0.01361782	0.00978976	0.01022828
2	0.02034523	0.02039627	0.01556970	0.01555956	0.01274576	0.01269312	0.00946063	0.00953374
3	0.01900610	0.01901130	0.01450160	0.01450301	0.01184259	0.01183122	0.00887581	0.00888637
4	0.01772007	0.01772036	0.01351788	0.01351821	0.01102940	0.01102784	0.00828170	0.00828295
5	0.01651708	0.01651709	0.01260023	0.01260027	0.01027915	0.01027901	0.00772041	0.00772051
6	0.01539552	0.01539552	0.01174467	0.01174467	0.00958103	0.00958103	0.00719626	0.00719626
7	0.01435012	0.01435012	0.01094717	0.01094717	0.00893044	0.00893044	0.00670761	0.00670761
8	0.01337569	0.01337569	0.01020382	0.01020382	0.00832404	0.00832404	0.00625214	0.00625214
9	0.01246744	0.01246744	0.00951095	0.00951095	0.00775881	0.00775881	0.00582760	0.00582760
10	0.01162086	0.01162086	0.00886512	0.00886512	0.00723196	0.00723196	0.00543189	0.00543189
15	0.00817604	0.00817604	0.00623719	0.00623719	0.00508817	0.00508817	0.00382169	0.00382169
20	0.00575238	0.00575238	0.00438827	0.00438827	0.00357985	0.00357985	0.00268881	0.00268881
25	0.00404717	0.00404717	0.00308744	0.00308744	0.00251866	0.00251866	0.00189175	0.00189175
30	0.00284745	0.00284745	0.00217221	0.00217221	0.00177204	0.00177204	0.00133097	0.00133097
35	0.00200337	0.00200337	0.00152829	0.00152829	0.00124675	0.00124675	0.00093642	0.00093642
40	0.00140950	0.00140950	0.00107525	0.00107525	0.00087717	0.00087717	0.00065883	0.00065883
45	0.00099167	0.00099167	0.00075651	0.00075651	0.00061714	0.00061714	0.00046353	0.00046353
50	0.00069771	0.00069771	0.00053225	0.00053225	0.00043420	0.00043420	0.00032612	0.00032612
55	0.00049088	0.00049088	0.00037447	0.00037447	0.00030549	0.00030549	0.00022945	0.00022945
60	0.00034537	0.00034537	0.00026347	0.00026347	0.00021493	0.00021493	0.00016143	0.00016143
65	0.00024299	0.00024299	0.00018536	0.00018536	0.00015121	0.00015121	0.00011358	0.00011358
70	0.00017095	0.00017095	0.00013041	0.00013041	0.00010639	0.00010639	0.00007991	0.00007991
75	0.00012028	0.00012028	0.00009175	0.00009175	0.00007485	0.00007485	0.00005622	0.00005622
80	0.00008462	0.00008462	0.00006455	0.00006455	0.00005266	0.00005266	0.00003955	0.00003955
85	0.00005953	0.00005953	0.00004542	0.00004542	0.00003705	0.00003705	0.00002783	0.00002783
90	0.00004189	0.00004189	0.00003195	0.00003195	0.00002606	0.00002606	0.00001958	0.00001958
95	0.00002947	0.00002947	0.00002248	0.00002248	0.00001834	0.00001834	0.00001377	0.00001377
100	0.00002073	0.00002073	0.00001581	0.00001581	0.00001290	0.00001290	0.00000969	0.00000969
110	0.00001026	0.00001026	0.00000783	0.00000783	0.00000638	0.00000638	0.00000479	0.00000479
120	0.00000508	0.00000508	0.00000387	0.00000387	0.00000316	0.00000316	0.00000237	0.00000237
130	0.00000251	0.00000251	0.00000191	0.00000191	0.00000156	0.00000156	0.00000117	0.00000117
140	0.00000124	0.00000124	0.00000009	0.00000009	0.00000007	0.00000007	0.00000005	0.00000005
150	0.00000006	0.00000006	0.00000004	0.00000004	0.00000003	0.00000003	0.00000002	0.00000002
160	0.00000003	0.00000003	0.00000002	0.00000002	0.00000001	0.00000001	0.00000001	0.00000001
170	0.00000001	0.00000001	0.00000001	0.00000001	0.00000000	0.00000000	0.00000000	0.00000000
180	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000
190	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000
200	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000
Sum	0.36027495	0.36066749	0.26734430	0.26374493	0.21365195	0.21515695	0.15872876	0.16160334
	$L_o^I = 4.74532061$,		$L_o^P = 3.62067623$,		$L_o^Q = 2.95361576$,		$L_o^R = 2.21767611$,	
	$L_o = 13.53728872$, $W_o = 1.80497183$							

Figures 7 and 8 illustrate that the probability I_0 increases as the retrial rate (α) rises. This happens because a higher retrial rate prompts customers in the orbit to retry more frequently and thereby increasing I_0. Additionally, Figure 7 shows that I_0 increases with the repair rate (δ). A higher repair rate makes the server available more quickly which allows orbiting customers to retry sooner which enhances the probability I_0. Conversely, Figure 8 demonstrates that an increase in the arrival rate of positive customers (λ_1) causes more customers to join the orbit leads to a decline in I_0. The effects of various values of b and p on I_0 and W_0 are depicted

TABLE 5. Exact and approximate (tail) orbit length distribution at post-departure epoch.

