

Reliability Analysis of a Limited-Capacity Dual-Repair System with Hybrid Vacation Policies

Sapna

Department of Mathematics and Statistics,
Manipal University Jaipur, Jaipur, 303007, Rajasthan, India.
E-mail: sapna.211051025@mujaipur.edu

Anamika Jain

Department of Mathematics,
RD & DJ College, Munger University, Munger, 811201, Bihar, India.
Corresponding author: anamikajain_02@rediffmail.com

Praveen Kumar Agrawal

Department of Applied Sciences & Humanities,
GL Bajaj Group of Institutions, Mathura-Delhi Road, Akbarpur, 281406, Uttar Pradesh, India.
E-mail: prkuag@gmail.com

(Received on June 21, 2025; Revised on July 23, 2025 & August 19, 2025 & September 18, 2025;
Accepted on September 26, 2025)

Abstract

This study investigates the reliability of a machine dual-repair system (MDRS) with limited capacity and multiple operational vacations (OV). Multiple operational vacation is a mechanism that involves the process of the two types of vacation, one as operational vacation (OV) and another one is non-operational vacation (NOV). In this machining system, there are M operating machines, to prevent redundant breakdowns of machines and S spare machines are used to ensure smooth functioning. The server can be in a busy state, an operational vacation state, or on Non-operational vacation state. State-transition equations are formulated based on the Chapman-Kolmogorov differential-difference equations, and with the help of these equations stationary probability distribution is obtained. The Runge-Kutta IV order numerical method is used to evaluate system performance measures. The cost has been optimized using Particle Swarm Optimization (PSO) and Grey Wolf Optimization (GWO) techniques by testing convergence as well as comparison of the outcomes of both PSO and GWO. The exploration of the essential performance measures and graphical representations has been conducted.

Keywords- Reliability, Retrial queues, Multiple operational vacation, Non-operational vacation.

1. Introduction

The multiple operational vacation (MOV) and machine dual-repair system (MDRS) models are used in scenarios where the service station continues to operate at a lower rate rather than shutting down completely due to reduced demand. Multiple operational vacation is a mechanism that processes the two types of vacation, such as server operating, but at a lower speed is acknowledged as operational vacation service. For example, in a CNC machine shop, a machine may go on "operational vacation" when the primary job queue is empty. During this time, the operator might perform routine diagnostics or lubricate the machine. Secondary non-operational vacation is a period when the server is completely idle or unavailable, doing no productive work related to the system. For example, a robotic arm in an assembly line goes on non-operational vacation when, power failure occurs, maintenance staff takes it offline for unexpected repairs. These models are widely applied in various fields such as cloud computing, production houses, hospitals, public transportation and maintenance, and telecommunications. Queueing theory enhances machine repair models by analysing and optimizing the flow of repair tasks, ensuring efficient resource allocation and

minimizing downtime. For instance, queueing theory can help establish the ideal quantity of spare parts to keep in stock or optimize the scheduling of maintenance personnel to minimize idle time and ensure timely repairs. This results in a balanced workload and prevents delays in machine repairs, ultimately improving overall productivity. Furthermore, queueing theory supports better decision-making by providing insights into the performance and reliability of the repair system under different conditions. It can simulate various scenarios, such as an increase in machine breakdowns or changes in repair times, enabling organizations to plan effectively for potential disruptions. Additionally, the literature on this topic explores aspects such as reliability, retrial queues, and multiple operational vacations. The reliability of any system plays an important role in the manufacturing industry. Yen et al. (2020), and Gao and Wang (2021) explore a reliability-based retrial model of the machine repair problem with working breakdowns under the F-policy. The author's study in this paper is about the reliability and sensitivity analysis of the retrial machine repair problem with F-policy, which is used to control the arrival process. Muthusamy et al. (2022) and Kumar et al. (2023) considered the reliability and optimization measures of retrial queues with different classes of customers under a working vacation. It is explained that if no failed units in the orbit, the server may take a working vacation. Ahuja and Jain (2023) investigated the fuzzy analysis of a queueing system with unreliable service using a direct search approach and geometric arrivals, incorporating a constant retrial policy and a delayed threshold policy. Li et al. (2024) investigate system reliability and optimization with preventive maintenance. Analysis of multi-objective optimization shows it is the best retrial system functioning with the help of warm standbys, even though the system has different types of failures. This research was done by Wu and Wang (2025).

The multiple operational working vacation model is used in scenarios where the service station continues operating at a lower rate rather than shutting down completely. Working vacations play an important role in repairing systems by improving energy efficiency and operational effectiveness. Meena et al. (2019) analysed a machine repair model incorporating vacation policies and standby provisioning mechanisms. Chakravarthy et al. (2020) obtained a queueing system that incorporates breakdowns, vacations, and a backup server during repairs. Deora et al. (2021) and Thakur et al. (2021) conducted a cost analysis and optimized the cost of a machine repair system with a working vacation, feedback, and particle swarm optimization. Bouchentouf et al. (2022) studied a multi-station unreliable machine system with a working vacation policy and customer impatience behaviour. Gao et al. (2023) derived the steady-state distribution and reliability of a redundant series system, where each primary unit is supported by a standby, under the supervision of a single repairman who takes a delayed vacation when no failed units are present. Sapna and Jain (2024) investigated the reliability of a machine-repair system characterized by multiple breakdowns and controllable phase-type repair mechanisms. Agrawal et al. (2023) analysed the reliability of a repairable system with multiple spare units under an "N-Policy" framework under the supervision of a single repairman. Bouchentouf et al. (2024) proposed a finite capacity redundant multi-server machine repairable system by using the matrix analytic method. The steady-state probabilities were computed to evaluate key performance measures. The cost function was optimized by combining the direct search method with particle swarm optimization. Kumar and Sharma (2025) analyze a queueing model of a manufacturing plant system with a server working vacation in association with an optimal policy. Anitha et al. (2025), and Zhu and Wang (2025) considered the vacation policy in a Markovian model and showed the results with different parameters.

The PSO optimization technique is stimulated by the social behavior done by the flock of birds and the school of fish while they travel to their perfect location. Jain and Raychaudhuri (2022) studied the Markovian model in which the cost is optimized by a Genetic Algorithm with the working vacation and multiple breakdowns. Chahal et al. (2024) explain the grey wolf optimization for a cloud computing repairable system with a threshold policy, along with discouragement and two-level Bernoulli feedback.

Sharma et al. (2025), Thakur and Jain (2025) analyze the cost effects for a machine repair system under the triadic policy with discouragement and multiple working vacations using metaheuristic optimization. **Table 1** presents a comparison between the proposed work and relevant studies.

Table 1. Comparison of the proposed work with relevant articles.

Authors	Keywords						
	Reliability	Retrial	MOV	Dual repair	Limited capacity	PSO	GWO
Wu and Wang (2025)	√	√	×	√	×	×	×
Chahal et al. (2024)	×	×	×	√	√	√	√
Li et al. (2024)	√	×	×	×	×	×	×
Kumar et al. (2023)	√	×	√	×	×	×	×
Deora et al. (2021)	×	×	√	√	√	√	×
Proposed model	√	√	√	√	√	√	√

This paper examines retrial machining systems with hybrid vacations, emphasizing the integration of maintainability strategies to enhance system reliability and availability. The structure of the paper is as follows: Section 2 outlines key definitions and assumptions, while Section 3 explores the model's application. Section 4 presents the governing equations, followed by Section 5, which deduces performance measures of the model. Section 6 discussed the cost function and optimization techniques. Section 7 analyses the impact of various parameters on sensitivity analysis. Section 8 provides the study's results, and finally, Section 9 discusses the conclusions of the work.

2. Definitions and Key Assumptions

The present study considers a retrial machine dual-repair system (RMDRS) in which once a unit fails, it is replaced by an available standby, and the failed unit is sent to a service facility for service. In this system, after providing service in a busy state, the server enters operational vacation OV_1 during the idle period. After completing the first operational vacation, if the system detects any failed units in the busy state, the server resumes service; otherwise, it proceeds to the next operational vacation, OV_2 . This process continues sequentially until OV_J (where $J=1, 2, 3, \dots, K-1$). After the $(K-1)^{th}$ operational vacation, if the server finds a failed unit in the busy state, it serves; otherwise, it proceeds to the next operational vacation OV_K . If, after completing the K^{th} operational vacation, the server does not find any failed machines in the busy state, it transits to a non-operational vacation (NOV). The failed machines will get the services according to the first-come, first -served (FCFS) pattern. The other notations and abbreviations are defined in **Table 2**.

Table 2. Notation and abbreviations used in the model.

Notations	Abbreviation
OV	Operational vacation
NOV	Non-Operational vacation
L	M+S, M is the number of operational machines, S is are number of standbys
λ	Operating rate of operational units
τ	Operating rate of standby units
θ_0	Retrial rate of failed units
λ_n	Rate of failed units in different states
μ_B	Service rate during the busy state
μ_{vi}	Service rate during OV_i ($i = 1, 2, 3, \dots, K$)
η	Switching rate from NOV to busy state
α	The breakdown rate of busy state, $OV_1, OV_2 \dots OV_K$ states
β	The repair rate of busy, $OV_1, OV_2 \dots OV_K$ states
$X_{n,0}$	The probability of 'n' failed units in the 1 st repair state
$X_{n,1}$	The probability of 'n' failed units in the 2 nd repair state
$X_{n,2}$	The probability of the retrial state
$X_{n,3}$	The probability that the server during a busy state
$X_{n,i}$	The probability of server during OV_i , $i = 4, 5, \dots, K+3$ state
$X_{n,K+4}$	The probability that the server during the NOV state

Arrival and retrial pattern: The arrival of failed units to the system follows a Poisson process. The failure rate, denoted by λ_n defined as follows

$$\lambda_n = \begin{cases} M\lambda + (S - n)\tau & ; n = 0, 1, 2, \dots (S - 1), S \\ (L - n)\lambda & ; n = S + 1, S + 2, \dots M + S \\ 0 & \text{otherwise} \end{cases}$$

The failed units in the orbit retry for the service follow the exponential distribution with retrial rate θ_0 .

