

Stochastic Modeling and Metaheuristic Optimization of Priority Retrial G-queue under Working Breakdown

NISHA¹, SHWETA UPADHYAYA^{2*}, DIVYA AGARWAL¹, CHANDRA SHEKHAR³

¹Department of Mathematics, Amity Institute of Applied Sciences,
Amity University, Uttar Pradesh, Noida
INDIA

²School of Computer Science Engineering and Technology,
Bennett University, Greater Noida,
INDIA

³Department of Mathematics, Birla Institute of Technology & Science,
Pilani, Rajasthan,
INDIA

**Corresponding Author*

Abstract: -The objective of this research is to investigate a novel retrial G-queue with one server. The concepts of preemptive priority, partial breakdown (working breakdown), feedback, and semi-active vacation (working vacation) are taken into consideration. There are two categories of consumers, which include priority consumers and ordinary consumers. Priority customers receive service before ordinary customers according to a non-preemptive priority discipline. Whenever the orbit becomes empty at the moment of service accomplishment for priority and ordinary consumers, the server takes several semi-active vacations. Firstly, the probability-generating functions for the framework/orbit by employing the supplementary variable approach have been deduced. Afterwards, several pertinent system performance metrics, viz., queue length and system size are addressed. Additionally, for the comparative analysis of optimal cost, we have employed the direct search method (DSM), particle swarm optimization (PSO), and teaching-learning-based optimization (TLBO). Convergence of various techniques is presented through graphs and finally various numerical instances have been given.

Key-Words: Particle Swarm Optimization, Teaching Learning Based Optimization, Retrial, G-queue, partial breakdown, Semi-active vacation, Priority consumers

Received: April 29, 2026. Revised: June 2, 2026. Accepted: July 5, 2026. Published: September 30, 2026.

1 Introduction

The assumption related to queuing theory in the majority of literature reviews is that an attendant (server) is always accessible in the framework (system). In most circumstances, the attendant is always reliable and ready to serve the consumers. We occasionally come across situations where an attendant may break down and need to be repaired. In some cases, throughout the industrial process, the framework will break down due to technical and network failures. When studying the waiting line framework (queuing system) with attendant failure, it is commonly believed that the attendant will stop serving whenever there are malfunctions in the framework. The concept of partial breakdown (working breakdown) in a single-server framework with infinite capacity was introduced in [1]. According to them, obstructive consumers (negative customers) could cause the framework to malfunction at any stage when the normal busy attendant remains operational. Therefore, the framework needs to be repaired with a backup (standby) attendant when the primary attendant malfunctions. The service rate of the backup attendant varies (lower than) that of the primary attendant. Once the repair is completed successfully, the primary attendant re-enters the framework and becomes functional. Later, a single server waiting line framework with

general distribution in the context of obstructive consumers and failure is explored in [2]. For an extensive literature review of the framework with attendant failure, readers may follow [3] and [4]. Recently, the queuing behavior of a single-server machine repair problem with a semi-active vacation, a retrieve waiting line (retrial queue), attendant failure, and dual-phase repair procedures has been addressed in [5].

Throughout the past several decades, an upsurge in curiosity has been observed in queuing theory with obstructive consumers, which is known as the G-queue. The concept of the G-queue to visualize the modeling of neural networks in [6] was initially introduced in queuing theory. In his study of a random neural network model, he represented excitatory and inhibitory neuron behaviour using positive and negative signals. In addition, he came up with a product-form solution that made it easier and more effective to analyze complex stochastic neural networks. Moreover, the obstructive consumers can exhibit behaviour such as breakdowns, call losses, load balancing, killing signals, etc. The obstructive consumers have the following range of consequences on the consumers' behaviours: (1) all the consumers are eliminated from the system. (2) Remove the consumers in front of the queue, even if one is currently being served.

(3) Towards the end of the queue, remove the consumers. In the multiprocessor computer system, telecommunication system, and computer networks, the queue with obstructive consumers has a manifold of applications. The theory of compositional reversed Markov processes with applications to G-networks has been addressed in [7]. Further, a review on the waiting line framework with G-networks and obstructive consumers, along with applications has been given in [8]. Later, several queuing models that operate with the simultaneous existence of obstructive consumers have been described in [9], [10], [11], [12]. Recently, researchers [13] evaluated the steady-state behaviour of a retrieve waiting line with features like balking, optional service, obstructive consumers, and general retrial time under a constant retrial policy. Also, in [14] the concept of a discrete-time Geo/Geo/1 G-queue with bernoulli feedback mechanisms, k-optional services, and an unstable server is given.

The term *semi-active vacation* refers to a classical queuing model generalization. In, [15] the M/M/1 waiting list framework with semi-active vacation was initially introduced. After that, the extension of M/M/1/WV to an M/G/1/WV was performed [16]. Lately, retrieve waiting lines with general retrial times and a single semi-active vacation have been investigated in [17]. For a variety of semi-active vacation queueing models, we refer the reader to the quick surveys by many researchers [18], [19]. Semi-active vacation is preferable to vacation since the attendant provides service at a decreased speed in place of shutting down completely during the vacation period. Moreover, if there are consumers at the time of service completion, the attendant can halt the vacation and return to the normal busy state. This type of policy is referred to as a vacation interruption policy. Lately, [20] presented the cost optimization and steady state behavior of only one attendant with semi-active vacation and partial breakdown using the matrix geometric approach.

Furthermore, when the high-priority consumer arrives, the lower-priority consumer may be pushed out in some system even if the service of the lower priority is still in progress. At the completion of the line, the consumer may receive feedback to ensure that they receive satisfactory service. Priority customers are served here before ordinary consumers. The investigation on the priority retrieve waiting line framework with bernoulli vacation and server malfunctions is provided in [21]. They employed the supplementary variable technique to obtain the steady-state probabilities for both the queuing framework and the reliability quantities. In [22] the retrieve waiting line framework with two

kinds of consumers was explored by the author. He derived the probability distribution and performance metrics by implementing the supplementary variable technique with the sojourn time of an arbitrary number of consumers. In [23] researchers focus on the performance evaluation of a single-server preemptive-priority retrieve waiting line framework with immediate Bernoulli feedback. In [24] a retrieve G-queue framework with preemptive priority arrivals in which the attendant takes a semi-vacation is presented. Moreover, looking into [25], the idea of a $Geo^X/G/1$ priority retrieve waiting line framework under the bernoulli vacation is inspected by the researchers. Priority retrieval queues are utilized in a variety of applications like manufacturing systems, operating platforms, real-time frameworks, and simulations.

Priority queues are ideal for customers who are pressed for time and want to avoid the queue. There are significant research gaps concerning priority consumers with a semi-active vacation and partial breakdown, according to the review of the literature stated above. This effort seeks to fill that gap, and the present research is concentrated on delivering performance evaluation by reducing the cost for state-dependent, bulk arrival reiterates G-queue with partial breakdown and semi-active vacation with priority. Nevertheless, no research has been done yet on priority queues with semi-active vacation and partial breakdown.

The following is a summary of the remaining sections: Section 2 describes the concise mathematical model and its applications in the real world after briefly presenting the necessary assumptions. Section 3 provides the notation and probability, the mathematical equations for the stability state, and the solution of the stability state. Section 4 includes metrics of the system performance, average system size, queue length. Numerical analysis with illustrations, some effects of parameters are discussed in Section 5. Next, Section 6 contains optimal cost comparison and convergence graphs using PSO (Particle swarm optimization) and TLBO (Teaching learning based optimization), as well as cost evaluation using DSM (direct search method). Finally, in Section 7, conclusion and potential outcomes are discussed.

2 Problem Formulation

The following details are included in this section. Here we inspect a single-server retrieve G-queue a waiting line framework with partial breakdown, semi-active vacation, and priority. Following is a more in-depth description of the model. Consumers enter the framework with a rate δ using a

Poisson process. We presumptively believe that the waiting area is not accessible. As a result, if a new consumer arrives and finds the attendant available, the consumer immediately gets the service. If a consumer encounters an attendant that is consistently busy, the arrivals will be added to a pool of blocked consumers known as orbit, with the FCFS (first-come, first-served) discipline. That is, only the first consumer in the orbit (waiting pool) has access to the server. The inter-retrieve epoch has a random distribution $\hat{\mathcal{R}}(t)$ with Laplace Stieltjes Transform (LST) $\hat{\mathcal{R}}^*(\vartheta)$.

The service times follow a general distribution during a usual busy period. The random variable $\mathcal{B}_b$ is used to represent it with distribution function (d.f) $\mathcal{B}_b(t)$ and LST $\mathcal{B}_b^*(\vartheta)$. Here, partial breakdown and semi-active vacation periods are considered lower speed service periods in this case.

Almost any time when an attendant is vacant then it starts semi-active vacation and an exponential distribution of vacation time is followed in this case, with parameter ϑ . If a consumer enters during a vacation moment, the attendant continues to operate at a lower (reduced) speed service rate $v_w < v_b$ (regular service rate). Suppose any consumer is in orbit at a lower service completion moment during vacation time. In that case, the attendant will end the vacation and return to the typical engaged period, resulting in vacation interruption, and if not, the vacation will continue. If there are consumers in the orbit when the vacation finishes, the attendant resumes normal operations; otherwise, the attendant will go on a vacation moment again.

At any time, obstructive consumers makes the system inoperable whenever the normal busy attendant is in operation with the exponentially distributed rate of $\tilde{\alpha}$. As a result of the obstructive consumers, all current consumers are compelled to abandon the framework, causing the primary attendant to fail. When the main attendant fails, it is immediately dispatched to be repaired, and the repair period begins. The exponential distribution with a rate of τ is followed by a repair time. The repaired server is expected to perform as a brand-new server. However, if obstructive consumers strike on a normal busy server, the server will stop running i.e., partial breakdown. The attendant behaves as a substitute server and provides a reduced level of service to arriving consumers during the partial breakdown $v_w < v_b$. If there are customers in the orbit when a repair is completed, the server returns to normal working level, and a new busy period is initiated. Otherwise, it is unoccupied and ready to serve new consumers. The service time follows a generic random variable $\mathcal{B}_w$ with d.f. $\mathcal{B}_w(t)$ and LST $\mathcal{B}_w^*(\vartheta)$ during semi-active

vacation and partial breakdown periods (reduced speed services). Feedback in queuing systems is a technique that allows users to return to the system (or a portion of it) for additional services or to try another service after receiving a service. This may be the result of incomplete service or discontent with the first service. We use feedback to optimize resource allocation, fulfill customer requests, and improve system performance. Here, we are taking the feedback probability, i.e., σ .

2.1 Justification (applications) for the Suggested Strategy

This analysis is very pertinent to the IT industry. Consider the TCSiON app as an example. Tata Consultancy Services has a strategic division called TCSiON that focuses on educational institutions, manufacturing industries (SMB) and testing bodies. There are so many features but we are here interested in the faculty management system. When we use this app there are so many solutions for faculty such as HRMS solutions, Payroll solutions, Procurement, inventory, self-service, and SIMS.

Consider a self-service facility under the HRMS solution sector in TCSiON, which was created to improve management in higher education. Here, a staff member (Positive consumer) applies for leave. This app has a single attendant (manager) on staff who handles all your inquiries. The consumer can hang up the app and depart the framework submitting a leave application request, or they can inquire about other features, such as purchase view inventory, direct dispatch, etc. (optional services). Newly joined faculties and other relieving faculties (priority consumers) are considered as priority consumers. An individual may need to request features again to confirm his joining/relieving if there was a communication error brought on by network ambiguity or any other similar problem (feedback). When the HR/manager is overseeing other areas of the business, he is unable to answer the request (in vacation mode). In these circumstances, the assistant manager (substitute server) typically serves relatively more slowly than usual (semi-active vacation). If there are inquiries in the framework during this phase, once all the inquiries are over (at the service completion), the HR/manager will return right away (i.e., vacation interruption occurs). However, if no inquiries arrive after the HR/manager has done his secondary task (vacation is over), he or she continues to manage other faculty management modules (vacation mode). When a consumer sends a request, there is a chance that the server will be busy; the consumer then tries again after a specific amount of time has passed (retrial). There is a chance that a customer could lose service due to

an improper signal, inadequate network coverage, or a malware attack (obstructive consumer) while using an app on a laptop or phone. This situation is a perfect match for the model we are considering. Therefore, the system manager will greatly benefit from all of the findings and analyses offered in our work. Section 5 provides a numerical illustration of the scenario that was covered.

