URAL MATHEMATICAL JOURNAL, Vol. 9, No. 2, 2023, pp. 60–76

DOI: 10.15826/umj.2023.2.005

HETEROGENEOUS SERVER RETRIAL QUEUEING MODEL WITH FEEDBACK AND WORKING VACATION USING ARTIFICIAL BEE COLONY OPTIMIZATION ALGORITHM

Divya Kothandaraman[†], **Indhira Kandaiyan**[††]

Department of Mathematics, School of Advanced Sciences,
Vellore Institute of Technology, Vellore-632014, Tamil Nadu, India

[†]divya.k2020@vitstudent.ac.in [††]kindhira@vit.ac.in

Abstract: This research delves into the dynamics of a retrial queueing system featuring heterogeneous servers with intermittent availability, incorporating feedback and working vacation mechanisms. Employing a matrix geometric approach, this study establishes the steady-state probability distribution for the queue size in this complex heterogeneous service model. Additionally, a range of system performance metrics is developed, alongside the formulation of a cost function to evaluate decision variable optimization within the service system. The Artificial Bee Colony (ABC) optimization algorithm is harnessed to determine service rates that minimize the overall cost. This work includes numerical examples and sensitivity analyses to validate the model's effectiveness. Also, a comparison between the numerical findings and the neuro-fuzzy results has been examined by the adaptive neuro fuzzy interface system (ANFIS).

Keywords: Retrial queue, Working vacation, MGA, ANFIS, ABC Optimization.

1. Introduction

In our modern, fast-paced society, it is crucial to prioritize the optimization of service systems due to the ever-changing needs and demands of diverse customers. Although traditional queueing models are valuable, they often fail to address the complexities of modern service environments. This research presents a queueing model that effectively addresses these challenges. This model fundamentally recognizes that service tasks can vary in nature and importance. It acknowledges the significance of selecting the appropriate service provider for a particular task. The concept of 'heterogeneous servers' is relevant here. Some servers specialize in handling routine requests, while others are particularly skilled at addressing complex issues.

Furthermore, we recognize that servers are not able to be available around the clock. Instead, they alternate between performing routine tasks and dedicating their attention to more specialized, secondary jobs. Intermittent availability optimizes resource allocation by ensuring that highly skilled servers are readily available when they are most needed. Finally, we have implemented a "working vacation" feature to guarantee uninterrupted service during periods of downtime. This feature ensures that customers are not left unattended, minimizing disruptions in service even when servers are on a break. This research goes beyond being a mere theoretical innovation; it tackles the actual challenges that service industries encounter in the real world. Our model provides a practical and competitive advantage in the dynamic landscape of modern service provision by enhancing efficiency, boosting customer satisfaction, and optimizing resource utilization.

The novelty of this work lies in its pioneering approach to designing service systems that can adapt to the multifaceted demands of contemporary industries. Unlike traditional queueing models, which rely on uniform servers and predictable service patterns, our model introduces heterogeneity, recognizing that not all service tasks are equal, which is shown in Fig. 1.

The remaining sections of this research paper: Section 2 outlines the model's development and the quasi-birth-death process framework. Moving on to Section 3, we delve into the matrix geometric approach, demonstrating the process of compute steady-state probabilities. In Section 4, we explore various performance metrics derived from the model and their practical implications. Section 5 is dedicated to a comprehensive discussion of sensitivity analysis and a cost assessment for the considered paradigm. Section 6 offers graphical representations of ANFIS and presents the numerical outcomes. The subsequent Section 7, delves into cost optimization strategies. Finally, Section 8 serves as the conclusion, where we summarize our investigation by highlighting the notable characteristics and real-world applications of our study.

1.1. Survey of literature

Our model incorporates two types of servers: one that handles routine tasks (server 1) and another that periodically shifts (server 2) its focus to secondary, more specialized tasks. The presence of heterogeneity enables a service delivery that is more customized and effective. In this heterogeneous queueing model, the servers provide service at a different rate. Morse [14] was the first to propose the notion of service heterogeneity. A queueing model with two classes and two servers is being discussed. A non-preemptive priority structure that is heterogeneous has been studied by Leemans [12]. According to [3], a heterogeneous two-server queueing system with feedback, reverse balking, and reneging and retaining renege customers can be analyzed. Markovian queueing model with discouraged arrivals, reneging customers, and retention of reneged customers was studied by [11] based on two heterogeneous servers finite capacities. A study presents an investigation of the heterogeneous queueing system $M/M/2$ with two types of server failures and catastrophes, along with their respective restoration processes, as conducted by the [16]. A queueing model with MAP arrivals and heterogeneous phase-type group services was researched by [4].