Services mechanism: Failed machines are serviced according to a first-come, first-served (FCFS) discipline. During the server's busy state, the service time for failed units follows an exponential distribution with rate μ_B . When the server enters an operational vacation, the service times remain exponentially distributed but with reduced service rates. Specifically, during the i^{th} operational vacation state OV_i (for $i=1, 2, \dots, K$), the service rate is μ_{vi} , where $\mu_{v1}, \mu_{v2}, \mu_{v3}, \dots, \mu_{vK}$ correspond to the respective operational vacation phases. Let $L=M+S$, where L represents the total number of machines in the system, M denotes the number of machines currently in operation, and S refers to the number of available spare machines. When any of the operating machines fail or go down, the system immediately checks for the availability of a spare machine. If a spare is available, it is replaced in the failed machine. The replacement process is assumed to be instantaneous, and the spare machine begins operating at the same performance rate as the original machine, thereby maintaining the system's operational capacity without delay.

Hybrid vacations pattern: If the server becomes idle after completing a service, it has two options. First, it can remain in the busy state with probability p , or switch to the first operational vacation state OV_1 with probability $\bar{p} = (1 - p)$. While in OV_1 , the server checks for failed units. If a failed unit is found, it begins servicing at a reduced rate μ_{v1} with probability b . If no failed unit is found, the server may transition to the next operational vacation state OV_2 , servicing at rate μ_{v2} , with probability $\bar{b} = (1 - b)$. This process continues through successive operational vacation states OV_i , for $3 \leq i \leq K+3$, with the same probability, each with its own service rate μ_{vi} . If no failed units are present during any of these operational vacation states and the server remains idle, it transitions to a non-operational vacation state, where it remains for an exponentially distributed time with rate η , as per the system policy explained in **Figure 1**.

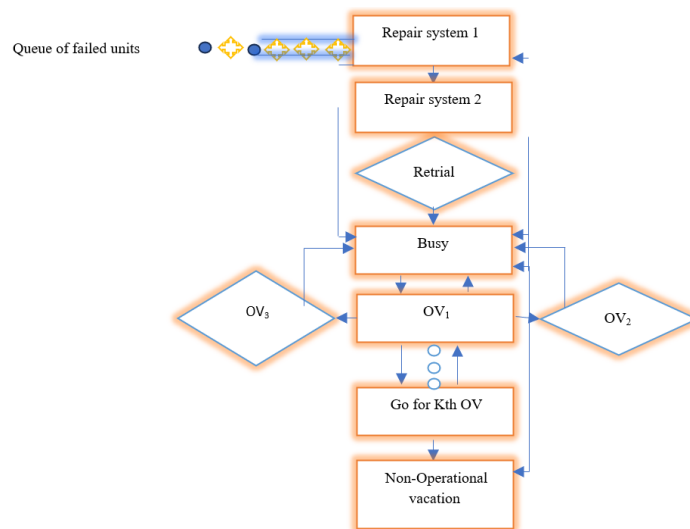


Figure 1. The flow chart of proposed model.

Breakdown pattern: Breakdown of the server is possible in a busy state or in the operational vacation state, and the server immediately sends to the repair facility for repair. The breakdown and the repair of the server follow the exponential distribution with rates α and β , respectively. There are two repair states in the model. After a breakdown, the server enters either the first repair state or the second repair state with probabilities $(1-r)$ and r , respectively, depending on the condition of the server (minor or major breakdown). After repair, the server returns from the first or second repair state with probabilities r' and $(1-r')$, respectively.

Steady state: Failure time of both the operating and standby units is supposed to be exponentially distributed with the rate λ and τ ($0 < \tau < \lambda$), respectively. If any operational unit failed, instantly replaced by the standbys if it is available. It is supposed that the switch of units is always perfect and the switchover time is always swift. When a standby unit becomes an operating unit, then its failure conditions will remain the same as the operational unit. For a machine repair system, α is the failure rate of the server, and β is the repair rate of the server. In the retrial state, ϑ_0 is the retrial rate accomplished with μ_B service rate and switching from retrial to busy state at rate λ . After repair of the failed unit, the server goes to the next state, opting for the first operational vacation. In between if the server finds any failed unit, it goes to the repair state. But if the server didn't find any failed unit in the system, it goes for the second operational vacation and continuously for the OV_{K+3} . Otherwise goes for the Non-operational vacation at state $K+4$. $L=M+S$, L is the total number of operating units.

$$(n, j) = \begin{cases} 0, & \text{the MRS under 1st repair (R)} \\ 1, & \text{the MRS under 2nd repair (R)} \\ 2, & \text{the retrial state of the MRS (RT)} \\ 3, & \text{is the busy state of the server} \\ 4, & \text{is the first operational vacation} \\ \dots & \dots \\ K+3, & \text{as the MRS at its (K+3)th operational vacation} \\ K+4, & \text{as the Non-operational vacation state of MRS} \end{cases},$$

where, $n = 0, 1, 2, \dots, M+S$, $j = 0, 1, 2, \dots, K+4$.

3. Application of the Model

The investigated model enhances public transportation systems by optimizing vehicle maintenance and scheduling to minimize service disruptions. Buses, trains, and metros are the service providers; these vehicles are busy in operation or providing services during the peak hours. But if the peak-hours are covered, then these servers can proceed with the operational vacation (OV). These vehicles operate at lower efficiency during off-peak times but still serve customers at a reduced rate as an operating vacation. During the maintenance period, all transport vehicles like buses, trains, and metros remain completely closed, hence it can be considered as a non-operational vacation (NOV). All the transport vehicles need regular attention for routine maintenance, and when a vehicle enters the queue, it must wait if all repair stations are occupied. Limited maintenance facilities and customer impatience make it crucial to balance repairs with operational efficiency. In modern transportation systems, vehicles such as buses, trains, and metro units can be successfully modeled as servers in a queueing or machine repair system. These vehicles rotate between being in service as a busy state, undergoing maintenance as a repair facility, or operating in reduced capacity as an operational working vacation. Such types of models are not only to assist in analyzing system performance metrics like reliability, availability, and breakdown time, but also enable cost optimization planning. This methodology finds direct application in industrial transport systems, urban mobility, and intelligent arrangement systems for large-scale transit networks.

During low-demand periods, some maintenance activities proceed at a slower rate under a hybrid vacation policy, while repair stations may close entirely when all vehicles are operational. By leveraging queueing theory and hybrid vacation policies, our model efficiently allocates maintenance resources, reduces downtime, and improves service reliability, ensuring smoother transit experiences for commuters. Model-based application shown in **Figure 2**.

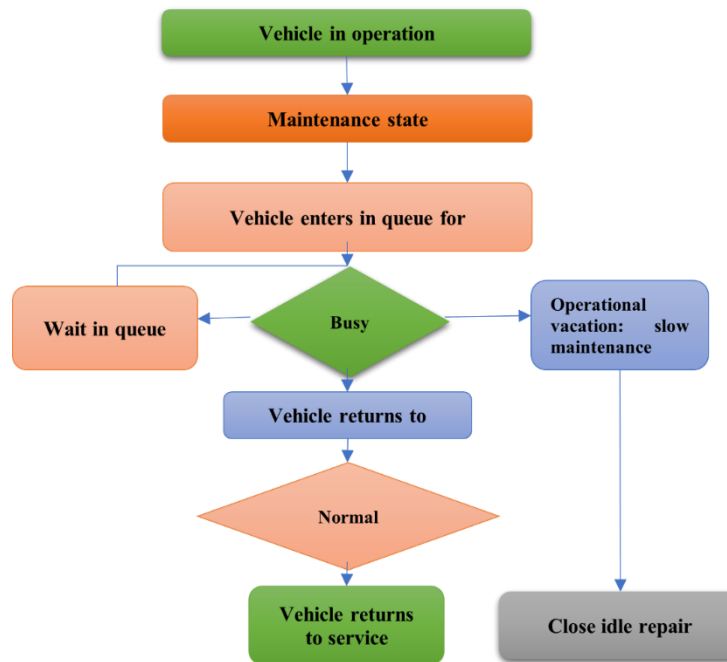


Figure 2. Model application.

4. Governing Equations

The steady state equations are formulated by using the transition rates diagram. The developed equations are formulated by the Chapman-Kolmogorov equation for inflow and outflow rates of different states. The inflow is indicated by the positive sign, and the outflow is denoted by a negative sign.