	$M/IG/1$ retrial G-queue		$M/Ga/1$ retrial G-queue	
n	Ω_n^+		Ω_n^+	
	Exact distribution	Tail distribution	Exact distribution	Tail distribution
0	0.04591633	0.04927793	0.07018020	0.06996490
1	0.04709637	0.04685715	0.06368898	0.06379032
2	0.04457376	0.04455530	0.05888796	0.05887628
3	0.04235757	0.04236653	0.05482958	0.05481913
4	0.04028266	0.04028528	0.05109449	0.05109113
5	0.03830584	0.03830627	0.04762237	0.04762187
6	0.03642443	0.03642448	0.04438823	0.04438819
7	0.03463513	0.03463513	0.04137408	0.04137408
8	0.03293364	0.03293369	0.03856464	0.03856464
9	0.03131582	0.03131582	0.03594597	0.03594597
10	0.02977744	0.02977744	0.03350512	0.03350512
20	0.01799377	0.01799377	0.01658519	0.01658519
30	0.01087319	0.01087319	0.00820974	0.00820974
40	0.00657040	0.00657040	0.00406386	0.00406386
50	0.00397033	0.00397033	0.00201163	0.00201163
60	0.00239917	0.00239917	0.00099576	0.00099576
70	0.00144976	0.00144976	0.00049290	0.00049290
80	0.00087605	0.00087605	0.00024399	0.00024399
90	0.00052937	0.00052937	0.00012077	0.00012077
100	0.00031989	0.00031989	0.00005978	0.00005978
110	0.00019330	0.00019330	0.00002959	0.00002959
120	0.00011680	0.00011680	0.00001464	0.00001464
130	0.00007058	0.00007058	0.00000725	0.00000725
140	0.00004265	0.00004265	0.00000358	0.00000358
150	0.00002577	0.00002577	0.00000177	0.00000177
160	0.00001557	0.00001557	0.00000008	0.00000008
170	0.00000941	0.00000941	0.00000004	0.00000004
180	0.00000568	0.00000568	0.00000002	0.00000002
190	0.00000343	0.00000343	0.00000001	0.00000001
200	0.00000207	0.00000207	0.00000000	0.00000000
210	0.00000125	0.00000125	0.00000000	0.00000000
220	0.00000007	0.00000007	0.00000000	0.00000000
230	0.00000004	0.00000004	0.00000000	0.00000000
240	0.00000002	0.00000002	0.00000000	0.00000000
250	0.00000001	0.00000001	0.00000000	0.00000000
260	0.00000000	0.00000000	0.00000000	0.00000000
	0.99999999	0.99999999	0.99999999	0.99999999

in Figures 9 and 10, respectively. Figure 9 shows that the probability I_0 decreases as the probability of opting for optional service p increases. This happens because a higher probability of optional service keeps the server busier. Thus, more arriving positive customers join the orbit which reduces I_0. In contrast, the values of W_o increase as shown in Figure 10. Further, Figure 9 depicts a decrease in I_0 as the parameter b increases. This occurs because the probability of balking decreases with an increase in b. Hence, more customers join the orbit which leads to a reduction in I_0. Additionally, the increase in the number of customers in the orbit results in a higher mean waiting time W_o as shown in Figure 10.

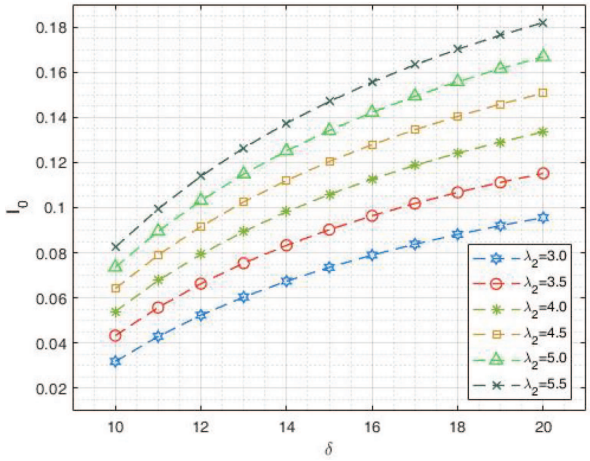

FIGURE 3. Effect of λ_2 on I_0 for different values of δ.

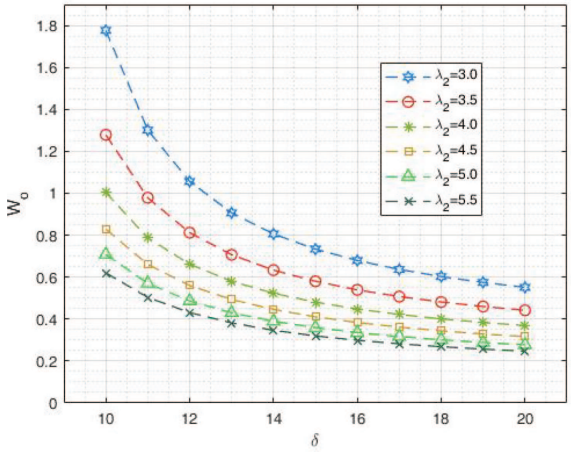

FIGURE 4. Effect of λ_2 on W_o for different values of δ.

Figure 11 demonstrates the effect of λ_2 on L_o for various values of δ. As observed in Figure 11, L_o decreases with an increase in δ. This is because a higher repair rate (δ) allows the server to become available more quickly which yields customers in the orbit to retry faster. As a result, the number of customers in the orbit decreases which yields a reduction in L_o. Additionally, as the arrival rate of negative customers (λ_2) increases, L_o also decreases. This occurs because frequent interruptions by negative customers cause the server to become idle more often after repairs, thereby reducing L_o. The effects of various values of b and p on L_o are illustrated in Figure 12. It can be observed that L_o increases as the probability of opting for the optional service p increases. This is because a higher probability of optional service keeps the server engaged for longer periods which yields more arriving positive customers to join the orbit, and consequently L_o increases. Furthermore, when the value of b increases, L_o also increases because a higher probability of joining the orbit leads to more customers entering it which in turn enlarges L_o. The effects of varying α and λ_1 on L_o are shown in Figure 13. As α increases L_o decreases because a higher retrial rate allows customers in the orbit to retry more quickly and thus shortens the orbit length. In contrast, increasing λ_1 brings more customers into the orbit which enlarges its length.