Agarwal [1] initially introduced the concept of a server with intermittent availability, where server 1 is consistently accessible while server 2 is periodically accessible. In this scenario, server 2 is responsible for executing a range of peculiar and unconventional tasks. Service interruptions may occur for a variable duration, but they are limited to instances when the ongoing task has been completed. This particular service is referred to as an intermittently available service. Sharda [19] investigated a queuing issue involving a server that is intermittently accessible, with entries and exits occurring in batches of varying sizes.

In recent times, queueing systems featuring server vacations have become increasingly popular. These "vacations" can arise from server outages or when the server is tasked with other responsibilities. Our model acknowledges the critical importance of maintaining continuous service, even during these working vacation periods. It is designed to ensure that customers are never left unattended, thus minimizing any disruptions in the quality of service provided. A recent trend in vacation queues has been working vacation, where service is provided at a lower rate during vacation periods than it is normally provided; i.e., while on vacation, the server provides service at a slower rate instead of ceasing completely. An initial proposal for a working vacation model has been made by Servi and Finn [13]. Madhu Jain [7] conducted a study on a single server working vacation queueing model that incorporates multiple types of server breakdowns. Sudhesh et al. [21] investigated the time-dependent dynamics of a single server queueing model featuring slow service. The researchers examined the effects of both single and multiple working vacations, as well as customers' impatience during periods of slow service. Krishnamoorthy et al. [9] discussed a queueing system with two heterogeneous servers. One server is always accessible, while the other takes vacations when no users are waiting. Laxmi et al. [23] examined a queuing system with several working vacations, incorporating elements of renewal input, balking, reneging, and heterogeneous servers. Two types of Working Vacations (WVs) and impatient clients were handled with in a multi-server queueing system by Yohapriyadharsini et al. [25]. Kumar et al. [10] examined a unreliable Markovian queueing model with two stage service, incorporating hybrid vacation.

The Matrix Geometric Method is a key technique in queueing theory. It simplifies the analysis of systems with varying service rates and transitions by using matrices, providing efficient solutions for steady-state probabilities and performance metrics. Initially introduced by Neuts [8], matrix-geometric models are the basis for stochastic computations. Recently, Divya [5] conducts an investigation on a Markovian queueing model that incorporates heterogeneous, intermittently available servers with feedback, operating under a hybrid vacation policy. ANFIS is an Adaptive Neuro Fuzzy Inference System. This ANFIS computer model uses fuzzy logic and neural networks to analyze and make decisions. ANFIS is famous for tackling complicated issues in numerous industries because it provides a foundation for building hybrid systems that can learn and adapt from data. ANFIS was established in the early 1980s by Professor Lotfi A. Zadeh [26]. Ahuja et al. [2] presented a comprehensive analysis of a single server queueing model with multiple stage service and functioning vacation, focusing on transient behavior. Additionally, they employed ANFIS computing techniques to enhance their analysis. Sethi et al. [17] conducted a study on the application of ANFIS in analyzing the performance of an unreliable $M/M/1$ queueing system. The study specifically focused on the impact of customers' impatience under N-policy. The ANFIS concept has garnered attention from a multitude of researchers in diverse fields of study [6], [18], [20], [22]. Wu and Yang [24] conducted optimization of a bi-objective queueing model that incorporates a two-phase heterogeneous service.