3 Steady State Analysis

This section contains the following information: By considering the lapsed (elapsed) retry time, the lapsed priority service time, lapsed service time, and the lapsed lower (reduced) speed service times as supplementary variable while using supplementary variable technique (SVT), we establish the steady state difference differential equations for the priority retrieve waiting framework. Next, we determine the probability generating functions (PGFs) of the number of consumers in the framework and orbit, along with the generating functions (GFs) of the orbit size for various states of the framework.

3.1 Probabilities and Annotations

We presume that $\hat{\mathcal{R}}(0) = 0$, $\hat{\mathcal{R}}(\infty) = 1$, $\mathcal{B}_p(0) = 0$, $\mathcal{B}_p(\infty) = 1$, $\mathcal{B}_b(0) = 0$, $\mathcal{B}_b(\infty) = 1$, $\mathcal{B}_w(0) = 0$, $\mathcal{B}_w(\infty) = 1$ are continuous at $h=0$ in a continuous state. In the next section, the notations and probabilities listed below are employed. (i) $\tilde{r}(x) \equiv$ the instantaneous rate for retrial time distribution; i.e., $\tilde{r}(x)dx = \frac{d\hat{\mathcal{R}}(x)}{1-\hat{\mathcal{R}}(x)}$. (ii) $v_b(h) \equiv$ the instantaneous rate for service time distribution; i.e., $v_b(h)dh = \frac{d\mathcal{B}_b(h)}{1-\mathcal{B}_b(h)}$. (iii) $v_w(h) \equiv$ the instantaneous rate for lower speed service time distribution i.e., $v_w(h)dh = \frac{d\mathcal{B}_w(h)}{1-\mathcal{B}_w(h)}$. (iv) $v_p(h) \equiv$ the instantaneous rate for priority service time distribution; i.e., $v_p(h)dx = \frac{d\mathcal{B}_p(h)}{1-\mathcal{B}_p(h)}$. (v) $\mathcal{Y}(t) \equiv$ Amount of overall consumers in the orbit. (vi) $\hat{\mathcal{R}}^0(t) \equiv$ the lapsed time of retry. (vii) $\mathcal{B}_b^0(t) \equiv$ the lapsed time of service. (viii) $\mathcal{B}_w^0(t) \equiv$ the lapsed time of lower speed service. (ix) $\mathcal{B}_p^0(t) \equiv$ the passing priority time. (x) $\check{\varphi}_0(t) \equiv$ At time t , the statistical possibilities that the framework is vacant. (xi) $\mathcal{W}_0(t) \equiv$ the statistical possibility that the framework is vacant and it is on vacation and experiencing a breakdown at a time t . (xii) $\hat{\mathcal{R}}_n(h, t) \equiv$ the statistical possibility that there are precisely n consumers in the orbit and that the test consumer's passing retry time falls amidst h and $h + dh$ at time t . (xiii) $\psi_{1,n}(h, t) \equiv$ the statistical possibility that there are precisely n consumers in the orbit and that the test consumer's passing normal service time with priority consumers falls amidst h and $h + dh$ at time t . (xiv) $\psi_{b,n}(h, t) \equiv$ the statistical possibility that

there are precisely n consumers in the orbit and that the test consumer's passing normal service time with ordinary consumers falls amidst h and $h + dh$ at time t . (xv) $\mathcal{W}_n(h, t) \equiv$ the statistical possibility that there are precisely n consumers in the orbit and that the test consumer's passing lower service time falls amidst h and $h + dh$ at time t .

3.2 The State Governing Mathematical Equations

We suppose the arbitrary variable so that the priority retrial G-queue model can be developed further as given below.

$$\mathcal{E}(t) = \begin{cases} 0, & \text{whenever attendant is not being} \\ & \text{used and is set to semi - active vacation} \\ & \text{and partially breakdown.} \\ 1, & \text{whenever attendant is inert and} \\ & \text{in normal service.} \\ 2, & \text{whenever attendant is engaged by} \\ & \text{a priority consumer without depriving} \\ & \text{an ordinary consumer and at} \\ & \text{time } t \text{ in a normal service period.} \\ 3, & \text{whenever attendant is engaged} \\ & \text{with an ordinary consumer and} \\ & \text{in normal service period at a time } t. \\ 4, & \text{whenever attendant is engaged} \\ & \text{and at a time } t \text{ in lower service} \\ & \text{period.} \end{cases}$$

To obtain the bivariate Markov procedure, we will introduce the supplementary variables denoted by $\hat{\mathcal{R}}^0(t)$, $\mathcal{B}_p^0(t)$, $\mathcal{B}_b^0(t)$ and $\mathcal{B}_w^0(t)$ corresponding to lapsed retrieve period, lapsed service period of priority without preemptive ordinary consumer, lapsed service period of ordinary consumer and lapsed lower speed service time, respectively. The server state (0,1,2,3,4) is denoted by $\mathcal{E}(t)$, which is characterized by the attendant being inactive, inert in normal service, engaged with priority, engaged with an ordinary consumer, and engaged with lower speed, respectively. $\hat{\mathcal{R}}^0(t)$ describes the lapse time of retrieve if we consider $\mathcal{Y}(t) > 0$ and $\mathcal{E}(t) = 1$. $\mathcal{B}_p^0(t)$ describes the lapse time of priority consumer without preemptive ordinary consumer is served during a busy normal (regular) service if we consider $\mathcal{Y}(t) \geq 0$ and $\mathcal{E}(t) = 2$. Further $\mathcal{B}_b^0(t)$ denotes the lapsed time of the ordinary consumer in normal service if we take $\mathcal{Y}(t) \geq 0$ and $\mathcal{E}(t) = 3$ and eventually on the condition that $\mathcal{Y}(t) \geq 0$ and $\mathcal{E}(t) = 4$ then $\mathcal{B}_w^0(t)$ demonstrate the lapsed time of lower (reduced) speed service. We will settle the possibility $\check{\varphi}_0(t) = \check{\varphi} \{ \mathcal{E}(t) =$

$0, \mathcal{Y}(t) = 0\}$ and $\check{Q}_0(t) = \check{\varphi} \{\mathcal{E}(t) = 0, \mathcal{Y}(t) = 0\}$ for the procedure $\{\mathcal{Y}(t), t \geq 0\}$. The sequence of epochs, which is defined by $\{t_n; n = 1, 2, 3, 4, \dots\}$, at which a lower service period completion or a normal service completion is observed. A Markov chain is formed by the sequence of random vectors $Z_n := \{\mathcal{E}(t_n+), \mathcal{Y}(t_n+)\}$ which is embedded in the retrieve waiting line framework. It follows from Appendix A that the $\{Z_n : n \in \mathcal{Y}\}$ is ergodic if and only if $\varrho < \hat{\mathcal{R}}^*(\check{\delta})$, where equation (1) represents ϱ , i.e.,

$$\varrho = \sigma \mathcal{B}_b^*(\check{\alpha} + \check{\delta}) - \check{\delta}. \quad (1)$$

Also, probability densities are as follows:

$$\begin{aligned} \hat{\mathcal{R}}_n(h, t)dh &= \check{\varphi} \{\mathcal{E}(t) = 1, \mathcal{Y}(t) = n, \\ &h \leq \hat{\mathcal{R}}^0(t) < h + dh\}, \quad (2) \\ \text{for } h \geq 0, t \geq 0, n \geq 1. \end{aligned}$$

$$\begin{aligned} \psi_{1,n}(h, t)dh &= \check{\varphi} \{\mathcal{E}(t) = 2, \mathcal{Y}(t) = n, \\ &h \leq \mathcal{B}_p^0(t) < h + dh\}, \quad (3) \\ \text{for } h \geq 0, t \geq 0, n \geq 0. \end{aligned}$$

$$\begin{aligned} \psi_{b,n}(h, t)dh &= \check{\varphi} \{\mathcal{E}(t) = 3, \mathcal{Y}(t) = n, \\ &h \leq \mathcal{B}_b^0(t) < h + dh\}, \quad (4) \\ \text{for } h \geq 0, t \geq 0, n \geq 0. \end{aligned}$$

$$\begin{aligned} \mathcal{W}_n(h, t)dh &= \check{\varphi} \{\mathcal{E}(t) = 4, \mathcal{Y}(t) = n, \\ &h \leq \mathcal{B}_w^0(t) < h + dh\}, \quad (5) \\ \text{for } h \geq 0, t \geq 0, n \geq 0 \end{aligned}$$

The stability criteria is assumed to be met in the sequel by which we can set $\check{\varphi}_0 = \lim_{t \rightarrow \infty} \check{\varphi}_0(t)$, $\check{Q}_0 = \lim_{t \rightarrow \infty} \check{Q}_0(t)$ limiting densities for $(h, f) > 0; n \geq 0$, Using (2) to (5), we will get (6) to (8)

$$\hat{\mathcal{R}}_n(h) = \lim_{t \rightarrow \infty} \hat{\mathcal{R}}_n(h, t), \psi_{1,n}(h) = \lim_{t \rightarrow \infty} \psi_{1,n}(h, t) \text{ and} \quad (6)$$

$$\psi_{b,n}(h) = \lim_{t \rightarrow \infty} \psi_{b,n}(h, t) \quad (7)$$

$$\mathcal{W}_n(h) = \lim_{t \rightarrow \infty} \mathcal{W}_n(h, t) \quad (8)$$

Through supplementary variable technique, we are able to derive the following framework of

mathematical equations that regulates the mechanics of the framework operation.

$$(\check{\delta} + \mathfrak{I})\check{\varphi}_0 = \tau \mathcal{W}_0 \quad (9)$$

$$\begin{aligned} (\check{\delta} + \mathfrak{I} + \vartheta + \tau) \mathcal{W} &= \vartheta \mathcal{W} + \kappa \int_0^\infty \psi_{b,0} v_b(h) dh \\ &+ \kappa \int_0^\infty \mathcal{W}_0(h) v(h) dh + \int_0^\infty \psi_{1,0}(h) v_p(h) dh \\ &+ (\check{\alpha} + \mathfrak{I}) \int_0^\infty \psi_{b,n}(h) dh + \check{\alpha} \int_0^\infty \psi_{1,n}(h) dh \quad (10) \end{aligned}$$

$$\frac{d}{dh} \hat{\mathcal{R}}_n(h) + (\check{\delta} + \mathfrak{I} + \check{r}(h)) \hat{\mathcal{R}}_n(h) = 0, \quad n \geq 1 \quad (11)$$

$$\frac{d}{dh} \psi_{1,0}(h) + (\check{\delta} + \check{\alpha} + v_p(h)) \psi_{1,0}(h) = 0, \quad n = 0 \quad (12)$$

$$\begin{aligned} \frac{d}{dh} \psi_{1,n}(h) + (\check{\delta} + \check{\alpha} + v_p(h)) \psi_{1,n}(h) \\ = \check{\delta} \psi_{1,n-1}(h), \quad n \geq 1 \quad (13) \end{aligned}$$

$$\begin{aligned} \frac{d}{dh} \psi_{b,0}(h) + (\check{\delta} + \mathfrak{I} + \check{\alpha} + v_b(h)) \psi_{b,0}(h) = 0 \\ , n = 0 \quad (14) \end{aligned}$$

$$\begin{aligned} \frac{d}{dh} \psi_{b,n}(h) + (\check{\delta} + \mathfrak{I} + \check{\alpha} + v_b(h)) \psi_{b,n}(h) \\ = \check{\delta} \psi_{b,n-1}(h) \quad n \geq 1 \quad (15) \end{aligned}$$

$$\begin{aligned} \frac{d}{dh} \mathcal{W}_0(h) + (\check{\delta} + \vartheta + \tau + v_w(h)) \mathcal{W}_0(h) = 0, \\ n = 0 \quad (16) \end{aligned}$$

$$\begin{aligned} \frac{d}{dh} \mathcal{W}_n(h) + (\check{\delta} + \vartheta + \tau + v_w(h)) \mathcal{W}_n(h) \\ = \check{\delta} \mathcal{W}_{n-1}(h) \quad n \geq 1 \quad (17) \end{aligned}$$