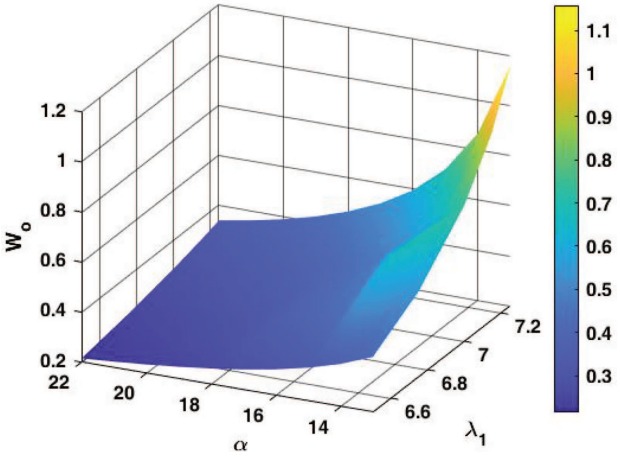

FIGURE 5. W_o for different pairs of α and λ_1.

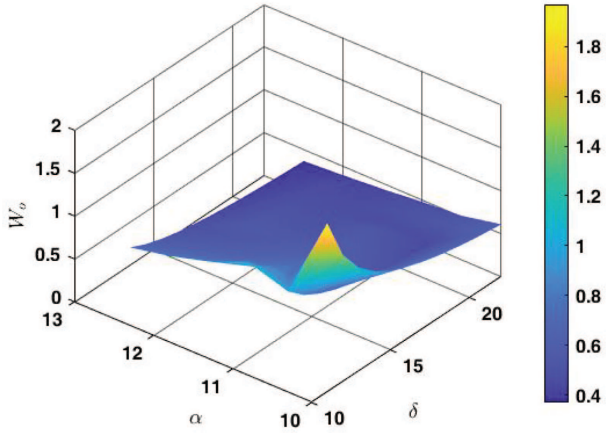

FIGURE 6. W_o for different pairs of α and δ.

6.2. Cost analysis

Cost analysis is an essential component in every real-world scenario that affects every aspect of operations. The holding cost, service cost, retrial cost, and repair cost are considered in this study. Naturally, management aims to minimize the total average cost (TAC) of the system. To address this objective, we develop a cost function for the proposed queueing model in this subsection and derive the TAC based on the following assumptions:

- C_0: The holding cost for each customer present in the orbit per unit time.
- C_1: The cost associated with maintaining the service rate κ_1 for FPS per unit time.
- C_2: The cost associated with maintaining the service rate κ_2 for SPS per unit time.
- C_3: The cost required to maintain the retrial rate α per unit time.
- C_4: The cost required to maintain the repair rate η per unit time.

The TAC per unit time for the system is defined as follows:

$$\text{TAC} = C_0 L_o + C_1 \kappa_1 + C_2 \kappa_2 + C_3 \alpha + C_4 \delta. \tag{69}$$

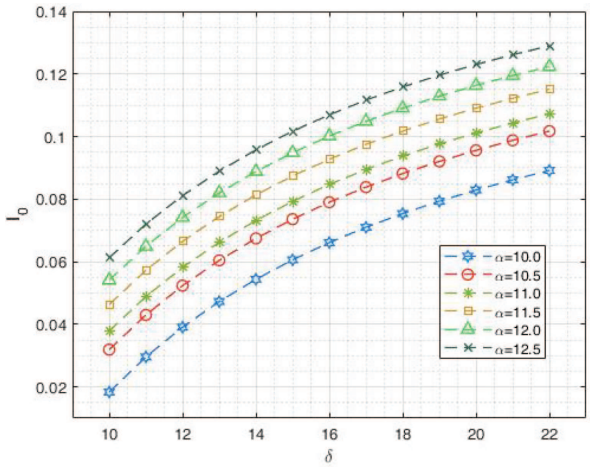

FIGURE 7. Effect of δ on I_0 for different values of α.

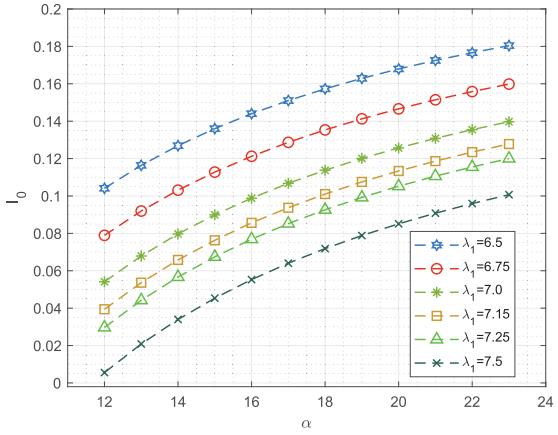

FIGURE 8. Effect of α on I_0 for different values of λ_1.

Tables 6, 7, and 8 provide a comprehensive cost analysis of the queueing system by evaluating key parameters such as the arrival rates of positive customers (λ_1) and negative customers (λ_2), the service rates of the first and second phases of service (κ_1 and κ_2), the retrial rate (α), and the repair rate (δ). The fixed cost parameters are set to $C_0 = 1.0$, $C_1 = 0.5$, $C_2 = 0.5$, $C_3 = 0.4$, and $C_4 = 0.3$ to make comparisons straightforward across different scenarios. These tables help to determine the most cost-effective parameter combinations by demonstrating the effects of changes in λ_1, λ_2, κ_1, κ_2, α, and δ on the total system cost. These tables emphasize the lowest TAC for different configurations, which are highlighted in bold to indicate the optimal parameter settings. For instance, Table 6 shows that the combination of $\lambda_2 = 3$ and $\delta = 16$ yields the lowest TAC of 21.28933, which makes it the optimal choice in that case. Similarly, Table 7 indicates that $\delta = 16$ and $\alpha = 11$ yield the lowest TAC of 21.05571. Table 8 further demonstrates that $\alpha = 19$ and $\lambda_1 = 7.5$ give the lowest TAC of 21.05571. These findings provide system managers with valuable guidance in selecting the optimal parameter values to minimize cost and enhance system performance.

