International Journal of Mathematical, Engineering and Management Sciences
Vol. 10, No. 4, 931-964, 2025
https://doi.org/10.33889/IJMEMS.2025.10.4.045

Ram Arti Publishers

Swarm based Heuristic Optimization of the Recurrent Customers and Standby Server Under General Retrial Times

N. Micheal Mathavavisakan
Department of Mathematics,
School of Advanced Sciences, Vellore Institute of Technology, Vellore, Tamil Nadu, India.
E-mail: michealmathava.visakan2020@vitstudent.ac.in

K. Indhira
Department of Mathematics,
School of Advanced Sciences, Vellore Institute of Technology, Vellore, Tamil Nadu, India.
E-mail: kindhira@vit.ac.in

Aliakbar Montazer Haghighi
Department of Mathematics,
Prairie View A&M University, Prairie View, Texas, USA.
Corresponding author: amhaghighi@pvamu.edu

(Received on June 22, 2024; Revised on September 26, 2024 & December 17, 2024; Accepted on January 24, 2025)

Abstract
As queueing theory and modeling deal with queue length, waiting time and busy period, that all affect costs for an in institution and/or a busing corporation, the optimization plays a crucial role in such models. This paper focuses on the performance modeling and optimal configuration of a single-server retrial queue with recurrent customers and a standby server, operating under Bernoulli working vacation conditions. The primary aim of the paper is to analyze the dynamics of this queueing model to achieve minimal operational costs while ensuring high performance. Using the supplementary variable technique (SVT), the probability generating functions (PGFs) and steady-state probabilities for the system's states, have been obtained enabling the development of comprehensive performance measures. These measures were rigorously validated through numerical examples. To complement the performance analysis, a cost function was formulated and optimized using advanced techniques, including the grey wolf optimizer (GWO), bat algorithm (BA), whale optimization algorithm (WOA), and cat swarm optimization (CSO). The results revealed that these algorithms successfully minimized operational costs while maintaining optimal system efficiency.

Keywords- Recurrent customer, Retrial queue, Working vacation, Heuristic optimization.

1. Introduction

Queuing systems are based on queueing theory. Applications of queueing theory have changed over the years since it was developed for a Danish telephone company by Erlang (1909). They are areas such as service network, inventory systems, and reliability studies, now days that are included in operations research. Queueing systems also quantitatively streamlined representations of congestion that retain its fundamental facets. When ‘clients’ began demanding ‘service’ from a particular resource, a queueing system evolved organically. Firms benefit from multiple real-world applications of queuing theory. Considering its capacity to enhance staff, planning, and customer service, it is widely utilized as a tool in operations management. In this regard, many researchers have analyzed various queueing models. For example, one of the pioneers of queueing theory Takacs (1960, 1962, 1980) who studies variety of models in queueing theory such as $M/G/1$, and combinatorics, see Haghighi and Mohanty (2019). Furthermore, queueing system strives to create economically viable systems that give clients prompt reliable service. Clients that have their access declined typically depart the service area for a brief length of time before returning later. This transpires all the time in both technical and real-world contexts. Clients whose work has been momentarily blocked will be held in orbit, a virtual waiting area, before attempting to occupy a server again. Retrial queues have been designed for this purpose. Because clients in an orbit are prone to working irrespective of one another, the retrial rate tends to be equivalent to the overall customers in the system.

Numerous researchers, including Falin and Templeton (1997), have investigated retrial queue (RQ) approaches. Similarly, Artalejo and Corral (1999) gathered extensive information on retrial waiting times. Recent studies by Rajadurai (2018a), Revathi (2022), and Jain and Kumar (2023) have contributed to the understanding of RQ. The widespread effectiveness of queues with negative customers, often referred to as G-queues, in sectors such as manufacturing, healthcare, transportation, and cloud computing has attracted the attention of researchers like Chao and Pinedo (1999) and Harrison and Marin (2014). The term "G-queue" was coined in recognition of Gelenbe (1989, 1991), who was a pioneer in this area, describing a queue where customers experience significant impacts. Many researchers have examined retrial G-queues in the context of server failures, including Peng et al. (2014), Li and Zhang (2017), Rajadurai (2018b), and Rajadurai et al. (2020).

According to Farahmand (1996), there are two scenarios that can occur when dealing with returning clients in a retrial queue: the constant scenario, the situation where every client's recall rate stays constant irrespective of the number of clients in orbit, and the discouraged scenario, wherein the recall rate falls as the number of clients in orbit goes up. The recurrent retry queues with service option on arrival were created by Farahmand and Livingstone (2001). However, Moreno (2004) investigated an $M/G/1$ retrial queue with a set number of recurring customers, K, $K > 0$, who upon receiving service, promptly re-enter the orbit, and transit users, who, shortly after been served, ultimately depart the system. Recently, Saggou et al. (2017) developed the performance metrics for such queueing system by considering two types of consumers using a server prone to failures and delayed repairs.

To implement a vacation queueing mechanism, a server must temporarily suspend all services and become unavailable to its primary customers. This time away from work is referred to as a vacation. However, during working vacation (WV) periods, the server continues to provide service to customers at a reduced rate. The integration of this queuing system has significant implications for various sectors, including networking, online services, file transfers, and postal deliveries. Rajadurai (2018b) introduced a novel retrial queue (RQ) model that incorporates breaks and WV. Li et al. (2019) examined an M/G/1 RQ with standard retry times and interruptions due to Bernoulli working vacations. Rajadurai (2019) discussed the steady-state analysis of a single-server priority RQ in the context of Bernoulli working vacations, where a typically busy server may be unavailable due to maintenance or failure. Gupta and Kumar (2021) evaluated different customer balking probabilities in high-traffic scenarios under the vacation interruption policy, recognizing both normal and Bernoulli working vacation modes. Additionally, Haghighi et al. (2025) studied time-dependent bulk arrivals with limited batch service, including reneging and setup times for service.

Moreover, it is challenging to predict when the system will experience a breakdown. In this research, it has been assumed that in the event of a system failure, the client whose service was interrupted is returned to the front of the queue. Additionally, the standby server is assigned to take over immediately once it becomes available for service. Radha et al. (2020) explored a queueing model that included batch arrivals with retrials, multiple optional service phases, extended Bernoulli vacations, and a standby server. Additionally, Maragathasundari et al. (2020) examined the effects of service disruptions and the function of the standby server within queuing models. Finally, Meena and Ayyappan (2024) conducted an analysis of a single-server preemptive priority queue that incorporated phase-type vacations, repairs, feedback, working breakdowns, closures, and impatient customers.

1.1 What is Novelty in this Article?

This article falls into the area of operations research, optimization techniques, and the queueing theory. Here are novelties of this article:

(i) Fulfilling the key gap in the cost optimization of our systems under complex scenarios like working vacations and server failures. While cost functions have been studied in retrial queues, there has been no significant work on developing advanced cost minimization techniques specifically tailored for retrial G-queues with recurrent customers in vacation modes.

(ii) The introduction of advanced metaheuristic algorithms such as Grey Wolf Optimizer (GWO), Bat Algorithm (BA), Whale Optimization (WO), and Cat Swarm Optimization (CSO) for minimizing the operational costs in these systems represents a novel contribution to the field. These algorithms can handle the complexity of nonlinear optimization problems, offering better convergence properties and solution quality compared to traditional methods, yet they have not been widely applied to retrial queues with vacation mechanisms.

(iii) To evaluate the orbit size and queue length distributions to aid in the development and evaluation of novel metrics to gauge system behavior.

(iv) Application of the supplementary variable technique (SVT) to analyze the recurrent customer vacation process marks a significant advancement in queueing theory. As expected, by employing SVT, the probability generating functions (PGFs) and steady-state probabilities for various system states have been successfully derived. Although SVT have been used in solving many queueing models, for our model, it is a novel approach not only provides a robust mathematical framework for understanding customer behavior but also enhances the analytical capabilities in evaluating complex queueing systems. The derived PGFs offer valuable insights into the expected number of customers in various states, facilitating better decision-making in operational management.

1.2 Development of Cost Optimization Models in Queue

"Optimization" refers to the process of identifying the best solution for a fitness function. Cost optimization (CO) is a business-oriented, ongoing process aimed at reducing expenses while enhancing corporate value. It involves securing the most favorable pricing and terms for all business transactions, as well as streamlining and rationalizing platforms, applications, processes, and services. In practice, operating costs are closely linked to revenue. To maximize financial success, system developers or administrators prioritize minimizing operational expenses per unit of time.

Sanga and Jain (2019) recently explored the admission strategy within a single-server queueing framework with finite capacity, where dissatisfied customers retry and the retry times are widely variable. They generated a cost function to evaluate the optimal service rate and its associated expected costs. To minimize the system's projected costs, they determined the optimal decision parameter for the service rate using a genetic algorithm (GA) and the Quasi-Newton method (QNM). Malik et al. (2021) investigated a bulk retrial G-queue incorporating repairman delays, Bernoulli feedback (BF), and a reliance on volunteer services. They also applied particle swarm optimization (PSO) to lower costs. Deora et al. (2021) assessed a machining system with standby provisioning, feedback, and a server vacation plan that allowed for regular working hours. They developed a CO function aimed at achieving optimal performance. The PSO approach was prominently utilized to obtain optimal operating conditions while considering commercial performance and minimizing projected costs. Laxmi and Jyothsna (2022) proposed a model for an unlimited buffer impatient client queue with a second optional service (SOS) and scheduled time off. They identified the ideal server service rates using the PSO method to further reduce the system's total expected costs. Certain realistic components, such as batch arrival, regular service, balking, WV, and unstable server, have been incorporated into queueing model Jain and Kumar (2022). A sensitivity analysis and a CO adopting the PSO approach have been created to further evaluate the model's potential. Vaishnawi et al. (2022) explored the ANFIS computing of an unstable single server queueing model that included several stage service and operational vacation. A nonlinear cost function is additionally created, and its reduction using the quasi-Newton technique is examined. Ahuja et al. (2022) analyzed the efficiency of a discrete queueing system for the recurrent customers with BF and two distinct types of vacations. On top of that, they employed graphical representations to assess the convergence of various optimization algorithms, that include direct search (DS), PSO, ABC, Cuckoo search (CS), and GA, in the quest for the best (optimal) system cost. Kumar and Jain (2023) discovered an unstable Markovian queueing model with a two-stage service process and an HV policy. Besides, a cost function has been built that reflects the best possibilities for the service system's decision variables. Applying PSO and ABC optimization algorithms, the optimal service rates at the lowest possible price are computed.

1.3 Motivation and Scope

The research gap identified in the current literature on retrial queue systems involves the lack of attention to the behavior of recurrent customers under vacation modes, particularly in the context of G-queues (queues with negative customers). While numerous studies have been conducted on retrial queues, most focus on standard customer behaviors and system conditions without incorporating vacation mechanisms, where the server may take periodic breaks or operate at reduced capacity (e.g., working vacations). These gaps are crucial because in many real-world applications, servers experience downtime, either planned or unplanned, and customers may attempt to rejoin the queue after being blocked or denied service. This dynamic has significant implications for system performance and cost efficiency, yet it has not been fully explored.

Specifically, while some works, like those of Farahmand (1996) and Moreno (2004), delve into the retry behavior of customers in retrial queues, none have investigated how recurrent customers behave when the system is in vacation mode, such as during a Bernoulli working vacation. This oversight leaves a gap in understanding how recurrent customers interact with a system that intermittently operates at reduced capacity and how this affects system performance, especially in terms of congestion and customer waiting times.

Additionally, another key gap lies in the cost optimization of such systems under complex scenarios like working vacations and server failures. While cost functions have been studied in retrial queues, there has been no significant work on developing advanced cost minimization techniques specifically tailored for retrial G-queues with recurrent customers in vacation modes. The introduction of advanced metaheuristic algorithms such as Grey Wolf Optimizer (GWO), Bat Algorithm (BA), Whale Optimization (WO), and Cat Swarm Optimization (CSO) for minimizing the operational costs in these systems represents a novel contribution to the field. These algorithms can handle the complexity of nonlinear optimization problems, offering better convergence properties and solution quality compared to traditional methods, yet they have not been widely applied to retrial queues with vacation mechanisms. Thus, the aim of this work is to evaluate the orbit size and queue length distributions to aid in the development and evaluation of novel metrics to gauge system behavior.

The structure of our article is as follows: Once these conditions are met, a comprehensive description of the queueing paradigm in Section 2 has been provided. Detailed explanations of the system's steady state (SS) behavior and the PGF of the queue size are provided in Section 3. Indicators of key system behaviors are compiled in Section 4. In Section 5, the results are shown numerically and graphically. In Section 6, GWO, BA, WO and CSO are used to produce the final numerical results and cost analysis. Section 7 provides a concise overview of the paper's main points.

1.4 Methodology
This study focuses on analyzing a single-server retrial queue with recurrent customers and a standby server, operating under Bernoulli working vacation conditions. The methodology followed these steps:

1.4.1 Queueing System Modeling
The system was modeled using the supplementary variable technique. The states of the system were defined based on customer arrivals, server status (busy, idle, or on vacation), and retrial behavior. Customer arrivals were assumed to follow a Poisson process, while service times were considered to follow an exponential distribution. The server's vacation periods were modeled using a Bernoulli process, where the server could either continue providing service at a reduced rate during vacations or remain idle.

1.4.2 State Transition Analysis
The system's behavior was analyzed through balance equations that represent the probabilities of transitions between different states. These equations were solved iteratively to obtain the steady-state probabilities for each system state.

1.4.3 Performance Measures
Key performance measures were derived from the steady-state probabilities. These included metrics such as the average number of customers in the orbit, server utilization, average waiting time in the system, and system throughput. These performance measures were validated through both graphical and tabular analyses to ensure their reliability.

1.4.4 Cost Function Development
A cost function was formulated to capture the trade-offs between operational efficiency and customer satisfaction. The cost function considered factors such as server maintenance costs, customer retrial delays, and the impact of server vacations.

1.4.5 Optimization Techniques
The cost function was minimized using multiple optimization algorithms, including the grey wolf optimizer, bat algorithm, whale optimization algorithm, and cat swarm optimization. Each algorithm's parameters were fine-tuned to ensure optimal performance, and multiple iterations were conducted to evaluate the effectiveness of each technique.

1.4.6 Validation and Comparison
The optimization results were validated through independent runs, and the convergence behavior of each algorithm was analyzed. Performance plots were generated to compare the efficiency, speed, and stability of the algorithms in minimizing the cost function.

1.4.7 Tools and Software
Numerical computations and optimization simulations were performed using MATLAB. Graphical representations of results, such as convergence curves and performance metrics, were also created using MATLAB's visualization tools.

1.4.8 Reproducibility
Detailed information about the parameters, initial conditions, and configurations of the optimization algorithms was documented to enable the study to be reproduced. Key parameter values, such as arrival rates, service rates, and vacation probabilities, were clearly specified in the corresponding tables.

This structured methodology ensures a comprehensive analysis of the retrial queue system while providing a foundation for further research in this area.

2. Description of the Model and its Real-world Application
Under Bernoulli working vacation policy, a *M/G/*1 retrial *G*-queue with recurrent customers and standby server model will be analyzed. A comprehensive outline of our model is given below:

2.1 Arrival Process
The system receives three types of consumers: regular consumers (also known as transmit consumers), disasters (negative consumers), and a fixed number of recurrent consumers (permanent consumers). It is assumed that transit and negative consumers enter the system from external sources at rates α and β, respectively, following independent Poisson processes.

2.2 Retrial Process
When a positive and transmit customer arrives at the server and sees that it is empty, the customer can begin service immediately. If the server is already at capacity or providing slower than expected service, new customers will be added to a group of blocked customers known as an orbit and will have to wait in queue behind the client at the front of the orbit queue before being granted access to the server. The periods between trials follow a random distribution, denoted by $H(x)$, which is transformed using the Laplace Stieltjes Transform (LST) as $H^*(\phi)$.

2.3 Regular Service Process
When a server is in an idle state, standard service is resumed as soon as a new positive or a recurrent customer arrives. The service times of the transit customers, represented by the random variable X is assumed to have a general distribution function denoted by $F_1(x)$, its LST as $F_1^*(\phi)$ and its n^{th} moments as ξ_{1n}.