2. Model description and assumptions

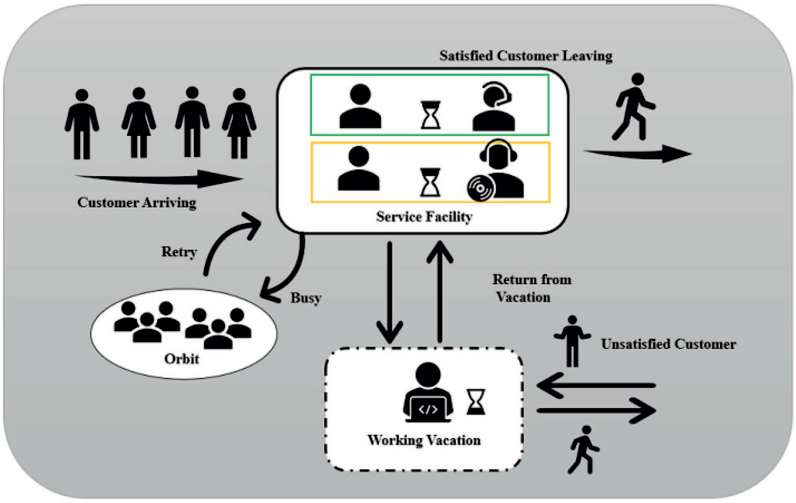


Figure 1. Model diagram.

1. **Arrival process.** In the RQ system, customers arrive according to a Poisson process with a rate of ω.

2. **Service process.** Server 1 is always obtainable, server 2 is intermittently obtainable. The servers provide service to customers with service rates of γ_1 and γ_2, respectively. The retrieval capacity time on server 2 follows an exponential distribution with a rate of β.

3. **Retrial process.** The retrial queuing mechanism enables customers to opt to orbit in the event that the servers are occupied upon their arrival. After a constant retrial rate ϕ, individuals may make another attempt to receive service following an exponentially distributed time.
4. **Vacation process.** A queuing system with working vacation is analyzed, If there are no customers in the orbits when the server 2 finishes servicing, it goes on a working vacation, where the server 2 serves customers with a reduced service rate γ_v during such periods, which follow an exponential distribution. When a vacation ends and there are customers waiting for service, the server 2 switches to regular service with retrieval rate τ. If there are no customers waiting, the server 2 retains in the same working vacation.
5. **Feedback rule.** During WV there are two possible outcomes for customers receiving service during working vacation: they may receive satisfactory service with probability p or unsatisfactory service with complementary probability $\overline{p}(1-p)$. In the event of unsatisfactory service, customers must undergo supplementary service, which follows an exponential distribution.

All stochastic processes in the system are independent of one another. The structure of the models transition diagram is depicted in the below Fig. 2. At time t, let $\chi(t)$ be the state of the server, which is defined as

$$\chi(t) = \begin{cases} 0, & \text{the server 2 is in WV \& it's free,} \\ 1, & \text{the server 2 is in WV \& it's busy,} \\ 2, & \text{the server 2 is in busy,} \\ 3, & \text{the server 2 is in intermittently obtainable} \end{cases}$$

and $\phi(t)$ be the number of customers in the system. The bi-variate process $\{(\phi(t), \chi(t)), t \geq 0\}$ that operates on a state space of $\{0, 1, 2, \dots\} \times \{0, 1, 2, 3\}$.

$$\Upsilon(t) = \{(l, m) | l \geq 0, \ m = 0, 1, 2, 3\}.$$

The state space of a Markov process is arranged in a lexicographical manner, as described below.

$$\Omega = \{(0,0) \bigcup \{(l, m) | l \geq 0, m = 0, 1, 2, 3\}.$$

2.1. Steady-state equation

To solve this problem and obtain effective and mathematically accurate model solutions, we employ the matrix-geometric method described in the following section. The matrix-geometric method is an effective method for obtaining steady-state probabilities when the state-space expands very quickly.

$$\overline{p}\gamma_v\pi_{0,0} = \omega\pi_{0,1}, \tag{2.1}$$

$$(\phi + \overline{p}\gamma_v)\pi_{l,0} = p\gamma_v\pi_{l-1,1} + \omega\pi_{l,1}, \quad l = 1, 2, 3\dots, \tag{2.2}$$

$$(2\omega + \xi + p\gamma_v)\pi_{0,1} = \overline{p}\gamma_v\pi_{0,0} + \phi\pi_{1,0} + \gamma_1\pi_{1,1}, \tag{2.3}$$

$$(2\omega + \gamma_1 + \tau + p\gamma_v)\pi_{l,1} = \omega\pi_{l-1,1}\overline{p}\gamma_v\pi_{l,0} + \phi\pi_{l+1,0} + \gamma_1\pi_{l+1,1}, \quad l = 1, 2, 3\dots, \tag{2.4}$$