Now, we determine the optimal retrial rate (α^*) and repair rate (δ^*) and the corresponding minimum total average cost (TAC*) by employing genetic algorithm and particle swarm optimization algorithm.

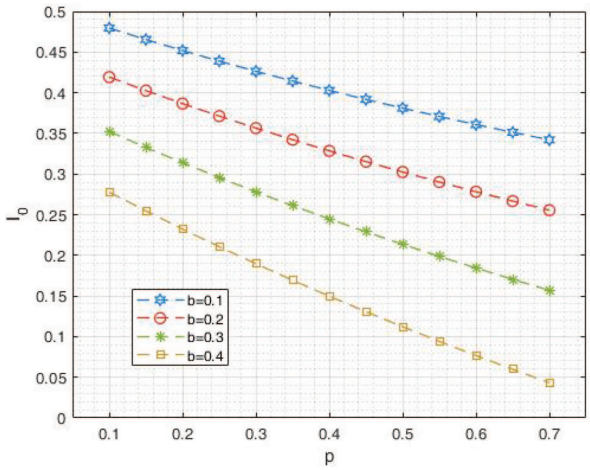

FIGURE 9. Effect of p on I_0 for different values of b.

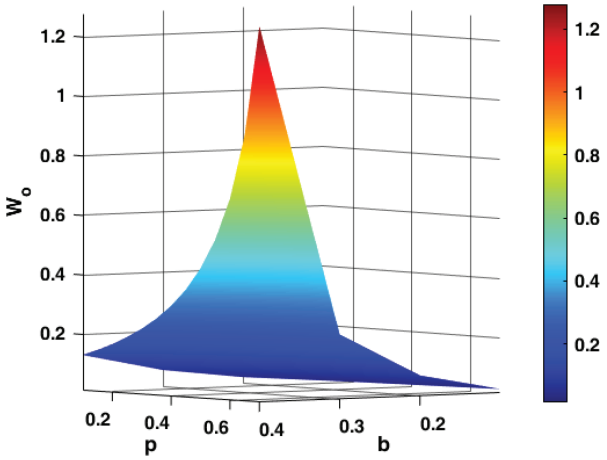

FIGURE 10. Effect of p on W_o for different values of b.

TABLE 6. Cost analysis for different pairs of λ_2 and δ.

δ	$\lambda_2 = 3.00$	$\lambda_2 = 3.5$	$\lambda_2 = 4.0$	$\lambda_2 = 4.5$	$\lambda_2 = 5$	$\lambda_2 = 5.5$
10	27.17084	23.68516	21.75816	20.53650	19.69356	19.07734
11	24.14462	21.88459	20.54372	19.65688	19.02741	18.55787
12	22.72823	21.02032	19.96768	19.25464	18.74016	18.35179
13	21.98152	20.58107	19.69720	19.08923	**18.64588**	**18.35179**
14	21.57699	20.37039	**19.59644**	**19.05841**	18.66311	18.36070
15	21.37065	**20.29667**	19.59964	19.17296	18.74619	18.47343
16	**21.28933**	20.31143	19.67110	19.21982	18.88502	18.62703
17	21.29191	20.38666	19.78976	19.36712	19.05252	18.80949
18	21.35350	20.50502	19.94245	19.54255	19.24410	19.01308
19	21.45815	20.65522	20.12035	19.73902	19.45376	19.23260
20	21.59520	20.82957	20.31760	19.95163	19.67735	19.46440

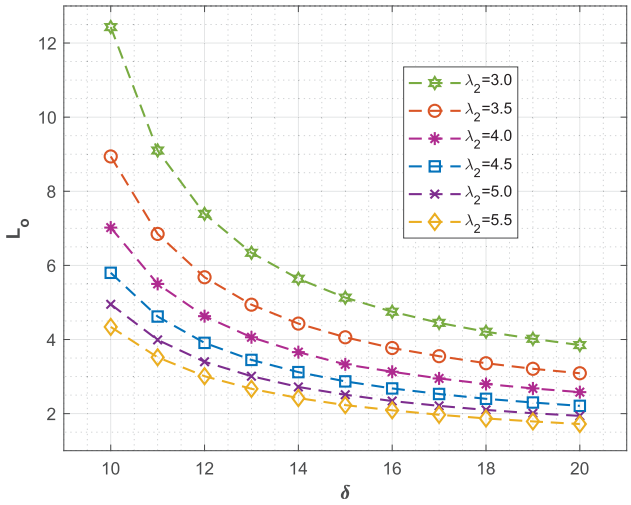

FIGURE 11. Effect of δ on L_o for different values of λ_2.

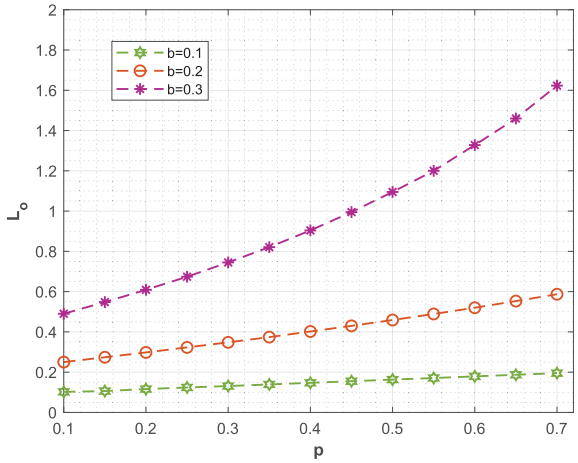

FIGURE 12. Effect of p on L_o for different values of b.