The system has a fixed number, M, of recurrent clients. Recurring clients instantly rejoin the retrial group after being served, in compliance with an FCFS discipline. It is presumed that access to the server is restricted to just the recurrent client at the top of the orbit. The group's top repeat client calls back after a period of time that follows an exponential distribution with a mean of $1/\gamma$. The service times of the recurrent customers represented by the random variable Y is assumed to have a general distribution function denoted by $F_2(y)$, its LST as $F_2^*(\phi)$ and its n^{th} moments as ξ_{2n}.

2.4 Working Vacation (WV, Service during the Server's Vacation)
The server goes on vacation if the orbit is free. Vacation time has an exponential distribution function with rate of ω. While on vacation, the server can yet perform service but with slower rate. This duration is referred to as the working vacation period and is denoted briefly by WV period. Furthermore, the vacation will be halted and the server will get back to normal operations with its general distribution $F_1(x)$ while on

vacation unless any orbital consumer joins the system.
Now, when the server's vacation ends and there is no consumer in the system, one of the following cases may happen according to Bernoulli distribution:

(i) With probability q, the server stays idle until a new consumer arrives, in which case the customer will receive the regular service, which in this case the server has taken a single WV with general service distribution function $F_1(x)$ and its LST by $F_1^*(\phi)$, otherwise,

(ii) With probability $p = 1- q$ departs for another WV, in which case the server is taking multiple WV. The service times during the multiple WV by V with a general distribution function denoted by $F_V(v)$, and its LST dented by $F_v^*(\phi)$ have been represented.

2.5 Removal Rule and Breakdown Process

The negative consumers (disasters) only arrive during the usual service periods for the positive clients. Negative customers will eliminate positive customers from the system while they are still in service, who cannot build up a wait and prevent them from receiving service. These kinds of unfavorable clients bring about server failure, which results in a brief outage of the service channel. The breakdown rate indicated by the symbol η.

2.6 Repair Process

The repair process begins right away following the breakdown and doesn't wait any longer. The repair times are independent and identical distributed random variables which are general distributed with PDF $G(x)$ and its LST as $G^*(\phi)$.

2.7 Standby Server

When the primary server is under maintenance, clients continue to receive service from a standby server. The standby service rates are $\vartheta > 0$, with mean standby service times of $1/\vartheta$, which leads to an exponential distribution for the standby service times. Upon the return of the primary server after an outage, the client who was being served by the standby server is transferred to the primary server to restore service.

The stochastic processes are regarded as completely independent from one another. **Figure 1** provides a visual representation of the model, demonstrating that the stochastic processes within the system operate independently.

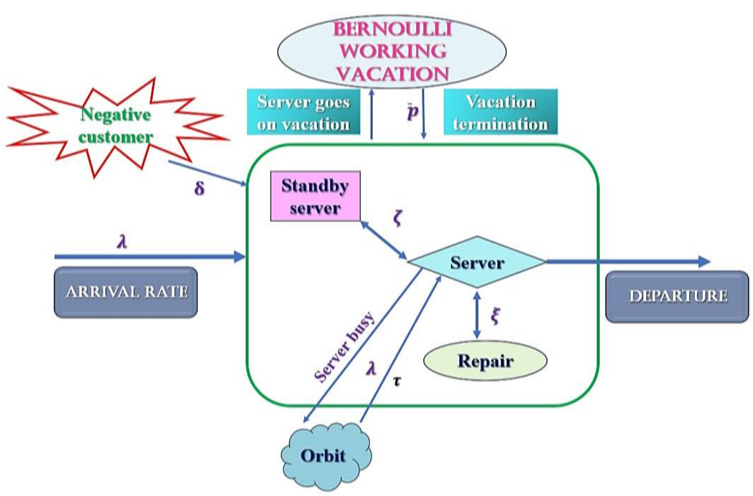


Figure 1. Pictorial representation of the model.

3. Practical Application of the Model

The primary domain of use for retrying queueing systems is in the realm of telecommunications. This research has implications for other information processing systems, including warehouse order processing systems. Client places an order for product (positive customer) in the production order system; other clients cancel purchases (negative customer) due to financial crisis or other concerns.

The Poisson process is assumed to govern the ordering scheme. Instead of wasting resources verifying the server's availability before allowing new customers to join the orbit, it may be more efficient to permit some new customers to join the orbit immediately upon arrival if the system is particularly busy. To address this, a policy of checking the server at intervals is implemented to prevent deadlock between commands examining the server. Repeat purchases can be represented as recurrent customers. In a queueing system where customers remain in constant contact with the server, they might reasonably expect the server to stay busy as long as there are people waiting in line. However, in systems where customers are not in continuous contact with the server, this expectation may not hold. There could be as many clients in the orbit as there are servers available to serve them if all clients must wait for the server to call them for assistance. If a breakdown occurs during service, all server operations will cease. In such cases, the server will be repaired as quickly as possible. Once maintenance is completed, the server will resume its normal operation, transitioning between busy and vacation modes as needed.

To further improve manufacturing plant productivity (during vacations), the management plan includes establishing a supplementary employment facility. Production will cease right away as the final client is serviced and the system is idle, and it won't resume unless the plant is needed for discretionary work (a single WV) at a reduced rate (a slower service rate). Following a disaster or the completion of an optional project, the factory will restart regular operations, albeit more slowly, until new orders (multiple WVs) are received. Once the factory's optional works are finished (vacation interruption), primary production can restart on a fresh order. The employment of such a tool is quite helpful for maximizing production facility productivity and preventing unnecessary strain.

4. Steady-state Probabilities

By factoring in the elapsed retrial times, elapsed regular service times, elapsed slower rate service are taken as a supplementary variable to formulates the steady-state (SS) equations for the retrial system. Next, the PGF for the total number of consumers in the system and orbit is calculated, as well as the orbit size generating functions (GFs) for each of the possible server states.

Below, most of the essential notations, including the new probabilities, being using are listed with their definitions for the comfort of readers:

Notations	Explanations
α	Arrival rate of transit consumers (Poisson process)
β	Arrival rate of negative consumers (Poisson process)
M	Number of recurrent customers
γ	Mean time for recurrent customer to return to orbit
θ	Rate of the server's vacation time (exponential distribution)
q	Probability that the server stays idle after a vacation
$p = 1 - q$	Probability that the server departs for another vacation
η	Breakdown rate of the server

ϑ	Service rate of the standby server
$h(x)$	Retrial completion rate function
$\sigma_i(x)$	Service completion rate function during normal service for the i^{th} server
$\sigma_v(x)$	Service completion rate function during working vacation
$\chi_i(x)$	Repair completion rate function for the i^{th} server
$\Phi_{\{i,n\}}(x,t)$	Probability of n customers with x during retrial time t
$\Omega_{\{i,n\}}(x,t)$	Probability of n customers with x during TCS (Transit Customer Service) and RCS (Recurrent Customer Service) at time t on normal busy period
$\Upsilon_{\{i,n\}}(x,t)$	Probability of n customers with x during vacation time t
$\Psi_{\{i,n\}}(x,t)$	Probability of n customers with x during repair in TCS and RCS at time t on repair period

4.1 The Steady-State Equations

By assumption, SS performs under the premise that $H(0) = 0$, $H(\infty) = 1$, $F_1(0) = 0$, $F_1(\infty) = 1$, $F_2(0) = 0$, $F_2(\infty) = 1$, $F_v(0) = 0$, $F_v(\infty) = 1$ and $G_i(0) = 0$, $G_i(\infty) = 1$, $i = 1,2$ are continuous at $x = 0$. Accordingly, the completion rates for retrial, regular service, vacation, and repair are represented by the functions $h(x)$, $\sigma_i(x)$, $\sigma_v(x)$ and $\chi_i(x), = 1,2$, and $\beta_{r_2}(x)$, respectively.

$$h(x)dx = \frac{dH(x)}{1 - H(x)}; \quad \sigma_i(x)dx = \frac{dF_i(x)}{1 - F_i(x)}, \qquad i = 1,2;$$
$$\sigma_v(x)dx = \frac{dF_v(x)}{1 - F_v(x)}; \quad \chi_i(x)dx = \frac{dG_i(x)}{1 - G_i(x)}, \qquad i = 1,2.$$

Apart from it, let $H^0(t)$, $F_i^0(t)$, $F_v^0(t)$ and $G^0{}_i(t)$ be the elapsed retrial times, the elapsed times of normal service and the elapsed vacation times, elapsed repair time respectively at time t. Furthermore, the random variable generated by,

$$\Pi(t) = \begin{cases} 0, & \text{the server is idle and in WV mode,} \\ 1, & \text{the server is idle and in normal service mode,} \\ 2, & \text{the server is full and transit customer service (TCS) mode,} \\ 3, & \text{the server is full and recurrent customer service (RCS)mode,} \\ 4, & \text{the server is full and in working vacation mode,} \\ 5, & \text{the server is in repair in transit mode, and} \\ 6, & \text{the server is in repair in recurrent mode.} \end{cases}$$

Thus, the SV $H^0(t)$, $F_i^0(t)$, $F_v^0(t)$ and $G_i^0(t), i = 1,2$ a bivariate Markov method should be constructed $\{\Pi(t), \aleph(t); t \geq 0\}$, here $\Pi(t)$ denote the system state $(0,1,2,3,4,5,6)$ depend on accessibility of the server during the idle or busy on regular service in TCS, RCS, WV and repair periods. $\aleph(t)$ denotes the number of customers in the orbit. If $\Pi(t) = 1$ and $\aleph(t) > 0$, then $H^0(t)$ is relates to the elapsed retrial time. If $\Pi(t) = 2$ and $\aleph(t) \geq 0$, then $F_1^0(t)$ is relates to the elapsed time of customer being served in TCS during regular busy period. If $\Pi(t) = 3$ and $\aleph(t) \geq 0$, then $F_2^0(t)$ is relates to the elapsed time of customer being served in RCS during regular busy period. If $\Pi(t) = 4$ and $\aleph(t) \geq 0$, then $F_v^0(t)$ is relates to the elapsed time of the customer being served in vacation rate service period. If $\Pi(t) = 5,6$ and $\aleph(t) \geq 0$, then $G_i^0(t), i = 1,2$ is relates to the elapsed time of the server is in repair mode during TCS and RCS.

Theorem 4.1

The embedded Markov chain (MC) $\{An; n\epsilon N\}$ is ergodic if and only if $\Theta < \alpha + \gamma$, which guarantees that the system remains stable. In other words, the system will operate without instability as long as Θ stays below the sum of α and γ, where,

$$\Theta = \left(\alpha + \gamma H^*(\alpha+\gamma)\right)\left\{\frac{\alpha}{\beta}\left(1+\eta E(G_1)\right)\left(1-F_1^*(\beta)\right)\right\} - \left(1-H^*(\alpha+\gamma)\right)$$
$$.\{\alpha + \gamma[F_2^*(\beta) + \frac{\alpha}{\beta}(1+\eta E(G_2))(1-F_2^*(\beta))]\}.$$

Proof

It is quite easy to verify the necessary condition of ergodicity by employing Pakes (1969) criterion, which asserts that the chain $\{\mathcal{A}_n; n \in \mathcal{N}\}$ is an irreducible and aperiodic. The mean value is provided below, with a assumption that a non-negative measure $e(\varepsilon)$, $\varepsilon \in \mathcal{N}$ and $\epsilon > 0$, and the MC is ergodic,

$$\varphi_\varepsilon = \mathcal{E}\left[e(v_{n+1}) - \frac{e(v_n)}{v_n} = \varepsilon\right],$$

with the limited exception ε's, $\varepsilon \in N$ and $\varphi_\varepsilon \leq -\in \forall\ \varepsilon \in \mathcal{N}$. In such scenario, $e(\varepsilon) = \varepsilon$ Is taken. As a result, the following is obtained:

$$\varphi_\varepsilon = \begin{cases} \Theta - 1, & \text{if } \varepsilon = 0, \\ \Theta - (\alpha+\gamma), & \text{if } \varepsilon = 1,2,\ldots \end{cases}.$$

Yet, it is evident that $\Theta < (\alpha + \gamma)$ requires ergodicity. Sennott et al. (1983) states that the prerequisite is satisfied if MC $\{\mathcal{A}_n; n\epsilon\mathcal{N}\}$ fits Kaplan's status, which is typically $\varphi_\varepsilon < \infty$, $\forall\ \varepsilon \geq 0$ and there exists an $\varepsilon_0 \in \mathcal{N}$ s.t $\varphi_\varepsilon \geq 0$ for $\varepsilon \geq \varepsilon_0$. Here, $\mathcal{W} = (w_{m\varepsilon})$ denotes the unit-step transition matrix (UTM) of $\{\mathcal{A}_n; n \in \mathcal{N}\}$ for $\varepsilon < m - i$ and $m > 0$. Thus, $\Theta \geq (\alpha + \gamma)$ provides the non-ergodicity of the MC.

The duration of transition between service intervals denoted by $\{t_n; n = 1,2,\ldots\}$. Consequently, using a number of arbitrary vectors $A_n = \{\Pi(t_n+), \aleph(t_n+)\}$, a Markov chain has formed and incorporated in the RQ system. As a result of Theorem 4.1 $\{A_n; n \in N\}$ is ergodic iff $\Theta < \alpha + \gamma$, to maintain the stable system, where:

$$\Theta = \left(\alpha + \gamma H^*(\alpha+\gamma)\right)\left\{\frac{\alpha}{\beta}\left(1+\eta E(G_1)\right)\left(1-F_1^*(\beta)\right)\right\} - (1-H^*(\alpha+\gamma))\{\alpha + \gamma[F_2^*(\beta) + \frac{\alpha}{\beta}(1 + \eta E(G_2))(1-F_2^*(\beta))]\}.$$

For the method $\{\aleph(t), t \geq 0\}$, the probabilities $\Phi_{0,M}(t) = P\{\Pi(t) = 0, \aleph(t) = 0\}$ and the probability density functions are specified as:

$$\Phi_n(x,t)dx = P\{\Pi(t) = 1, \aleph(t) = n, x \leq H^0(t) < x + dx\},$$
$$\text{for } t \geq 0,\ x \geq 0 \text{ and } n \geq 1.$$

$$\Omega_{i,n}(x,t)dx = P\{\Pi(t) = 2,3, \aleph(t) = n, x \leq F_i^0(t) < x + dx\},$$
$$\text{for } t \geq 0,\ x \geq 0,\ i = 1,2 \text{ and } n \geq 0.$$

$$\Upsilon_{v,n}(x,t)dx = P\{\Pi(t) = 4, \aleph(t) = n, x \leq F_v^0(t) < x + dx\},$$
$$\text{for } t \geq 0,\ x \geq 0 \text{ and } n \geq 0.$$

$$\Psi_{i,n}(x,t)dx = P\{\Pi(t) = 5,6, \aleph(t) = n, x \leq G_i^0(t) < x + dx\},$$
$$\text{for } t \geq 0,\ x \geq 0,\ i = 1,2 \text{ and } n \geq 0.$$

The resources may be alloted, if the sequel meets the stability requirement $\Phi_{0,M} = lim_{t\to\infty}\Phi_{0,M}(t)$ and the density functions that limit are:

$$\Phi_n(x) = lim_{t\to\infty}\Phi_n(x,t); \quad \Omega_{i,n}(x) = lim_{t\to\infty}\Omega_{i,n}(x,t),\ i = 1,2;$$

$$\Upsilon_{v,n}(x) = lim_{t\to\infty}\Upsilon_{v,n}(x,t); \quad \Psi_{i,n}(x) = lim_{t\to\infty}\Psi_{i,n}(x,t),\ i = 1,2.$$