Server is in 1st repair state

$$\frac{dX_{0,0}(t)}{dt} = -[(K+2)\beta + \lambda_0]X_{0,0}(t) + \bar{r}\alpha \sum_{i=3}^{K+2} X_{0,i}(t) \quad (1)$$

$$\frac{dX_{n,0}(t)}{dt} = -[(K+2)\beta + \lambda_n]X_{n,0}(t) + \lambda_{n-1}X_{n-1,0}(t) + \bar{r}\alpha \sum_{i=3}^{K+2} X_{n,i}(t), \quad 1 \leq n \leq L-1 \quad (2)$$

$$\frac{dX_{L,0}(t)}{dt} = -[(K+2)\beta]X_{L,0}(t) + \lambda_{L-1}X_{L-1,0}(t) + \bar{r}\alpha \sum_{i=3}^{K+2} X_{L,i}(t) \quad (3)$$

Server is in 2nd repair state

$$\frac{dX_{0,1}(t)}{dt} = -[(K+2)\beta + \lambda_0]X_{0,1}(t) + r\alpha \sum_{i=3}^{K+2} X_{n,i}(t) \quad (4)$$

$$\frac{dX_{n,1}(t)}{dt} = -(K+2)\beta + \lambda_n X_{n,1}(t) + \lambda_{n-1} X_{n-1,1}(t) + r\alpha \sum_{i=3}^{K+2} X_{n,i}(t), \quad 1 \leq n \leq L-1 \quad (5)$$

$$\frac{dX_{L,1}(t)}{dt} = -(K+2)\beta X_{L,1}(t) + \lambda_{L-1} X_{L-1,1}(t) + r\alpha \sum_{i=3}^{K+2} X_{L,i}(t) \quad (6)$$

Server is in a retrial state

$$\frac{dX_{0,2}(t)}{dt} = -\lambda X_{0,2}(t) + p\mu_B X_{0,3}(t) \quad (7)$$

$$\frac{dX_{n,2}(t)}{dt} = -[\lambda + \vartheta_0] X_{n,2}(t) + p\mu_B X_{n,3}(t), \quad 1 \leq n \leq L \quad (8)$$

The server is in a busy state

$$\frac{dX_{0,3}(t)}{dt} = -[\lambda_0 + p\mu_B + \alpha] X_{0,3}(t) + \lambda X_{0,2}(t) + r'\beta X_{0,1}(t) + \bar{r}'\beta X_{0,0}(t) + \vartheta_0 X_{1,2}(t) + \eta X_{0,K+4}(t) \quad (9)$$

$$\begin{aligned} \frac{dX_{n,3}(t)}{dt} &= -[\lambda_n + \mu_B + \alpha] X_{n,3}(t) + \lambda_{n-1} X_{n-1,3}(t) + \lambda X_{n,2}(t) + r'\beta X_{n,1}(t) \\ &+ \bar{r}'\beta X_{n,0}(t) + \vartheta_0 X_{n+1,2}(t) + \eta X_{n,K+4}(t), \quad 1 \leq n \leq L-1 \end{aligned} \quad (10)$$

$$\frac{dX_{L,3}(t)}{dt} = -[\mu_B + \alpha] X_{L,3}(t) + \lambda_{L-1} X_{L-1,3}(t) + \lambda X_{L,2}(t) + r'\beta X_{L,1}(t) + \bar{r}'\beta X_{L,0}(t) + \eta X_{L,K+4}(t) \quad (11)$$

Server is in 1st operational vacation state OV1

$$\frac{dX_{0,4}(t)}{dt} = -[\lambda_0 + \alpha] X_{0,4}(t) + r'\beta X_{0,1}(t) + \bar{r}'\beta X_{0,0}(t) + \bar{p}\mu_B X_{1,3}(t) + b\mu_{v1} X_{1,4}(t) \quad (12)$$

$$\begin{aligned} \frac{dX_{n,4}(t)}{dt} &= -[\lambda_n + \alpha + \mu_{v1}] X_{n,4}(t) + r'\beta X_{n,1}(t) + \bar{r}'\beta X_{n,0}(t) + \bar{p}\mu_B X_{n+1,3}(t) \\ &+ b\mu_{v1} X_{n+1,4}(t) + \lambda_{n-1} X_{n-1,4}(t), \quad 1 \leq n \leq L-1 \end{aligned} \quad (13)$$

$$\frac{dX_{L,4}(t)}{dt} = -[\alpha + \mu_{v1}] X_{L,4}(t) + r'\beta X_{L,1}(t) + \bar{r}'\beta X_{L,0}(t) + \lambda_{L-1} X_{L-1,4}(t) \quad (14)$$

Server is in 2nd operational vacation state OV2

$$\frac{dX_{0,5}(t)}{dt} = -[\lambda_0 + \alpha] X_{0,5}(t) + r'\beta X_{0,1}(t) + \bar{r}'\beta X_{0,0}(t) + \bar{b}\mu_{v1} X_{1,4}(t) + b\mu_{v2} X_{1,5}(t) \quad (15)$$

$$\begin{aligned} \frac{dX_{n,5}(t)}{dt} &= -[\lambda_n + \alpha + \mu_{v2}] X_{n,5}(t) + r'\beta X_{n,1}(t) + \bar{r}'\beta X_{n,0}(t) + \bar{b}\mu_{v1} X_{n+1,4}(t) \\ &+ b\mu_{v2} X_{n+1,5}(t) + \lambda_{n-1} X_{n-1,5}(t), \quad 1 \leq n \leq L-1 \end{aligned} \quad (16)$$

$$\frac{dX_{L,5}(t)}{dt} = -[\alpha + \mu_{v2}] X_{L,5}(t) + r'\beta X_{L,1}(t) + \bar{r}'\beta X_{L,0}(t) + \lambda_{L-1} X_{L-1,5}(t) \quad (17)$$

Server is in i^{th} operational vacation state OV i ($i = 6, 7, \dots, K+3$)

$$\frac{dX_{0,i}(t)}{dt} = -[\lambda_0 + \alpha] X_{0,i}(t) + r'\beta X_{0,1}(t) + \bar{r}'\beta X_{0,0}(t) + \bar{b}\mu_{vi-4} X_{1,i-1}(t) + b\mu_{vi-3} X_{1,i}(t) \quad (18)$$

$$\frac{dX_{n,i}(t)}{dt} = -[\lambda_n + \alpha + \mu_{vi-3}]X_{n,i}(t) + r'\beta X_{n,1}(t) + \bar{r}'\beta X_{n,0}(t) + \bar{b}\mu_{vi-4}X_{n+1,i-1}(t) + b\mu_{vi-3}X_{n+1,i}(t) + \lambda_{n-1}X_{n-1,i}(t), 1 \leq n \leq L-1 \quad (19)$$

$$\frac{dX_{L,i}(t)}{dt} = -[\alpha + \mu_{vi-3}]X_{L,i}(t) + r'\beta X_{L,1}(t) + \bar{r}'\beta X_{L,0}(t) + \lambda_{L-1}X_{L-1,i}(t) \quad (20)$$

Server is in a non-operational vacation state, NOV

$$\frac{dX_{0,K+4}(t)}{dt} = -[\lambda_0 + \eta]X_{0,K+4}(t) + \bar{b}\mu_{vK}X_{1,K+3}(t) \quad (21)$$

$$\frac{dX_{n,K+4}(t)}{dt} = -[\lambda_n + \eta]X_{n,K+4}(t) + \bar{b}\mu_{vK}X_{n+1,K+3}(t) + \lambda_{n-1}X_{n-1,K+4}(t), 1 \leq n \leq L-1 \quad (22)$$

$$\frac{dX_{L,K+4}(t)}{dt} = -\eta X_{L,K+4}(t) + \lambda_{L-1}X_{L-1,K+4}(t) \quad (23)$$

5. Performance Measures

The system state probabilities are added to understand the distribution of failed units.

$$\text{❖ Probability of MRS when server is in 1st repair state: } X_{r1} = \sum_{n=0}^{M+S} X_{n,0}(t) \quad (24)$$

$$\text{❖ Probability of MRS when server is in 2nd repair state: } X_{r2} = \sum_{n=0}^{M+S} X_{n,1}(t) \quad (25)$$

$$\text{❖ Probability of MRS when it is in retrial state: } X_{rt} = \sum_{n=0}^{M+S} X_{n,2}(t) \quad (26)$$

$$\text{❖ Probability of MRS when server is in busy state: } X_b = \sum_{n=0}^{M+S} X_{n,3}(t) \quad (27)$$

$$\text{❖ Probability of MRS when server is in } OV_1 \text{ state: } X_{ov1} = \sum_{n=0}^{M+S} X_{n,4}(t) \quad (28)$$

$$\text{❖ Probability of MRS when server is in } OV_2 \text{ state: } X_{ov2} = \sum_{n=0}^{M+S} X_{n,5}(t) \quad (29)$$

$$\text{❖ Probability of MRS when server is in } OV_i \text{ state: } X_{ovi} = \sum_{n=0}^{M+S} X_{n,i}(t), i = 6, 7 \dots K+3 \quad (30)$$

$$\text{❖ Probability of MRS when server is in NOV state: } X_{nov} = \sum_{n=0}^{M+S} X_{n,K+4}(t) \quad (31)$$

$$\text{For evaluation of } P_{0,0} \text{ we use the normalizing condition that is: } \sum_{n=0}^{M+S} \sum_{i=0}^{K+4} X_{n,i} = 1 \quad (32)$$

System of reliability analysis

Let Y be the arbitrary variable that indicates the system's time to failure. Where $\sum_{n=0}^{M+S} \sum_{i=0}^{K+4} X_{n,i}$ is the probability function, is given by

$$R_Y(t) = 1 - [\sum_{n=0}^{M+S} X_{n,3}(t) + \sum_{n=0}^{M+S} \sum_{i=0}^{K+4} X_{n,i}(t)] \quad (33)$$