At $h=0$ and $f=0$, boundary conditions of the steady state are as follows:

$$\begin{aligned} \hat{\mathcal{R}}_n(0) &= \int_0^\infty \psi_{1,n}(h) v_p(h) dh + \kappa \int_0^\infty \psi_{b,n}(h) \\ &v_b(h) dh + \sigma \int_0^\infty \psi_{b,n-1}(h) dh + \kappa \int_0^\infty \mathcal{W}_n(h) \\ &v_w(h) dh + \sigma \int_0^\infty \mathcal{W}_{n-1}(h) v_w(h) dh, \quad n \geq 1 \quad (18) \end{aligned}$$

$$\psi_{1,n}(0) = \mathfrak{I} \int_0^\infty \hat{\mathfrak{R}}_n(h) dh, \quad n \geq 1 \quad (19)$$

$$\psi_{b,0}(0) = \int_0^\infty \hat{\mathfrak{R}}_1(h) \tilde{r}(h) dh + (\vartheta + \tau) \times \int_0^\infty \mathcal{W}_0(h) dh + (\delta + \mathfrak{I}) \check{\varphi}_0, \quad n = 0 \quad (20)$$

$$\psi_{b,n}(0) = \int_0^\infty \hat{\mathfrak{R}}_{n+1} \tilde{r}(h) dh + \delta \int_0^\infty \hat{\mathfrak{R}}_n(h) dh + (\vartheta + \tau) \int_0^\infty \mathcal{W}_n(h) dh, \quad n \geq 1 \quad (21)$$

$$\mathcal{W}_n(0) = \begin{cases} (\delta + \mathfrak{I}) \mathcal{W}_0, & n = 0 \\ 0, & n \geq 1 \end{cases} \quad (22)$$

Here is the normalizing condition,

$$\sum_{n=1}^\infty \int_0^\infty \hat{\mathfrak{R}}_n(h) dh + \sum_{n=0}^\infty \left[\int_0^\infty \psi_{1,n}(h) dh + \int_0^\infty \psi_{b,n}(h) dh + \int_0^\infty \mathcal{W}(h) dh \right] + \check{\varphi}_0 + \mathcal{W}_0 = 1 \quad (23)$$

3.3 The Stability State Response

The generating functions approach is used to acquire the stability state response of the equations written above, the details of the generating functions are mentioned below for $|m| \leq 1$:

$$\hat{\mathfrak{R}}(m) = \sum_{h=1}^\infty \sum_{n=1}^\infty \hat{\mathfrak{R}}_n(h) m^n, \quad (24)$$

$$\psi_1(m) = \sum_{h=1}^\infty \sum_{n=1}^\infty \psi_{1,n}(h) m^n \quad (25)$$

$$\psi_b(m) = \sum_{h=1}^\infty \sum_{n=1}^\infty \psi_{b,n}(h) m^n, \quad (26)$$

$$\mathcal{W}(m) = \sum_{h=1}^\infty \sum_{n=1}^\infty \mathcal{W}_n(h) m^n \quad (27)$$

We develop an expression for the PGF of the number of consumers in the priority retrieve obstructive advent waiting list framework with semi-active vacation, partial breakdown and in orbit at random epochs in this section which is given by (24), (25), (26), (27).

Theorem 3.1 *At arbitrary defined intervals, these are the partial generating functions whenever the server is vacant throughout the retrial time and on WV and WB, free in normal service, overloaded due to priority consumer and normal service, overloaded due to ordinary consumer and normal service, overloaded due to service period of lower speed.*

$$\hat{\mathfrak{R}}(h, m) = \hat{\mathfrak{R}}(0, m) [1 - \hat{\mathfrak{R}}(h)] e^{-(\delta + \mathfrak{I})} \quad (28)$$

$$\psi_1(h, m) = \psi_1(0, m) [1 - \mathcal{B}_p(h)] e^{-\mathcal{H}_p(m)h} \quad (29)$$

$$\psi_b(h, m) = \psi_b(0, m) [1 - \mathcal{B}_b(h)] e^{-\mathcal{H}_b(m)h} \quad (30)$$

$$\mathcal{W}(h, m) = \mathcal{W}(0, m) [1 - \mathcal{B}_w(h)] e^{-\mathcal{H}_w(m)h} \quad (31)$$

here,

$$\mathcal{H}_p(m) = \delta(1-m) + \tilde{\alpha}; \quad \mathcal{H}_b(m) = \delta(1-m) + \tilde{\alpha} + \mathfrak{I}, \quad (32)$$

$$\mathcal{H}_w(m) = \delta(1-m) + \vartheta + \tau, \quad (33)$$

$$\psi_b(0, m) = \frac{\hat{\mathfrak{R}}(0, m)}{m} [\hat{\mathfrak{R}}^*(\delta + \mathfrak{I}) + \delta m \bar{\mathcal{R}}^*(\delta + \mathfrak{I})] + (\tilde{\lambda} + \tilde{\delta}) \mathcal{W}_0 \mathcal{V}(m) + (\delta + \mathfrak{I}) \check{\varphi}_0 \quad (34)$$

here,

$$\mathcal{V}(m) = \frac{(\vartheta + \tau) (1 - \mathcal{B}_w^*(\mathcal{H}_w(m)))}{\mathcal{H}_w(m)}, \quad (35)$$

$$\hat{\mathfrak{R}}(0, m) = \psi_1(0, m) \mathcal{F}(m) + \psi_b(0, m) \mathcal{C}(m) + (\sigma m + \kappa) \times \mathcal{B}_w^*(\mathcal{H}_w(m)) \mathcal{W}(0, m) (\delta + \mathfrak{I}) \mathcal{W}_0 \tau \mathcal{W}_0 \quad (36)$$

here,

$$\mathcal{C}(m) = [(\sigma m + \kappa) \mathcal{B}_b^*(\mathcal{H}_b(m)) + \mathcal{S}(m)], \quad (37)$$

$$\mathcal{S}(m) = (\tilde{\alpha} + \mathfrak{I}) \bar{\mathcal{B}}^*(\mathcal{H}_b(m)), \quad (38)$$

$$\mathcal{F}(m) = [\mathcal{B}_p^*(\mathcal{H}_p(m)) + \tilde{\alpha} \bar{\mathcal{B}}_p^*(\mathcal{H}_p(m))] \quad (39)$$

$$\psi_1(0, m) = \mathfrak{I} \hat{\mathfrak{R}}(0, m) \hat{\mathfrak{R}}^*(\delta + \mathfrak{I}) \bar{\mathcal{B}}^*(\tilde{\alpha}) \quad (40)$$

$$\mathcal{W}(0, z) = (\delta + \mathfrak{I}) \mathcal{W}_0 \quad (41)$$

Proof: On multiplying equations (10) to (22) by the appropriate power of m and summing over n , also, using equations (9) and (22), by solving, we get these above equations from (28) to (41). (For detail explanation of the proof, see [12])

Theorem 3.2 Whenever the attendant is vacant and set to WV and WB, the attendant is vacant and in normal service, overloaded due to a priority consumer without preemptive an ordinary consumer and in normal service, overloaded due to ordinary consumer and in normal service period, overloaded due to period of lower speed, the stationary distribution of the number of consumers in the orbit under the stability condition $\rho < \mathcal{R}^*(\delta)$ is given by

$$\hat{\mathcal{R}}(m) = \frac{\text{Num1}(m)}{\text{Deno1}(m)} \quad (42)$$

$$\psi_1(m) = \mathfrak{I} \bar{\mathcal{B}}^*(\tilde{\alpha}) \frac{\text{Num1}(m)}{\text{Deno1}(m)} \quad (43)$$

$$\psi_b(m) = \psi_b(0, m) \bar{\mathcal{B}}_b^*(\mathcal{H}_b(m)) \quad (44)$$

$$\mathcal{W}(m) = \mathcal{W}(0, m) \bar{\mathcal{B}}_w^*(\mathcal{H}_w(m)) \quad (45)$$

$$\begin{aligned} \text{Num1}(m) = m \mathcal{W}_0 \hat{\mathcal{R}}^*(\delta + \mathfrak{I}) \{ ((\delta + \mathfrak{I}) \mathcal{V}(m) \\ + \tau) C(m) + (\delta + \mathfrak{I}) ((\sigma m + \kappa) \times \mathcal{B}_w^*(\mathcal{H}_w(m)) \\ - 1) - \tau \} \end{aligned} \quad (46)$$

$$\begin{aligned} \text{Deno1}(m) = m - m \mathfrak{I} \hat{\mathcal{R}}^*(\delta + \mathfrak{I}) \mathcal{F}(m) - \{ \hat{\mathcal{R}}^* \\ (\delta + \mathfrak{I}) + \delta m \hat{\mathcal{R}}^*(\delta + \mathfrak{I}) \} C(m) \end{aligned} \quad (47)$$

$$\begin{aligned} \text{Num1}'(1) = \mathcal{W}_0 (1 - \hat{\mathcal{R}}^*(\delta + \mathfrak{I})) \left\{ (1 - \mathcal{B}_w^*(\vartheta \\ + \tau)) \left[\rho + \frac{\delta}{\vartheta + \tau} - \sigma \right] + \frac{\tau}{\delta + \mathfrak{I}} \rho \right\} \end{aligned} \quad (48)$$

$$\begin{aligned} \text{Deno1}'(1) = (1 - \hat{\mathcal{R}}^*(\delta + \mathfrak{I})) \rho - \hat{\mathcal{R}}^*(\delta + \mathfrak{I}) \\ (\delta \rho + \delta + \mathfrak{I} (1 - \delta)) \end{aligned} \quad (49)$$

$$\text{Where, } \rho = \sigma \mathcal{B}_b^*(\tilde{\alpha} + \delta) - \delta$$

$$\begin{aligned} \text{Numb}'(1) = \mathcal{W}_0 ((\delta + \mathfrak{I}) \beta \{ \sigma \mathcal{B}_w^*(\vartheta + \tau) - \sigma \\ - \delta \mathcal{B}_w^*(\vartheta + \tau) \} + \gamma + \zeta + \chi) \times \{ \bar{\mathcal{B}}_b^*(\tilde{\alpha} + \mathfrak{I}) \} \end{aligned} \quad (50)$$

$$\begin{aligned} \mathcal{W}_0 = \frac{\text{Deno1}'(1)}{\left\{ \text{Deno1}'(1) \left[\left(\frac{\tau + \delta + \mathfrak{I}}{\delta + \mathfrak{I}} \right) + \frac{(\delta + \mathfrak{I}) \mathcal{V}(z)}{\vartheta + \tau} \right] \right.} \\ \left. \times \frac{1}{+(1 + \mathfrak{I}) \times \frac{\bar{\mathcal{B}}_p^*(\tilde{\alpha}) \text{Num1}'(1)}{\mathcal{W}_0} + \frac{\text{Numb}'(1)}{\mathcal{W}_0}} \right\}} \end{aligned} \quad (51)$$

$$\begin{aligned} \check{\varphi}_0 = \frac{\tau}{(\delta + \mathfrak{I})} \\ \times \left[\frac{\text{Deno1}'(1)}{\text{Deno1}'(1) \left[\left(\frac{\tau + \delta + \mathfrak{I}}{\delta + \mathfrak{I}} \right) + \frac{(\delta + \mathfrak{I}) \mathcal{V}(z)}{\vartheta + \tau} \right] + (1 + \mathfrak{I})} \right. \\ \left. \times \frac{1}{\bar{\mathcal{B}}_p^*(\tilde{\alpha}) \frac{\text{Num1}'(1)}{\mathcal{W}_0} + \frac{\text{Numb}'(1)}{\mathcal{W}_0}} \right] \end{aligned} \quad (52)$$

Proof: By integrating the equations from (28) to (41) with respect to h and assuming that the partial generating function is defined as subsequent as;

$$\begin{aligned} \hat{\mathcal{R}}(m) = \int_0^\infty \hat{\mathcal{R}}(h, m) dh, \psi_1(m) = \int_0^\infty \psi_1(h, m) \\ dh, \psi_b(m) = \int_0^\infty \psi_b(h, m) dh, \mathcal{W}(m) = \int_0^\infty \mathcal{W}(h, \\ m) dh, \end{aligned}$$

also making use of the normalizing condition (23), we can figure out the possibility that the attendant is inactive (idle) (52) when there is no consumer in the orbit by setting $m=1$ in equations (42) to (45), using equations (46) to (50), we will get (42) to (45), and after applying the L'Hospital rule where it is necessary then we acquire equation (53)

$$\check{\varphi}_0 + \mathcal{W}_0 + \hat{\mathcal{R}}(1) + \psi_1(1) + \psi_b(1) + \mathcal{W}(1) = 1. \quad (53)$$

We will also get (51) by using normalizing condition (23). (For detail explanation of the proof, see [12].)