$$\omega\pi_{0,2} = (\gamma_1 + \gamma_2)\pi_{1,2} + \beta\pi_{0,3} + \tau\pi_{0,1}, \tag{2.5}$$

$$\omega\pi_{l,2} = \omega\pi_{l-1,2} + \tau\pi_{l,1} + (\gamma_1 + \gamma_2)\pi_{l+1,2} + \beta\pi_{l,3}, \quad l = 1, 2, 3\dots, \tag{2.6}$$

$$(\beta + \omega)\pi_{0,3} = \gamma_1\pi_{1,3}, \tag{2.7}$$

$$(\gamma_1 + \beta + \omega)\pi_{l,3} = \omega\pi_{l-1,3} + \gamma_1\pi_{l+1,3}, \quad l = 1, 2, 3.\dots \tag{2.8}$$

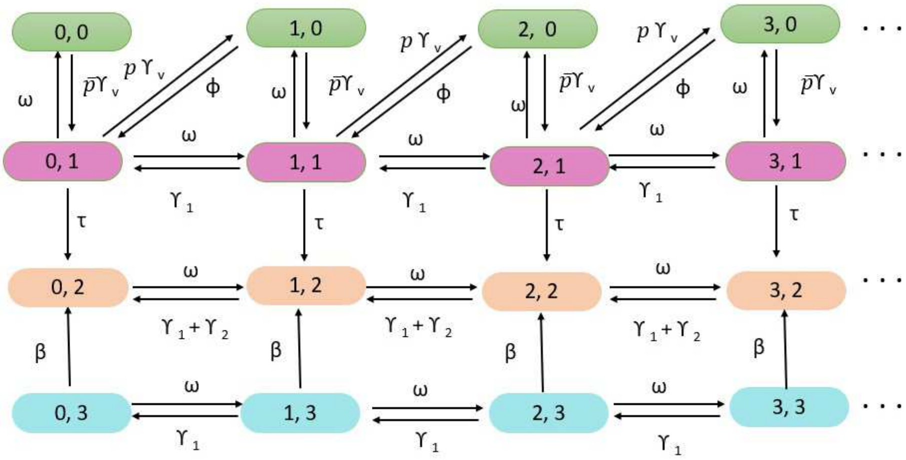


Figure 2. Transition diagram of the model.

3. Matrix-geometric solution

To calculate the steady-state probabilities of the model using the matrix-geometric approach, we utilize a system of equations denoted as (2.1) to (2.8). These equations help determine the probabilities at a steady state. The transition rate matrix Q, which represents the Markov chain in this model, is structured as a block tridiagonal matrix. The matrix Q is subdivided into sub matrices.

$$Q = \begin{bmatrix} S_0 & T_0 & & & & & \\ U_0 & V_0 & T_0 & & & & \\ & U_0 & V_0 & T_0 & & & \\ & & U_0 & V_0 & T_0 & & \\ & & & \ddots & \ddots & \ddots & \\ & & & & \ddots & \ddots & \ddots \end{bmatrix}$$

$$S_0 = \begin{bmatrix} -\omega & \omega & 0 & 0 \\ \overline{p}\gamma_v & -(\omega+\gamma_v+\tau) & \tau & 0 \\ 0 & 0 & -\omega & 0 \\ 0 & 0 & \beta & -(\omega+\beta) \end{bmatrix};$$

$$T_0 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ p\gamma_v & \omega & 0 & 0 \\ 0 & 0 & \omega & 0 \\ 0 & 0 & 0 & \omega \end{bmatrix}; \quad U_0 = \begin{bmatrix} 0 & \phi & 0 & 0 \\ 0 & \gamma_1 & 0 & 0 \\ 0 & 0 & \gamma_1+\gamma_2 & 0 \\ 0 & 0 & 0 & \gamma_1 \end{bmatrix};$$

$$V_0 = \begin{bmatrix} -(\omega+\phi) & \omega & 0 & 0 \\ \overline{p}\gamma_v & -(\omega+\gamma_1+\tau+\gamma_v) & \tau & 0 \\ 0 & 0 & -(\omega+\gamma_1+\gamma_2) & 0 \\ 0 & 0 & \beta & -(\omega+\beta+\gamma_1) \end{bmatrix}.$$

The steady-state probability vector Π for Q is partitioned as $\Pi = (\Pi_0, \Pi_1, \Pi_2, \ldots)$, where the sub-vectors $\Pi_l = \{\pi_{l,0}, \pi_{l,1}, \pi_{l,2}, \pi_{l,3}\},\ l \geq 0$.