❖ Then the transformed reliability function is:

$$R_Y^*(s) = \int_0^\infty e^{-st} R_Y(t) dt \quad (34)$$

❖ The mean time to failure (MTTF) of the system is:

$$MTTF = \int_0^\infty R_Y(t) dt = \lim_{s \rightarrow 0} R_Y^*(s) \quad (35)$$

❖ The steady-state availability is calculated as:

$$AV = 1 - \sum_{n=0}^{M+S} X_{n,3}(\infty) - \sum_{n=0}^{M+S} \sum_{i=0}^{K+4} X_{n,i}(\infty) \quad (36)$$

The average expected number of failed units in the system is-

❖ The Average (expected) number of failed units in the 1st repair state:

$$E_n(r) = \sum_{n=0}^L n X_{n,0} \quad (37)$$

❖ The Average (expected) number of failed units in the 2nd repair state:

$$E_n(r) = \sum_{n=0}^L n X_{n,1} \quad (38)$$

❖ The Average (expected) number of failed units in the retrial state:

$$E_n(rt) = \sum_{n=0}^L n X_{n,2} \quad (39)$$

❖ The Average (expected) number of failed units in busy state:

$$E_n(b) = \sum_{n=0}^L n X_{n,3} \quad (40)$$

❖ The Average (expected) number of failed units in OVi state:

$$E_n(ovi) = \sum_{n=0}^L \sum_{i=4}^{K+3} n X_{n,i} \quad (41)$$

❖ The Average (expected) number of failed units in NOV state:

$$E_n(nov) = \sum_{n=0}^L n X_{n,K+4} \quad (42)$$

❖ The average number of failed units waiting in the services area is L_S :

$$L_S = \sum_{n=0}^L \sum_{i=0}^{K+4} n X_{n,i} \quad (43)$$

❖ The average number of total failed units waiting in the queue L_q is:

$$L_q = \sum_{n=0}^L \sum_{i=0}^{K+4} (n-1) X_{n,i} \quad (44)$$

❖ The average number waiting time W_S of failed units in the system is:

$$W_S = \frac{L_S}{\lambda_{eff}} \quad (45)$$

$$\text{where, the } \lambda_{eff} = \sum_{n=0}^L n \lambda_n X_n \quad (46)$$

6. Cost Optimization

Once the model is developed and after completing its computational and numerical simulation, the next step is to move towards the calculation of its cost factor. The cost function is used to calculate the optimal repair rate and the total optimum cost of the system. The notation for the total cost function is defined as:

C_{HC} : Cost hold for each failed unit present in the system.

C_{r1} : Cost per unit time for the 1st repair state present in the system.

C_{r2} : Cost per unit time for the 2nd repair state present in the system.

C_{rt} : Cost per unit time for failed units present retrial state in the system.

C_b : Cost per unit time for failed units in the busy state present in the system.

C_{ovi} : Cost per unit time for failed units present in OVi, $i=1, 2, \dots, K$, state of the system.

C_{nov} : Cost per unit time for failed units present in the NOV state of the system

$C_{\mu B}$: Cost per unit time, while μ_B is the service rate.

$C_{\mu vi}$: Cost per unit in operational vacation time while μ_{vi} is the service rate, $i=1, 2, \dots, K$.

C_{bb} : Cost per unit time, while β is the repair rate of the busy state

C_{ovir} : Cost per unit time while β is the repair rate of OVi, $i=1, 2, \dots, K$.

The expected total cost (TC) function of the Model based on these parameters is

$$TC(F, \mu, \beta) = C_{HC}L_S + C_rX_r + C_{rt}X_{rt} + C_bX_b + \sum_{i=1}^K C_{ovi}X_{ovi} + C_{\mu B}\mu_B + \sum_{i=1}^K C_{\mu vi}\mu_{vi} + \beta(C_{bb} + \sum_{i=1}^K C_{ovir}) \quad (47)$$

Min $TC(F, \mu, \beta)$; $\mu \geq 0, \beta \geq 0$, minimize the total cost by classical optimized cost using the total cost formula with respect to μ is the service rate, and β is the repair rate. Obtained cost convergence curve through Grey Wolf Optimizations and Particle Swarm Optimization with respect to the two different sets of costs.

The cost function was optimized by Particle swarm optimization (PSO) and Grey wolf optimization (GWO) techniques, and has been carried out in order to compare the cost of the proposed model. Both the optimizations have been depicted cost convergence curve.

Particle swarm optimization

The PSO optimization technique is stimulated by the social behavior done by the flock of birds and the school of fish as they travel to their perfect location. So this optimization is inspired by nature, and this shows the coordination of animals by their movement to find food and shelter. This process, based on the iteration, updates the position and its velocity done by each individual based on its own best position found by the swarm. So this algorithm follows the iterative process, particle improve their position with two best values, the first one is known as the pbest (personal best) value, and the second best one is called the gbest (global best value). The particles modify their position and velocity using the equations-

$$x_{n+1} = x_n + v_{n+1} \quad (48)$$

$$v_{n+1} = v_n + c_1r_1 * (p_{best,n} - x_{n+1}) + c_2r_2(g_{best,n} - x_{n+1}) \quad (49)$$

where,

v_n = velocity at n^{th} iteration,

x_n = position at n^{th} iteration,

$r_{1,2}$ = position number chosen within (0, 1), and

c_1, c_2 are the acceleration constants

Grey wolf optimizations

Grey wolf algorithm (GWO) is a nature-influenced optimization algorithm based on the social hierarchy and hunting behaviour of Grey wolves in the wild. GWO is a hunting strategy of Grey wolves, in which the wolves encircle the prey and the iteration of positions of these wolves during the hunt. There are alpha wolves, which are stronger in terms of strategy and position than all in the group. Wolves are followed by beta wolves, and tracked then similar by delta, then by omega wolves. The position of the prey is important in hunting, and the wolves update their positions based on the positions of the alpha, beta, and delta wolves. In the algorithm, encircling the prey and the hunting process is mathematically modelled to iteratively improve the solution to the optimization problem. Alpha is the optimal solution, and beta and delta are the second-best solutions. This is represented by adjusting the position of candidate solutions based on the positions of the α, β, δ , and ω , wolves. The GWO algorithm balances inspection and exploitation to find the optimal or near-optimal solution.

$$\vec{D} = \left| \vec{C} \cdot \vec{X}_p - \vec{X}(t) \right| \quad (50)$$

$$\vec{X}_{t+1} = |\vec{X}_t - \vec{G} \cdot \vec{D}|, \text{ The } \vec{D} \text{ is the distance vector which defines the distance between the prey and the grey wolf, and the } \vec{X}_p \text{ is a vector which defines the position of the prey. } \vec{X} \text{ defines the position of the grey wolves, and variable } t \text{ represents the current situation of iteration.}$$

$$\vec{G} = \vec{C} \cdot (2\vec{r}_1 - \vec{a}) \quad (51)$$

$\vec{C} = (2\vec{r}_2)$, $\vec{a}$ is from 2 to 0 for each iteration, and locate the position between the two specific points; furthermore, the components $\vec{r}_1, \vec{r}_2$ are in the range of $[0, 1]$. The position of the wolves changes iteratively, and it is based on their hunting tactics, resulting in the aiming converging towards the optimal solution. A solution can be found with the help of different equations of solutions.

$$\vec{D}_\alpha = |\vec{C}_1 \cdot \vec{X}_\alpha - \vec{X}|, \vec{D}_\beta = |\vec{C}_2 \cdot \vec{X}_\beta - \vec{X}|, \vec{D}_\delta = |\vec{C}_3 \cdot \vec{X}_\delta - \vec{X}| \quad (52)$$

where, $\vec{X}_1, \vec{X}_2, \vec{X}_3, \vec{C}_1, \vec{C}_2, \vec{C}_3$, and $\vec{G}_1, \vec{G}_2, \vec{G}_3$ are the coefficient vectors.

$$\vec{X}_1 = |\vec{X}_\alpha - \vec{D}_\alpha \cdot \vec{G}_1|, \vec{X}_2 = |\vec{X}_\beta - \vec{D}_\beta \cdot \vec{G}_2|, \vec{X}_3 = |\vec{X}_\delta - \vec{D}_\delta \cdot \vec{G}_3| \quad (53)$$

The grey wolf changes their position during the hunting as the following iteration shows as-

$$\vec{X}_{t+1} = \frac{\vec{X}_1 + \vec{X}_2 + \vec{X}_3}{3} \quad (54)$$

Table 3. Cost set with different cost variables.

Cost parameters	C_{HC}	C_r	C_{rt}	C_b	C_{ov1}	C_{ov2}	C_{ov3}	C_{nov}
Set I	5	6	7	8	5	3	2	1
Set II	50	30	40	20	5	3	2	1
Cost parameters	$C_{\mu B}$	$C_{\mu w1}$	$C_{\mu w2}$	$C_{\mu w3}$	C_{bb}	C_{ov1r}	C_{ov2r}	C_{ov3r}
Set I	20	10	14	15	12	4	3	2
Set II	8	3	5	4	10	2	4	6

7. Sensitivity Analysis

To perform the sensitivity analysis of the proposed Machine Repair System (MRS), where $L = M + S$ (with L representing the total number of machines, M the number of operating machines, and S the number of standby machines), it is essential to fix certain parameters while varying others. Numerical results can then be obtained using MATLAB software. By fixing the parameters as follows: $M = 6, S = 3, K = 5, \eta = 0.7, \tau = 0.2, \lambda = 0.5, \mu_B = 15, \mu_{vi} = \mu_B - i, 1 \leq i \leq K, p = 0.2, b = 0.8, \alpha = 0.5, \beta = 1, \theta_0 = 0.7, \eta = 0.2$, others are in **Table 3**, this section investigates how the system responds to variations in key parameters. Specifically, the analysis explores the effects of the machine arrival rate (λ), the arrival rate of failed units (λ_n), the service rate ($\mu_i, i = 1, 2, \dots, K$), the probability of operational vacation (b), and the breakdown and repair rates (α and β), retrial rate of failed units (θ_0), and switching rate from NOV to busy state (η).