4 System Performance Metrics

We estimate various zestful framework possibilities or (probabilities) while the framework is in different states in this section, as well as system performance metrics, and the average system size of this model.

4.1 System State Probabilities

Theorem 4.1 When the attendant is inactive (idle) during the retrieve time and in WV and WB, free in normal service, busy with priority consumer and in regular service, engaged with an ordinary consumer and in regular service, engaged in lower speed service period, the marginal probability generating functions are respectively given by:

(i) Let $\hat{\mathcal{R}}$ be the steady-state possibility of the attendant being inactive (idle) while retrieve time.

$$\hat{\mathcal{K}} = \hat{\mathcal{K}}(1) = \frac{\mathcal{W}_0 (1 - \hat{\mathcal{K}}^*(\delta + \mathfrak{I}))}{\{1 - \hat{\mathcal{K}}^*(\delta + \mathfrak{I})\varrho - \bar{\mathcal{K}}^*(\delta + \mathfrak{I})\}} \times \frac{\left\{ (1 - \mathcal{B}_w^*(\vartheta + \tau)) [\varrho + \frac{\delta}{\vartheta + \tau} - \sigma] + \frac{\tau}{\delta + \mathfrak{I}} \varrho \right\}}{(\delta \varrho + \delta + \mathfrak{I} (1 - \delta))} \quad (54)$$

where $\varrho = \sigma \mathcal{B}_b^*(\tilde{\alpha} + \mathfrak{I}) - \delta$

(ii) Let ψ_1 be the steady-state possibility of the attendant being engaged with a priority consumer in a normal service period at a time t.

$$\psi_1 = \psi_1(1) = \mathfrak{I} \bar{\mathcal{B}}_p^*(\tilde{\alpha}) \times \frac{\mathcal{W}_0 (1 - \hat{\mathcal{K}}^*(\delta + \mathfrak{I})) \left\{ (1 - \mathcal{B}_w^*(\vartheta + \tau)) \right\}}{\{1 - \hat{\mathcal{K}}^*(\delta + \mathfrak{I})\varrho - \bar{\mathcal{K}}^*(\delta + \mathfrak{I})\}} \times \frac{\left\{ [\varrho + \frac{\delta}{\vartheta + \tau} - \sigma] + \frac{\tau}{\delta + \mathfrak{I}} \varrho \right\}}{(\delta \varrho + \delta + \mathfrak{I} (1 - \delta))} \quad (55)$$

(iii) Let ψ_b be the steady-state possibility of the attendant being engaged with an ordinary consumer and in a normal service period at a time t.

$$\psi_b = \psi_b(1) = \mathcal{W}_0 \times \left\{ \frac{((\delta + \mathfrak{I}) \beta \{ \sigma \mathcal{B}_w^*(\vartheta + \tau) - \sigma - \delta \mathcal{B}_w^{*'}(\vartheta + \tau) \})}{(1 - \hat{\mathcal{K}}^*(\delta + \mathfrak{I})\varrho - \bar{\mathcal{K}}^*(\delta + \mathfrak{I}))} \times \frac{\mathcal{B}_w^{*'}(\vartheta + \tau) + \gamma + \zeta + \chi}{(\delta \varrho + \delta + \mathfrak{I} (1 - \delta))} \right\} \times \{\bar{\mathcal{B}}_b^*(\tilde{\alpha} + \mathfrak{I})\} \quad (56)$$

where,

$$\beta = \{\hat{\mathcal{K}}^*(\delta + \mathfrak{I}) + \delta \bar{\mathcal{K}}^*(\delta + \mathfrak{I})\}; \quad (57)$$

$$\gamma = \{((\mathcal{B}_w^*(\vartheta + \tau) - 1) - \tau) \delta (\delta + \mathfrak{I} \bar{\mathcal{K}}^*(\delta + \mathfrak{I}))\}; \quad (58)$$

$$\zeta = [\delta (\delta + \mathfrak{I}) (\mathcal{B}_w^{*'}(\vartheta + \tau) + \mathcal{B}_w^{*'}(\vartheta + \tau)) (1 - \mathfrak{I} \bar{\mathcal{K}}^*(\delta + \mathfrak{I}))]; \quad (59)$$

and

$$\chi = [(\delta + \mathfrak{I}) (1 - \mathcal{B}_w^*(\vartheta + \tau) + \tau) (1 - \mathfrak{I} \bar{\mathcal{K}}^*(\delta + \mathfrak{I}))]; \quad (60)$$

(iv) Let $\mathcal{W}$ be the steady-state possibility of the attendant being engaged during a lower speed service period at time t.

$$\mathcal{W} = \mathcal{W}(1) = (\delta + \mathfrak{I}) \mathcal{W}_0 \left\{ \frac{1 - \mathcal{B}_w^*(\vartheta + \tau)}{\vartheta + \tau} \right\} \quad (61)$$

(v) Let $\mathcal{W}_{wv}$ be the steady-state possibility of the attendant being on partial breakdown and semi-active vacation.

$$\mathcal{W}_{wv} = \mathcal{W} + \mathcal{W}_0 = \mathcal{W}_0 ((\vartheta + \tau) + (\delta + \mathfrak{I})) \times \frac{(1 - \mathcal{B}_w^*(\vartheta + \tau))}{\vartheta + \tau} \quad (62)$$

(vi) Let $\mathcal{P}_f$ be the stability state possibility of the attendant being failure:

$$\mathcal{P}_f = (\tilde{\alpha} + \mathfrak{I}) \psi_b(1) = \mathcal{W}_0 \times \left\{ \frac{((\delta + \mathfrak{I}) \beta \{ \sigma \mathcal{B}_w^*(\vartheta + \tau) - \sigma - \delta \mathcal{B}_w^{*'}(\vartheta + \tau) \})}{(1 - \hat{\mathcal{K}}^*(\delta + \mathfrak{I})\varrho - \bar{\mathcal{K}}^*(\delta + \mathfrak{I}))} \times \frac{(\gamma + \zeta + \chi)}{(\delta \varrho + \delta + \mathfrak{I} (1 - \delta))} \right\} \times \{1 - \mathcal{B}_b^*(\tilde{\alpha} + \mathfrak{I})\} \quad (63)$$

Proof: From the foregoing equations in Theorem 3.2, with setting $m \rightarrow 1$ and employing the L'Hospital rule, we procure the above-mentioned steady state probabilities results (54)-(63). (For detail explanation of the proof, see [12])

4.2 Average System Size

Theorem 4.2 (a) In steady state, the probability generating function of the number of consumers in the orbit $\mathcal{K}_0(1)$ and of the framework $\mathcal{K}_s(1)$ as given by:

$$\hat{L}_q = \lim_{m \rightarrow 1} \mathcal{K}_0'(m) = \mathcal{K}_0'(1) = \hat{\mathcal{K}}'(1) + \psi_1'(1) + \psi_b'(1) + \mathcal{W}'(1) \quad (64)$$

Here,

$$\hat{\mathcal{K}}'(1) = \frac{\mathcal{N}_r''(1)}{\mathcal{D}_r''(1)}, \quad (65)$$

$$\mathcal{N}_r''(1) = \mathcal{W}_0 \frac{(1 - \hat{\mathcal{K}}^*(\delta + \mathfrak{I}))}{(\delta + \mathfrak{I})} (\mathcal{D}'' + \mathcal{D}'), \quad (66)$$

$$\mathcal{D}' = \{(\delta + \mathfrak{I})(1 - \mathcal{B}_w^*(\vartheta + \tau))[\varrho + \frac{\delta}{\vartheta + \tau} - \sigma], \\ + \tau \varrho\} \quad (67)$$

$$\mathcal{D}'' = (\delta^2 - 2 \delta \sigma \mathcal{B}_b^*(\delta + \mathfrak{I})) ((\delta + \mathfrak{I})(1 - \mathcal{B}_w^*(\vartheta + \tau)) + \tau) + 2 \delta^2 \frac{\delta + \mathfrak{I}}{\vartheta + \tau} \times [\mathcal{B}_w^*(\vartheta + \tau) + \bar{\mathcal{B}}_w^*(\vartheta + \tau)] + 2 \delta (\delta + \mathfrak{I}) \varrho (\mathcal{B}_w^*(\vartheta + \tau) + \bar{\mathcal{B}}_w^*(\vartheta + \tau)) + (\delta + \mathfrak{I})[\delta^2 \mathcal{B}_w^{*''}(\vartheta + \tau) - 2 \delta \sigma \mathcal{B}_w^{*'}(\vartheta + \tau)] \quad (68)$$

$$\mathcal{D}_r''(1) = (-\hat{\mathcal{R}}^*(\delta + \mathfrak{I}) - \delta \hat{\mathcal{R}}^*(\delta + \mathfrak{I}))(\delta - 2 \delta \sigma \mathcal{B}_b^*(\tilde{\alpha} + \mathfrak{I})) - \mathfrak{I} \hat{\mathcal{R}}^*(\delta + \mathfrak{I}) (2 \delta^2 - \delta) - 2 \delta (\sigma \mathcal{B}_b^*(\tilde{\alpha} - 2 \delta (\sigma \mathcal{B}_b^*(\tilde{\alpha} + \mathfrak{I}) - \delta)), \quad (69)$$

$$\psi_1'(1) = \mathfrak{I} \bar{\mathcal{B}}_p^*(\tilde{\alpha}) \frac{\mathcal{N}_r''(1)}{\mathcal{D}_r''(1)}, \quad (70)$$

$$\psi_b'(1) = \frac{\mathcal{N}_{rmd}''(1)}{\mathcal{D}_r''(1)}, \quad (71)$$

$$\mathcal{N}_{rmd}''(1) = (First' \times (-\delta) \bar{\mathcal{B}}_b^*(\tilde{\alpha} + \mathfrak{I}) + First'' \times \bar{\mathcal{B}}_b^*(\tilde{\alpha} + \mathfrak{I})) \bar{\mathcal{B}}_b^*(\tilde{\alpha} + \mathfrak{I}), \quad (72)$$

$$First' = \mathcal{W}_0 (\delta + \mathfrak{I}) [(-\delta) \mathcal{B}_w^{*'}(\vartheta + \tau) + \sigma \mathcal{B}_w^*(\vartheta + \tau) - \sigma] \times [\hat{\mathcal{R}}^*(\delta + \mathfrak{I}) + \delta \hat{\mathcal{R}}^*(\delta + \mathfrak{I})] + \delta \hat{\mathcal{R}}^*(\delta + \mathfrak{I}) (\mathcal{B}_w^*(\vartheta + \tau) - 1) - \tau + (1 - \delta) (1 - \mathfrak{I} \hat{\mathcal{R}}^*(\delta + \mathfrak{I})) ((\delta + \mathfrak{I})(1 - \mathcal{B}_w^*(\vartheta + \tau) + \tau) + \delta (\delta + \mathfrak{I})(1 - \mathfrak{I} \hat{\mathcal{R}}^*(\delta + \mathfrak{I})) \times (\mathcal{B}_w^{*'}(\vartheta + \tau) + \bar{\mathcal{B}}_w^*(\vartheta + \tau)), \quad (73)$$

$$First'' = \mathcal{W}_0 [-\delta (\delta + \mathfrak{I}) \sigma \mathcal{B}_w^{*'}(\vartheta + \tau) + (\delta)^2 (\delta + \mathfrak{I}) \mathcal{B}_w^{*''} + (\delta + \mathfrak{I}) \sigma \mathcal{B}_w^{*'}(\vartheta + \tau)] [\hat{\mathcal{R}}^*(\delta + \mathfrak{I}) + \delta \hat{\mathcal{R}}^*(\delta + \mathfrak{I})] + 2 \delta \hat{\mathcal{R}}^*(\delta + \mathfrak{I}) (\sigma \mathcal{B}_w^*(\vartheta + \tau) - \delta \mathcal{B}_w^{*'}(\vartheta + \tau) - \sigma) + (\delta + \mathfrak{I}) (1 - \mathcal{B}_w^*(\vartheta + \tau) + \tau) (2 \delta - (\delta)^2) \mathfrak{I} \hat{\mathcal{R}}^*(\delta + \mathfrak{I}) + (\delta + \mathfrak{I}) (1 + \mathfrak{I} \delta \hat{\mathcal{R}}^*(\delta + \mathfrak{I})) [(\mathcal{B}_w^{*'}(\vartheta + \tau) + \bar{\mathcal{B}}_w^*(\vartheta + \tau))(\delta + \frac{(2 (\delta)^2)}{\vartheta + \tau})] \quad (74)$$

$$\mathcal{W}'(1) = \frac{\delta + \mathfrak{I}}{\vartheta + \tau} \mathcal{W}_0 \delta (\mathcal{B}_w^{*'}(\vartheta + \tau) + \bar{\mathcal{B}}_w^*(\vartheta + \tau)), \quad (75)$$

$$\hat{\mathcal{L}}_s = \lim_{z \rightarrow 1} \mathcal{K}_s'(z) = \mathcal{K}_s'(1) = \mathcal{K}_0'(1) + \psi_1(1) + \psi_b(1) + \mathcal{W}(1) \quad (76)$$

Proof: Putting (66), (67), (68), and (69) together, we get (65) and (70). Similarly, putting (72), (73), (74) and (69) together, we get (71). Differentiating equations (65), (70), (71) and (75) with respect to m and putting $m \rightarrow 1$, we acquire the expected number of consumers in orbit (64) and expected number of consumers in framework (76).