By employing SVT, the following system of equations will be generated:

$$(\alpha+\gamma)\Phi_{0,M} = \bar{\kappa}\int_0^\infty \Omega_{1,M}(x)\sigma_1(x)dx + \bar{\kappa}\int_0^\infty \Omega_{2,M-1}(x)\sigma_2(x)dx + \int_0^\infty \Upsilon_{v,0}(x)\sigma_v(x)dx \quad (1)$$

$$\frac{d}{dx}\Phi_n(x) + (\alpha+\gamma+h(x))\Phi_n(x) = 0, n \geq M+1 \quad (2)$$

$$\frac{d}{dx}\Omega_{1,n}(x) + (\alpha+\beta+\eta+\sigma_1(x))\Omega_{1,n}(x) = \alpha(1-\delta_{n,M})\Omega_{1,n-1}(x) + \int_0^\infty \Psi_{1,n}(x)\chi_1(x)dx, n \geq M \quad (3)$$

$$\frac{d}{dx}\Omega_{2,n}(x) + \left(\alpha+\beta+\eta+\sigma_2(x)\right)\Omega_{2,n}(x) = \alpha\left(1-\delta_{n,M-1}\right)\Omega_{2,n-1}(x) + \int_0^\infty \Psi_{2,n}(x)\chi_2(x)dx, n \geq M-1 \quad (4)$$

$$\frac{d}{dx}\Upsilon_{v,n}(x) + \left(\alpha+\omega+\sigma_v(x)\right)\Upsilon_{v,n}(x) = \alpha\left(1-\delta_{n,M}\right)\Upsilon_{v,n-1}(x), n \geq M \quad (5)$$

$$\frac{d}{dx}\Psi_{1,n}(x) + \left(\alpha+\vartheta+\chi_1(x)\right)\Psi_{1,n}(x) = \vartheta\Psi_{1,n}(x) + \alpha(1-\ \delta_{n,M})\Psi_{1,n-1}(x), n \geq M \quad (6)$$

$$\frac{d}{dx}\Psi_{2,n}(x) + \left(\alpha+\vartheta+\chi_2(x)\right)\Psi_{2,n}(x) = \vartheta\Psi_{2,n}(x) + \alpha(1-\delta_{n,M})\Psi_{2,n-1}(x), n \geq M-1 \quad (7)$$

Then, at $x = 0$, the SS boundary conditions are as follows:

$$\Phi_n(0) = \int_0^\infty \Omega_{1,n}(x)\sigma_1(x)dx + \int_0^\infty \Omega_{2,n-1}(x)\sigma_2(x)dx + \int_0^\infty \Upsilon_{v,n}(x)\sigma_v(x)dx, n \geq M+1 \quad (8)$$

$$\Omega_{1,n}(0) = \int_0^\infty \Phi_{n+1}(x)h(x)dx + \alpha\left(1-\delta_{n,M}\right)\int_0^\infty \Phi_n(x)dx\ +\ \omega\ \int_0^\infty \Upsilon_{v,n}(x)dx + \alpha\delta_{n,M}\Phi_{0,M}, n \geq M \quad (9)$$

$$\Omega_{2,n}(0) = (1-\delta_{n,M-1})\gamma\int_0^\infty \Phi_{n+1}(x)dx + \gamma\delta_{n,M-1}\Phi_{0,M}, n = 0 \quad (10)$$

$$\Upsilon_{v,n}(0) = \begin{cases}\alpha\Phi_0, & n = 0,\\ 0, & n \geq 1,\end{cases} \quad (11)$$

$$\Psi_{1,n}(0) = \eta\Omega_{1,n}(x), n \geq M \quad (12)$$

$$\Psi_{2,n}(0) = \eta\Omega_{2,n}(x), n \geq M-1 \quad (13)$$

where, $\delta_{i,j}$ denotes the Kroneker's delta function.

The normalizing condition is:

$$\Phi_{0,M} + \sum_{n>M}^{\infty}\int_0^\infty \Phi_n(x)dx + \sum_{n\geq M}^{\infty}\left(\int_0^\infty \Omega_{1,n}(x)dx + \int_0^\infty \Upsilon_{v,n}(x)dx + \int_0^\infty \Psi_{1,n}(x)dx\right) + \sum_{n\geq M-1}^{\infty}\left(\int_0^\infty \Omega_{2,n}(x)dx + \int_0^\infty \Psi_{2,n}(x)dx\right) = 1.$$

4.2 The Steady-State Solution

The probability generating function (PGF) approach is applied to determine the SS solution to the RQ model. The PGFs for $|\tilde{z}| \leq 1$ in the preceding equations are described accordingly as:

$$\Phi(x,\tilde{z}) = \sum_{n>M}^{\infty} \Phi_n(x)\tilde{z}^n; \quad \Phi(0,\tilde{z}) = \sum_{n>M}^{\infty} \Phi_n(0)\tilde{z}^n;$$

$$\Omega_1(x,\tilde{z}) = \sum_{n\geq M}^{\infty} \Omega_{1,n}(x)\tilde{z}^n; \quad \Omega_1(0,\tilde{z}) = \sum_{n\geq M}^{\infty} \Omega_{1,n}(0)\tilde{z}^n;$$

$$\Omega_2(x,\tilde{z}) = \sum_{n\geq M-1}^{\infty} \Omega_{1,n}(x)\tilde{z}^n; \quad \Omega_2(0,\tilde{z}) = \sum_{n\geq M-1}^{\infty} \Omega_{1,n}(0)\tilde{z}^n;$$

$$\Upsilon_v(x,\tilde{z}) = \sum_{n\geq M}^{\infty} \Upsilon_{v,n}(x)\tilde{z}^n; \quad \Upsilon_v(0,\tilde{z}) = \sum_{n\geq M}^{\infty} \Upsilon_{v,n}(0)\tilde{z}^n;$$

$$\Psi_1(x,\tilde{z}) = \sum_{n\geq M}^{\infty} \Psi_{1,n}(x)\tilde{z}^n; \quad \Psi_1(0,\tilde{z}) = \sum_{n\geq M}^{\infty} \Psi_{1,n}(0)\tilde{z}^n, i = 1,2;$$

$$\Psi_2(x,\tilde{z}) = \sum_{n\geq M-1}^{\infty} \Psi_{1,n}(x)\tilde{z}^n; \quad \Psi_2(0,\tilde{z}) = \sum_{n\geq M-1}^{\infty} \Psi_{1,n}(0)\tilde{z}^n, i = 1,2.$$

Now, from (2) to (13), multiply the SS equation and SS boundary conditions by $\tilde{z}^n$ and sum over n, $(n = 0,1,2,\dots)$, are obtain as:

$$\frac{\partial}{\partial x}\Phi(x,\tilde{z}) + \big(\alpha + \gamma + h(x)\big)\Phi(x,\tilde{z}) = 0 \tag{15}$$

$$\frac{\partial}{\partial x}\Omega_1(x,\tilde{z}) + \big(\beta + \eta + \alpha(1-\tilde{z}) + \sigma_1(x)\big)\Omega_1(x,\tilde{z}) - \int_0^\infty \Psi_{1,n}(x)\chi_1(x)dx = 0 \tag{16}$$

$$\frac{\partial}{\partial x}\Omega_2(x,\tilde{z}) + \big(\beta + \eta + \alpha(1-\tilde{z}) + \sigma_2(x)\big)\Omega_2(x,\tilde{z}) - \int_0^\infty \Psi_{2,n}(x)\chi_2(x)dx = 0 \tag{17}$$

$$\frac{\partial}{\partial x}\Upsilon_v(x,\tilde{z}) + \big(\omega + \alpha(1-\tilde{z}) + \sigma_v(x)\big)\Upsilon_v(x,\tilde{z}) - \int_0^\infty \Psi_{2,n}(x)\chi_2(x)dx = 0 \tag{18}$$

$$\frac{\partial}{\partial x}\Psi_1(x,\tilde{z}) + (\vartheta(1-\tfrac{1}{\tilde{z}}) + \alpha(1-\tilde{z}) + \chi_1(x))\Psi_1(x,\tilde{z}) = 0 \tag{19}$$

$$\frac{\partial}{\partial x}\Psi_2(x,\tilde{z}) + (\vartheta(1-\tfrac{1}{\tilde{z}}) + \alpha(1-\tilde{z}) + \chi_2(x))\Psi_2(x,\tilde{z}) = 0 \tag{20}$$

$$\Phi(0,\tilde{z}) = \int_0^\infty \Omega_1(x,\tilde{z})\sigma_1(x)dx + \tilde{z}\int_0^\infty \Omega_2(x,\tilde{z})\sigma_2(x)dx - \beta\{\int_0^\infty \Omega_1(x,\tilde{z})dx + \tilde{z}\int_0^\infty \Omega_2(x,\tilde{z})dx\} + \tilde{z}\int_0^\infty \Upsilon_v(x,\tilde{z})\sigma_v(x)dx \ - (\alpha+\gamma)\Phi_{0,M}\tilde{z}^M \tag{21}$$

$$\Omega_1(0,\tilde{z}) = \frac{1}{\tilde{z}}\int_0^\infty \Phi(x,\tilde{z})h(x)dx + \alpha\int_0^\infty \Phi(x,\tilde{z})dx + \omega\int_0^\infty \Upsilon_v(x,\tilde{z})dx + \alpha\Phi_{0,M}\tilde{z}^M \tag{22}$$

$$\Omega_2(0,\tilde{z}) = \frac{\gamma}{\tilde{z}}\int_0^\infty \Phi(x,\tilde{z})dx + \gamma\Phi_{0,M}\tilde{z}^{M-1} \tag{23}$$

$$\Upsilon_v(0,\tilde{z}) = \alpha\Phi_{0,M}\tilde{z}^M \tag{24}$$

$$\Psi_1(0,\tilde{z}) = \eta\Omega_1(x,\tilde{z}) \tag{25}$$

$$\Psi_2(0,\tilde{z}) = \eta\Omega_2(x,\tilde{z}) \tag{26}$$

Solving the partial differential Equations (15) to (20), the following are obtained:

$$\Phi(x,\tilde{z}) = \Phi(0,\tilde{z})[1-H(x)]e^{-(\alpha+\gamma)x} \tag{27}$$

$$\Omega_1(x,\tilde{z}) = \Omega_1(0,\tilde{z})[1-F_1(x)]e^{-L_1(\tilde{z})x} \tag{28}$$

$$\Omega_2(x,\tilde{z}) = \Omega_2(0,\tilde{z})[1-F_2(x)]e^{-L_2(\tilde{z})x} \tag{29}$$

$$\Upsilon_v(x,\tilde{z}) = \Upsilon_v(0,\tilde{z})[1-F_v(x)]e^{-L_v(\tilde{z})x} \tag{30}$$

$$\Psi_1(x,\tilde{z}) = \Psi_1(0,\tilde{z})[1-G_1(x)]e^{-L_r(\tilde{z})x} \tag{31}$$

$$\Psi_2(x,\tilde{z}) = \Psi_2(0,\tilde{z})[1-G_2(x)]e^{-L_r(\tilde{z})x} \tag{32}$$

where,

$$L(\tilde{z}) = \alpha(1-\tilde{z}),\ L_v(\tilde{z}) = \omega + L(\tilde{z}),\ L_r(\tilde{z}) = L(\tilde{z}) + \vartheta(1-\frac{1}{\tilde{z}}),$$

and

$$L_1(\tilde{z}) = \beta + L(\tilde{z}) + \eta - \eta G_1^*(L(\tilde{z})),\ L_2(\tilde{z}) = \beta + L(\tilde{z}) + \eta - \eta G_2^*(L(\tilde{z})).$$

Inserting the Equations (27) to (29) in (22) and making some calculations, finally, the following is obtained:

$$\Omega_1(0,\tilde{z}) = \frac{\Phi(0,\tilde{z})}{\tilde{z}}\left\{H^*(\alpha+\gamma) + \frac{\alpha\tilde{z}}{\alpha+\gamma}[1-H^*(\gamma)]\right\} + V(\tilde{z})\Upsilon_v(0,\tilde{z}) + \alpha\Phi_{0,M}\tilde{z}^M \tag{33}$$

From Equation (29), the following is attained:

$$\Omega_2(0,\tilde{z}) = \frac{\gamma}{\tilde{z}(\alpha+\gamma)}[1-H^*(\alpha+\gamma)]\Phi(0,\tilde{z}) + \frac{\gamma}{\tilde{z}}\Phi_{0,M}\tilde{z}^M \tag{34}$$

Equation (20) implies that:

$$\Phi(0,\tilde{z}) = [F_1(L_1^*(\tilde{z})) + W_1(\tilde{z})]\Omega_1(0,\tilde{z}) + [F_2(L_2^*(\tilde{z})) + \frac{W_2(\tilde{z})}{\tilde{z}}]\tilde{z}\Omega_2(0,\tilde{z}) - (\alpha+\gamma)\Phi_{0,M}\tilde{z}^M \tag{35}$$

where

$$W_1(\tilde{z}) = \frac{\beta}{L_1(\tilde{z})}(1-F_1^*(L_1(\tilde{z}))),\ W_2(\tilde{z}) = \frac{\beta}{L_2(\tilde{z})}(1-F_2^*(L_2(\tilde{z}))),$$

$$V(\tilde{z}) = \frac{\omega}{\omega+\alpha(1-\tilde{z})}(1-F_v^*(L_v(\tilde{z}))).$$

Combining (33) and (34) in (35), leads to the following:

$$\Phi(0,\tilde{z})\{\tilde{z}(\alpha+\gamma) - [F_1^*(L_1(\tilde{z})) + W_1(\tilde{z})]\{(\alpha+\gamma)H^*(\alpha+\gamma) + \alpha\tilde{z}(1-H^*(\alpha+\gamma))\} - \gamma(1-H^*(\alpha+\gamma))[\tilde{z}F_2^*(L_2(\tilde{z})) + W_2(\tilde{z}) = (\alpha+\gamma)\Phi_{0,M}\tilde{z}^{M+1}\{[\alpha(V(\tilde{z})+1)][F_1^*(L_1(\tilde{z})) + W_1(\tilde{z})] + \gamma[\tilde{z}F_2^*(L_2(\tilde{z})) + W_2(\tilde{z})]\} \tag{36}$$

Next, the marginal orbit size distributions formed by the server's current state in a subsequent theorem is examinned.

Theorem 4.2

According to the stability condition, $\Theta < \alpha + \gamma$, the stationary distribution of the number of clients in orbit is computed for the server's inactive, occupied, slow service, and repair periods, along with the probability of the server being inactive are computed below:

$$\Phi(\tilde{z}) = \frac{Ne(\tilde{z})}{De(\tilde{z})} \tag{37}$$

where,

$$Ne(\tilde{z}) = (\alpha+\gamma)\Phi_{0,M}\tilde{z}^{M+1}\{[\alpha(V(\tilde{z})+1)][F_1^*(L_1(\tilde{z})) + W_1(\tilde{z})] + \gamma[\tilde{z}F_2^*(L_2(\tilde{z})) + W_2(\tilde{z})]\},$$

$$De(\tilde{z}) = \tilde{z}(\alpha+\gamma) - [F_1^*(L_1(\tilde{z})) + W_1(\tilde{z})]\{(\alpha+\gamma)H^*(\alpha+\gamma) + \alpha\tilde{z}(1-H^*(\alpha+\gamma))\} - \gamma(1-H^*(\alpha+\gamma))[\tilde{z}F_2^*(L_2(\tilde{z})) + W_2(\tilde{z})].$$

$$\Omega_1(\tilde{z}) = \frac{\Phi_{0,M}\tilde{z}^M(1-F_1^*(L_1(\tilde{z})))}{L_1(\tilde{z})De(\tilde{z})}\{\tilde{z}(\alpha+\gamma)[\alpha(V(\tilde{z})+1)] + \gamma[\tilde{z}F_2^*(L_2(\tilde{z})) + W_2(\tilde{z})]\{(\alpha+\gamma)H^*(\alpha+\gamma) + \alpha(1-H^*(\alpha+\gamma))[\tilde{z}-V(\tilde{z})+1]\}\} \tag{38}$$

$$\Omega_2(\tilde{z}) = \frac{\gamma\Phi_{0,M}\tilde{z}^{M-1}(1-F_2^*(L_2(\tilde{z})))}{(\alpha+\gamma)L_2(\tilde{z})De(\tilde{z})}\{(1-H^*(\alpha+\gamma)) + (\alpha+\gamma)\{\tilde{z}(\alpha+\gamma) - [F_1^*(L_1(\tilde{z})) + W_1(\tilde{z})]((\alpha+\gamma)H^*(\alpha+\gamma) + \alpha\tilde{z}(1-H^*(\alpha+\gamma))) - \gamma(1-H^*(\alpha+\gamma))[\tilde{z}F_2^*(L_2(\tilde{z})) + W_2(\tilde{z})]\}\} \tag{39}$$

$$\Upsilon_v(\tilde{z}) = \frac{\alpha\Phi_{0,M}\tilde{z}^{M}(1-F_v^*(L_v(\tilde{z})))}{L_v(\tilde{z})} \tag{40}$$