3.1. Stability criteria

Theorem 1. *The inequality*

$$\rho = \frac{\gamma_1 + \gamma_2}{\omega} < 1$$

is the necessary and sufficient condition for the system to be stable.

Proof. Let us define the matrix $\mathcal{E} = T_0 + V_0 + U_0$ given by

$$\mathcal{E} = \begin{bmatrix} -\xi_1 & \xi_1 & 0 & 0 \\ \gamma_v & -\xi_2 & \tau & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & \beta & -\beta \end{bmatrix}.$$

Where $\xi_1 = (\omega + \phi)$; $\xi_2 = (\tau + \gamma_v)$. There exists a stationary probability $\Pi = [\Pi_0, \Pi_1, \Pi_2, \Pi_3]$ of $\mathcal{E}$ such that

$$\Pi\mathcal{E} = 0, \quad \Pi e = 1, \tag{3.1}$$

where $e = [1, 1, 1, 1]^T$. Using Theorem 3.1.1 of Netus [8], the necessary and sufficient condition for the stability of the system is as follows:

$$\Pi T_0 e < \Pi U_0 e. \tag{3.2}$$

Solving (3.1) and (3.2), we get

$$\frac{\gamma_1 + \gamma_2}{\omega} < 1. \tag{3.3}$$

□

3.2. Stationary probability distribution

Let Π_{lm} be the steady-state probability that the process is in state (l, m), which is defined as follows:

$$\Pi_{lm} = \lim_{t\to 0} Pr\big[\phi(t) = l, \ \chi(t) = m\big], \quad l = 0, 1, 2, 3\ldots \quad \text{and} \quad m = 0, 1, 2, 3.$$

We denote the steady state probability vector of Q by $\Pi = (\Pi_0, \Pi_1, \Pi_2, \ldots)$. Where $\Pi_l = \big(\pi_{l,1}, \pi_{l,1}, \pi_{l,2}, \pi_{l,3}\big)$ for $l \geq 0$. Under the stability condition (3.3). The steady-state equations can be expressed in matrix form as follows,

$$\Pi\mathrm{Q} = 0. \tag{3.4}$$

Equation (3.4) can be written as

$$\begin{gathered} \Pi_0 S_0 + \Pi_1 U_0 = 0, \\ \Pi_0 T_0 + \Pi_1 V_0 + \Pi_2 U_0 = 0, \\ \vdots \\ \Pi_{i-1} T_0 + \Pi_i V_0 + \Pi_{i+1} U_0 = 0, \quad i = 1, 2, 3, \ldots. \end{gathered}$$

Based on the matrix-geometric method [8, 15], we obtain

$$\Pi_i = \Pi_0 \mathcal{R}^i \quad \text{for} \quad i \geq 1 \tag{3.5}$$

and Π_0 satisfies the set of equations

$$\Pi_0 \left(S_0 + \mathcal{R} U_0 \right) = 0. \tag{3.6}$$

Where $\mathcal{R}$ is referred to as the rate matrix which satisfies

$$T_0 + \mathcal{R} V_0 + \mathcal{R}^2 U_0 = 0,$$

and 0 denotes a zero squared matrix of an appropriate order.

The rate matrix can be approximate iteratively by considering one sequence with initialization $\mathcal{R}_0 = 0$ and calculating

$$\mathcal{R}_{i+1} = -\left(T_0 + \mathcal{R}_i^2 U_0 \right) V_0^{-1}, \quad i = 1, 2, \ldots.$$

Thus, $\lim_{i \to \infty} \mathcal{R}_i$ is an approximate solution of the rate matrix $\mathcal{R}$. From the normalization condition, we obtain the following:

$$\sum_{i=0}^{\infty} \Pi e = \sum_{i=0}^{\infty} \Pi_0 \mathcal{R}^i e = \Pi_0 \left(I - \mathcal{R} \right)^{-1} e = 1.$$

Combining (3.6) with the normalization condition yields

$$\Pi_0 = [\pi_{0,0}, \pi_{0,1} \ldots].$$

Once the steady-state probability vector Π_0 is available, then Π_i (i $\geq$ 1) can be determined using (3.5).