Corollary 4.2 (b): The Little's formula (77) is utilised to evaluate the average amount of time a consumer stays in the orbit and in the framework.

Proof:

$$\mathcal{W}_q = \frac{\hat{\mathcal{L}}_q}{\delta} \text{ and } \mathcal{W}_s = \frac{\hat{\mathcal{L}}_s}{\delta}. \quad (77)$$

Corollary 4.3 The system's reliability measures for the steady state, namely the service provider's availability and failure frequency, are provided by equation (78).

Proof:

$$\mathcal{A}_s = 1 - \mathcal{P}_f \text{ and } \mathcal{F}_s = \alpha [\psi_1(1) + \psi_b(1) + \mathcal{W}(1) + \mathcal{W}_{wv}(1)] \quad (78)$$

5 Sensitivity Analysis

In order to forecast how the possible parameters considered in our model would affect the performance indicators, numerical examples are absolutely necessary. Any model's behavior in real-world situations can be predicted through numerical analysis.

5.1 Numerical Simulation for Real-world Illustration

We have specifically used the example from Section 2.1 to demonstrate this section. i.e., by conducting a numerical simulation of how the TCSion software for the university responds to various scenarios. We affirmatively receive a request from the consumer at a pace of $\delta = 1/\text{min}$ to take a leave application. A pace of $\nu_b = 2/\text{min}$ is used by the management to respond to the consumers. The management prioritizes consumers which are faculty or teaching personnel. They arrive at a pace of $\mathfrak{I} = 0.12$. The probability of a consumer calling back for feedback is $\sigma = 0.28$ whenever there has been a breakdown in communication. The regular attendant lower service rate is $\nu_w = 2$. The attendant works at a lower pace of $\vartheta = 0.35$ per min when the management takes a semi-active vacation. Most of the time, it is believed that the attendant (server) is always accessible. However, in many real-world scenarios, attendants experience failures that result in service interruptions. The repair rate that we will use is $\tau = 0.23$. Obstructive consumers

could cause the system to malfunction at any time when the normal busy server is active. We use this software on our phones and laptops, which allows viruses to get into the system. Here, the obstructive consumer rate is $\tilde{\alpha}=0.02$. The possibilities are thus obtained as follows: $\hat{\mathbf{R}}=0.0032$, $\psi_1=0.0027$, $\psi_b=0.1320$, $\mathcal{W}=0.2662$, $\mathcal{W}_{wv}=0.7606$, $\mathcal{P}_f=0.0026$.

5.2 System Parameters' Impact on Long-term Probabilities and Average Queue Size

Additionally, we looked into various other numerical outcomes. These findings demonstrate how several factors might affect system performance indicators. The following outcomes are demonstrated using MATLAB software: Table 1, Table 2, Table 3 signify that whenever we make a slight increment in either lower service rate or vacation rate, or obstructive arrival rate, there is a modest increase in the possibility of being inactive (idle), i.e., $\check{\rho}_0$. There is a small decline in the likelihood that the attendant being inactive during retrieve, i.e., $\hat{\mathbf{R}}$. We observe a steady decline in the likelihood that the attendant will be engaged with a priority consumer and provide normal service at time 't', i.e., ψ_1 , and an upsurge is visualized in the possibility that the attendant is on vacation or experiences a partial breakdown, i.e., $\mathcal{W}_{wv}$. Moreover, from these above tables, we observe that the possibility of the attendant being busy with an ordinary consumer and in a normal service period i.e., ψ_b is falling rapidly with the increase in (ν_w) or ($\mathfrak{I}$), or ($\tilde{\alpha}$). Additionally, the possibility of the attendant being busy with an ordinary consumer and in a normal service period, i.e., ψ_b , increases with the increase in ϑ . We have noticed that the queue length decreases as we increase the lower service rate and vacation rate. In contrast, the queue length increases as we increase the obstructive arrival rate and priority rate. Further, Table 4 represents that $\hat{L}_q$, $\hat{\mathbf{R}}$, ψ_1 , and $\mathcal{W}_{wv}$ are going to rise by increasing $\mathfrak{I}$ (priority rate). Meanwhile $\check{\rho}_0$ and ψ_1 , are going toward decline on rising the priority rate.

Table 1: Impact of reduced service (lower service) rate ν_w on system performance measures, when $\delta=1$, $\mathfrak{I}=0.12$, $\tilde{\alpha}=0.02$, $\nu_b=2$, $\vartheta=0.35$, $\tau=0.23$, $\sigma=0.28$.

L.S.R (ν_w)	$\check{\rho}_0$	$\hat{L}_q$	$\hat{\mathbf{R}}$	ψ_1	ψ_b	$\mathcal{W}_{wv}$
1.1	0.0928	0.1838	0.0060	0.0059	0.1433	0.7528
1.2	0.0952	0.1627	0.0052	0.0052	0.1402	0.7549
1.3	0.0974	0.1427	0.0045	0.0046	0.1373	0.7569
1.4	0.0995	0.1236	0.0038	0.0040	0.1345	0.7588
1.5	0.1015	0.1053	0.0032	0.0034	0.1320	0.7606

Source: created by the authors

Table 2: Impact of vacation rate ϑ on system performance measures, when $\delta=1$, $\mathfrak{I}=0.12$, $\tilde{\alpha}=0.02$, $\nu_b=2$, $\nu_w=1.5$, $\tau=0.23$, $\sigma=0.28$.

V.R (ϑ)	$\check{\rho}_0$	$\hat{L}_q$	$\hat{\mathbf{R}}$	ψ_1	ψ_b	$\mathcal{W}_{wv}$
0.15	0.0970	0.1157	0.0118	0.0115	0.1275	0.7536
0.20	0.0982	0.1083	0.0096	0.0094	0.1287	0.7554
0.25	0.0993	0.1050	0.0074	0.0074	0.1298	0.7572
0.30	0.1004	0.1044	0.0053	0.0054	0.1309	0.7589
0.35	0.1015	0.1053	0.0032	0.0034	0.1320	0.7606

Source: created by the authors

Table 3: Impact of negative (obstructive) arrival rate $\tilde{\alpha}$ on system performance measures, when $\delta=1$, $\nu_b=2$, $\nu_w=1.5$, $\vartheta=0.35$, $\tau=0.23$, $\sigma=0.28$

N.R ($\tilde{\alpha}$)	$\check{\rho}_0$	$\hat{L}_q$	$\hat{\mathbf{R}}$	ψ_1	ψ_b	$\mathcal{W}_{wv}$
0.02	0.1015	0.1053	0.0032	0.0027	0.1320	0.7606
0.04	0.1017	0.1332	0.0031	0.0023	0.1308	0.7620
0.06	0.1019	0.1556	0.0030	0.0020	0.1296	0.7634
0.08	0.1021	0.1741	0.0030	0.0018	0.1285	0.7647
0.10	0.1022	0.1898	0.0029	0.0016	0.1273	0.7660

Source: created by the authors

Table 4: Impact of priority rate on system performance measures, when $\delta=1$, $\tilde{\alpha}=0.02$, $\nu_b=2$, $\nu_w=1.5$, $\vartheta=0.35$, $\tau=0.23$, $\sigma=0.28$.

P.R ($\mathfrak{I}$)	$\check{\rho}_0$	$\hat{L}_q$	$\hat{\mathbf{R}}$	ψ_1	ψ_b	$\mathcal{W}_{wv}$
0.12	0.1015	0.1053	0.0032	0.0027	0.1320	0.7606
0.14	0.0994	0.1114	0.0033	0.0029	0.1318	0.7626
0.16	0.0973	0.1181	0.0035	0.0031	0.1315	0.7646
0.18	0.0953	0.1253	0.0036	0.0033	0.1313	0.7665
0.20	0.0934	0.1329	0.0037	0.0034	0.1311	0.7684

Source: created by the authors

5.3 Effect of Critical System Parameters on Queue Length of the Framework

In this subsection, we discuss the impact of parameters on the framework's queue length. The primary goal of the framework's managers is to reduce the queue length, which validates the queuing model in a real-life situation. We have set the arbitrary default parameters as $\delta=1$, $\mathfrak{I}=0.12$, $\nu_b=3$, $\vartheta=0.009$, $\tau=0.17$, $\sigma=0.28$. In Figure 1, the queue length slopes down as the lower service rate increases and queue length increases as the regular arrival rate rises. Figure 2 depicts that the queue length upsurges as the regular arrival rate rises and the queue length decreases as the probability of semi-active vacation

increases. Furthermore, Figure 3 demonstrates that the queue length declines whenever there is a boost in the lower service rate of the framework. Also, as we hike up the priority arrival rate, the queue length upsurges. Moreover, we found that in Figure 4, as we increase the rate of regular arrival rate, the queue length increases. However, as we upsurge the partial breakdown rate queue length decreases.