$$\Psi_1(\tilde{z}) = \frac{\eta\Phi_{0,M}\tilde{z}^{M}\left(1-F_1^*(L_1(\tilde{z}))\right)\left(1-G_1^*(L_r(\tilde{z}))\right)}{L_1(\tilde{z})L_r(\tilde{z})De(\tilde{z})}\{\tilde{z}(\alpha+\gamma)[\alpha(V(\tilde{z})+1)] + \gamma\,[\tilde{z}F_2^*(L_2(\tilde{z})) + W_2(\tilde{z})]\{(\alpha+\gamma)H^*(\alpha+\gamma) + \alpha(1-H^*(\alpha+\gamma))[\tilde{z} - V(\tilde{z}) + 1]\}\} \tag{41}$$

$$\Psi_2(\tilde{z}) = \frac{\eta\Phi_{0,M}\tilde{z}^{M-1}(1-F_2^*(L_2(\tilde{z})))(1-G_2^*(L_r(\tilde{z})))}{(\alpha+\gamma)L_2(\tilde{z})L_r(\tilde{z})De(\tilde{z})}\{(1-H^*(\alpha+\gamma)) + (\alpha+\gamma)\{\tilde{z}(\alpha+\gamma) - [F_1^*(L_1(\tilde{z})) + W_1(\tilde{z})]((\alpha+\gamma)H^*(\alpha+\gamma) + \alpha\tilde{z}\ (1-H^*(\alpha+\gamma))) - \gamma(1-H^*(\alpha+\gamma))[\tilde{z}F_2^*(L_2(\tilde{z})) + W_2(\tilde{z})]\}\} \tag{42}$$

where,

$$\Phi_{0,M} = \frac{Nr(\Phi_{0,M})}{Dr(\Phi_{0,M})} \tag{43}$$

$$Nr(\Phi_{0,M}) = (\alpha+\gamma) - \Theta, \text{ and}$$

$$\begin{gathered}
Dr(\Phi_{0,M}) = (\omega + \alpha(1-F_v^*(\omega)))(\frac{(\alpha+\gamma)-\Theta}{\omega}) + (1-H^*(\alpha+\gamma)) \\
\{(M+1)\{1+\alpha(2-F_v^*(\omega))\} + [\alpha(2-F_v^*(\omega))][\frac{\alpha}{\beta}(1+\eta E(G_1))(1-F_1^*(\beta))] \\
+\frac{\alpha\gamma}{\beta}(1+\eta E(G_2))(1-F_2^*(\beta)) + \alpha[E(F_v) + \frac{\alpha}{\omega}(1-F_v^*(\omega))] + \gamma\} \\
+(E(F_1)(\eta E(G_1)-1))\{[\alpha E(F_v) + \frac{\alpha}{\omega}(1-F_v^*(\omega))](\alpha\gamma(1-H^*(\alpha+\gamma)) \\
-(\alpha+\gamma)) - (\alpha+\gamma)(M+1)[\alpha(2-F_v^*(\omega))] - \alpha\gamma(1-H^*((\alpha+\gamma))) \\
-\gamma[(\alpha+\gamma)H^*((\alpha+\gamma)) - \alpha(F_v^*(\omega)-1)(1-H^*(\alpha+\gamma))]\{M + F_2^*(\beta) \\
+\frac{\alpha}{\beta}(1+\eta E(G_2))(1-F_2^*(\beta))\}\} + (\gamma E(F_2)(\eta E(G_2)-1)) \\
\{[(\alpha+\gamma)H^*(\alpha+\gamma) + \alpha(1-H^*(\alpha+\gamma))][\frac{\alpha}{\beta}(1+\eta E(G_1))(1-F_1^*(\beta))] \\
-(1-H^*(\alpha+\gamma))\{\frac{M-1}{\alpha+\gamma} + 2\alpha(M-1) - \alpha - \gamma[F_2^*(\beta) + \frac{\alpha}{\beta}(1+\eta E(G_2)) \\
(1-F_2^*(\beta))]\} - (\alpha+\gamma)\}.
\end{gathered}$$

Proof

By integrating Equations (27)-(32) with respect to x and defining the partial probability generating functions as follows,

$$\Phi(\tilde{z}) = \int_0^\infty \Phi(x,\tilde{z})dx,\ \ \Omega_1(\tilde{z}) = \int_0^\infty \Omega_1(x,\tilde{z})dx,\ \ \Omega_2(\tilde{z}) = \int_0^\infty \Omega_2(x,\tilde{z})dx,$$

$$\Upsilon_v(\tilde{z}) = \int_0^\infty \Upsilon_v(x,\tilde{z})dx,\ \ \Psi_1(\tilde{z}) = \int_0^\infty \Psi_1(x,\tilde{z})dx, \text{ and }\ \Psi_2(\tilde{z}) = \int_0^\infty \Psi_2(x,\tilde{z})dx.$$

Considering the idle unknown (Φ_0), the probability that the server is inactive as long as there is no client in orbit, it could potentially be determined by employing the normalizing condition. Therefore, by placing $\tilde{z} = 1$ in (37)-(42) and adhering to the L-hospitals rule when required, the following is obtained:

$$\Phi_{0,M} + \Phi(1) + \Omega_1(1) + \Omega_2(1) + \Upsilon_v(1) + \Psi_1(1) + \Psi_2(1) = 1.$$

Theorem 4.3
The probability generating function of the number of clients in the system and the orbit length at a stationary instant, based on the stability criterion $\Theta<\alpha+\gamma$, is derived as follows:

$$K_s(\tilde{z}) = \Phi_{0,M}\{\frac{Ne_s(\tilde{z})}{De_s(\tilde{z})}\} \tag{44}$$

$$\begin{gathered} Ne_s(\tilde{z}) = L_1(\tilde{z})L_2(\tilde{z})L_r(\tilde{z})\{De(\tilde{z})[1 + \frac{\alpha\tilde{z}^M\left(1 - F_v^*\left(L_v(\tilde{z})\right)\right)}{L_v(\tilde{z})} \\ +\tilde{z}^{M+1}[1 - H^*(\alpha + \gamma)]\{[\alpha(V(\tilde{z}) + 1)][F_1^*(L_1(\tilde{z})) + W_1(\tilde{z})] + \gamma[\tilde{z}F_2^*(L_2(\tilde{z})) \\ + W_2(\tilde{z})]\}\} + L_2(\tilde{z})\left[L_r(\tilde{z}) + \eta\left(1 - G_1^*\left(L_r(\tilde{z})\right)\right)\right][1 - F_1^*\left(L_1(\tilde{z})\right)] \\ \tilde{z}^{M+1}\{\tilde{z}(\alpha+\gamma)[\alpha(V(\tilde{z}) + 1)] \\ +\gamma[\tilde{z}F_2^*(L_2(\tilde{z})) + W_2(\tilde{z})]\{(\alpha+\gamma)H^*(\alpha+\gamma) + \alpha(1 - H^*(\alpha+\gamma)) \\ [\tilde{z} - V(\tilde{z}) + 1]\}\} + L_1(\tilde{z})\left[L_r(\tilde{z}) + \eta\left(1 - G_2^*\left(L_r(\tilde{z})\right)\right)\right] \\ [1 - F_2^*(L_2(\tilde{z}))]\frac{\gamma\tilde{z}^M}{(\alpha+\gamma)}\{(1 - H^*(\alpha+\gamma)) \\ +(\alpha+\gamma)\{\tilde{z}(\alpha+\gamma) - [F_1^*(L_1(\tilde{z})) + W_1(\tilde{z})]((\alpha+\gamma)H^*(\alpha+\gamma) + \alpha\tilde{z} \\ (1 - H^*(\alpha+\gamma))) - \gamma(1 - H^*(\alpha+\gamma))[\tilde{z}F_2^*(L_2(\tilde{z})) + W_2(\tilde{z})]\}\}, \end{gathered}$$

$$\begin{gathered} De_s(\tilde{z}) = L_1(\tilde{z})L_2(\tilde{z})L_r(\tilde{z})\{\tilde{z}(\alpha+\gamma) - [F_1^*(L_1(\tilde{z})) + W_1(\tilde{z})] \\ \{(\alpha+\gamma)H^*(\alpha+\gamma) \ + \alpha\tilde{z}(1 - H^*(\alpha+\gamma))\} \\ -\gamma(1 - H^*(\alpha+\gamma))[\tilde{z}F_2^*(L_2(\tilde{z})) + W_2(\tilde{z})]\}. \end{gathered}$$

$$K_0(\tilde{z}) = \Phi_{0,M}\{\frac{Ne_0(\tilde{z})}{De_s(\tilde{z})}\} \tag{45}$$

$$\begin{gathered} Ne_0(\tilde{z}) = L_1(\tilde{z})L_2(\tilde{z})L_r(\tilde{z})\{De(\tilde{z})[1 + \frac{\alpha\tilde{z}^M\left(1 - F_v^*\left(L_v(\tilde{z})\right)\right)}{L_v(\tilde{z})}] \\ +\tilde{z}^{M+1}[1 - H^*(\alpha + \gamma)]\{[\alpha(V(\tilde{z}) + 1)][F_1^*(L_1(\tilde{z})) + W_1(\tilde{z})] \\ +\gamma[\tilde{z}F_2^*(L_2(\tilde{z})) + W_2(\tilde{z})]\}\} \\ +L_2(\tilde{z})\left[L_r(\tilde{z}) + \eta\left(1 - G_1^*\left(L_r(\tilde{z})\right)\right)\right][1 - F_1^*\left(L_1(\tilde{z})\right)] \\ \tilde{z}^M\{\tilde{z}(\alpha+\gamma)[\alpha(V(\tilde{z}) + 1)] + \gamma[\tilde{z}F_2^*(L_2(\tilde{z})) + W_2(\tilde{z})] \\ \{(\alpha+\gamma)H^*(\alpha+\gamma) + \alpha(1 - H^*(\alpha+\gamma))\ [\tilde{z} - V(\tilde{z}) + 1]\}\} \\ +L_1(\tilde{z})\left[L_r(\tilde{z}) + \eta\left(1 - G_2^*\left(L_r(\tilde{z})\right)\right)\right][1 - F_2^*\left(L_2(\tilde{z})\right)] \\ \frac{\gamma\tilde{z}^{M-1}}{(\alpha+\gamma)}\{(1 - H^*(\alpha+\gamma)) + (\alpha+\gamma)\{\tilde{z}(\alpha+\gamma) \\ - [F_1^*(L_1(\tilde{z})) + W_1(\tilde{z})]((\alpha+\gamma)H^*(\alpha+\gamma) \\ +\alpha\tilde{z}(1 - H^*(\alpha+\gamma))) - \gamma(1 - H^*(\alpha+\gamma))[\tilde{z}F_2^*(L_2(\tilde{z})) + W_2(\tilde{z})]\}\}, \end{gathered}$$

where, $\Phi_{0,M}$ is denoted by Equation (43).

Proof
The probability genering function of the total number of clients in the system and the orbit ($K_s(\tilde{z})$) and

$(K_0(\tilde{z}))$ is calculated by utilizing

$$K_s(z) = \Phi_{0,M} + \Phi(\tilde{z}) + \Upsilon_v(\tilde{z}) + \tilde{z}\{\Omega_1(\tilde{z}) + \Omega_2(\tilde{z}) + \Psi_1(\tilde{z}) + \Psi_2(\tilde{z})\}, \text{ and}$$

$$K_0(z) = \Phi_{0,M} + \Phi(\tilde{z}) + \Upsilon_v(\tilde{z}) + \Omega_1(\tilde{z}) + \Omega_2(\tilde{z}) + \Psi_1(\tilde{z}) + \Psi_2(\tilde{z}).$$

The Equations (37)-(42) may be added to the findings from earlier to get equations (44) and (45).

5. System Performance Measures

This section contains the system estimated average full service periods for various system states, as well as some important system probability and system efficiency measurements.

5.1 System State Probabilities

The preceding findings are obtained by solving the previous systems of Equations (37)-(42) with $\tilde{z} \to 1$, while employing l'Hospital's rule wherever applicable.

(i) Let Φ be the steady state probability that the server is idle during the retrial time as follows,

$$\Phi = \Phi(1) = \Phi_{0,M}(1 - H^*(\alpha+\gamma)) \times \left\{ \frac{\begin{matrix}(M+1)\{1+\alpha(2-F_v^*(\omega))\}+[\alpha(2-F_v^*(\omega))]\left[\frac{\alpha}{\beta}(1+\eta E(G_1))(1-F_1^*(\beta))\right] \\ +\frac{\alpha\gamma}{\beta}(1+\eta E(G_2))(1-F_2^*(\beta))+\alpha[E(F_v)+\frac{\alpha}{\omega}(1-F_v^*(\omega))]+\gamma\end{matrix}}{(\alpha+\gamma)-\Theta} \right\} \tag{46}$$

(ii) Let Ω_1 be the steady state probability that the server is busy with transit customer as follows,

$$\Omega_1 = \Omega_1(1) =$$

$$\Phi_{0,M}E(F_1)\text{x} \left\{ \frac{\begin{matrix}\left[\alpha E(F_v)+\frac{\alpha}{\omega}(1-F_v^*(\omega))\right]\left(\alpha\gamma(1-H^*(\alpha+\gamma))-(\alpha+\gamma)\right)-(\alpha+\gamma)(M+1)[\alpha(2-F_v^*(\omega))]-\alpha\gamma\left(1-H^*((\alpha+\gamma))\right) \\ -\gamma[(\alpha+\gamma)H^*((\alpha+\gamma))-\alpha(F_v^*(\omega)-1)(1-H^*(\alpha+\gamma))]\{M+F_2^*(\beta)+\frac{\alpha}{\beta}(1+\eta E(G_2))(1-F_2^*(\beta))\}\end{matrix}}{(\alpha+\gamma)-\Theta} \right\} \tag{47}$$

(iii) Let Ω_2 be the steady state probability that the server is busy with recurrent customer is given by,

$$\Omega_2 = \Omega_2(1) = \gamma E(F_2)\Phi_{0,M} \times$$

$$\left\{ \frac{\begin{matrix}[(\alpha+\gamma)H^*(\alpha+\gamma)+\alpha(1-H^*(\alpha+\gamma))][\frac{\alpha}{\beta}(1+\eta E(G_1))(1-F_1^*(\beta))]-(1-H^*(\alpha+\gamma))\{\frac{M-1}{\alpha+\gamma}+2\alpha(M-1)-\alpha-\gamma[F_2^*(\beta) \\ +\frac{\alpha}{\beta}(1+\eta E(G_2))(1-F_2^*(\beta))]\}-(\alpha+\gamma)\end{matrix}}{(\alpha+\gamma)-\Theta} \right\} \tag{48}$$

(iv) Suppose Υ_v is the steady state probability that the server is on Bernoulli working vacation is given by,

$$\Upsilon_v = \Upsilon_v(1) = \frac{\alpha\Phi_{0,M}}{\omega}[1 - F_v^*(\omega)] \tag{49}$$

(v) Suppose Ψ_1 is steady-state probability that the server is repair during transit customer service is given by,

$$\Psi_1 = \Psi_1(1) = \Phi_{0,M}\eta E(F_1)E(G_1) \times$$

$$\left\{\frac{\begin{array}{c}(\alpha+\gamma)(M+1)\left[\alpha\left(2-F_v^*(\omega)\right)\right]+\alpha\gamma\left(1-H^*\left((\alpha+\gamma)\right)\right)-\left[\alpha E(F_v)+\frac{\alpha}{\omega}(1-F_v^*(\omega))\right]\left(\alpha\gamma\left(1-H^*(\alpha+\gamma)\right)-(\alpha+\gamma)\right)\\ +\gamma\left[(\alpha+\gamma)H^*\left((\alpha+\gamma)\right)-\alpha(F_v^*(\omega)-1)\left(1-H^*(\alpha+\gamma)\right)\right]\\ \{M+F_2^*(\beta)+\frac{\alpha}{\beta}(1+\eta E(G_2))(1-F_2^*(\beta))\}\end{array}}{(\alpha+\gamma)-\Theta}\right\} \tag{50}$$

(vi) Suppose Ψ_2 is steady state probability that the server is repair with recurrent customer service is given by,

$$\Psi_2 = \Psi_2(1) = \Phi_{0,M}\eta E(F_2)E(G_2)$$

$$\times\left\{\frac{\begin{array}{c}(1-H^*(\alpha+\gamma))\{\frac{M-1}{\alpha+\gamma}+2\alpha(M-1)-\alpha-\gamma[F_2^*(\beta)+\frac{\alpha}{\beta}(1+\eta E(G_2))(1-F_2^*(\beta))]\}-[(\alpha+\gamma)H^*(\alpha+\gamma)\\ +\alpha(1-H^*(\alpha+\gamma))][\frac{\alpha}{\beta}(1+\eta E(G_1))(1-F_1^*(\beta))]+(\alpha+\gamma)]\end{array}}{(\alpha+\gamma)-\Theta}\right\} \tag{51}$$

5.2 Mean Queue Length

When the system satisfied steady state condition,

(i) By differentiating $\tilde{z}$, (45) and providing $\tilde{z} = 1$ results in the average number of customers in the orbit (L_q)

$$L_q = K_0'(1) = \lim_{\tilde{z}\to 1}\frac{d}{d\tilde{z}}K_0(\tilde{z}) = \Phi_{0,M}\left[\frac{Ne_q'''(1)De_q''(1)-De_q'''(1)Ne_q''(1)}{3(De_q''(1))^2}\right] \tag{52}$$

(ii) The average number of customers in the system (L_s) is computed by differentiating (44) with respect to $\tilde{z}$, and further by assigning $\tilde{z} = 1$

$$L_s = K_s'(1) = \lim_{\tilde{z}\to 1}\frac{d}{d\tilde{z}}K_s(\tilde{z}) = \Phi_{0,M}\left[\frac{Ne_s'''(1)De_q''(1)-De_q'''(1)Ne_q''(1)}{3(De_q''(1))^2}\right] \tag{53}$$

(iii) Little's law yields the mean duration of a consumer in the queue (W_q) and the mean duration of a consumer in the system (W_s),

(i.e.,) $W_s = \frac{L_s}{\alpha}$ and $W_q = \frac{L_q}{\alpha}$.