Genetic algorithm for TAC minimization

The concept of genetic algorithm (GA) was introduced by Bremermann [8] and his colleagues during the 1960s and 1970s. The GA is based on the principle of natural selection, where the fittest individuals are chosen for reproduction to produce successive generations of improved solutions. By simulating biological evolution through selection, crossover, and mutation, GAs enable adaptation to changing environments. Many researchers, including Mathavavisakan *et al.* [33], Sanga and Antala [40], and Sanga and Jain [41], have successfully used it because of its adaptability. The GA is particularly suitable for this optimization because TAC is nonlinear and depends on interpolated data. The algorithm operates with a randomly generated population, evaluates the fitness function, and uses the standard GA operators: selection, crossover, and mutation. Elitism is applied to preserve the best solution, and the process continues until the stopping criterion (number of generations) is met. GA parameters used for TAC minimization are given in Table 9 and L_0 is obtained from interpolation of tabulated values provided in Table 10.

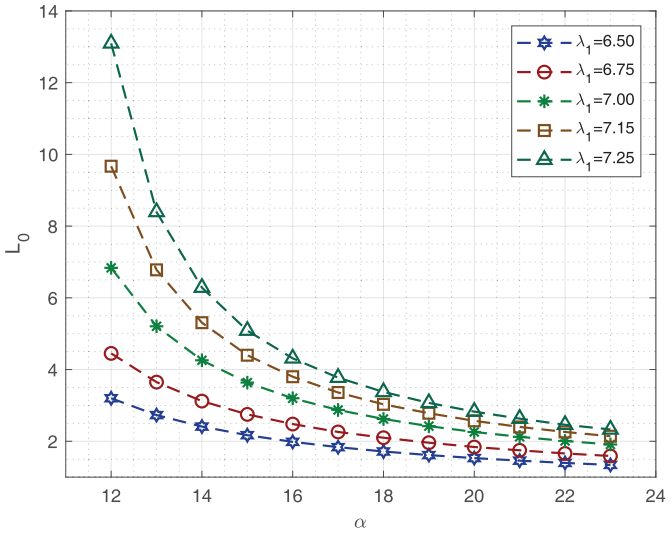

FIGURE 13. Effect of α on L_o for different values of λ_1.

TABLE 7. Cost analysis for different pairs of α and δ.

δ	$\alpha = 10.0$	$\alpha = 10.5$	$\alpha = 11.0$	$\alpha = 11.5$	$\alpha = 12$	$\alpha = 12.5$
10	37.02462	27.17084	25.19078	23.28613	22.13669	21.39782
11	28.58668	24.14462	23.06729	21.93829	21.22171	20.75136
12	25.41265	22.72823	22.03228	21.26624	20.77094	20.44656
13	23.84841	21.98152	21.48762	20.92553	20.56145	**20.32708**
14	22.98621	21.57699	21.20543	20.77172	**20.49327**	20.31926
15	22.49269	21.37065	21.08002	**20.73359**	20.51491	20.38387
16	22.21665	**21.28933**	**21.05571**	20.77203	20.77203	20.49812
17	22.07958	21.29191	21.10037	20.86373	20.86373	20.64784
18	**22.03668**	21.35350	21.19415	20.99396	20.99396	20.82380
19	22.06054	21.45815	21.32416	21.15298	21.15298	21.01970
20	22.13342	21.59520	21.48164	21.33405	21.33405	21.23109
21	22.24335	21.75722	21.66046	21.53238	21.53238	21.45478
22	22.38198	21.93890	21.85616	21.74449	21.69404	21.68838

The algorithm employs the following steps:

(1) **Initialization of population:** A population of candidate solutions, each representing a pair (α, δ), is randomly generated within the bounds $\alpha \in [10.0, 12.5]$ and $\delta \in [10, 22]$.
(2) **Fitness evaluation:** The fitness of each candidate is evaluated using the TAC function given in (69). The value of L_0 corresponding to each (α, δ) pair is obtained by interpolation of tabulated values.
(3) **Selection:** The fittest candidates are selected to produce offspring for the next generation using tournament selection.
(4) **Crossover:** Blend crossover is applied to generate new candidates by combining the selected parents, ensuring both exploration and exploitation of the search space.
(5) **Mutation:** Small random perturbations are applied to maintain diversity and avoid premature convergence. The variables are clamped within bounds after mutation to ensure valid interpolation.

TABLE 8. Cost analysis for different pairs of α and λ_1.

α	$\lambda_1 = 6.5$	$\lambda_1 = 6.75$	$\lambda_1 = 7.0$	$\lambda_1 = 7.15$	$\lambda_1 = 7.25$	$\lambda_1 = 7.5$
13	23.84841	21.98152	20.90687	22.48753	24.09575	34.34250
14	22.98621	21.57699	20.35801	21.40759	22.38542	27.07889
15	22.49269	21.37065	20.13709	20.90703	21.58677	24.42334
16	22.21665	**21.28933**	**20.09914**	20.70130	21.21396	23.18038
17	22.07958	21.29191	20.17368	**20.66599**	**21.07430**	22.55435
18	**22.03668**	21.35350	20.32230	20.73805	21.07610	22.25270
19	22.06054	21.45815	20.52229	20.88212	21.17019	**22.14260**
20	22.13342	21.59520	20.75936	21.07682	21.32782	22.15502
21	22.24335	21.75722	21.02408	21.30848	21.53105	22.25057
22	22.38198	21.93890	21.30996	21.56795	21.76811	22.40507

TABLE 9. GA parameters used for TAC minimization.

Parameter	Value
Population size P_s	80
Number of generations	200
Crossover method	Blend crossover
Mutation probability (p_m)	0.15
Mutation step	Small local perturbation
α bounds	10.0–12.5
δ bounds	10–22
Elitism	Best individual preserved each generation

(6) **Elitism and stopping criteria:** The best solution from each generation is preserved (elitism). The GA stops after a predefined number of generations (200 generations in this study).

The proposed GA successfully determined the optimal values α^* and δ^* that minimize the TAC. The algorithm yields $\alpha^* = 12.5000$ and $\delta^* = 13.6032$ with a corresponding minimum $\text{TAC}^* = 20.302813$. The convergence of the GA over successive generations is shown in Figure 14 which demonstrates steady improvement and stability of the optimization process. Similarly, when the search range for δ is restricted to the range 16–22, the GA identifies $\alpha^* = 12.5000$ and $\delta^* = 16.0000$ which gives a minimum $\text{TAC}^* = 20.490000$. Both results are in close agreement with the values reported in Table 7 which validate the accuracy of our approach.