4. Performance measures

4.1. Performance measures

Based on the steady-state probabilities, we give numerous performance metrics for the model under evaluation.

- Prob that the servers are in idle

$$P_i = \pi_{0,0}.$$

- Prob that the server is in busy state

$$P_b = P[m = 2] = \sum_{l=1}^{\infty} \pi_{l,2}.$$

- Prob that the server 2 is in WV state

$$P_{wv} = P[m = 0] + P[m = 1] = \pi_{0,0} + \sum_{l=0}^{\infty} \pi_{l,1}.$$

- Prob that the server 2 is in IO state

$$P_{Io} = P[m = 3] = \sum_{l=0}^{\infty} \pi_{l,3}.$$

- Average system length

$$ASL = \sum_{l=0}^{\infty} l\pi_{l,1} + \sum_{l=0}^{\infty} l\pi_{l,0} + \sum_{l=0}^{\infty} l\pi_{l,2} + \sum_{l=0}^{\infty} l\pi_{l,3}.$$

- Average queue length

$$AQL = \sum_{l=0}^{\infty} l - 1\pi_{l,1} + \sum_{l=0}^{\infty} l - 1\pi_{l,0} + \sum_{l=0}^{\infty} l - 1\pi_{l,2} + \sum_{l=0}^{\infty} l - 1\pi_{l,3}.$$

4.2. Practical application

The proposed queueing model is applicable to a semi-attended self-checkout system in retail stores.

Server 1 – Always Available (Self-Checkout Machine): This server is designed to operate continuously and is accessible to customers at all times, similar to the self-checkout machines commonly found in retail shops. Customers have the freedom to use it at any time without any interruptions.

Server 2, also known as the Intermittently Obtainable (Human Billing Counter), functions similarly to a human cashier at a retail store. The service is available intermittently, which means that it serves customers but may take breaks or go on working vacations. A working vacation refers to planned breaks or vacations for server 2, during which it is temporarily unavailable to serve customers. For instance, a cashier may take a lunch break or have a scheduled time off during their shift. There are numerous systems similar to the queueing model, such as call centers, healthcare triage, banking services, online customer support, and restaurant service.

5. Sensitivity and cost analysis

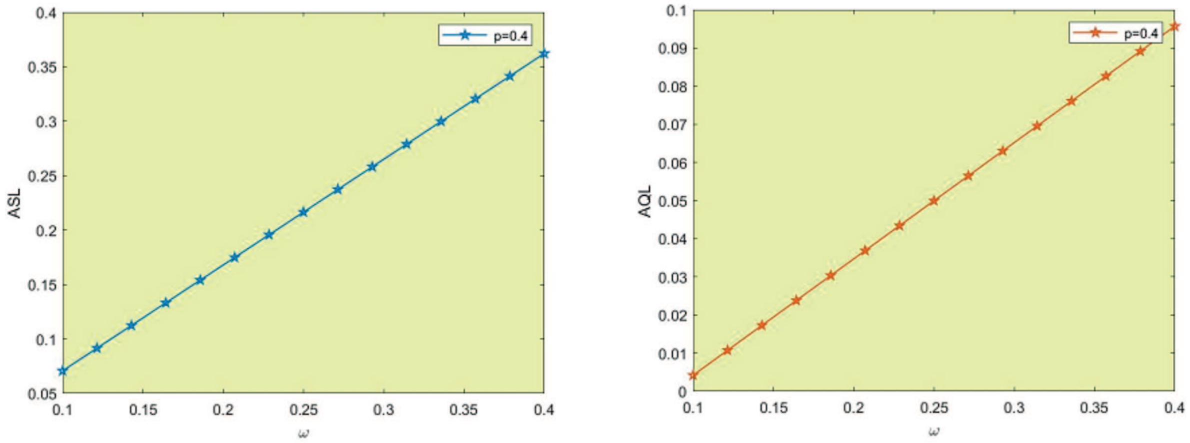
5.1. Sensitivity analysis

Here, we provide numerical examples to demonstrate the influence of various system settings on three distinct efficiency metrics (ω, γ_1, and γ_2) are applied to the following scenarios:
Case 1: $\gamma_1 = 0.5, \gamma_2 = 1$, and vary the value of ω from 0.1 to 0.4.
Case 2: $\omega = 0.05, \gamma_2 = 2$, and vary the value of γ_1 from 0.5 to 1.2.
Case 3: $\omega = 0.06, \gamma_1 = 0.5$, and vary the value of γ_2 from 1 to 3.