Further, Appendix A comprises all of the values stated above.

6. Numerical Application

In this study, MATLAB has been utilized to illustrate the precise impact of various parameters on the tangible characteristics of the system. The analysis has been conducted for exponentially distributed retry times, delayed repairs, and slow service times. To ensure compliance with stability requirements, the numerical measurements have been selected arbitrarily.

Table 1 shows that the retrial rate $h(x)$ and $\Phi_{0,M}$ increases, L_q, $\Phi(1)$, $\Omega_1(1)$ and Υ_v are diminishes. **Table 2** shows that lower service rate (ω), $\Phi_{0,M}$, L_q, $\Omega_1(1)$ and $\Psi_2(1)$ are escalates Υ_v diminishes. **Table 3** shows that repair rate in recurrent state $\chi_2(x)$, and L_s increases, $\Phi_{0,M}$, L_s, $\Omega_1(1)$ and W_s decreases.

Table 1. $\Phi_{0,M}$ and L_q for distinct retrial rate $h(x)$ in regard to the values of $M = 5$, $\alpha = 0.5$, $\gamma = 0.1$, $\omega = 0.7$, $\vartheta = 0.5$, $\eta = 1$, and $\beta = 2$.

Retrial rate $h(x)$	$\Phi_{0,M}$	L_q	$\Phi(1)$	$\Omega_1(1)$	Υ_v
3.1	0.0412	3.7718	1.0769	1.5899	0.0029
3.2	0.0421	3.5054	1.0688	1.5114	0.0030
3.3	0.0428	3.2763	1.0614	1.4408	0.0031
3.4	0.0434	3.0771	1.0548	1.3769	0.0032
3.5	0.0440	2.9026	1.0487	1.3188	0.0033
3.6	0.0446	2.7484	1.0432	1.2658	0.0034
3.7	0.0451	2.6112	1.0382	1.2172	0.0035

Table 2. $\Phi_{0,M}$ and L_s for different lower service rate (ω) for the values of $M = 5$, $\alpha = 0.5$, $\gamma = 0.1$, $\vartheta = 0.5$, $\eta = 1$, and $\beta = 2$.

Lower service rate (ω)	$\Phi_{0,M}$	L_q	$\Omega_1(1)$	Υ_v	$\Psi_2(1)$
1.1	0.0385	0.2454	1.2900	0.0087	0.8293
1.2	0.0386	0.2464	1.2903	0.0080	0.8322
1.3	0.0387	0.2473	1.2910	0.0075	0.8349
1.4	0.0388	0.2482	1.2919	0.0069	0.8375
1.5	0.0390	0.2490	1.2931	0.0065	0.8399
1.6	0.0391	0.2497	1.2944	0.0061	0.8423
1.7	0.0392	0.2505	1.2958	0.0057	0.8446

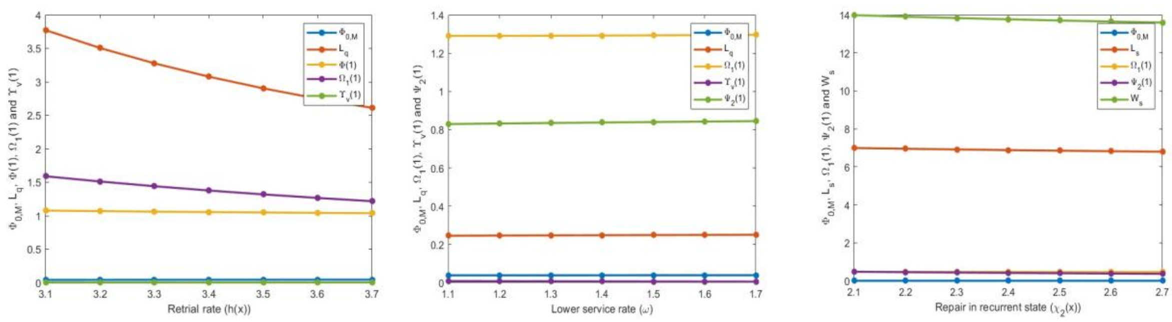
Table 3. $\Phi_{0,M}$ and L_q for different Repair rate in recurrent state $\chi_2(x)$ for the ranges of $M = 5$, $\alpha = 0.5$, $\gamma = 0.1$, $\omega = 0.7$, $\vartheta = 0.5$, $\eta = 1$, and $\beta = 2$.

Repair rate in recurrent state $\chi_2(x)$	$\Phi_{0,M}$	L_s	$\Omega_1(1)$	$\Psi_2(1)$	W_s
2.1	0.0013	6.9925	0.4694	0.4788	13.9851
2.2	0.0012	6.9527	0.4683	0.4560	13.9055
2.3	0.0011	6.9163	0.4674	0.4353	13.8326
2.4	0.0010	6.8829	0.4666	0.4165	13.7658
2.5	0.0009	6.8521	0.4658	0.3991	13.7043
2.6	0.0008	6.8237	0.4657	0.3832	13.6474
2.7	0.0007	6.7974	0.4644	0.3685	13.5948

As a result of the parameters $M = 5$, $\alpha = 0.5$, $\gamma = 0.1$, $\omega = 0.7$, $\vartheta = 0.5$, $\eta = 1$, and $\beta = 2$, **Figure 2** shows the 2-D graph that explored the performance measures. In **Figure 2(a)**, shows the increase the retrial rate $h(x)$ and $\Phi_{0,M}$ escalates, L_q, $\Phi(1)$, $\Omega_1(1)$ and Υ_v are diminishes. In **Figure 2(b)**, it was found that lower service rate (ω), $\Phi_{0,M}$, L_q, $\Omega_1(1)$ and $\Psi_2(1)$ are escalates Υ_v diminishes. In **Figure 2(c)**, it was found that repair rate in recurrent state $\chi_2(x)$, and L_s increases, $\Phi_{0,M}$, L_s, $\Omega_1(1)$ and W_s decreases.

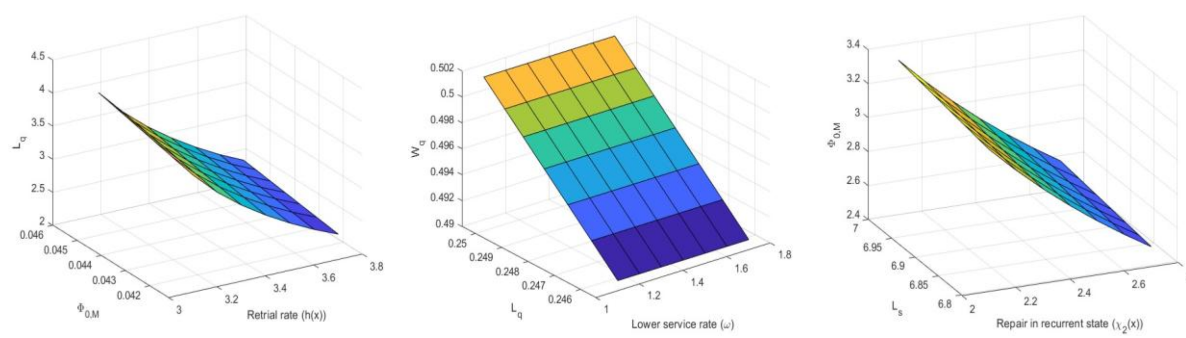
Figure 3 represent the 3-D graph that displays the system effectiveness. In **Figure 3(a)**, the surface displays the escalation of the retrial rate $h(x)$ and $\Phi_{0,M}$, (L_q) diminishes. In **Figure 3(b)**, (L_q), (W_q) elevates as the lower service rate (ω) rises. In **Figure 3(c)**, it can be observed that $\Phi_{0,M}$ and L_s decreases while increasing the repair rate in recurrent state $\chi_2(x)$.

The aforementioned numerical data is often relied upon by researchers to determine the impacts of behaviours on the system's evaluation procedures with assurance that their results are indicative of real-world settings.



(a) Φ_0, M, L_q, Φ (1), $\Omega_1(1)$, $\Upsilon_1(1)$, verses retrial rate $h(x)$ (b) Φ_0, M, L_q, $\Omega_1(1)$, $\Upsilon_1(1)$, $\Psi_2(1)$, verses lower service rate ω (c) Φ_0, M, L_s, $\Omega_1(1)$, $\Psi_2(1)$, W_s verses repair rate in recurrent $\chi_2(x)$

Figure 2. Impact of system performance measures on $h(x)$, ω, $\chi_2(x)$ in a $2D$ visualization.



(a) $\Phi_{0,M}$, L_q verses retrial rate $h(x)$ (b) L_q, W_q verses lower service rate ω (c) L_s, $\Phi_{0,M}$, verses repair rate $\chi_2(x)$

Figure 3. Impact of system performance measures on $h(x)$, ω, $\chi_2(x)$ in a $3D$ visualization.

7. Cost Optimization

"Optimization" refers to the act of selecting the combination of inputs to an objective function that yields the maximum or minimum result. Cost optimization (CO) is the method of constantly concentrating on a company's developments in the aim of saving expenditures and charges, when enhancing the company's value. It entails standardizing, streamlining, and rationalizing platforms, applications, procedures, and services, as well as obtaining the most competitive price and terms on all business transactions. The link between the system's profit and operating expenditures is fairly close in a condition that more closely mimics real life. As a result, system administrators' major task is to reduce the sum of funds required for operations every hour to maximize the profit. Our major goal here is to identify the factors that compute the good average cost per unit of time (CUT). To accomplish this aim, the section of the model that was constructed is made more cost-effective by adding a cost function.

Now, a CO method is employed to obtain the optimal values for the elements, which are comprised the normal service rate and recurrent service rate (ζ_1, ζ_2). It is assumed that the various system performance and the difference cost sets related to those activities have a linear connection in the predicted cost function.

Every variable inside the cost factor used to calculate the predicted total cost function (TC) (ζ_1, ζ_2) for per unit time is defined below:

R_h => keeping the costs of every customer in the system for a predetermined amount of time.
R_b => CUT for the server is busy mode.
R_v => Cost per unit time when the server is vacation mode.
R_f => CUT for there pair of the server after breakdown.
R_1 => CUT during busy period.
R_2 => CUT during recurrent mode.

Cost function predictions shown as

$$TC(\zeta_1, \zeta_2) = R_h L_q + R_b\{\Omega_1 + \Omega_2\} + R_v \Upsilon_v + R_f\{\Psi_1 + \Psi_2\} + R_1\zeta_1 + R_2\zeta_2 \quad (54)$$

The substantial non-linearity of the cost function indicated in (54) makes it difficult to optimize by an analytical method. As a result, the heuristic procedure is used to optimise the overall cost, which is intended to depend on the busy and recurrent service rate ζ_1 and ζ_2 respectively.
Finding the best service rate (ζ_1^*)in busy mode and the best service rate (ζ_2^*) in recurrent mode while lowering the overall cost function is our main objective.

$$TC(\zeta_1^*, \zeta_2^*) = \min_{\zeta_1^*, \zeta_2^*} TC(\zeta_1, \zeta_2).$$

Numerous optimization techniques have been evolved via extensive research and development since the early 1960s. Numerous algorithms have proven their ability to handle a wide range of optimization issues. A global optimization strategy should be applied if the objective function has local optima or if there is not enough knowledge to figure out its structure. Nevertheless, if our objective function has a single optima, as in the case of a unimodal function, or if being near the global optima is assured, a local optimization technique should be picked. Implementing a local search algorithm to a problem that needs a global search algorithm will yield insufficient results because local optima would fool the local search. In this work, three global search optimization algorithms are employed: Grey wolf optimizer (GWO), bat algorthim (BA), whale optimization (WO) and cat swarm optimization (CSO).

In one of the three distinct subsections that make up this section, each of these three algorithms is covered in detail. These algorithms are employed because of the crucial role and requirement of CO in mind. As long as the algorithm's presumptions hold true, local search techniques usually reduce the computing cost associated with finding the global best solution.

To provide a graphical depiction of the cost function's sensitivity analysis, the cost elements in accordance with **Table 4** are organized as follows:

Table 4. Cost sets adopted in the proposed cost analysis.

Cost set	R_h	R_b	R_v	R_f	R_1	R_2
Set 1	10	40	15	25	20	15
Set 2	5	35	10	20	15	10
Set 3	15	30	5	10	10	5

7.1 Grey Wolf Optimizer (GWO)

The Grey Wolf Optimizer (GWO), introduced by Mirjalili et al. (2014), is a swarm intelligence algorithm inspired by the social hierarchy and hunting behavior of grey wolves in the wild. GWO's simplicity, minimal parameter requirements, and effective balance between exploration and exploitation make it a popular choice for various optimization problems. In this study, GWO has been used to determine the optimal service rate pairs and minimize total costs. Its performance has been analyzed using various parameters, as shown in **Table 5**.

7.2 Bat Algorithm (BA)

The Bat Algorithm (BA), proposed by Yang (2010), is inspired by the echolocation behavior of bats. By employing frequency tuning and automatic zooming, BA maintains a balance between exploration and exploitation, enabling effective searches for optimal solutions. Its applications span diverse domains, including scheduling and feature selection. In this work, BA has been utilized to optimize service rates and costs, with results summarized in **Table 6**.

7.3 Whale Optimization (WO)

The Whale Optimization Algorithm (WO), developed by Mirjalili and Lewis (2016), simulates the bubble-net hunting strategy of humpback whales. It effectively balances global and local search mechanisms, making it suitable for addressing constrained and unconstrained optimization problems. This study applies WO to optimize service rates and total costs, with findings detailed in **Table 7**.

7.4 Cat Swarm Optimization (CSO)

Cat Swarm Optimization (CSO), introduced by Chu and Tsai (2007), models the seeking and tracing behaviors of cats. This dual-phase approach ensures efficient exploration of the solution space. CSO has been applied in this study to optimize service rates and costs, with results presented in **Table 8**.

By focusing on the key contributions of these algorithms, the methodology section has been streamlined without compromising essential details. The pseudocodes and detailed algorithmic parameters are included in Appendix B and Appendix C for reference.

7.5 Comparison of Optimization Algorithm

Here, it is assessed which of the fore cost-finding algorithms, GWO, BA, WO and CSO -produces the best results when implemented in MATLAB. Four different cost sets and three different pairings of optimal parameters $(\alpha, \omega, \vartheta)$ are taken into consideration in **Table 4**. Next, the MATLAB code for each of the previously stated techniques repeatedly is run. As a result, this process has been carried out once again and produced **Tables 5** through **8**. It was found that there was a remarkable degree of similarity in the outcomes from all four courses. Consequently, for these four strategies, the optimal solutions and related expenses are similar near to another one. This demonstrates that the previously discussed meta-heuristics offer trustworthy (local) optimum solutions.