Table 4 presents the variation in state probabilities corresponding to variations in the service rate μ_B . It is observed that as the service rate increases, the probabilities also increase. With respect to time, as time progresses from $t = 5$ to $t = 30$, the probability of the retrial state increases significantly, indicating a rising tendency for units to enter the retrial phase. Meanwhile, the probabilities associated with the repair and busy states decrease gradually. In contrast, the probabilities of the operational vacation states and the Nonoperational vacationed state decline at a much slower rate. This suggests that while the system becomes more likely to enter retrial states over time, the influence of operational vacation phases remains relatively steady and slowly diminishing. These trends highlight the critical role of the service rates in shaping the dynamics of retrial and repair behavior, especially under a busy server condition.

Similarly, **Table 5** presents the variation in state probabilities corresponding to different values of the parameter b . It is observed that increasing the value of b leads to a general decrease in the state probabilities, indicating a diminishing likelihood of the system occupying those states as b increases. As time advances, the probability of the retrial state rises sharply, reflecting an increasing propensity for units to re-enter the retrial phase. Meanwhile, the probabilities of the repair and busy states diminish gradually. By contrast, the operational vacation states and the Non-operational vacationed state exhibit only modest declines. Together, these patterns indicate that although the system increasingly favours retrial states over time, the impact of operational vacation phases remains relatively constant and recedes slowly. The results indicate a gradual change in the operational vacation states, whereas the repair and retrial state probabilities exhibit more rapid variation.

Figures 3 to Figure 6 depict the queue length, represented by the expected number of failed units $E[N]$, in relation to key system parameters. The sensitivity analysis considers variations in the arrival rate λ (from 1 to 3), the operational vacation probability b (from 0.3 to 0.9), the breakdown rate α (from 0.5 to 1.5), and the repair rate β (from 0.3 to 0.7). The results indicate that the queue length increases sharply in the early time interval (from $t = 0$ to $t = 1$), followed by only marginal changes thereafter.

The reliability results are obtained using MATLAB simulations. **Figures 7 to Figure 12** illustrate the system reliability (RT) in relation to various parameters: failure rate λ ranging from 1 to 3, breakdown rate α from 0.5 to 1.5, repair rate β from 0.3 to 0.7, failure rate λ_0 from 0.05 to 0.25, service rate μ_B and μ_v from 5 to 15. The results show that system reliability decreases as the time interval progresses, indicating reduced performance over time under varying parameter conditions.

Figure 13 to Figure 16 illustrate the relationship between waiting time (WT) and various system parameters, including the arrival rate λ (ranging from 1 to 3), α (0.5 to 1.5), β (0.3 to 0.7), and λ_0 (0.3 to 0.7). These graphs demonstrate how the arrival rate λ influences the waiting time over time t . Lower values of λ tend to result in higher waiting times, likely due to increased server idle periods. Conversely, higher λ values reduce waiting times, although excessive arrival rates may lead to system overload.

Figures 17 and 18 show how the service rate (μ_B) and repair rate (β) varying with time (t) affect the total cost. It was observed that when the service rate increases, the total cost first goes down but then starts to rise again. This shows that there is an optimal service rate; if the service is too slow, machines stay idle longer, and if it's too fast, it becomes too expensive. As time goes on, the total cost usually goes down and then levels off. This suggests that the system becomes more efficient over time, possibly because of learning, fewer early problems, or better maintenance.

In **Figure 19 and 20** depict the results compared getting by the PSO convergence curve and GWO convergence curve, and observed that both algorithms reached similar final costs quickly, with no further improvement after 10 iterations for 1st cost set with different cost variables. **Figures 21 and 22** with cost set II show the comparison. After the illustration, it was found that PSO outperformed GWO by achieving a lower final cost, although both algorithms exhibited fast convergence behavior.

Overall, we conclude that Particle Swarm Optimization (PSO) and Grey Wolf Optimizer (GWO) have gained lots of attention in recent years for solving complex reliability and cost optimization problems in queueing theory and machine repair systems. Rani et al. (2023) applied PSO to optimize the cost-reliability trade-off in a series-parallel repairable system, achieving under the admission-controlled policy. A study by Agarwal et al. (2024) demonstrated the effectiveness of GWO in observation of optimal working vacation and preventive maintenance to minimize total downtime and cost. GWO has shown better

performance in balancing and exploration, making it well-suited for nonlinear, multimodal reliability optimization problems. Researchers have used GWO and PSO in optimizing reliability in systems with multiple failure modes and vacation types for learning in a wireless area network, Bedi et al. (2022). Using both PSO and GWO in either a comparative or hybrid manner brings diversity in cost optimization. While PSO offers faster convergence in smooth search spaces, it may get trapped in local optima. GWO, with its leader-based structure, improves global search capability and diversity in solutions. Thus, the combination of these two enhances the strength and improves the quality of results, specifically in complex machine repair models involving multiple conflicting objectives like cost, breakdown, and reliability. PSO typically exhibits rapid initial convergence due to its strong adsorption behavior; particles quickly cluster around promising regions. Convergence curves often display a steep initial decline in fitness, followed by a plateau. In GWO the convergence is more gradual than PSO, with slow initial progress but better refinement in later iterations. This algorithm is less prone to premature convergence, especially for nonlinear or bounded problems.

Table 4. State probabilities for different values of service rate.

$\mu_B = 20$	t	X_{rt}	X_r	X_b	X_{ov1}	X_{ov2}	X_{ov3}	X_{ov4}	X_{nov}
	5	0.525037	0.242938	0.004491	0.053716	0.053891	0.053845	0.053832	0.011985
	10	0.761858	0.120749	0.002969	0.026804	0.026711	0.026647	0.026632	0.007205
	15	0.876341	0.062345	0.002208	0.013801	0.013712	0.013674	0.013664	0.003744
	20	0.931639	0.034189	0.001838	0.007517	0.007443	0.007422	0.007416	0.001979
	25	0.958359	0.020585	0.001659	0.004480	0.004413	0.004401	0.004398	0.001118
	30	0.971270	0.014010	0.001573	0.003012	0.002949	0.002941	0.002939	0.000701
	t	X_{rt}	X_r	X_b	X_{ov1}	X_{ov2}	X_{ov3}	X_{ov4}	X_{nov}
$\mu_B = 25$	5	0.531474	0.239857	0.003611	0.053148	0.053309	0.053260	0.053247	0.011882
	10	0.768566	0.117492	0.002375	0.026150	0.026054	0.025989	0.025975	0.007062
	15	0.882196	0.059505	0.001762	0.013215	0.013131	0.013092	0.013083	0.003612
	20	0.936610	0.031791	0.001467	0.007018	0.006951	0.006930	0.006925	0.001869
	25	0.962677	0.018518	0.001325	0.004049	0.003990	0.003979	0.003976	0.001026
	30	0.975166	0.012157	0.001257	0.002626	0.002572	0.002565	0.002563	0.000621
$\mu_B = 30$	t	X_{rt}	X_r	X_b	X_{ov1}	X_{ov2}	X_{ov3}	X_{ov4}	X_{nov}
	5	0.535840	0.237766	0.003019	0.052761	0.052913	0.052862	0.052850	0.011813
	10	0.773085	0.115297	0.001977	0.025708	0.025611	0.025546	0.025531	0.006966
	15	0.886120	0.057601	0.001466	0.012821	0.012740	0.012702	0.012693	0.003524
	20	0.939930	0.030189	0.001220	0.006684	0.006622	0.006602	0.006597	0.001796
	25	0.965557	0.017138	0.001103	0.003760	0.003708	0.003697	0.003695	0.000965
	30	0.977763	0.010921	0.001047	0.002368	0.002320	0.002314	0.002312	0.000568

Table 5. State probabilities for different values of probability service rate.