Particle swarm optimization for TAC minimization

The stochastic optimization technique known as particle swarm optimization (PSO) was first introduced by Kennedy and Eberhart [26] for solving optimization problems involving complex non-linear functions. PSO algorithm is employed in the present study to determine the optimal values of the decision variables α and δ that minimize TAC. PSO is a population-based metaheuristic designed by the social behavior of birds and fish, where each candidate solution, called a particle, adjusts its position in the search space based on its own experience and the experience of neighboring particles. In this approach, particles are initialized randomly within the bounds of $\alpha \in [10.0, 12.5]$ and $\delta \in [10, 22]$, and the TAC corresponding to each particle is evaluated using the interpolated values of L_0 from Table 7. The particles update their velocities and positions iteratively, moving toward the global best solution while maintaining diversity in the swarm. The process continues until a predefined number of iterations is reached with the best solution being preserved throughout. Using this method, PSO efficiently identifies the optimal α^* and δ^* values, and the convergence of the algorithm over iterations is illustrated in Figure 15. PSO parameters used for TAC minimization are given in Table 11 and L_0

TABLE 10. L_o for different pairs of α and δ.

δ	$\alpha = 10.0$	$\alpha = 10.5$	$\alpha = 11.0$	$\alpha = 11.5$	$\alpha = 12$	$\alpha = 12.5$
10	22.52	12.43	10.29	8.18	6.83	5.89
11	13.78	9.10	7.86	6.53	5.62	4.95
12	10.31	7.39	6.53	5.56	4.87	4.34
13	8.44	6.34	5.68	4.92	4.36	3.92
14	7.28	5.64	5.10	4.47	3.99	3.61
15	6.49	5.13	4.68	4.13	3.71	3.38
16	5.91	4.75	4.35	3.87	3.49	3.19
17	5.47	4.45	4.10	3.66	3.32	3.04
18	5.13	4.21	3.89	3.49	3.17	2.92
19	4.86	4.02	3.72	3.35	3.05	2.81
20	4.63	3.85	3.58	3.23	2.95	2.73
21	4.44	3.72	3.46	3.13	2.86	2.65
22	4.28	3.60	3.35	3.04	2.79	2.58

TABLE 11. PSO parameters used for TAC minimization.

Parameter	Value
Number of particles (N)	30
Number of iterations	300
Inertia weight (w)	0.6
Cognitive coefficient (c1)	1.5
Social coefficient (c2)	1.5
Position bounds (α)	[10.0, 12.5]
Position bounds (δ)	[10, 22]
Velocity initialization	0
Penalty for invalid interpolation	1×10^6

TABLE 12. Comparative performance of GA and PSO under different parameter settings.

Algorithm	Parameter setting	Mean conv. iteration	α^*	δ^*	TAC*
GA	$P_s = 60$, $p_m = 0.10$	37	12.50	13.6032	20.3028
GA	$P_s = 80$, $p_m = 0.15$	29	12.50	13.6032	20.3028
GA	$P_s = 100$, $p_m = 0.20$	31	12.50	13.6032	20.3028
PSO	$N = 50$, $w = 0.6$	16	12.50	13.6032	20.3028
PSO	$N = 100$, $w = 0.7$	14	12.50	13.6032	20.3028
PSO	$N = 150$, $w = 0.8$	13	12.50	13.6032	20.3028

is obtained from interpolation of tabulated values provided in Table 10. Using PSO, the algorithm converges to the optimal values $\alpha^* = 12.5000$ and $\delta^* = 13.6032$ with a corresponding minimum TAC$^* = 20.302813$ which are same as obtained by GA. This demonstrates that both algorithms yield consistent and reliable results for the minimization of TAC.

To further assess the robustness of the proposed optimization framework, both GA and PSO were executed multiple times under different parameter settings. As reported in Table 12, both algorithms consistently converged to the same optimal decision variables (α^*, δ^*) and the corresponding minimum TAC value across all

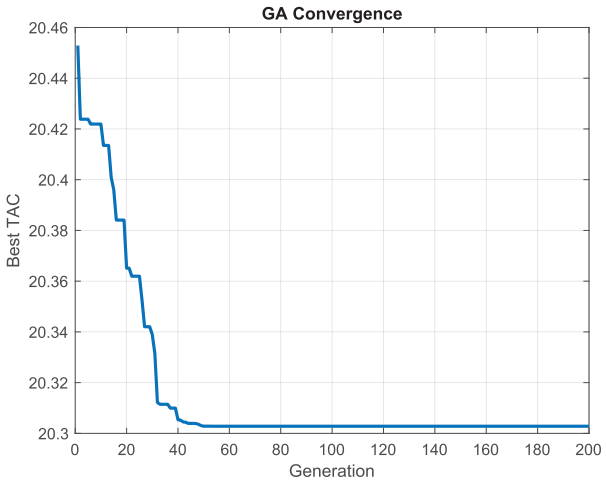

FIGURE 14. GA convergence curve.

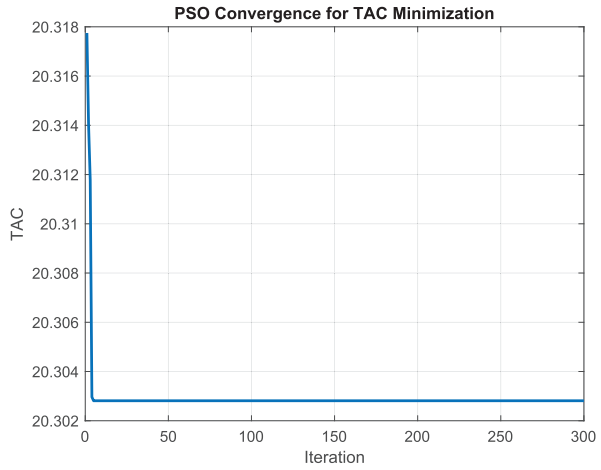

FIGURE 15. PSO convergence curve.

parameter configurations. This indicates the presence of a strong global minimum and confirms the reliability of the obtained solutions.