A lot iterations needed by CSO is far less than that needed by other approaches, as shown in **Tables 5-8**. Any method can help us find the optimal cost, but when different approaches are evaluateed for our model, CSO turns out to be the most effective one. Out of all of these tactics, it was found that the CSO strategy is the greatest one because it provides a lot of advantages to its consumers. It works well in global searches, is simple to configure, needs some elements, and is not impacted by the scale of different elements. CSO often results in an early and quick convergence in center optimal sites with little search capacity. On the other hand, in an area where the search has been improved, CSO typically results in a progressive convergence.

7.6 Convergence of Optimization Algorithm

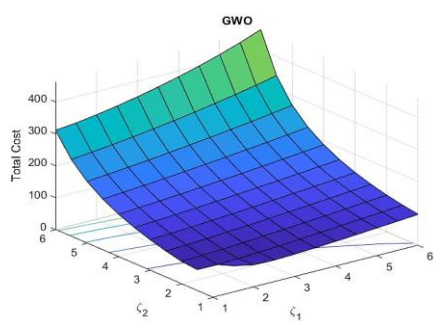
The elements are not in a steady circumstance at the start of the optimization process when any of the following techniques is used: BA, CSO, GWO and WO Determining whether the elements come back to its usual condition and it will keep looking for a better solution is crucial. Convergence is therefore a crucial element of cost analysis. The convexity and optimality of the cost function with respect to the three cost sets considered in the optimization study are shown in **Figure 4**. The optimization techniques BA, CSO, GWO and WO were used to create this figure. The elements are converge to the optimal cost in CSO the fastest, with less convergence in BA, CSO, GWO and WO as seen in **Figure 5**. During the same period of time, this happens.

One may draw the following conclusions from the previously mentioned convergent nature of various optimization techniques:

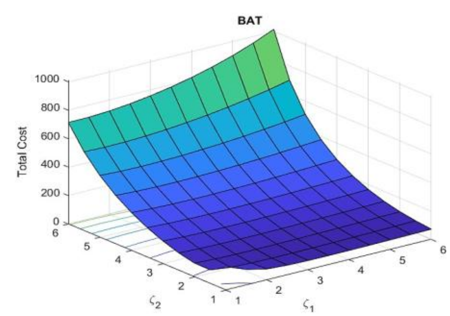
(i) The total cost of parameters and their optimal values determined by BA, CSO, GWO and WO is the same.

(ii) The research findings indicate that the model that was provided is in line with the real-world scenarios. Some of their financial worries will be partially resolved by the cost optimization, which takes this system up to its whole cost.

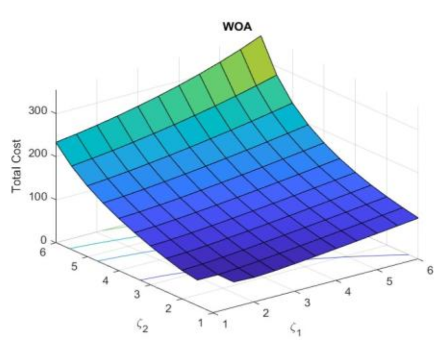
(iii) Under the current conditions, the generated cost-benefit analysis can be largely trusted to show the reasoning behind our model and help network managers and experts lessen the complex of the problem that stopping presents to particular communication services.



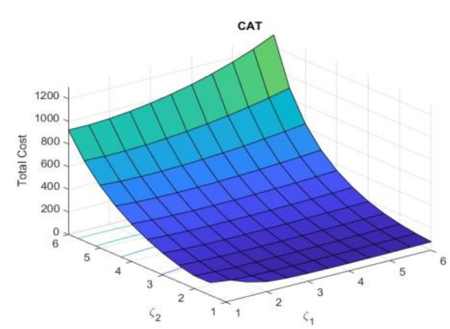
(a) TC vs ζ_1, ζ_2 using GWO



(b) TC vs ζ_1, ζ_2 using BA

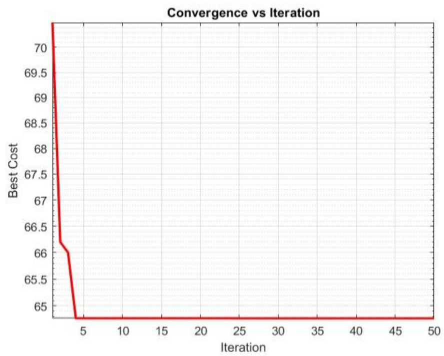


(c) TC vs ζ_1, ζ_2 using WO

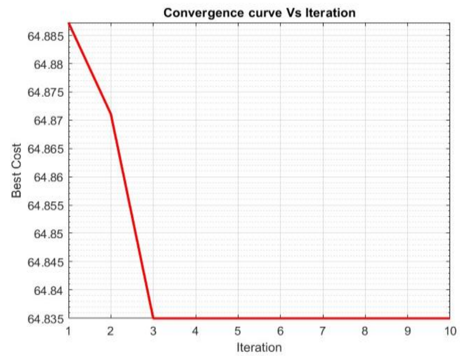


(d) TC vs ζ_1, ζ_2 using GA

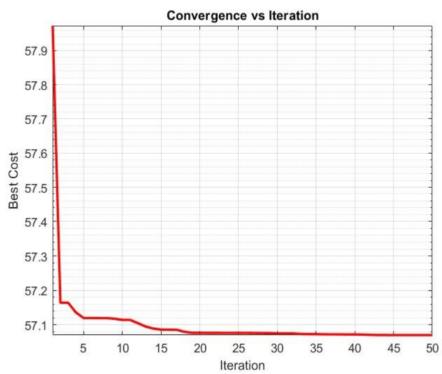
Figure 4. Optimality of the cost function.



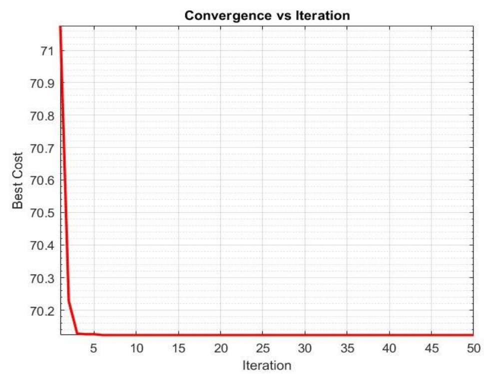
(a) Convergence vs Iteration using GWO



(b) Convergence vs Iteration using BA



(c) Convergence vs iteration using WO



(d) Convergence vs iteration using CSO

Figure 5. Convergence of the cost function.

7.7 Results and Discussion

This study effectively modeled and analyzed the performance of a single-server retrial queue system featuring recurrent customers and a standby server under Bernoulli working vacation conditions. The founding highlights crucial insights into the behavior of such systems, emphasizing strategies for minimizing operational costs and optimizing performance.

Below are the key findings derived from the analysis:

7.7.1 Innovative Approach to Recurrent Customer Modeling

The introduction of the supplementary variable technique (SVT) to analyze the recurrent customer vacation process marks a significant advancement in queueing theory. By employing SVT, the probability generating functions (PGFs) and steady-state probabilities for various system states were successfully derived. This novel approach not only provides a robust mathematical framework for understanding customer behavior but also enhances the analytical capabilities in evaluating complex queueing systems. The derived PGFs offer valuable insights into the expected number of customers in various states, facilitating better decision-making in operational management.

7.7.2 Impact of Working Vacation on System Performance

The analysis revealed that the Bernoulli working vacation policy substantially influences the system's performance metrics. During working vacations, the server can continue providing service at a reduced rate, which mitigates the impact of customer arrivals during non-standard operating conditions. This dual-functionality of the server significantly reduces waiting times and enhances customer throughput compared to traditional vacation models where service is entirely halted. The graphical representation of these metrics demonstrates how effective management of working vacations can lead to improved service levels and customer satisfaction.

7.7.3 Optimization of Operational Costs

A major highlight of the study is the development and optimization of a cost function, which quantitatively assesses operational expenses associated with the retrial queue system. By employing advanced optimization algorithms-namely, the Grey Wolf Optimizer (GWO), Bat Algorithm (BA), Whale Optimization (WO), and Cat Swarm Optimization (CSO), it was demonstrated that significant cost reductions could be achieved while maintaining optimal system performance. The comparative analysis of these algorithms illustrated their effectiveness in reaching lower cost configurations, showcasing their utility in real-world applications. The visualizations provided in the study clearly depict the convergence rates and effectiveness of each algorithm, highlighting their strengths and potential areas for further refinement.

7.7.4 Numerical Analysis and Validation of Performance Measures

Comprehensive numerical analyses and graphical presentations confirmed the validity of the derived performance measures. Key metrics such as the mean number of customers in the system, average waiting times, and system utilization were meticulously calculated and presented. These findings not only corroborate the theoretical underpinnings of the model but also serve as a basis for practical applications in various service environments. The results indicated that even slight adjustments to system parameters could lead to significant variations in performance outcomes, underscoring the necessity of continual monitoring and adjustment of queue management strategies.

7.7.5 Trade-offs between Service Rate and Customer Satisfaction

The results also highlighted the trade-offs between service rates during working vacations and overall customer satisfaction. While increasing service rates can lead to enhanced throughput, it is essential to consider the associated costs and potential impacts on server performance. The optimization models developed in this study provide a framework for balancing these factors, enabling managers to make informed decisions regarding service strategies that align with their operational goals.

7.7.6 Future Research Directions

The insights gained from this study pave the way for future research avenues, particularly in extending the model to incorporate more complex customer behaviors, such as heterogeneous service times or varying arrival rates. Investigating the effects of additional operational factors, such as customer priority levels or alternative service configurations, could yield valuable insights into improving queue management practices. Furthermore, applying machine learning techniques to dynamically adjust service strategies based on real-time data could enhance system responsiveness and efficiency.

8. Conclusion

In this study, the evaluation, oversight, and efficiency of an M/G/1 retrial G-queue with a standby server under Bernoulli working vacation were thoroughly analyzed. By employing the supplementary variable technique, the steady-state equations were developed that describe the size of both the system and the orbit, providing key performance indicators such as the mean system length and mean orbit length. Through extensive numerical analysis, the effects of various system parameters on performance, highlighting their influence on system behavior and identifying the optimal configurations to achieve efficient system control were explored.

The cost minimization aspect of the study was tackled by integrating four advanced metaheuristic algorithms - Grey Wolf Optimizer (GWO), Bat Algorithm (BA), Whale Optimization (WO), and Cat Swarm Optimization (CSO). These algorithms were employed to determine the most advantageous cost configuration for the system. Our analysis revealed that the Grey Wolf Optimizer (GWO) consistently provided the most effective cost reductions compared to the other strategies, demonstrating its superiority in this context. Additionally, the Bat Algorithm (BA) and Whale Optimization (WO) also produced competitive results, showing their viability for cost optimization in queueing models. These results offer significant insights for decision-makers in operations management, particularly when aiming to enhance system performance while minimizing costs.

The findings from this study can be utilized to make informed decisions for managing and improving retrial queue systems with vacation policies. The introduction of metaheuristic techniques for cost optimization in queueing theory marks a notable advancement, especially in terms of real-world applications that involve system downtime and service interruptions.

Lastly, the authors propose that future research could extend this work by investigating bulk service queueing systems with recurrent customers over hybrid vacation policies. Additionally, an exploration of the transient solution for retrial queues with hybrid vacations using the supplementary variable technique could offer valuable insights into short-term system behavior, complementing the steady-state results presented in this study. These avenues for future research could further enhance the applicability of retrial queue models in practical scenarios.

Conflict of Interest

The authors confirm that there is no conflict of interest to declare for this publication.

AI Disclosure

The author(s) declare that no assistance is taken from generative AI to write this article.

Acknowledgments

The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve the quality of this paper.

Appendices

Appendix A

$$Ne_q''(1) = \{2L_2(\beta)(\alpha(1+\eta E(G_1)))(\alpha+\vartheta) + 2L_1(\beta)(\alpha(1+\eta E(G_2)))(\alpha+\gamma)$$
$$+\vartheta L_1(\beta)L_2(\beta)\}\{1+\frac{\alpha}{\omega}(1-F_v^*(\omega)) + (1-H^*(\alpha+\gamma))$$
$$(\alpha(2-F_v^*(\omega))+\gamma)\}$$
$$-2L_1(\beta)L_2(\beta)(\alpha+\vartheta)\{((\alpha+\gamma)-\Theta)(1+\alpha(1-F_v^*(\omega))) + \alpha M(1-F_v^*(\omega))$$
$$+E(F_v) + \frac{\alpha}{\omega}(1-F_v^*(\omega))\} + (1-H^*(\alpha+\gamma))\{(M+1)(1+\alpha(2-F_v^*(\omega)))$$
$$+\left[\alpha\left(2-F_v^*(\omega)\right)\right]\left[\frac{\alpha}{\beta}\left(1+\eta E(G_1)\right)\left(1-F_1^*(\beta)\right)\right]$$
$$+\frac{\alpha\gamma}{\beta}(1+\eta E(G_2))(1-F_2^*(\beta))$$
$$+\alpha[E(F_v)+\frac{\alpha}{\omega}(1-F_v^*(\omega))] + \gamma\} + 2\vartheta L_1(\beta)L_2(\beta)\{1+\frac{\alpha}{\omega}(1-F_v^*(\omega))$$
$$+(1-H^*(\alpha+\gamma)) + ((\alpha(2-F_v^*(\omega))+\gamma))\} + \{2L_2(\beta)(1-F_1^*(\beta))(\alpha+\vartheta)$$
$$(\eta E(G_1^{(2)})) + 2\alpha[1+\eta E(G_2)](1-F_1^*(\beta))(\alpha+\vartheta)(1+\eta E(G_1)) + 2\alpha L_2(\beta)$$
$$(\alpha+\vartheta)(1+\eta E(G_1))^2 E(F_1) - 2ML_2(\beta)(1-F_1^*(\beta))(\alpha+\vartheta)(1+\eta E(G_1))\}$$
$$\{(\alpha+\gamma)(\alpha)(2-F_v^*(\omega)) + \gamma\{(\alpha+\gamma)H^*(\alpha+\gamma) + \alpha(1-H^*(\alpha+\gamma))$$
$$(F_v^*(\omega)-1)\}\} - 2L_2(\beta)(1-F_1^*(\beta))((\alpha+\vartheta)(1+\eta E(G_1)))\{(\alpha+\gamma)$$
$$[\alpha(2-F_v^*(\omega)) + \alpha E(F_v) - \frac{\alpha}{\omega}(1-F_v^*(\omega))] + \gamma[(F_2^*(\beta)+\frac{\alpha}{\beta}(1+\eta E(F_2))$$
$$(1-F_2^*(\beta)))((\alpha+\gamma)H^*(\alpha+\gamma) + \alpha(1-H^*(\alpha+\gamma))(F_v^*(\omega)-1))]$$
$$+\alpha\gamma(1-H^*(\alpha+\gamma))[1-\alpha E(F_v) - \frac{\alpha}{\gamma}(1-F_v^*(\omega))]\} + \gamma\{-2(M-1)L_1(\beta)$$
$$(1-F_2^*(\beta))(\alpha+\vartheta)(1+\eta E(G_2)) + 2(\alpha(1+\eta E(G_1)))(1-F_2^*(\beta))((\alpha+\gamma)$$
$$(1+\eta E(G_2))) + 2L_1(\beta)(\alpha(1+\eta E(G_2))E(F_2))(\alpha+\vartheta)(1+\eta E(G_2))$$
$$+L_1(\beta)(1-F_2^*(\beta))((\alpha+\vartheta)(\eta E(G_2^{(2)})))\}\{(\frac{(1-H^*(\alpha+\gamma))}{\alpha+\gamma})(\gamma(\alpha+\gamma-1))\}$$
$$-2L_1(\beta)(1-F_2^*(\beta))(\alpha+\vartheta)(1+\eta E(G_2))\{(\alpha+\gamma) - \alpha(1-H^*(\alpha+\gamma))$$
$$+(\alpha+\gamma H^*(\alpha+\gamma))[\frac{\alpha}{\beta}(1+\eta E(G_1))(1-F_1^*(\beta))] - \gamma(1-H^*(\alpha+\gamma))$$
$$[F_2^*(\beta) + \frac{\alpha}{\beta}(1+\eta E(G_2))(1-F_2^*(\beta))]\}.$$