$b = 0.1$	t	X_{rt}	X_r	X_b	X_{ov1}	X_{ov2}	X_{ov3}	X_{ov4}	X_{nov}
	5	0.532790	0.234600	0.004607	0.049586	0.052171	0.052356	0.052352	0.021261
	10	0.771856	0.112971	0.003002	0.024043	0.025080	0.025092	0.025064	0.012443
	15	0.884219	0.056956	0.002207	0.012089	0.012567	0.012565	0.012547	0.006311
	20	0.936968	0.030755	0.001829	0.006480	0.006707	0.006706	0.006696	0.003272
	25	0.961752	0.018452	0.001651	0.003844	0.003955	0.003954	0.003949	0.001826
$b = 0.4$	30	0.973398	0.012669	0.001568	0.002605	0.002661	0.002661	0.002658	0.001144
	t	X_{rt}	X_r	X_b	X_{ov1}	X_{ov2}	X_{ov3}	X_{ov4}	X_{nov}
	5	0.525037	0.242938	0.004491	0.053716	0.053891	0.053845	0.053832	0.011985
	10	0.761858	0.120749	0.002969	0.026804	0.026711	0.026647	0.026632	0.007205
	15	0.876341	0.062345	0.002208	0.013801	0.013712	0.013674	0.013664	0.003744
	20	0.931639	0.034189	0.001838	0.007517	0.007443	0.007422	0.007416	0.001979
$b = 0.6$	25	0.958359	0.020585	0.001659	0.004480	0.004413	0.004401	0.004398	0.001118
	30	0.971270	0.014010	0.001573	0.003012	0.002949	0.002941	0.002939	0.000701
	t	X_{rt}	X_r	X_b	X_{ov1}	X_{ov2}	X_{ov3}	X_{ov4}	X_{nov}
	5	0.517166	0.251448	0.004370	0.058275	0.055386	0.055336	0.055334	0.002432
	10	0.751381	0.128988	0.002930	0.029891	0.028322	0.028292	0.028291	0.001503
	15	0.867823	0.068245	0.002207	0.015764	0.014904	0.014888	0.014888	0.000801
$b = 0.6$	20	0.925710	0.038057	0.001847	0.008737	0.008237	0.008228	0.008227	0.000433
	25	0.954487	0.023048	0.001667	0.005243	0.004922	0.004916	0.004916	0.000248
	30	0.968791	0.015585	0.001578	0.003506	0.003273	0.003270	0.003270	0.000156

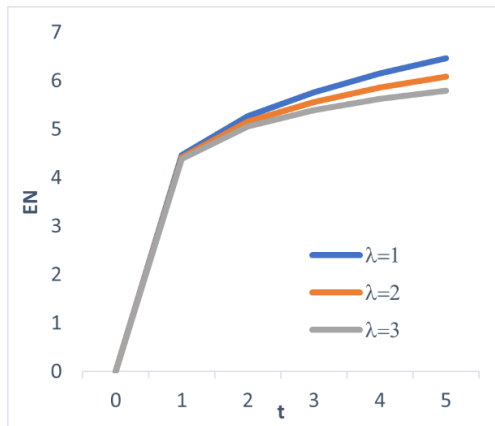


Figure 3. Expected Length with varying t vs. Parameter- λ .

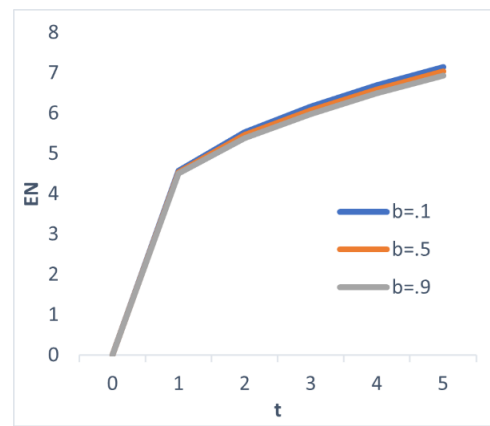


Figure 4. Expected Length with varying t vs. Parameter- b .

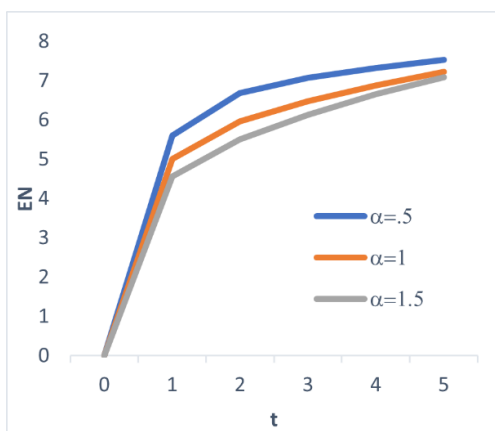


Figure 5. Expected length with varying t vs. parameter- α .

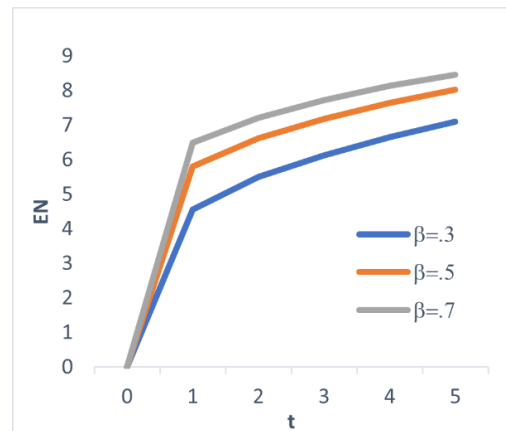


Figure 6. Expected length with varying t vs. parameter- β .

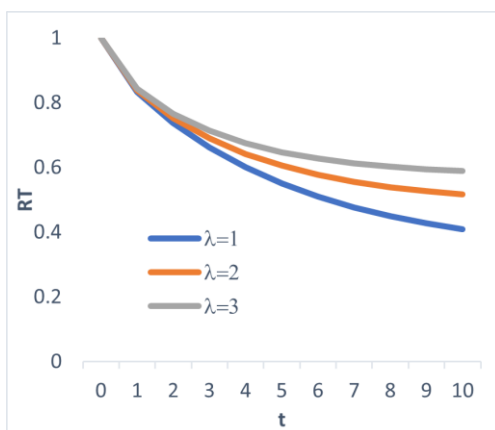


Figure 7. Reliability with varying t vs parameter - λ .

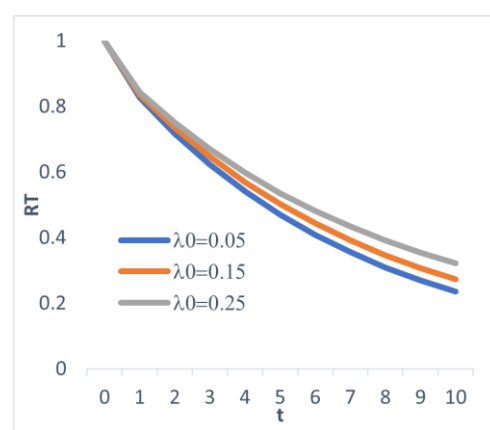


Figure 8. Reliability with varying t vs parameter - λ_0 .

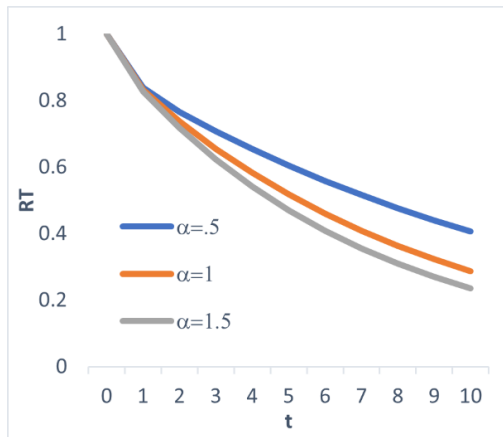


Figure 9. Reliability with varying t vs parameter - α .

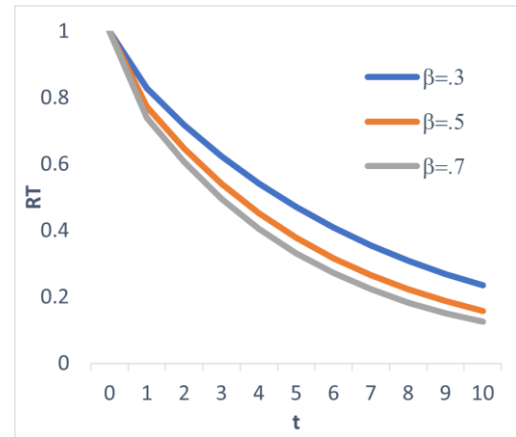


Figure 10. Reliability with varying t vs parameter - β .

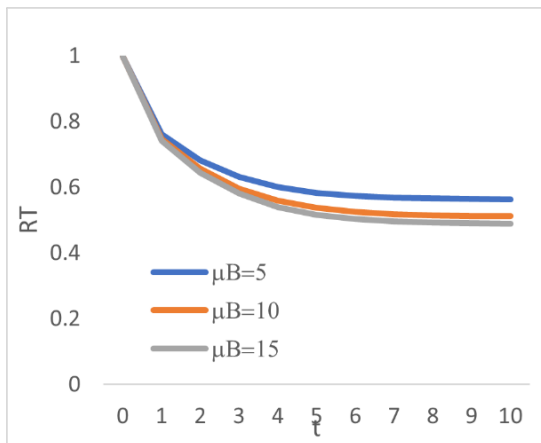


Figure 11. Reliability with varying t vs parameter - μ_B .

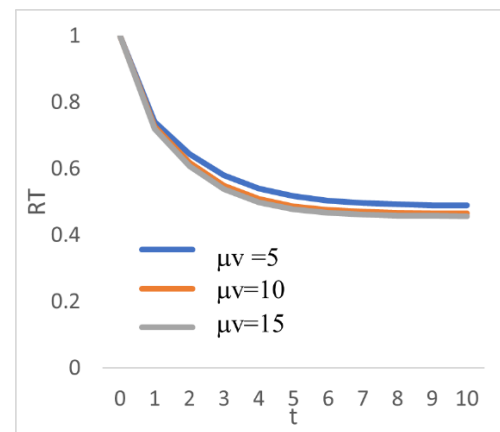


Figure 12. Reliability with varying t vs parameter - μ_v .