All values assigned to the system parameters in our numerical analysis satisfy the stability condition as described by (3.3). The curves depicting the performance measures against the parameters are illustrated in Fig. 3 to 5. Table 1 presents numerical results showing that increasing arrival rate (ω) corresponds to increasing system size (ASL), queue size (AQL) and server busy state probability (P_b). leads to, decreases the probabilities of other states. Similarly, as we increase the service rates (γ_1 and γ_2), the server busy probability (P_b), queue size (AQL), and system size (ASL) decreases, the probabilities of other states increase.

6. ANFIS implementation and results

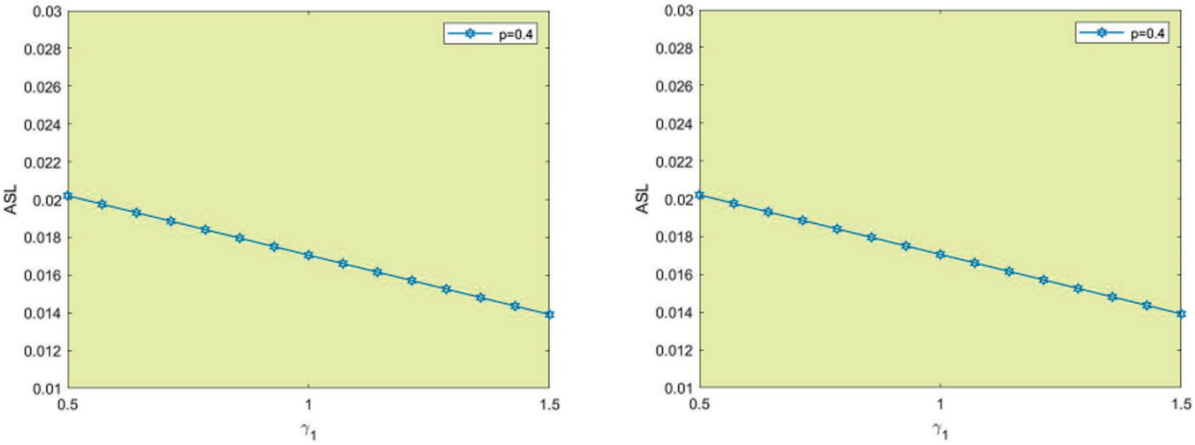
An Adaptive Neuro-Fuzzy Inference System (ANFIS) is a computational model that combines the principles of neural networks and fuzzy logic to perform complex tasks, such as pattern recognition and system modeling. ANFIS utilizes a hybrid approach that blends the adaptability of neural



(a) ω *Vs. ASL*

(b) ω *Vs. AQL*

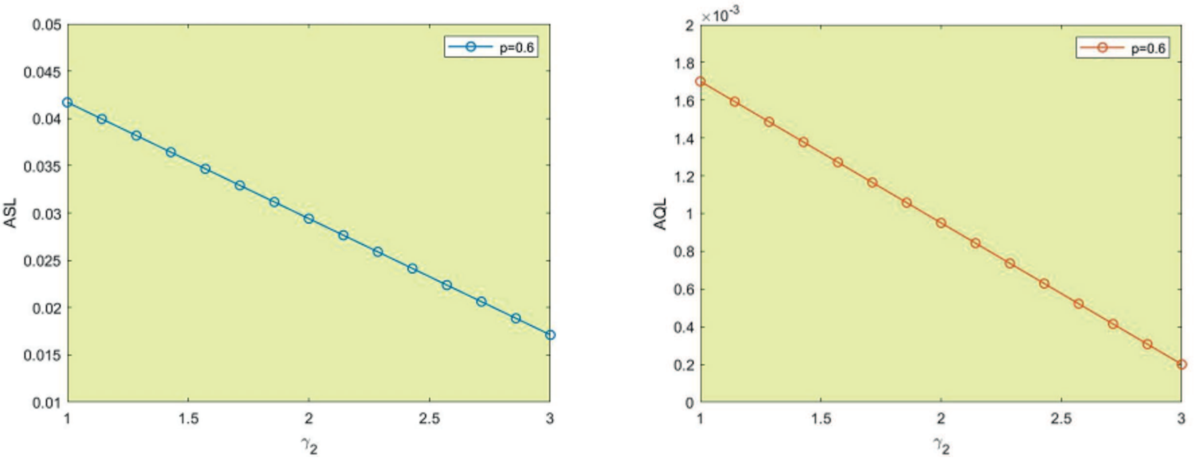
Figure 3. The curves depicting the performance measures against the parameter ω.