$$De_q''(1) = -2\beta^2(\alpha+\gamma)[(\alpha+\gamma)-\Theta].$$
$$Ne_q'''(1) = \{6\alpha(1+\eta E(G_1))L_2(\beta)(\alpha+\vartheta) + 6L_1(\beta)\alpha(1+\eta E(G_2))(\alpha+\vartheta)$$
$$+2L_1(\beta)L_2(\beta)\vartheta\}\{((\alpha+\gamma)-\Theta)(1+\alpha(1-F_v^*(\omega))) + \alpha M(1-F_v^*(\omega))$$
$$+E(F_v) + \frac{\alpha}{\omega}(1-F_v^*(\omega))\} + (1-H^*(\alpha+\gamma))\{(M+1)(1+\alpha(2-F_v^*(\omega)))$$
$$+\left[\alpha\left(2-F_v^*(\omega)\right)\right]\left[\frac{\alpha}{\beta}\left(1+\eta E(G_1)\right)\left(1-F_1^*(\beta)\right)\right]$$
$$+\frac{\alpha\gamma}{\beta}(1+\eta E(G_2))(1-F_2^*(\beta))$$

$$+\alpha[E(F_v)+\frac{\alpha}{\omega}(1-F_v^*(\omega))]+\gamma\}-\{3L_1(\beta)\Gamma_4(\alpha+\gamma)+3\vartheta\alpha(1+\eta E(G_1))$$
$$L_2(\beta)+3\Gamma_3L_2(\beta)(\alpha+\vartheta)+6\alpha^2\big(1+\eta E(G_1)\big)\big(1+\eta E(G_2)\big)(\alpha+\vartheta)+3\vartheta L_1(\beta)(1+\eta E(G_2))\}\{1$$
$$+\frac{\alpha}{\omega}(1-F_v^*(\omega))+(1-H^*(\alpha+\gamma))$$
$$+((\alpha(2-F_v^*(\omega))$$
$$+\gamma))\}-2L_1(\beta)L_2(\beta)(\alpha+\vartheta)\{\{(1+\alpha(1-F_v^*(\omega)))\{-2\alpha(1-H^*(\alpha+\gamma))$$
$$[-\alpha(1+\eta E(G_1))E(F_1)+\frac{\alpha}{\beta}(1+\eta E(G_1))(\beta E(F_1)-F_1^*(\beta)+1)]$$
$$-(\alpha+\gamma H^*(\alpha+\gamma))(\Gamma_1+W_1'')-\gamma(1-H^*(\alpha+\gamma))-2\alpha(1+\eta E(G_2))E(F_2)$$
$$+\Gamma_2+W_2''(1)\}+2(1+\alpha(1-F_v^*(\omega)))((\alpha+\gamma)-\Theta)+\alpha M^2(1-F_v^*(\omega))$$
$$+M(\frac{\alpha}{\omega}(\omega E(F_v)-F_v^*(\omega)+1))+V''(1)\}+(1-H^*(\alpha+\gamma))$$
$$\{M(M+1)(1+\alpha(2-F_v^*(\omega)))+2(M+1)\{(1+\alpha(2-F_v^*(\omega)))$$
$$+\Big(\alpha\big(2-F_v^*(\omega)\big)\Big)\left(\frac{\alpha}{\beta}\big(1+\eta E(G_1)\big)\big(1-F_1^*(\beta)\big)\right)$$
$$+\frac{\gamma\alpha}{\beta}(1+\eta E(G_2))(1-F_2^*(\beta))$$
$$+\alpha(E(F_v)+\frac{\alpha}{\omega}(1-F_v^*(\omega)))+\gamma\}+\alpha V''(1)+2\alpha(\frac{\alpha}{\omega}(\omega E(F_v)-F_v^*(\omega)+1))$$
$$(\frac{\alpha}{\beta}(1+\eta E(G_1))(1-F_1^*(\beta)))+\alpha(2-F_v^*(\omega))(\Gamma_1+W_1''(1))$$
$$+\frac{2\gamma\alpha}{\beta}(1+\eta E(G_2))(1-F_2^*(\beta))+(\Gamma_2+W_2''(1))\}\}+\{6M(\alpha(1+\eta E(G_2)))$$
$$(1-F^*(\beta))(\alpha+\vartheta)(1+\eta E(G_1))+6ML_2(\beta)(\alpha(1+\eta E(G_2)))$$
$$(\alpha+\vartheta)\Big(\big(1+\eta E(G_1)\big)\Big)+6\Big(\alpha\big(1+\eta E(G_2)\big)\Big)\big(1-F_1^*(\beta)\big)$$
$$(\alpha+\vartheta)(1+\eta E(G_1))$$
$$+3M(M-1)L_2(\beta)\big(1-F_1^*(\beta)\big)(\alpha+\vartheta)\big(1+\eta E(G_1)\big)$$
$$+3ML_2(\beta)(1-F_1^*(\beta))$$
$$(\alpha+\vartheta)\left(\eta E\left(G_1^{(2)}\right)\right)-3\Gamma_4\big(1-F_1^*(\beta)\big)(\alpha+\vartheta)\big(1+\eta E(G_1)\big)$$
$$+3\alpha(1+\eta E(G_1))$$
$$(1-F_1^*(\beta))(\alpha+\vartheta)(\eta E(G_1^{(2)}))-3L_2(\beta)\Gamma_1(\alpha+\vartheta)(1+\eta E(G_1))+3L_2(\beta)$$
$$(1-F_1^*(\beta))(\alpha+\vartheta)(\eta E(G_1^{(2)}))-L_2(\beta)(1-F_1^*(\beta))(\alpha+\vartheta)(\eta E(G_1^{(3)}))\}$$
$$\{(\alpha+\gamma)(\alpha)(2-F_v^*(\omega))+\gamma\{(\alpha+\gamma)H^*(\alpha+\gamma)+\alpha(1-H^*(\alpha+\gamma))$$
$$(F_v^*(\omega)-1)\}\}+\{6(\alpha)(1+\eta E(G_2))$$
$$(1-F_1^*(\beta))(\alpha+\vartheta)(1+\eta E(G_1))-6ML_2(\beta)(1-F_1^*(\beta))(\alpha+\vartheta)$$
$$+6L_2(\beta)(\alpha)(1+\eta E(G_1))E(F_1)(\alpha+\vartheta)(1+\eta E(G_1))+2L_2(\beta)(1-F_1^*(\beta))$$
$$(\alpha+\vartheta)(\eta E(G_1^{(2)}))\}\{(\alpha+\gamma)[\alpha(2-F_v^*(\omega))+\alpha E(F_v)-\frac{\alpha}{\omega}(1-F_v^*(\omega))]$$
$$+\gamma[(F_2^*(\beta)+\frac{\alpha}{\beta}(1+\eta E(F_2))(1-F_2^*(\beta)))((\alpha+\gamma)H^*(\alpha+\gamma)$$
$$+\alpha(1-H^*(\alpha+\gamma))(F_v^*(\omega)-1))]+\alpha\gamma(1-H^*(\alpha+\gamma))[1-\alpha E(F_v)$$
$$-\frac{\alpha}{\gamma}(1-F_v^*(\omega))]\}-3L_2(\beta)(1-F_1^*(\beta))(\alpha+\vartheta)(1+\eta E(G_1))$$
$$\{(\alpha+\gamma)(\frac{2\alpha}{\omega}(\omega E(F_v)-F_v^*(\omega)+1)+\alpha V''(1))+2\gamma(F_2^*(\beta)+\frac{\alpha}{\beta}(1+\eta E(G_2))$$

$$
\begin{gathered}
(1-F_2^*(\beta)))+\alpha(1-H^*(\alpha+\gamma))(1-\alpha E(F_v)-\frac{\alpha}{\omega}(1-F_v^*(\omega)))\\
+\gamma(-2\alpha(1+\eta E(G_2))E(F_2)+\Gamma_2+W_2''(1))((\alpha+\gamma)H^*(\alpha+\gamma)\\
+\alpha(1-H^*(\alpha+\gamma))(F_v^*(\omega)-1))-\alpha\gamma(1-H^*(\alpha+\gamma))V''(1)\}\\
+\gamma\{6(M-1)\\
(\alpha(1+\eta E(G_1)))(1-F_2^*(\beta))((\alpha+\vartheta)(1+\eta E(G_2)))+6(M-1)L_1(\beta)\\
(\alpha(1+\eta E(G_2))E(F_2))((\alpha+\vartheta)(1+\eta E(G_2)))-6((\alpha)(1+\eta E(G_1)))\\
((\alpha)(1+\eta E(G_2))E(F_2))((\alpha+\vartheta)(1+\eta E(G_2)))-3(M-1)(M-2)L_1(\beta)\\
(1-F_2^*(\beta))((\alpha+\vartheta)(1+\eta E(G_2)))+3(M-1)L_1(\beta)(1-F_2^*(\beta))\\
((\alpha+\vartheta)(\eta E(G_2^{(2)})))-3\Gamma_3(1-F_2^*(\beta))((\alpha+\vartheta)\\
(1+\eta E(G_2)))-3\alpha(1+\eta E(G_1))(1-F_2^*(\beta))((\alpha+\vartheta)(\eta E(G_2^{(2)})))\\
-3L_1(\beta)\Gamma_2(\alpha+\vartheta)(1+\eta E(G_2))+3L_1(\beta)(1-F_2^*(\beta))(\alpha+\vartheta)\\
(\eta E(G_2^{(2)}))-L_1(\beta)(1-F_2^*(\beta))(\alpha+\vartheta)(\eta E(G_2^{(3)}))\}\\
\{(\frac{(1-H^*(\alpha+\gamma))}{\alpha+\gamma})(\gamma(\alpha+\gamma-1))\}+\gamma\{6(\alpha)(1+\eta E(G_1))\\
(1-F_2^*(\beta))(\alpha+\vartheta)(1+\eta E(G_2))-6(M-1)L_1(\beta)(1-F_2^*(\beta))\\
(\alpha+\vartheta)(1+\eta E(G_2))+6L_1(\beta)(\alpha)(1+\eta E(G_1))E(F_2)(\alpha+\gamma)(1+\eta E(G_2))\\
+3L_1(\beta)(1-F_2^*(\beta))(\alpha+\gamma)(\eta E(G_2^{(2)}))\}\{(\alpha+\gamma)-\alpha(1-H^*(\alpha+\gamma))\\
+(\alpha+\gamma H^*(\alpha+\gamma))[\frac{\alpha}{\beta}(1+\eta E(G_1))(1-F_1^*(\beta))]\\
-\gamma(1-H^*(\alpha+\gamma))[F_2^*(\beta)+\frac{\alpha}{\beta}(1+\eta E(G_2))(1-F_2^*(\beta))]\}-3\gamma L_1(\beta)\\
(1-F_2^*(\beta))(\alpha+\gamma)(1+\eta E(G_2))\{(\alpha+\gamma)-[(\alpha+\gamma)H^*(\alpha+\gamma)\\
+\alpha(1-H^*(\alpha+\gamma))](\Gamma_1+W_1''(1))-2(1-H^*(\alpha+\gamma))\{\frac{\alpha}{\beta}(1+\eta E(G_1))\\
(1-F_1^*(\beta))\}-\gamma(1-H^*(\alpha+\gamma))[-2\alpha(1+\eta E(G_2))E(F_2)+\Gamma_2+W_2''(1)]\}.
\end{gathered}
$$

$$
\begin{gathered}
De_q'''(1)=6\beta(\alpha+\vartheta)[\alpha(1+\eta E(G_1))][(\alpha+\gamma)-\Theta]+6\beta(\alpha+\vartheta)[\alpha(1+\\
\eta E(G_2))][(\alpha+\gamma)-\Theta]+3\vartheta\beta^2[(\alpha+\gamma)-\Theta]+(3\beta^2(\alpha+\\
\vartheta))\{(\alpha+\quad\gamma H^*(\alpha+\gamma))[\Gamma_1+W_1''(1)]+2(\frac{\alpha}{\beta}(1+\eta E(G_1)(1-F_1^*(\beta)))\\
[\alpha(1-H^*(\alpha+\quad\gamma))]+\gamma(1-H^*(\alpha+\gamma))\{2\alpha(1+\eta E(G_2))\\
E(F_2)+\Gamma_2+W_2''(1)\}+1\}.
\end{gathered}
$$

where,

$$W_1''(1)=\frac{1}{\beta^2}\{\beta^3\Gamma_1-\beta^2(1-F_1^*(\beta))\Gamma_3-2\beta^2(1+\eta E(G_1))E(F_1)-2\alpha^2(1+\eta E(G_1))(1-F_1^*(\beta))\},$$

$$W_2''(1)=\frac{1}{\beta^2}\{\beta^3\Gamma_1-\beta^2(1-F_2^*(\beta))\Gamma_3-2\beta^2(1+\eta E(G_2))E(F_2)-2\alpha^2(1+\eta E(G_2))(1-F_2^*(\beta))\},$$

$$V''(1)=\frac{1}{\omega^2}\{\alpha\omega^3E(F_v)-2\omega\alpha E(F_v)-2\alpha(1-F^*(\omega))\},$$

$$\Gamma_1=\alpha^2[(1+\eta E(G_1))^2E(F_1^{(2)})+\eta E(G_1^{(2)})E(F_1)],$$

$$\Gamma_2=\alpha^2[(1+\eta E(G_2))^2E(F_2^{(2)})+\eta E(G_2^{(2)})E(F_2)],$$

$$\Gamma_3=\alpha(1+\eta E(G_1))+\alpha^2\eta E(G_1^{(2)}),$$

$$\Gamma_4 = \alpha(1 + \eta E(G_2)) + \alpha^2 \eta E(G_2^{(2)}),$$
$$Ne_s'''(1) = Nr_q'''(1) + \{6(\alpha(1 + \eta E(G_1)))(1 - F_2^*(\beta))(\alpha + \vartheta)(1 + \eta E(G_2))$$
$$-6(M - 1)L_1(\beta)(1 - F_2^*(\beta))(\alpha + \vartheta)(1 + \eta E(G_2)) + 6L_1(\beta)(\alpha(1 + \eta E(G_2))$$
$$E(F_2))(\alpha + \vartheta)(1 + \eta E(G_2)) + 3L_1(\beta)(1 - F_2^*(\beta))(\alpha + \vartheta)(\eta E(G_2^{(2)}))\}$$
$$\{(\frac{(1 - H^*(\alpha + \gamma))}{\alpha + \gamma})(\gamma(\alpha + \gamma - 1))\} - [6L_1(\beta)(1 - F_2^*(\beta))(\alpha + \vartheta)$$
$$(1 + \eta E(G_2))]\{(\alpha + \gamma) - \alpha(1 - H^*(\alpha + \gamma)) + (\alpha + \gamma H^*(\alpha + \gamma)$$
$$)[\frac{\alpha}{\beta}(1 + \eta E(G_1))(1 - F_1^*(\beta))] - \gamma(1 - H^*(\alpha + \gamma))[F_2^*(\beta)$$
$$+\frac{\alpha}{\beta}(1 + \eta E(G_2))(1 - F_2^*(\beta))]\}$$

Appendix B

Algorithm 1 Pseudo Code for GWO Algorithm

INPUT: Objective function $=TC(\zeta_1,\zeta_2),a,B,D$
OUTPUT: The cost function's value $TC(\zeta_1^*, \zeta_2^*)$
Initiate the grey wolf population Y_i, $i = 1,2,...,n$
Initiate a, B, D
Estimate the fitness of every search factor
Y_α = the best search factor
Y_β = the second-best search factor Y_γ = the third best search factor while t < maximum number of iterations do for every search factor do
    Arbitrarily initiate c_1 and c_2
    Update the place of the present search factor
    Update a, B and D
    Estimate the fitness of every search factors
    Update Y_α, Y_β and Y_γ $t = t + 1$
  end for
end while return Y_α

Algorithm 2 Pseudo Code for BA Algorithm

INPUT: Objective function = $TC(\zeta_1, \zeta_2)$, number of trials, frequencies, velocities
OUTPUT: The cost function's value $TC(\zeta_1^*, \zeta_2^*)$
Set the bat group y_i and v_i ($i = 1, 2,..., n$)
Set frequencies f_i, pulse rates r_i and the loudness B_i while
t < Max number of trials do
  Create new responses by altering frequencies
    Improve velocities and locations/responses if rand > r_i then
      Choose a response among the best responses
      Create a local response around the chosen best response end if
    Create a new response by flying randomly if rand< B_i & $f(y_i) < f(Y^*)$ then
      Accept the new response

Boost r_i and decrease B_i end if
Rank the bats and find the current best X^* end while

Algorithm 3 Pseudo Code for WO Algorithm

INPUT: Objective function = $TC(\zeta_1,\zeta_2)$, Number of maxiter and Population etc.