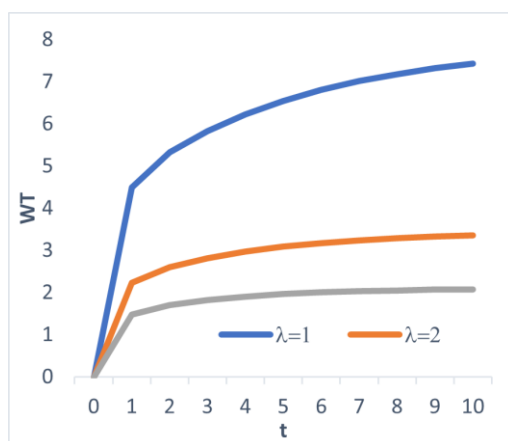


Figure 13. Waiting time with varying t vs parameters - λ .

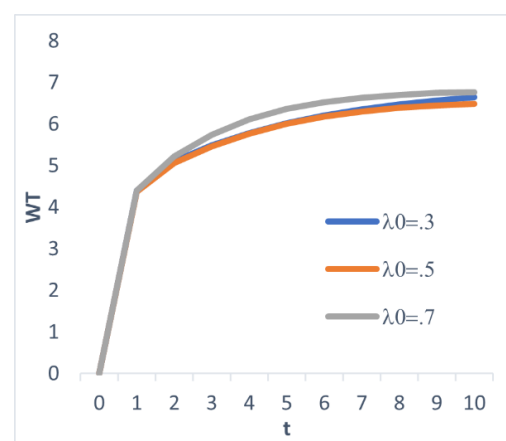


Figure 14. Waiting time with varying t vs parameters - λ_0 .

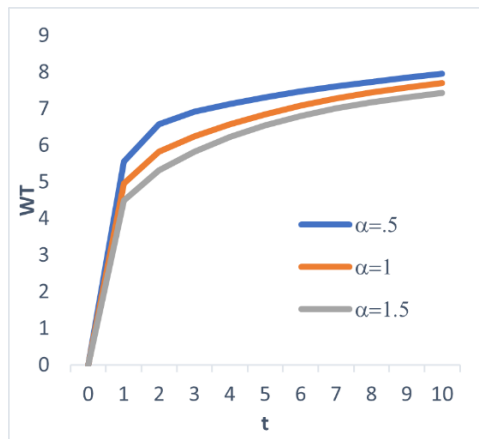


Figure 15. Waiting time with varying t vs parameters - α .

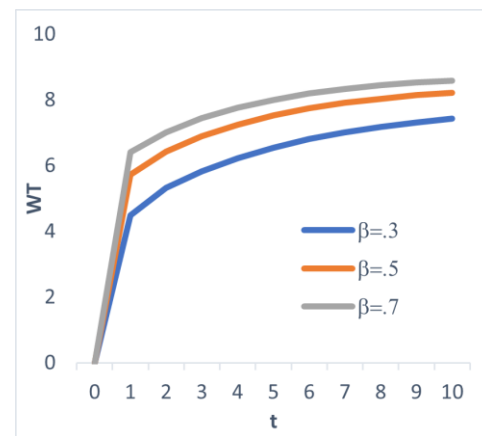


Figure 16. Waiting time with varying t vs parameters - β .

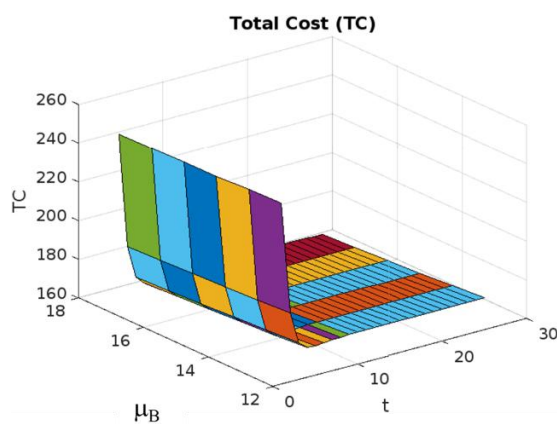


Figure 17. Total cost (TC) w. r. t. μ_B cost set I.

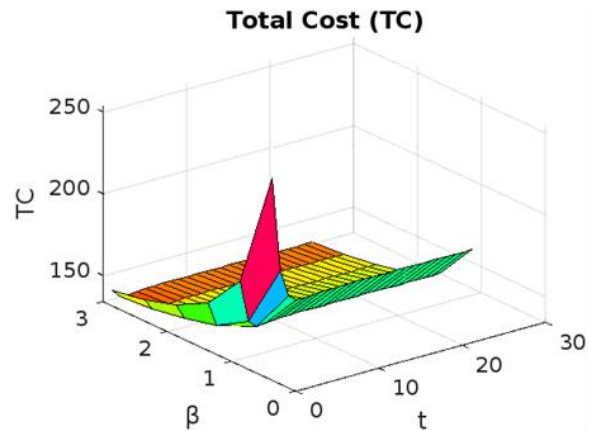


Figure 18. Total cost (TC) w. r. t. β cost set I.

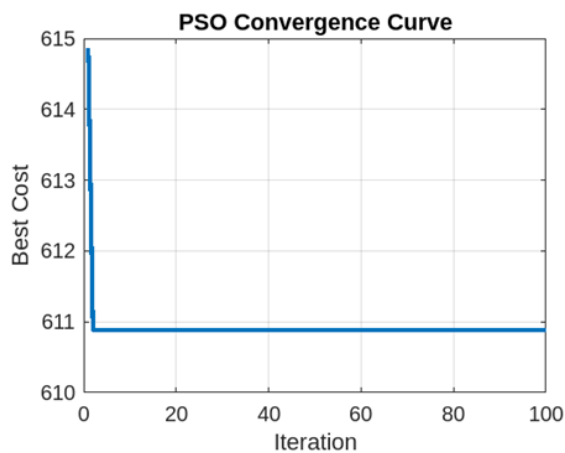


Figure 19. PSO cost convergence curve w. r. t. cost set I.

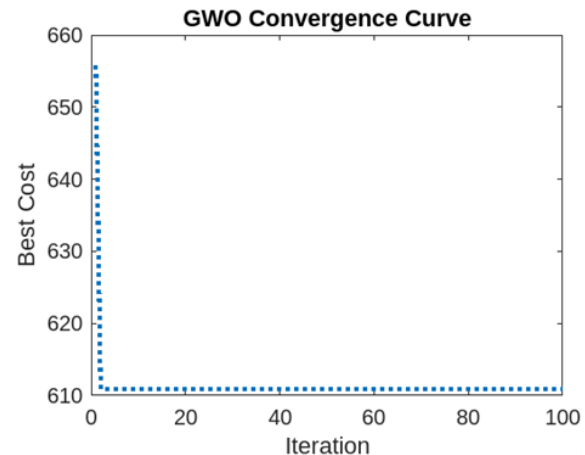


Figure 20. GWO cost convergence curve w. r. t. cost set I.

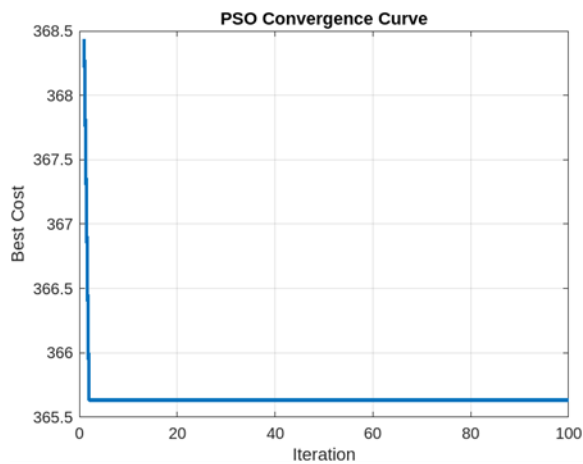


Figure 21. PSO cost convergence curve w. r. t. cost set II.

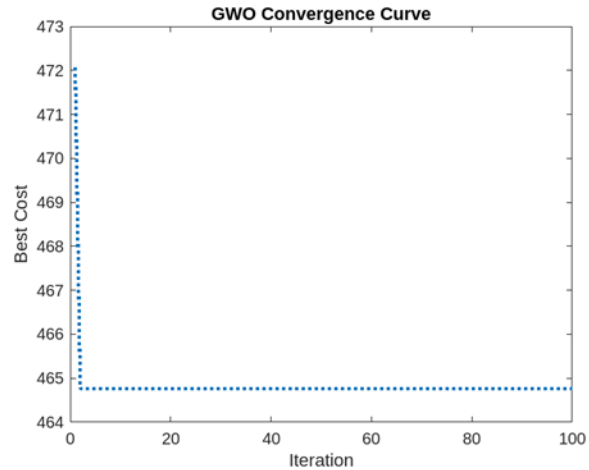


Figure 22. PSO cost convergence curve w. r. t. cost set II.

8. Results Discussion

To ensure the practical relevance of the proposed model, we have shown analytical and optimization results against both simulation and reference data. When the vacation parameter is disabled, the system returns to the standard M/M/1 model without any breakdowns, reproducing the published reliability measurements, thereby confirming the validity of the proposed framework.