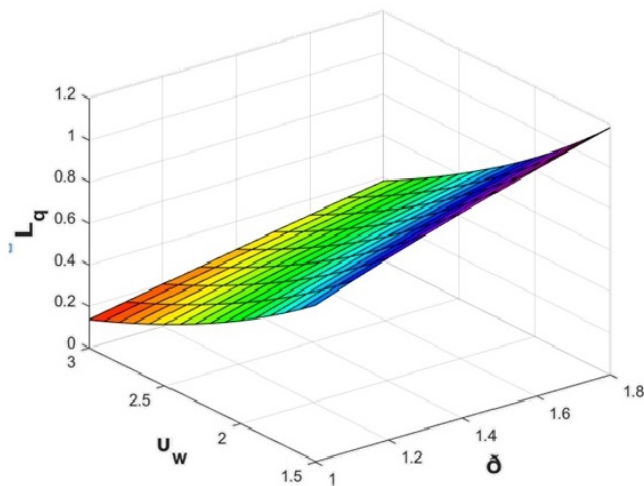


Figure 1: Lower service rate u_w vs. queue length by fluctuating δ

Source: created by the authors

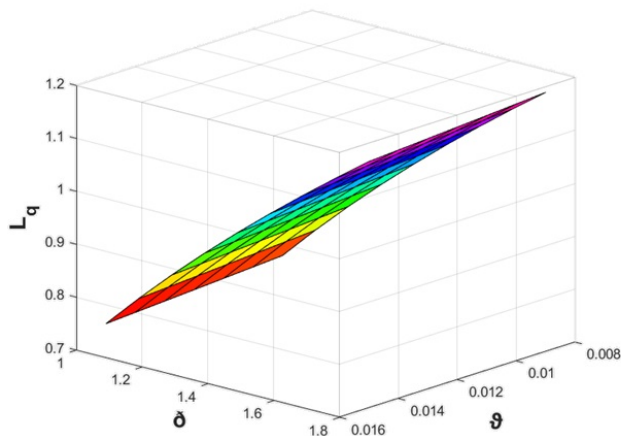


Figure 2: Regular arrival rate δ vs. queue length by fluctuating ϑ

Source: created by the authors

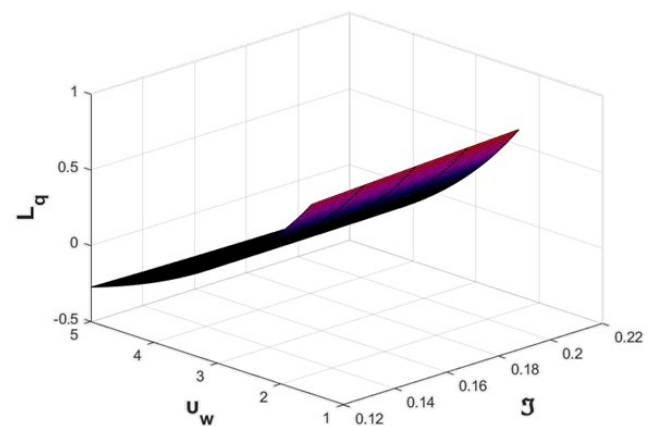


Figure 3: Lower service rate u_w vs. queue length by fluctuating γ

Source: created by the authors

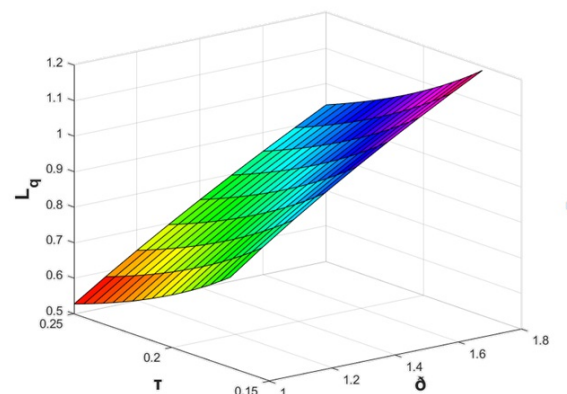


Figure 4: Partial breakdown rate τ vs. queue length by fluctuating δ

Source: created by the authors

6 Cost Optimization Analysis

As soon as a model is built and ready to use, the maker's or analyst's primary concern is usually whether the model is cost-effective or not. As a result, "cost analysis" plays a crucial role in determining how effective the model is. It lowers the risk factors and aids in system designers' decision-making. Concerning the necessity and importance of cost optimization, we have carried out this work by utilizing two different optimization strategies, namely the direct search method (DSM),

particle swarm optimization (PSO) and teaching learning based optimization (TLBO). In contrast to many other optimization approaches, PSO does not require the objective function to be differentiable, which is an additional benefit. This strategy can be used in many different situations. Signal processing, telecommunications, designing, management, and other areas are a few of them.

6.1 Direct Search Method

In [26] the direct search method is developed by authors, which is also referred to as the "Pattern Search Method". It is one of the most often used techniques that doesn't include inclination. To determine the value of our cost function (TC), we first select a decision variable (let's say d), after which we fix the values of every other parameter except d . We now modify the values for d and see how the values for TC change. The lowest Value of the cost function (let's say TC^*) provides us with the optimal cost for a given value of d . Secondly, using Figure 5, we demonstrate how the average cost of the consumers varies depending on various criteria using Direct search method. The suggested default values are $\delta = 1$, $\mathfrak{I} = 0.12$, $\tilde{\alpha} = 0.09$, $\nu_b = 3$, $\vartheta = 0.009$, $\tau = 0.17$, $\sigma = 0.2$. Figure 5 represents the graph of expected cost by varying ν_b for different values of τ ; it also shows that the expected cost initially falls when we elevate the value of ν_b (regular/normal service rate); afterward, it will obtain an optimal value. However, in Figure 6, the expected cost also initially falls when we elevate the value of τ (partial breakdown rate), and it will obtain the optimal value afterward; it shows how expected cost varies with respect to priority arrival rate $\mathfrak{I}$. Both graphs exhibit convexity behavior. Later on, it experiences an upsurge and remains consistent afterwards for both figures. Similarly, in Figure 6, the predicted cost initially declines when we raise the value of either ν_b or $\mathfrak{I}$ (priority service rate).

6.2 Particle Swarm Optimization

The PSO approach is utilized in this part to address the cost optimization issue. It is a computational technique that optimizes a tricky problem. It will optimize by refining the candidate solution. The exploitation and exploration search concepts as a part of the PSO algorithm is provided in [27]. Every object that has a location and velocity is a candidate solution. The finest local and global positions of each object in the search area have an impact on its movement. PSO solved problems by having a community (called a swarm) of candidate solutions (objects).

In this segment, we address, for each consumer logged into the system, C_2 is holding the monetary

value per unit of time. Whenever an attendant offers service throughout a typical engaged period, C_3 is the expenditure incurred per unit of time; Whenever an attendant delivers service during a partial breakdown and vacation. C_1 is a holding cost per unit time for each consumers present in the framework, there is a price per unit of time called C_4 which represents the cost per unit time when the attendant is in failure state; C_5 represents the expenditure per client served at the average service rate of ν_b ; C_6 represents the expenditure per client served at the average lower service rate ν_w ; When the priority arrival rate is $\mathfrak{I}$ then C_7 represents the expenditure of consumer per unit time. The anticipated cost (expenditure) function per unit of time under a linear cost structure is represented by equation (38) using the above-mentioned expenditure variables and accompanying framework performance measurement.

$$TC = C_1 \hat{L}_s + C_2 \psi_b + C_3 \mathcal{W}_{wv} + C_4 \mathcal{P}_f + C_5 \nu_b + C_6 \nu_w + C_7 \mathfrak{I} \quad (79)$$

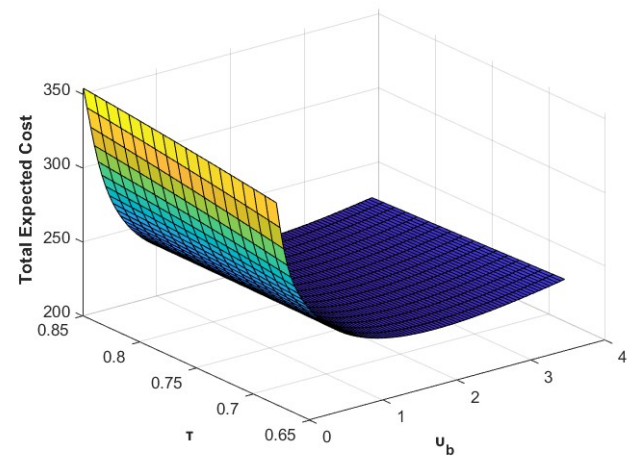


Figure 5: Regular service rate ν_b Vs total expected cost by fluctuating τ

Source: created by the authors

where, respectively, equations (56), (62), (63), and (76) give ψ_b , $\mathcal{W}_{wv}$, $\mathcal{P}_f$ and $\hat{L}_s$ is accessible from Theorem 4.1 and Theorem 4.2:(a). In addition, we look at how the predicted cost function behaves when the cost parameters are varied. We make the following assumptions for service rate: exponential regular service, exponential lower speed service during semi-active vacation and partial breakdown, and exponential retrial. The cost elements and other parameters for subsequent variables are $\delta = 1$, $\mathfrak{I} = 0.15$, $\tilde{\alpha} = 0.09$, $\nu_b = 3$, $\nu_w = 0.50$, $\vartheta = 0.009$,

$\tau = 0.70$, $\sigma = 0.28$, $C_1 = 10$, $C_2 = 400$, $C_3 = 150$,
 $C_4 = 250$, $C_5 = 15$, $C_6 = 18$, $C_7 = 50$.

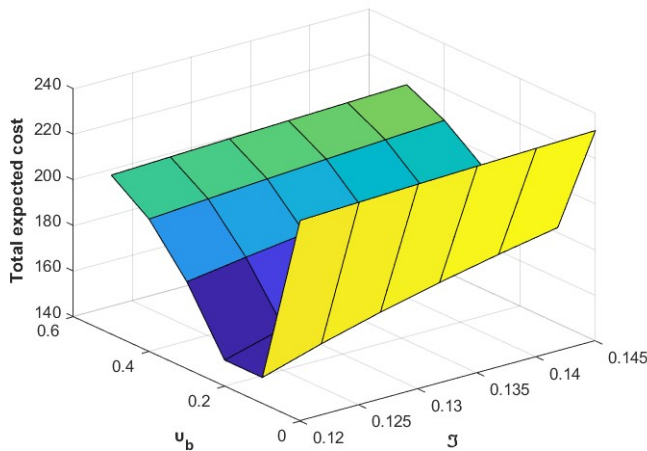


Figure 6: Regular service rate ν_b Vs total expected cost by fluctuating $\mathfrak{J}$

Source: created by the authors

Here all cost are in a dollar. We determine the overall anticipated cost (expenditure) per unit of time $TC = \$222.34$. Furthermore, we investigate the behavior of the anticipated cost function for various cost parameter values. Following are the fixed system parameters: $\delta = 1$, $\mathfrak{J} = 0.15$, $\tilde{\alpha} = 0.09$, $\nu_b = 3$, $\nu_w = 0.50$, $\vartheta = 0.009$, $\tau = 0.70$, we will take exponential regular service and different values of $\sigma = 0.28$. Table 5 shows, in turn, how $(\mathfrak{J}, \nu_w)$ and (ν_w, τ) , affect the predicted cost function. It is apparent that the estimated cost function shows a linear growth as the cost parameters increase. The framework parameters are given as: $\delta = 1$, $\mathfrak{J} = 0.15$, $\tilde{\alpha} = 0.09$, $\nu_b = 3$, $\nu_w = 0.50$, $\vartheta = 0.009$, $\tau = 0.70$. We will also use different values of p in Table 6. Furthermore, Table 6 shows how the predicted cost function is affected by $(\mathfrak{J}, \nu_w)$ and (ν_w, τ) , in turn, for different values of ν_b . In addition, we examine how the predicted cost function behaves for various cost parameter values and also compare Table 5 with Table 6 below. It is clear that when the cost parameters increase, the predicted cost function exhibits a linear trend.

It is crucial to understand whether objects simmer down or wander around in search of the ultimate solution. When applying an optimization approach, it is frequent for the objects not to always be in equilibrium. Due to this, identifying solutions for these issues requires convergence in cost optimization. It is apparent that, after the same number of iterations, the objects eventually achieve the recommended cost.

To a certain extent, the mathematical analysis of the framework cost and perfect design parameters will offer coordinators useful knowledge to address their financial issues more effectively. The subsequent graphics in Figure 7 demonstrate how an object fits the best outcome following a number of repetitions (generations) in PSO, which demonstrates the best expenditure of the M/G/1 retrieve G-queue framework for various expenditure sets. Moreover, we have performed computations to find the joint optimal value of (ν_w, τ) by varying p with an exponential service rate.

The framework's parameters for PSO optimization are given as: $\delta = 1$, $\mathfrak{J} = 0.15$, $\tilde{\alpha} = 0.09$, $\nu_b = 3$, $\nu_w = 0.50$, $\vartheta = 0.009$, $\tau = 0.70$. According to the PSO algorithm's convergence analysis, the objective function value slightly declines during the first few iterations. The particles quickly reach the optimal area near iteration 5, demonstrating the algorithm's rapid convergence. The convergence curve becomes rather stable with only slight variations in the fitness value after iteration 5. This shows that the PSO algorithm effectively strikes a balance between exploration and exploitation, generating a near-optimal solution in fewer iterations. In Figure 7, we acquire the optimal cost of \$223.97 near iteration 5, which is for a single-server bulk-arrival general service distribution retrieve G-queue with partial breakdown and semi-active vacation. Additionally, Figure 8 demonstrates how an object fits the best outcome following several repetitions (generations) in PSO. We have done computations to find joint optimal values of $(\mathfrak{J}, \nu_w)$ by varying ν_b . Also, in Figure 8, we have employed an Erlang service rate for a single-server bulk-arrival retrieve G-queue with partial breakdown and semi-active vacation. Here, we obtain the optimal cost of \$240.57 at iteration 5.