OUTPUT: The cost function's value $TC(\zeta_1^*, \zeta_2^*)$
Initialize a, A, C, l and p
Determine the fitness of every searching agent X^* = the best search agent while (it <Maxiter) do for every search agent do if ($p < 0.5$) then if ($|A| < 1$) then
Adjust the current search agent's coordinates using (1) else
($|A| \geq 1$)
Choose an arbitrary search agent (X_{rand})
Adjust the present search agent's coordinates using (3) end if
($p \geq 0.5$)
Adjust the present search agent's coordinates using (2) end if
end for
Find the fitness of every search agent
Update X^* if a better way can be found to do this it=it+1
Update a, A, C, l and p end while return X^*

Algorithm 4 Pseudo Code of CSO Algorithm

INPUT: Objective function = $TC(\zeta_1,\zeta_2)$, Number of maxiter and Population etc
OUTPUT: The cost function's value $TC(\zeta_1^*, \zeta_2^*)$
Create an initial population of cats $Y_i(i = 1,2,...,n)$, v and self-position consideration (SPC) while (Failure to meet the termination criteria or $I < I_{max}$) do
Sort all the cats by their calculated fitness function values Y_g = cat with the best answer for $i = 1: N$ do
if (SPC = 1) then
Switch to the search mode
else
Start tracing mode end if
end for i end while Results analysis and representation

Appendix C

Table 5. Effect of $\alpha, \omega, \vartheta$ on $(TC^*, \zeta_1^*, \zeta_2^*)$ using GWO.

Parameters		$(TC^*, \zeta_1^*, \zeta_2^*)$		
		Cost set 1	Cost set 2	Cost set 3
α	0.7	(81.3174,1.1022,1.7952)	(57.3218,1.0039,1.8769)	(57.3633,1.0223,2.6693)
	00.8	(95.6608,1.2396,2.1464)	(68.6276,1.0521,2.6324)	(83.0626,1.7339,1.2723)
	00.9	(108.2312,1.1378,3.0516)	(71.4557,1.1901,2.4962)	(78.9805,1.1153,4.8485)
ω	4.1	(81.8595,1.9675,2.0721)	(57.2833,1.0113,1.8459)	(57.4192,1.0320,2.6441)
	4.2	(81.1499,1.0959,1.8201)	(57.4122,1.2294,1.4465)	(59.4770,1.2546,2.0392)
	4.3	(80.9845,1.0922,1.8287)	(56.9892,1.1240,1.6043)	(58.1516,1.1563,2.2758)
ϑ	0.1	(81.3999,1.1602,1.6745)	(57.7852,1.2833,1.3722)	(59.9257,1.2831,2.0013)
	0.2	(86.1982,1.0699,2.1004)	(62.4646,1.5268,1.1278)	(63.8065,1.5440,1.5310)
	0.3	(90.7340,1.0774,2.1609)	(62.4744,1.1693,1.7665)	(66.0212,1.0847,2.8799)

Table 6. Effect of $\alpha, \omega, \vartheta$ on $(TC^*, \zeta_1^*, \zeta_2^*)$ using BA.

Parameters		$(TC^*, \zeta_1^*, \zeta_2^*)$		
		Cost set 1	Cost set 2	Cost set 3
α	0.7	(82.9629,1.3890,1.3239)	(57.1935,1.0856,1.6896)	(57.3633,1.0223,2.6693)
	0.8	(99.0555,1.4104,2.6855)	(64.8841,1.3306,1.6417)	(73.5202,1.4242,2.3087)
	0.9	(109.0222,1.1300,3.1492)	(73.1651,1.1671,2.6794)	(79.7424,1.3290,3.3216)
ω	4.1	(87.6229,1.7966,2.9427)	(57.3645,1.9923,1.9027)	(60.1713,1.2917,1.9589)
	4.2	(93.6000,1.7523,3.5375)	(63.8529,1.7341,3.1435)	(59.3421,1.2523,2.0709)
	4.3	(81.1391,1.1947,1.6397)	(57.1487,1.1857,1.5009)	(58.8434,1.2164,2.1218)
ϑ	0.1	(82.8683,1.3796,1.3353)	(57.4241,1.1965,1.4898)	(59.1324,1.2205,2.1499)
	0.2	(86.5766,1.0039,2.2274)	(60.0518,1.2348,1.5634)	(63.3058,1.2106,2.3318)
	0.3	(90.8545,1.2398,1.7950)	(64.8528,1.4899,1.2232)	(66.4365,1.1105,2.7538)

Table 7. Effect of $\alpha, \omega, \vartheta$ on $(TC^*, \zeta_1^*, \zeta_2^*)$ using WO.

Parameters		$(TC^*, \zeta_1^*, \zeta_2^*)$		
		Cost set 1	Cost set 2	Cost set 3
α	0.7	(81.3410,1.0769,1.8496)	(57.2291,1.1310,1.6220)	(57.1163,1.0152,2.9237)
	0.8	(94.3177,1.4026,1.6436)	(64.8996,1.4791,1.2850)	(71.1598,1.1808,3.4498)
	0.9	(100.8677,1.4042,1.8665)	(68.2073,1.4388,1.5497)	(78.0862,1.2137,3.9676)
ω	4.1	(81.3216,1.0942,1.8202)	(57.2469,1.1419,1.6040)	(57.4192,1.0320,2.6441)
	4.2	(82.5587,1.3662,1.3557)	(57.0634,1.0733,1.7189)	(56.9322,1.1732,2.9190)
	4.3	(80.9913,1.0806,1.8392)	(56.9892,1.1240,1.6043)	(56.7504,1.2011,2.9145)
ϑ	0.1	(81.6981,1.2444,1.5576)	(57.1937,1.0695,1.7309)	(57.1163,1.0731,2.9236)
	0.2	(86.2642,1.2479,1.6657)	(59.8157,1.1332,1.7351)	(61.0453,1.1566,3.2015)
	0.3	(90.7627,1.2221,1.8282)	(62.4431,1.1144,1.8870)	(65.0118,1.1921,3.4790)

Table 8. Effect of $\alpha, \omega, \vartheta$ on $(TC^*, \zeta_1^*, \zeta_2^*)$ using CSO.

Parameters		$(TC^*, \zeta_1^*, \zeta_2^*)$		
		Cost set 1	Cost set 2	Cost set 3
α	0.7	(81.3152,1.1109,1.7885)	(57.1912,1.0810,1.7086)	(60.7831,1.0219,3.2291)
	0.8	(94.3125,1.4146,1.6129)	(64.7666,1.4300,1.4698)	(77.4248,1.1171,4.1272)
	0.9	(106.2876,1.5931,1.4124)	(71.1618,1.6106,1.1491)	(94.1367,1.2449,4.9402)
ω	4.1	(81.3152,1.1108,1.7886)	(57.1912,1.0813,1.7084)	(60.7831,1.0442,3.2288)
	4.2	(81.1444,1.1088,1.7890)	(57.0628,1.0796,1.7088)	(60.5809,1.1981,3.2241)
	4.3	(80.9753,1.1065,1.7909)	(56.9358,1.0775,1.7095)	(60.3813,1.2731,3.2184)
ϑ	0.1	(81.3152,1.1108,1.7891)	(57.1912,1.0815,1.7076)	(60.7831,1.0761,3.2292)
	0.2	(85.9144,1.130,1.8864)	(59.8066,1.1110,1.7753)	(65.4298,1.1131,3.5518)
	0.3	(90.5667,1.1421,1.9924)	(62.4379,1.1297,1.8556)	(70.1241,1.5310,3.8742)

References

Ahuja, A., Jain, A., & Jain, M. (2022). Transient analysis and ANFIS computing of unreliable single server queueing model with multiple stage service and functioning vacation. *Mathematics and Computers in Simulation*, *192*, 464-490.

Artalejo, J.R., & Corral, A.G. (1999). *Retrial queueing systems, mathematical and computer modelling*. Springer, Berlin, Heidelberg.

Chao, X., & Pinedo, M. (1999). *Queueing networks: customers, signals, and product form solutions*. Wiley, New York.

Chu, S.C., & Tsai, P.W. (2007). Computational intelligence based on the behavior of cats. *International Journal of Innovative Computing, Information and Control*, *3*(1), 163-73.

Deora, P., Kumari, U., & Sharma, D.C. (2021). Cost analysis and optimization of machine repair model with working vacation and feedback–policy. *International Journal of Applied and Computational Mathematics*, *7*, 1-4.

Erlang, A.K. (1909). The theory of probabilities and telephone conversations. *Nyt Tidsskrift for Matematik B*, *20*, 33-39.

Falin, G.I., & Templeton, J.G. (1997). *Retrial queues*. Chapman & Hall, London.

Farahmand, K. (1996). Single line queue with recurrent repeated demands. *Queueing Systems*, *22*, 425-435.

Farahmand, K., & Livingstone, N. (2001). Recurrent retrial queues with service option on arrival. *European Journal of Operational Research*, *131*(3), 530-535.

Gelenbe, E. (1989). Random neural networks with negative and positive signals and product form solution. *Neural Computation*, *1*(4), 502-510.

Gelenbe, E. (1991). Product-form queueing networks with negative and positive customers. *Journal of Applied Probability*, *28*(3), 656-663.

Gupta, P., & Kumar, N. (2021). Cost optimization of single server retrial queueing model with Bernoulli schedule working vacation, vacation interruption and balking. *Journal of Mathematical and Computational Science*, *11*(3), 2508-2523.

Haghighi, A.M., & Mohanty, S.G. (2019). Professor lajos takács: a tribute. In: Andrews, G., Krattenthaler, C., Krinik, A. (eds) *Lattice Path Combinatorics and Applications* (pp. 1-28), Springer International Publishing. ISBN: 978-3-030-11101-4.

Haghighi, Montazer, A., Dimitar, P., & Mishev (2025). A Transient $M^{[X]}$ / $M^{[k, K]}$ /1 Queue with reneging and setup time for service. *International Journal of Mathematics in Operational Research*.

Harrison, P.G., & Marin, A. (2014). Product-forms in multi-way synchronizations. *The Computer Journal*, *57*(11),1693-1710.

Jain, M., & Kumar, A. (2022). 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*, *8*(3), 210-235.

Jain, M., & Kumar, A. (2023). Unreliable server $M^{[X]}$ /G/1 retrial feedback queue with balking, working vacation and vacation interruption. *Proceedings of the National Academy of Sciences, India Section A: Physical Sciences*, *93*(1), 57-73.

Kumar, A., & Jain, M. (2023). Cost optimization of an unreliable server queue with two stage service process under hybrid vacation policy. *Mathematics and Computers in Simulation*, *204*, 259-281.

Laxmi, P.V., & Jyothsna, K. (2022). Cost and revenue analysis of an impatient customer queue with second optional service and working vacations. *Communications in Statistics-Simulation and Computation*, *51*(85), 4799-4814.

Li, T., & Zhang, L. (2017). An M/G/1 retrial G-queue with general retrial times and working breakdowns. *Mathematical and Computational Applications*, *22*(1), 15-27.

Li, T., Zhang, L., & Gao, S. (2019). An *M*/*G*/1 retrial queue with balking customers and Bernoulli working vacation interruption. *Quality Technology & Quantitative Management*, *16*(5), 511-530.

Malik, G., Upadhyaya, S., & Sharma, R. (2021). Particle swarm optimization and maximum entropy results for mx/g/1 retrial g-queue with delayed repair. *International Journal of Mathematical, Engineering and Management Sciences*, *6*(2), 541-559.

Maragathasundari, S., Manimala, M., & Lingisetty, A. (2020). Queuing system on service interruption and standby server. *Network Security*, *8*(2), 8-12.

Meena, S., & Ayyappan, G. (2024). Analysis of MMAP/PH (1), PH (2)/1 preemptive priority queueing model with single vacation, repair and impatient customers. *Applications & Applied Mathematics, 19*(1)*,* 1.

Mirjalili, S., & Lewis, A. (2016) The whale optimization algorithm. *Advances in Engineering Software*, *95*, 51-67.

Mirjalili, S., Mirjalili, S.M., & Lewis, A. (2014). Grey wolf optimizer, *Advances in Engineering Software*. *69*, 46-61.

Moreno, P. (2004). An M/G/1 retrial queue with recurrent customers and general retrial times, *Applied Mathematics and Computation, 159*(3), 651-666.

Pakes, A.G. (1969). Some conditions for ergodicity and recurrence of Markov chains, *Operations Research*, *17*(6), 1058-1061.

Peng, Y., Liu, Z., & Wu, J. (2014). 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*, *44*, 187-213.

Radha, J., Indhira, K., & Chandrasekaran, V.M. (2020). Analysis of M [X]/Gk/1 retrial queue with Bernoulli-2 vacation, search for customers and hot stand-by. *International Journal of Process Management and Benchmarking*, *10*(2), 196-223.

Rajadurai, P. (2018a). A study on an *M*/*G*/1 retrial G-queue with unreliable server under variant working vacations policy and vacation interruption. *Songklanakarin Journal of Science & Technology*, *40*(1), 231-242.

Rajadurai, P. (2018b). Sensitivity analysis of an *M*/*G*/1 retrial queueing system with disaster under working vacations and working breakdowns. *RAIRO-Operations Research*, *52*(1), 35-54.

Rajadurai, P. (2019). A study on *M*/*G*/1 preemptive priority retrial queue with Bernoulli working vacations and vacation interruption. *International Journal of Process Management and Benchmarking*, *9*(2), 193-215.

Rajadurai, P., Sundararaman, M., & Narasimhan D. (2020). Performance analysis of an *M*/*G*/1 retrial G-queue with feedback under working breakdown services. *Songklanakarin Journal of Science & Technology*, *42*(1), 236-247.

Revathi, C. (2022). Search of arrivals of an *M*/*G*/1 retrial queueing system with delayed repair and optional re-service using modified Bernoulli vacation. *Journal of Computational Mathematica*, *6*(1), 200-209.

Saggou, H., Lachemot, T., & Ourbih-Tari, M. (2017). Performance measures of *M*/*G*/1 retrial queues with recurrent customers, breakdowns, and general delays. *Communications in Statistics-Theory and Methods*, *46*(16), 7998-8015.

Sanga, S.S., & Jain, M. (2019). Cost optimization and ANFIS computing for admission control of *M*/*M*/1/*K* queue with general retrial times and discouragement. *Applied Mathematics and Computation*, *363*, 124624.

Sennott, L.I., Humblet, P.A., & Tweedie, R.L. (1983). Mean drifts and the non-ergodicity of Markov chains. *Operations Research*, *31*(4), 783-789.

Takács, L. (1960). Transient behavior of single-server queuing processes with recurrent input and exponentially distributed service times. *Operations Research*, *8*(2), 231-245.

Takács, L. (1962). A single-server queue with Poisson input. *Operations Research*, *10*(3), 388-394.

Takács, L. (1980). Queues with infinitely many servers. *RAIRO-Operations Research*, *14*(2), 109-113.

Vaishnawi, M., Upadhyaya, S., & Kulshrestha, R. (2022). Optimal cost analysis for discrete-time recurrent queue with Bernoulli feedback and emergency vacation. *International Journal of Applied and Computational Mathematics*, *8*(5), 254.

Yang, X.S. (2010). A new metaheuristic bat-inspired algorithm. In: Juan, R., González, D.A.P., Cruz, C., Terrazas, G., & Krasnogor, N. (eds) *Nature inspired cooperative strategies for optimization* (pp. 65-74). Springer, Berlin, Heidelberg.