In this study, we compare both PSO and GWO to discourse reliability cost optimization in a machine repair model with operational vacations and non-operational vacations, and demonstrate their comparative effectiveness through simulation and performance metrics. In order to derive the optimal parameter and the corresponding optimum cost for the system, MATLAB code is created for both PSO and GWO. The convergence analysis shows that PSO achieves faster convergence within 10 iterations but settles at a slightly higher cost (~ 610.9), whereas GWO starts from a worse initial solution but eventually outperforms PSO, achieving a lower final cost (~ 610). While the cost set II is observed as the PSO archives the convergence within 5 iterations at a lower cost (~ 365.5), whereas GWO eventually performs high at a cost of (~ 464). This demonstrates GWO's superior global search capability, while PSO offers faster local convergence, and GWO is simpler to apply in comparison with PSO because it needs some input parameters. Therefore, combining both can balance exploration and exploitation in reliability-based cost optimization.

9. Conclusion

This study analyzed the reliability of a machine repair system (MRS) with limited capacity, multiple operational vacations, and Non-operational vacations. By incorporating spare machines, the model ensures continuous operation and minimizes system downtime. The state-transition equations provide a comprehensive framework for understanding system dynamics. The numerical techniques were successfully applied to evaluate key performance measures, and numerical results were validated through MATLAB simulations. The findings highlight the effectiveness of hybrid vacation policies in optimizing repair system performance, reducing congestion, and improving reliability. Future research can extend this model by incorporating additional real-world constraints, such as variable repair rates and priority-based servicing, to enhance its applicability. The optimized cost presents the most economical arrangements of the system, balancing maintenance schedules, service availability, and good operational attempts. This

provides an understanding of how to reduce service failures and costs simultaneously, making the system more efficient, reliable, and available in industrial or transportation settings.

Conflicts of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We do not analyze or generate any datasets, because our work proceeds within a theoretical and mathematical approach. One can obtain the relevant materials from the references below.

AI Disclosure

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

References

- Agarwal, D., Agarwal, R., & Upadhyaya, S. (2024). Detection of optimal working vacation service rate for retrial priority G-queue with immediate Bernoulli feedback. *Results in Control and Optimization*, 14, 100397. <https://doi.org/10.1016/j.rico.2024.100397>.
- Agrawal, P.K., Jain, A. & Thakur, S. (2023). Optimization of multicomponent machine system with reneging. *EVERGREEN Joint Journal of Novel Carbon Resource Sciences & Green Asia Strategy*, 10(4), 2638-2644. <https://doi.org/10.5109/7160921>.
- Ahuja, A., & Jain, A. (2023). Fuzzy analysis of queueing system with unreliable service and geometric arrival by incorporating constant retrial policy and delayed threshold recovery. *Journal of Ambient Intelligence and Humanized Computing*, 14(6), 7499-7518. <https://doi.org/10.1007/s12652-022-04455-y>.
- Anitha, K., Poongothai, V., & Godhandaraman, P. (2025). Performance analysis of a queueing system with tandem nodes, retrial and server vacations. *Results in Control and Optimization*, 18, 100520. <https://doi.org/10.1016/j.rico.2025.100520>.
- Bedi, P., Das, S., Goyal, S.B., Shukla, P.K., Mirjalili, S., & Kumar, M. (2022). A novel routing protocol based on grey wolf optimization and Q learning for wireless body area. *Expert Systems with Applications*, 210, 118477. <https://doi.org/10.1016/j.eswa.2022.118477>.
- Bouchentouf, A.A., Boualem, M., Yahiaoui, L., & Ahmad, H. (2022). A multi-station unreliable machine model with working vacation policy and customer impatience. *Quality Technology and Quantitative Management*, 19(6), 766-796. <https://doi.org/10.1080/16843703.2022.2054088>.
- Bouchentouf, A.A., Kumar, K., & Chahal, P.K. (2024). Particle swarm optimization for a redundant repairable machining system with working vacations and impatience in a multi-phase random environment. *Swarm and Evolutionary Computation*, 90, 101688, <https://doi.org/10.1016/j.swevo.2024.101688>.
- Chahal, P.K., Kumar, K. & Soodan, B.S. (2024). Grey wolf algorithm for cost optimization of cloud computing repairable system with N-policy, along with the discouragement and two-level Bernoulli feedback. *Mathematics and Computers in Simulation*, 225, 545-569. <https://doi.org/10.1016/j.matcom.2024.06.005>.
- Chakravarthy, S.R., Shruti, & Kulshrestha, R. (2020). A queueing model with server breakdowns, repairs, vacations, and backup server. *Operations Research Perspectives*, 7, 100131. <https://doi.org/10.1016/j.orp.2019.100131>.
- Deora, P., Kumari, U., & Sharma, D.C. (2021). Cost analysis and optimization of machine repair system with working vacation and feedback-policy. *International Journal of Applied and Computational Mathematics*, 7, 250. <https://doi.org/10.1007/s40819-021-01185-1>.

- Gao, S., & Wang, J. (2021). Reliability and availability analysis of a retrial system with mixed standbys and an unreliable repair facility. *Reliability Engineering and System Safety*, 205, 107240. <https://doi.org/10.1016/j.res.2020.107240>.
- Gao, S., Wang, J., & Zhang, J. (2023). Reliability analysis of a redundant series system with common cause failures and delayed vacation. *Reliability Engineering & System Safety*, 239, 109467. <https://doi.org/10.1016/j.res.2023.109467>.
- Jain, A., & Raychaudhuri, C. (2022). Cost optimization using genetic algorithm in customers' intolerance Markovian model with working vacation and multiple working breakdowns. *International Journal of Mathematical, Engineering and Management Sciences*, 7(5), 656-669. <https://doi.org/10.33889/ijmems.2022.7.5.043>.
- Kumar, K., & Sharma, S. (2025). Modelling and computation for a queueing manufacturing plant system with server working vacation under control policies. *Journal of System Science and System Engineering*. <https://doi.org/10.1007/s11518-025-5664-x>.
- Kumar, P., Jain, M., & Meena, R.K. (2023). Transient analysis and reliability modeling of a fault-tolerant system operating under an admission control policy with double retrial features and working vacation. *ISA Transactions*, 134, 183-199. <https://doi.org/10.1016/j.isatra.2022.09.011>.
- Li, J., Hu, L., Wang, Y., & Kang, J. (2024). Reliability and optimization design of a repairable k-out-of-n retrial system with two failure modes and preventive maintenance. *Communications in Statistics-Theory and Methods*, 53(15), 5524-5552. <https://doi.org/10.1080/03610926.2023.2165822>.
- Meena, R.K., Jain, M., Sanga, S.S., & Assad, A. (2019). Fuzzy modeling and harmony search optimization for machining system with general repair, standby support and vacation. *Applied Mathematics and Computation*, 361, 858-873. <https://doi.org/10.1016/j.amc.2019.05.053>.
- Muthusamy, S., Devadoss, N., & Ammar, S.I. (2022). Reliability and optimization measures of a retrial queue with different classes of customers under a working vacation schedule. *Discrete Dynamics in Nature and Society*, 2022(1), 6806104. <https://doi.org/10.1155/2022/6806104>.
- Rani, S., Jain, M., & Meena, R.K. (2023). Queueing modeling and optimization of a fault-tolerant system with reboot, recovery, and vacationing server operating under admission control policy. *Mathematics and Computer in Simulation*, 209, 408-425. <https://doi.org/10.1016/j.matcom.2023.02.015>.
- Sapna, & Jain, A. (2024). Reliability analysis in multiple breakdown with controllable phase repair machining system. *International Journal of Information Technology*, 16(16), 3623-3632. <https://doi.org/10.1007/s41870-024-01805-1>.
- Sharma, S., Kumar, K., & Garg, S. (2025). Cost analysis for machine repair problem under triadic policy with discouragement and multiple working vacations. *International Journal of Management Science and Engineering Management*, 20(1), 122-140. <https://doi.org/10.1080/17509653.2024.2381741>.
- Thakur, S., & Jain, A. (2025). Markovian queueing model under Bernoulli vacation and servers' malfunctioning: metaheuristic optimization technique. *Mathematics and Computers in Simulation*, 231, 259-275. <https://doi.org/10.1016/j.matcom.2024.12.016>.
- Thakur, S., Jain, A., & Jain, M. (2021). ANFIS and cost optimization for Markovian queue with operational vacation. *International Journal of Mathematical, Engineering and Management Sciences*, 6(3), 894-910. <https://doi.org/10.33889/ijmems.2021.6.3.053>.
- Wu, C.H., & Wang, K.H. (2025). Multi-objective retrial warm-standby system machine repair system with two failure modes. *Quality Technology and Quantitative Management*, 1-23. <https://doi.org/10.1080/16843703.2025.2499801>.
- Yen, T.C., Wang, K.H., & Wu, C.H. (2020). Reliability-based measure of a retrial machine repair problem with working breakdowns under the F-policy. *Computers & Industrial Engineering*, 150, 106885. <https://doi.org/10.1016/j.cie.2020.106885>.

Zhu, S., & Wang, J. (2025). Strategic joining in a single-server Markovian queue with Bernoulli working vacations. *Journal of Systems Science and Complexity*, 38(4), 1683-1706. <https://doi.org/10.1007/s11424-025-3215-7>.



Original content of this work is copyright © Ram Arti Publishers. Uses under the Creative Commons Attribution 4.0 International (CC BY 4.0) license at <https://creativecommons.org/licenses/by/4.0/>

Publisher's Note- Ram Arti Publishers remains neutral regarding jurisdictional claims in published maps and institutional affiliations.