6.3 Teaching Learning Based Algorithm

In [28], the Teaching-Learning-Based Optimization (TLBO) algorithm was introduced in detail. It draws inspiration from the teaching-learning process in a classroom, emulating the impact a teacher has on learners' outcomes. Like other swarm intelligence algorithms (SIs), TLBO is a population-based heuristic stochastic optimization algorithm. However, it stands out because it does not require any algorithm-specific parameters. In this algorithm, learners are considered as search points spread across the decision variable space, similar to the population of solutions in evolutionary algorithms (EAs) and swarm intelligence (SIs). The learner with the highest fitness level is designated as the class teacher. Unlike conventional EAs and SIs, TLBO's iterative evolution process comprises both a teaching phase and a learning phase. In

order to raise the class's average level of knowledge, students learn from the teacher during the teaching phase and from another student chosen at random during the learning period. TLBO has been effectively extended and widely used to handle a wide range of real-world problems from various disciplines of science and technology because of its certain characteristics, such as its straightforward concept, without algorithm-specific parameters, quick convergence, ease of implementation, and efficacy.

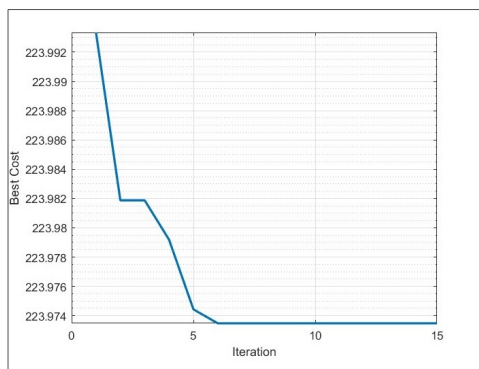


Figure 7: Optimal cost of M/G/1 retrieve G-queue framework with partial breakdown and semi-active vacation for cost set II when $p=0.28$, (ν_w, τ) , optimal cost=223.97 in dollar (For normal exponential service rate)

Source: created by the authors

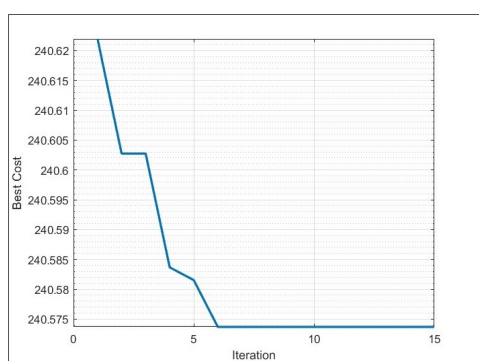


Figure 8: Optimal cost of M/E_k/1 retrieve G-queue framework with partial breakdown and semi-active vacation for cost set II when $p=0.30$, $(\mathfrak{J}, \nu_w)$, optimal cost=240.57 in dollar (for Erlang service rate)

Source: created by the authors

6.4 Comparative Analysis Between Particle Swarm Optimization and Teaching Learning Based Algorithm

A crucial component of cost analysis is convergence. There is no symmetry in the findings at the beginning of any optimization procedure. Working on cost optimization makes it significant that the result must head in some particular direction and settle down at some point. Using two distinct approaches, we are now examining the particle's convergence towards the optimal solution. We used the framework parameters for PSO as follows: $\delta = 2$, $\tilde{\alpha} = 0.09$, $\nu_b = 3$, $\vartheta = 0.009$, $\tau = 0.20$. The convergence analysis of the PSO method indicates that during the initial iterations, the objective function value marginally reduces. Near iteration 5, the particles swiftly arrive at the optimal region, highlighting the algorithm's rapid convergence behavior. After iteration five, the convergence curve becomes quite steady with only minor changes in the fitness value. This displays that the PSO algorithm successfully balances exploration and exploitation to produce a near-optimal solution in fewer iterations. In Figure 9, we get the optimal cost, i.e., \$290.7893, which is nearly the same as we obtained from TLBO. On the other hand, $\delta = 2$, $\tilde{\alpha} = 0.09$, $\nu_b = 5$, $\vartheta = 0.009$, $\tau = 0.20$ are the parameters for TLBO. The TLBO algorithm's convergence analysis exhibits that the objective function value quickly drops in the early iterations. A noticeable improvement in the fitness value is seen close to iteration 5, indicating that the learner's effective progress towards the optimal solution while being influenced by the learner and teacher phases. The convergence curve becomes steadier and smoother after iteration five, suggesting that the solution has been gradually improved. This behavior shows how the TLBO algorithm can produce near-optimal outcomes with quick convergence and less computational work. The framework's parameters are given as: $\delta = 2$, $\tilde{\alpha} = 0.09$, $\nu_b = 5$, $\vartheta = 0.009$, $\tau = 0.20$. Figure 9 indicates that the result converges to the optimal value in PSO and TLBO after a few iterations. Figure 10 reveals that the optimal solution is \$290.7894 at iteration 5. Besides, Figure 10 specify that the result converges to the optimal value in PSO and TLBO after a few iterations. We have used four different cost sets in Table 7. For the cost set given in Table 7, these graphs reflect the information provided in Table 8. Here, we also perform a comparative analysis between PSO and TLBO, as shown in Table 8 below. Based on the graphs, it can be determined that PSO and TLBO converge nearly simultaneously after different iterations. By analyzing two distinct scenarios in the table, we demonstrate that the ideal cost for each of

the two methods coincides.

Consequently, we may arrive at a general conclusion as:

(i) The optimal value of the expected cost obtained via DSM method agrees with that obtained using PSO and TLBO techniques.

(ii) While the two cost optimization strategies are nearly equal, Also, TLBO requires fewer iterations to get the model's optimal value than PSO.

(iii) Since there is a high degree of compatibility between our model and real-world issues, system managers can use it to determine the minimal cost of the framework.

Table 5: Optimum cost with optimal parameters using PSO for different cost sets (total expenditure) by employing regular exponential service rate.

Optimal Parameters and Optimal cost	Parameters	Total Expenditure (I) in (dollar)	Total Expenditure (II) in (dollar)	Total Expenditure (III) in (dollar)	Total Expenditure (IV) in (dollar)
$(\mathcal{J}, v_w); ETC^*$ $(v_w, \tau) ETC^*$	$p=0.28$	[0.12,0.64]; 219.65 [0.65,0.7]; 222.34	[0.12,0.62]; 221.28 [0.65,0.7]; 223.97	[0.12,0.69]; 230.60 [0.68,0.7]; 233.31	[0.12,0.7]; 233.30 [0.7,0.7]; 236.02
$(\mathcal{J}, v_w); ETC^*$ $(v_w, \tau) ETC^*$	$p=0.29$	[0.12,0.63]; 219.63 [0.65,0.7]; 222.32	[0.12,0.62]; 221.28 [0.65,0.7]; 223.96	[0.12,0.69]; 230.60 [0.68,0.7]; 233.30	[0.12,0.7]; 233.26 [0.7,0.7]; 235.98
$(\mathcal{J}, v_w); ETC^*$ $(v_w, \tau) ETC^*$	$p=0.30$	[0.12,0.63]; 219.61 [0.65,0.7]; 222.29	[0.12,0.62]; 221.28 [0.65,0.7]; 223.95	[0.12,0.69]; 230.59 [0.68,0.7]; 233.29	[0.12,0.7]; 233.23 [0.7,0.7]; 235.94

Source: created by the authors

Table 6: Optimum cost with optimal parameters using PSO for different cost sets (total expenditure) by employing Erlagian regular service rate.

Optimal Parameters and Optimal cost	Parameters	Total Expenditure (I) in (dollar)	Total Expenditure (II) in (dollar)	Total Expenditure (III) in (dollar)	Total Expenditure (IV) in (dollar)
$(\mathcal{J}, v_w); ETC^*$ $(v_w, \tau) ETC^*$	$v_b=3$	[0.12,0.64]; 220.35 [0.65,0.70]; 223.13	[0.12,0.63]; 222.04 [0.65,0.70]; 224.82	[0.12,0.69]; 231.39 [0.69,0.70]; 234.21	[0.12,0.70]; 233.99 [0.70,0.70]; 236.80
$(\mathcal{J}, v_w); ETC^*$ $(v_w, \tau) ETC^*$	$v_b=4$	[0.12,0.59]; 227.30 [0.65,0.70]; 230.16	[0.12,0.57]; 228.33 [0.65,0.70]; 213.21	[0.12,0.63]; 239.30 [0.65,0.70]; 242.16	[0.12,0.68]; 245.18 [0.67,0.70]; 248.04
$(\mathcal{J}, v_w); ETC^*$ $(v_w, \tau) ETC^*$	$v_b=5$	[0.12,0.52]; 237.62 [0.65,0.70]; 239.96	[0.12,0.59]; 250.35 [0.65,0.70]; 240.61	[0.12,0.59]; 250.35 [0.65,0.70]; 253.25	[0.12,0.65]; 259.06 [0.65,0.70]; 261.95

Source: created by the authors

Table 7: Distinct cost components connected to the cost set.

Cost Set	$\mathcal{C}_1$	$\mathcal{C}_2$	$\mathcal{C}_3$	$\mathcal{C}_4$	$\mathcal{C}_5$	$\mathcal{C}_6$	$\mathcal{C}_7$
Cost set I	10	400	150	250	15	18	50
Cost set II	14	400	150	250	19	18	50
Cost set III	15	450	200	300	20	20	60
Cost set IV	20	500	250	350	25	22	70

Source: created by the authors

Table 8: Comparing the optimal cost (in dollars) for each cost set in two distinct scenarios using PSO and TLBO

Case 1: For suggested parameters $\delta = 1, \tilde{\alpha} = 0.09, v_b = 3, \theta = 0.009, \tau = 0.17, \sigma = 0.28$,

Algorithm	Cost set I	Cost set II	Cost set III	Cost set IV
PSO	\$242.5994	\$262.7167	\$314.0067	\$385.5175
TLBO	\$242.5944	\$262.7167	\$314.0067	\$385.5175

Case 2: For suggested parameters $\delta = 2, \tilde{\alpha} = 0.09, v_b = 5, \theta = 0.009, \tau = 0.20, \sigma = 0.35$,

Algorithm	Cost set I	Cost set II	Cost set III	Cost set IV
PSO	\$ 290.7894	\$324.9336	\$381.9160	\$472.7253
TLBO	\$290.7894	\$324.9336	\$381.9160	\$472.7253

Source: created by the authors

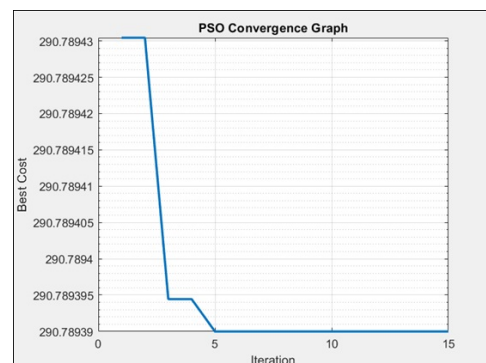


Figure 9: Convergence graph for PSO

Source: created by the authors

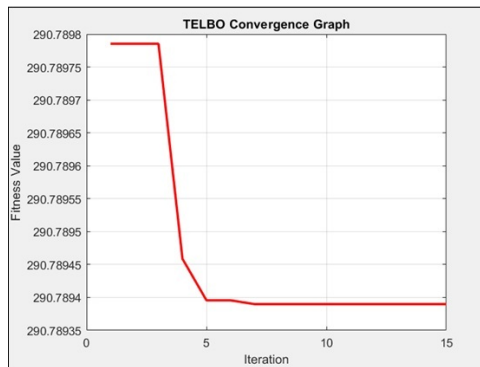


Figure 10: Convergence graph for TLBO

Source: created by the authors

7 Conclusion and Potential Outcomes

This analysis includes a novel type of single server retrial G-queue, feedback under partial breakdown, and semi-active vacations with preemptive priority. We cover performance metrics such as long-run probabilities and the system's average queue size, which can control and diminish queue size. The primary goal of our investigation is to present nearly equivalent research that relates to the precise and approximate expected waiting times for the developed model so that system planners and manufacturers can shorten the time of consumers who wait in a line, which in turn improves the quality of service provided by their respective systems. The proposed approach is suitable for industrial waits, management and production industries, communication networks, supermarkets, health care administration phone consultations, and the telecommunication system, etc. Apparently, it is nearly impossible to design a model in any of these vast industries where the server never malfunctions or becomes inactive. Our concept is made to assist in preventing everyday overpopulation issues that networking and communication platforms encounter. The real-world congestion circumstances that are used to validate the conclusions are shown in the concrete example and numerical simulation presented in Section 5. We have also compared the optimal cost using DSM, PSO and TLBO, also evaluated that these optimal costs are nearly the same. State-dependent arrivals, vacation interruption, and j-vacation can be added to this study to further enhance it, but doing so will make the model more difficult and intricate and require a significant amount of computational work.