However, noticeable differences are observed in the convergence behavior of the two algorithms. PSO achieves convergence in significantly fewer iterations compared to GA, reflecting its faster exploitation capability driven by social and cognitive learning mechanisms. In contrast, GA requires a larger number of generations to converge and exhibits higher variability in convergence iterations due to its stronger exploration characteristics introduced by crossover and mutation operations.

Overall, the comparative results demonstrate that while both GA and PSO are effective and robust for TAC minimization, PSO offers faster convergence, whereas GA provides a more exploratory search. These complementary behaviors further validate the effectiveness of the proposed optimization approach.

6.3. Managerial insights

The numerical analysis of the queueing system provides valuable information on system performance and cost behavior. Based on these findings, the following managerial insights can help improve efficiency and guide decision-making:

(1) The probability of having any number of requests in the orbit at a given moment can be calculated. As shown in Table 3, the probability of more than 230 requests in the orbit is almost negligible, which assures managers that extreme congestion is unlikely.
(2) In Table 3, it is found that the system maintains an average orbit size of about 20 requests with a mean waiting time of around 3 seconds and both within acceptable performance limits.
(3) In Table 3, it is also observed that the server remains busy approximately 67% of the time, idle about 22%, and in repair around 11%. This indicates that while the system achieves good utilization, a notable fraction of time is lost due to idleness and repairs. Reducing repair duration and better balancing arrival/retrial dynamics could improve efficiency further.
(4) By analyzing the cost from Tables 6 to 8, manager can identify the most cost-effective parameter settings. Even small adjustments in parameters can significantly reduce the TAC. For instance, as shown in Table 6, the minimum cost occurs for $\delta \in [13, 16]$ which indicates that manager should aim for balanced parameters to minimize cost.

7. Conclusion

This study has examined the steady-state behavior of an $M/G/1$ retrial queue with balking, optional service, negative customers, and general retrial times under constant retrial policy. We derive differential-difference equations with supplementary variables to determine the probability generating function and orbit length distribution at random epoch. Stability criteria is established, and the PASTA principle is utilized to connect post-departure and random epochs for determining the orbit length distribution at post-departure epoch. Comprehensive performance measures are evaluated through numerical analyses which highlight the impact of retrial time, service time, and repair time distributions. A detailed cost analysis underscores economic implications and practical relevance of the model. This study has used advanced metaheuristic algorithms including particle swarm optimization and the genetic algorithm to optimize total cost. The proposed methodology can also be extended to the analysis of the $M^X/G/1$ retrial G-queue with balking, optional service, and general retrial times under vacation policies.

Acknowledgments

The authors would like to acknowledge the anonymous reviewers and editors for their insightful comments and suggestions to improve the quality of this manuscript.

References

[1] D. Arivudainambi and P. Godhandaraman, Retrial queueing system with balking, optional service and vacation. *Ann. Oper. Res.* **229** (2015) 67–84.
[2] J.R. Artalejo, G-networks: a versatile approach for work removal in queueing networks. *Eur. J. Oper. Res.* **126** (2000) 233–249.
[3] J.R. Artalejo, Accessible bibliography on retrial queues: progress in 2000–2009. *Math. Comput. Modell.* **51** (2010) 1071–1081.
[4] J.R. Artalejo and G. Choudhury, Steady state analysis of an $M/G/1$ queue with repeated attempts and two-phase service. *Qual. Technol. Quant. Manage.* **1** (2004) 189–199.
[5] I. Atencia and P. Moreno, A single-server G-queue in discrete-time with geometrical arrival and service process. *Perform. Eval.* **59** (2005) 85–97.

[6] S.P. Bala Murugan and A.J. Pauline Veronica, An $M/G/1$ retrial queue with recurrent customers, general retrial times and working vacation. *App. Appl. Math.* **20** (2025).

[7] B. Bank and S.K. Samanta, Analytical and computational studies of the $BMAP/G^{(a,Y)}/1$ queue. *Commun. Stat.-Theory Methods* **50** (2021) 3586–3614.

[8] H.J. Bremermann, The Evolution of Intelligence: The Nervous System as a Model of its Environment. University of Washington, Department of Mathematics (1958).

[9] H. Bruneel, B. Steyaert, E. Desmet and G.H. Petit, Analytic derivation of tail probabilities for queue lengths and waiting times in ATM multiserver queues. *Eur. J. Oper. Res.* **76** (1994) 563–572.

[10] G. Choudhury, Some aspects of an $M/G/1$ queueing system with optional second service. *Top* **11** (2003) 141–150.

[11] G. Choudhury, An $M/G/1$ retrial queue with an additional phase of second service and general retrial time. *Int. J. Inf. Manage. Sci.* **20** (2009) 1–14.

[12] E. Desmet, B. Steyaert, H. Bruneel and G.H. Petit, Tail distributions of queue length and delay in discrete-time multiserver queueing models, applicable in ATM networks, in Proc. ITC'13 (1991) 1–6.

[13] I. Dimitriou, The $M/G/1$ retrial queue with event-dependent arrivals. Preprint arXiv:2203.02757 (2022).

[14] B. Doshi, Analysis of a two phase queueing system with general service times. *Oper. Res. Lett.* **10** (1991) 265–272.

[15] G.I. Falin, Single-line repeated orders queueing systems. *Optimization* **17** (1986) 649–667.

[16] G.I. Falin and J.G.C. Templeton, Retrial Queues. Chapman and Hall, London (1997).

[17] S. Gao and J. Wang, Stochastic analysis of a preemptive retrial queue with orbital search and multiple vacations. *RAIRO-Oper. Res.* **54** (2020) 231–249.