Acknowledgment:

We express our sincere gratitude to the anonymous reviewers for their valuable comments and insightful suggestions.

References:

- [1] K. Kalidass and R. Kasturi, "A queue with working breakdowns", *Computers and Industrial Engineering*, vol.63, no.4, pp.779-783, 2012, <https://doi.org/10.1016/j.cie.2012.04.018>.
- [2] BK. Kim and DH. Lee, "The M/G/1 queue with disasters and working breakdowns", *Applied Mathematical Modelling*, vol.38, no.(5-6), pp.1788-1798, 2014, <https://doi.org/10.1016/j.apm.2013.09.016>.
- [3] Nidhi and S.S. Sanga, "Analysis for an Unreliable Retrial Queueing System Under Threshold-Based Recovery Using ANFIS Approach", *In International Conference on Soft Computing for Problem-Solving Singapore: Springer Nature Singapore*, 2025, pp. 55-68, https://doi.org/10.1007/978-981-95-5044-9_5.
- [4] R. Agarwal, D. Agarwal and S. Upadhyaya, "Study on Optimization of Unreliable Server Queueing Systems: A PSO Based Survey", *In Proceedings of International Conference on Scientific and Natural Computing, Springer, Singapore* 2021, pp. 189-197, https://doi.org/10.1007/978-981-16-1528-3_16.
- [5] K. S. Antala, V. Panchal, S. S. Sanga, "Transient analysis and cost optimization of a machine repair system with dual-phase repair, working vacation, and retrial", *Operational Research*, vol.26, no.2, pp. 42, 2026, <https://doi.org/10.1007/s12351-026-01034-3>.
- [6] E. Gelenbe, "Random neural networks with negative and positive signals and product form solution", *Neural Computation*, vol.1, no.4, pp.502-510, 1989, <https://doi.org/10.1162/neco.1989.1.4.502>.
- [7] PG. Harrison, "Compositional reversed Markov processes with applications to G-networks", *Perform Evaluation*, vol.57, no.3, pp.379-408, 2004, <https://doi.org/10.1016/j.peva.2004.02.002>.
- [8] T. Van Do, "Bibliography on G-networks, negative customers and applications", *Mathematical and Computer Modelling*, vol.53, no.(1-2), pp.205-212, 2011, <https://doi.org/10.1016/j.mcm.2010.08.006>.
- [9] B. Krishna Kumar, S. Pavai Madheswari and S. R. Anantha Lakshmi, "An M/G/1 Bernoulli feedback retrial queueing system with negative customers", *Operational Research*, vol.13, no.2, pp.187-210, 2013, <https://doi.org/10.1007/s12351-011-0107-5>.

- [10] S. Gao, and J. Wang, "Performance and reliability analysis of an M/G/1-G retrial queue with orbital search and non-persistent customers", *European Journal of Operational Research*, vol. 236, no.2, pp.561-572, 2014, <https://doi.org/10.1016/j.ejor.2014.01.065>.
- [11] Y. Peng, Z. Liu, and J. Wu, "An M/G/1 retrial G-queue with preemptive resume priority and collisions subject to the server breakdowns and delayed repairs", *Journal of Applied Mathematics and Computing*, vol.44, no.1, pp.187-213, 2014, <https://doi.org/10.1007/s12190-013-0688-7>.
- [12] P. Rajadurai, V. M. Chandrasekaran, M. C. Saravananarajan, "Analysis of an $M^X/G/1$ unreliable retrial G-queue with orbital search and feedback under Bernoulli vacation schedule", *Opsearch*, vol.53, no.1, pp.197-223, 2016, <https://doi.org/10.1007/s12597-015-0226-5>.
- [13] K. M. Rashmi, S. K. Samanta, "Performance analysis of an M/G/1 retrial G-queue with balking, optional service and general retrial policy", *RAIRO-Operations Research*, vol.60, no.3, pp.1439-1473, 2026, <https://doi.org/10.1051/ro/2026040>.
- [14] R. Kulshrestha, A. Singh, S. Goel, "Economic and reliability analysis of discrete-time G-queue with multi-optional services and Bernoulli feedback", *Operational Research*, vol.25, no.3, pp.83, 2025, <https://doi.org/10.1007/s12351-025-00963-9>.
- [15] L. D. Servi and S.G. Finn, "M/M/1 queues with working vacations", *Performance Evaluation*, vol.50, no.1, pp.41-52, 2002, [https://doi.org/10.1016/S0166-5316\(02\)00057-3](https://doi.org/10.1016/S0166-5316(02)00057-3).
- [16] D. Wu. and H. Takagi, "M/G/1 queue with multiple working vacations", *Performance Evaluation*, vol.63, no.7, pp.654-681, 2006, <https://doi.org/10.1016/j.peva.2005.05.005>.
- [17] D. Arivudainambi, P. Godhandaraman and P. Rajadurai, "Performance analysis of a single server retrial queue with working vacation", *Opsearch*, vol.51, no.3, pp.434-462, 2014, <https://doi.org/10.1007/s12597-013-0154-1>.
- [18] P. Deora, U. Kumari and D. C. Sharma, "Cost analysis and optimization of machine repair model with working vacation and feedback-policy", *International Journal of Applied and Computational Mathematics*, vol.7, no.6, pp.1-14, 2021, <https://doi.org/10.1007/s40819-021-01185-1>.
- [19] M. Jain and A. Kumar, "Unreliable Server $M^X/G/1$ Queue with Working Vacation and Multi-phase Repair with Delay in Verification", *International Journal of Applied and Computational Mathematics*, vol.8, no.4, pp.1-24, 2022, <https://doi.org/10.1007/s40819-022-01383-5>.
- [20] X. Yang, Y. Zhang, B. Wang, B and X. J. Li, "Performance Analysis and Cost Optimization of the M/M/1/N Queueing System with Working Vacation and Working Breakdown", *Mathematics*, vol.13, no.18, pp.2980, 2025, <https://doi.org/10.3390/math13182980>.
- [21] J. Wu and Z. Lian, "A single-server retrial G-queue with priority and unreliable server under Bernoulli vacation schedule", *Computers and Industrial Engineering*, vol.64, no.1, pp.84-93, 2013, <https://doi.org/10.1016/j.cie.2012.08.015>.
- [22] S. Gao, "A preemptive priority retrial queue with two classes of customers and general retrial times", *Operational Research*, vol.15, no.2, pp.233-251, 2015, <https://doi.org/10.1007/s12351-015-0175-z>.
- [23] P. Rajadurai, S. Yuvarani and M.C. Saravananarajan, "Performance analysis of preemptive priority retrial queue with immediate Bernoulli feedback under working vacations and vacation interruption", *Songklanakarin Journal of Science and Technology*, vol.38, no.5, pp.507-520, 2016, <https://sjst.psu.ac.th/article/1623> [Access Date: 01/01/2026].
- [24] D. Agarwal, R. Agarwal, and S. Upadhyaya, "Detection of optimal working vacation service rate for retrial priority G-queue with immediate Bernoulli feedback," *Results in Control and Optimization*, vol. 14, pp. 100397, 2024, <https://doi.org/10.1016/j.rico.2024.100397>.
- [25] S. Upadhyaya, S. Vaishnawi, D. Agarwal, and I. Ahmad, "Performance Forecasting of Discrete-Time Priority Retrial Queue With Its Application in Cognitive Radio Networks", *International Journal of Communication Systems*, vol.38, no.4, pp. e6136, 2025, <https://doi.org/10.1002/dac.6136>.
- [26] R. Hooke and T.A. Jeeves, "Direct Search: Solution of Numerical and Statistical Problems", *Journal of the ACM (JACM)*, vol.8, no.2, pp.212-229, 1961, <https://doi.org/10.1145/321062.321069>.

- [27] J. Kennedy and R. Eberhart, "Particle swarm optimization", *In Proceedings of ICNN'95-International Conference on Neural Networks, IEEE*, 1995, pp.1942-1948, <https://ieeexplore.ieee.org/document/488968> [Access Date: 06/08/2026].
- [28] R.V. Rao, V.J. Savsani and D.P. Vakharia, "Teaching-learning-based optimization: an optimization method for continuous non-linear large-scale problems", *Information Sciences*, vol.183, no.1, pp.1-15, 2012, <https://doi.org/10.1016/j.ins.2011.08.006>.
- [29] A. G. Pakes, "Some conditions for ergodicity and recurrence of Markov chains", *Operations Research*, vol.17, no.6, pp.1058-1061, 1969, <https://doi.org/10.1287/opre.17.6.1058>. <https://www.jstor.org/stable/168325> [Access Date: 06/08/2026].
- [30] L. I. Sennott, P. A. Humblet, and R. L. Tweedie, "Mean drifts and the non-ergodicity of Markov chains", *Operations Research*, vol.31, no.4, pp.783-789, 1983, <https://doi.org/10.1287/opre.31.4.783>.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

The authors equally contributed in the present research, at all stages from the formulation of the problem to the final findings and solution.

Sources of Funding for Research Presented in Scientific Article or Scientific Article Itself

No funding was received for conducting this study.

Conflicts of Interest

The authors have no conflicts of interest to declare that are relevant to the content of this article.

Creative Commons Attribution License 4.0 (Attribution 4.0 International, CC BY 4.0)

This article is published under the terms of the Creative Commons Attribution License 4.0 https://creativecommons.org/licenses/by/4.0/deed.en_US

Appendix A: The $\{Z_n : n \in \mathcal{Y}\}$ is ergodic if and only if $\varrho < \hat{\mathfrak{K}}^*(\delta)$, where ϱ is given by (1).

Proof: To prove the sufficient condition of ergodicity equation given by (1), it is easier way to use the

Foster's criterion(See [29]), which asserts that the chain $\{Z_n : n \in \mathcal{Y}\}$ is an irreducible and aperiodic Markov chain, and it is ergodic if there exists a non-negative function $\tilde{f}(\tilde{j}), \tilde{j} \in \mathcal{Y}$, and $\epsilon > 0$, such that the mean drift $\Phi_j = E[\tilde{f}(z_{n+1}) - \tilde{f}(z_n) | z_n = \tilde{j}]$ is finite for all $\tilde{j} \in \mathcal{Y}$ and $\Phi_j \leq -\epsilon$ for all $\tilde{j} \in \mathcal{Y}$, except for all $\tilde{j}'$ s. In our case, we assume the function $\tilde{f}(\tilde{j}) = \tilde{j}$ then we have equation (80) which is given below

$$\Phi_j = \begin{cases} \varrho - 1, & \text{if } \tilde{j} = 0, \\ \varrho - \hat{\mathfrak{K}}^*(\delta) & \text{if } \tilde{j} = 1, 2, \dots \end{cases} \quad (80)$$

Evidently, the inequality $\varrho < \hat{\mathfrak{K}}^*(\delta)$ is the sufficient condition for ergodicity

As clearly observed in [30], to prove the necessary condition, if the Markov Chain $\{Z_n : n \in \mathcal{Y}\}$ satisfies Kaplan's condition, namely as $\Phi_j < \infty \forall \tilde{j} \geq 0$ and $\exists \tilde{j}_0 \in \mathcal{Y}$ such that $\Phi_j \geq 0$ for $\tilde{j} \geq \tilde{j}_0$. We clearly observed that in our case, Kaplan's condition is satisfied since there is a k such that $m_{ij} = 0$ for $\tilde{j} < i - k$ and $i > 0$. Where, $M = m_{ij}$ is the one step transition matrix of $\{Z_n : n \in \mathcal{Y}\}$. So, the ergodicity of the Markov chain is implied by $\varrho < \hat{\mathfrak{K}}^*$.