[18] E. Gelenbe, Random neural networks with negative and positive signals and product form solution. *Neural Comput.* **1** (1989) 502–510.

[19] E. Gelenbe, The first decade of G-networks. *Eur. J. Oper. Res.* **126** (2000) 231–232.

[20] M.M.N. GnanaSekar and I. Kandaiyan, Analysis of an $M/G/1$ retrial queue with delayed repair and feedback under working vacation policy with impatient customers. *Symmetry* **14** (2022) 2024.

[21] A. Gómez-Corral, Stochastic analysis of a single server retrial queue with general retrial times. *Nav. Res. Logistics (NRL)* **46** (1999) 561–581.

[22] D. Gross, J.F. Shortle, J.M. Thompson and C.M. Harris, Fundamentals of Queueing Theory. Vol. 627. John Wiley & Sons (2011).

[23] R. Harini and K. Indhira, Dynamical behaviour and meta heuristic optimization of a hospital management software system. *Heliyon* **10** (2024). (In Press)

[24] M. Jain and S.S. Sanga, Admission control for finite capacity queueing model with general retrial times and state-dependent rates. *J. Ind. Manage. Opt.* **16** (2020).

[25] V.A. Kapyrin, A study of the stationary distributions of a queueing system with recurring demands. *Cybernetics* **13** (1977) 584–590.

[26] J. Kennedy and R. Eberhart, Particle swarm optimization, in Proceedings of ICNN'95-International Conference on Neural Networks (2002).

[27] J. Kim and B. Kim, A survey of retrial queueing systems. *Ann. Oper. Res.* **247** (2016) 3–36.

[28] B.K. Kumar, A. Vijayakumar and D. Arivudainambi, An $M/G/1$ retrial queueing system with two-phase service and preemptive resume. *Ann. Oper. Res.* **113** (2002) 61–79.

[29] Z. Liu, J. Wu and G. Yang, An $M/G/1$ retrial G-queue with preemptive resume and feedback under N-policy subject to the server breakdowns and repairs. *Comput. Math. App.* **58** (2009) 1792–1807.

[30] T.H. Liu, J.C. Ke, C.C. Kuo and F.M. Chang, On the retrial queue with imperfect coverage and delay reboot. *RAIRO-Oper. Res.* **55** (2021) S1229–S1248.

[31] K.C. Madan, An $M/G/1$ queue with second optional service. *Queue. Syst.* **34** (2000) 37–46.

[32] S.P. Madheswari, B.K. Kumar and P. Suganthi, Analysis of $M/G/1$ retrial queues with second optional service and customer balking under two types of Bernoulli vacation schedule. *RAIRO-Oper. Res.* **53** (2019) 415–443.

[33] N.M. Mathavavisakan, K. Indhira and S.I. Ammar, Cost optimization and analysis of retrial queues with hybrid vacation policies and fault correction: a metaheuristic approach. *RAIRO-Oper. Res.* **59** (2025) 2683–2720.

[34] J. Medhi, A single server Poisson input queue with a second optional channel. *Queue. Syst.* **42** (2002) 239–242.

[35] S. Meziani and T. Kernane, Extended generator and associated martingales for $M/G/1$ retrial queue with classical retrial policy and general retrial times. *Prob. Eng. Inf. Sci.* **37** (2023) 206–213.

[36] C. Morfopoulou, Network routing control with G-networks. *Perform. Eval.* **68** (2011) 320–329.

[37] S. Pradhan, U.C. Gupta and S.K. Samanta, Queue-length distribution of a batch service queue with random capacity and batch size dependent service: $M/G_r^Y/1$. *Opsearch* **53** (2016) 329–343.

[38] S.K. Samanta and B. Bank, Analysis of stationary queue-length distributions of the $BMAP/R^{(a,b)}/1$ queue. *Int. J. Comput. Math. Comput. Syst. Theory* **5** (2020) 198–223.

[39] S.K. Samanta and K.M. Rashmi, Analytical and computational aspects of a batch arrival retrial queue with a constant retrial policy. *Qual. Technol. Quant. Manage.* (2025) 1–28.

[40] S.S. Sanga and K.S. Antala, Performance analysis and cost investigations for state-dependent single unreliable server finite queue under F-policy using GA and QNM. *J. Comput. Appl. Math.* **441** (2024) 115679.

[41] S.S. Sanga and M. Jain, Cost optimization and ANFIS computing for admission control of $M/M/1/K$ queue with general retrial times and discouragement. *Appl. Math. Comput.* **363** (2019) 124624.

[42] K.M. Rashmi and S.K. Samanta, Analysis of two-way communication in $M_1^X, M_2/G_1, G_2/1$ retrial queue under the constant retrial policy. *Methodol. Comput. Appl. Probab.* **26** (2024) 1–32.

[43] J.F. Shortle, P.H. Brill, M.J. Fischer, D. Gross and D.M. Masi, An algorithm to compute the waiting time distribution for the $M/G/1$ queue. *INFORMS J. Comput.* **16** (2004) 152–161.

[44] J. Wu and X. Yin, An $M/G/1$ retrial G-queue with non-exhaustive random vacations and an unreliable server. *Comput. Math. App.* **62** (2011) 2314–2329.

[45] J. Xu, L. Liu and K. Wu, Analysis of a retrial queueing system with priority service and modified multiple vacations. *Commun. Stat.-Theory Methods* **52** (2023) 6207–6231.

[46] F. Xu, R. Tian and Q. Shao, Optimal pricing strategy in an unreliable $M/G/1$ retrial queue with Bernoulli preventive maintenance. *RAIRO-Oper. Res.* **57** (2023) 2639–2657.

[47] T. Yang, M.J.M. Posner, J.G. Templeton and H. Li, An approximation method for the $M/G/1$ retrial queue with general retrial times. *Eur. J. Oper. Res.* **76** (1994) 552–562.