(a) γ_1 *Vs. ASL*

(b) γ_1 *Vs. AQL*

Figure 4. The curves depicting the performance measures against the parameter γ_1.



(a) γ_2 *Vs. ASL*

(b) γ_2 *Vs. AQL*

Figure 5. The curves depicting the performance measures against the parameter γ_2.

Table 1. Various performance measures by varying ω, γ_1, γ_2, γ_v.

p	ω	γ_1	γ_2	γ_v	ASL	AQL	P_b
0.4	0.1	0.5	1	0.5	0.0707	0.0042	1.0007
	0.2				0.1535	0.0202	0.9995
	0.3				0.2500	0.0500	1.0000
	0.4				0.3622	0.0957	1.0000
0.6	0.1				0.0717	0.0049	1.0010
	0.2				0.1534	0.0202	0.9995
	0.3				0.2499	0.0499	1.0000
	0.4				0.3610	0.0947	0.9998
0.4	0.05				0.0202	0.0003	0.9999
		0.7			0.0187	0.0003	0.9981
		0.9			0.0174	0.0003	0.9954
0.6		0.5			0.0379	0.0004	1.0009
		0.7			0.0384	0.0013	0.9980
		0.9			0.0432	0.0040	0.7762
0.4	0.06	0.5			0.0417	0.0017	1.0000
			1.5		0.0307	0.0008	0.9999
			2		0.0244	0.0005	0.9999
			2.5		0.0204	0.0004	1.0000
			3		0.0171	0.0002	0.9952
0.6			1		0.0417	0.0017	1.0000
			1.5		0.0309	0.0009	1.0000
			2		0.0246	0.0006	1.0000
			2.5		0.0204	0.0004	1.0000
			3		0.0173	0.0003	0.9955

networks with the interpretability of fuzzy logic. It consists of a layered architecture where input data is passed through a series of nodes, each representing a fuzzy membership function. These nodes calculate membership values based on the input data's similarity to predefined linguistic terms. ANFIS learns and adjusts its parameters using a combination of gradient descent and least-squares methods. This enables it to fine-tune the strengths of its fuzzy rules and the connection weights between nodes to accurately model intricate relationships within the data. The model is particularly useful when dealing with non-linear and uncertain data, making it suitable for applications in various fields, including control systems, prediction, and optimization. By incorporating both neural networks and fuzzy logic, ANFIS provides a balance between the strengths of both approaches, offering a powerful tool for researchers and practitioners to tackle complex problems effectively.

Table 2. Values of the MF for the linguistics based on input parameters.

Input parameters	No. of membeship function	Linguistic Values
p, ω, γ_1, γ_2	5	Very Low, Low, Medium, High, Very High

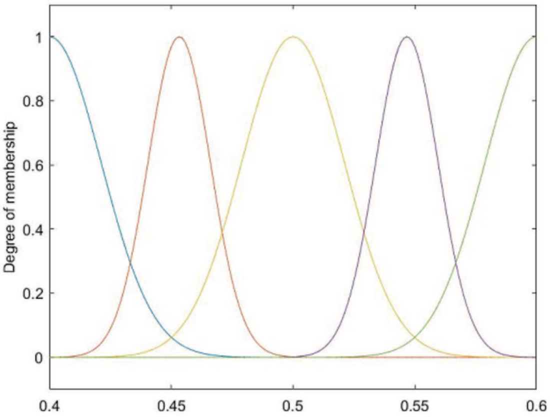
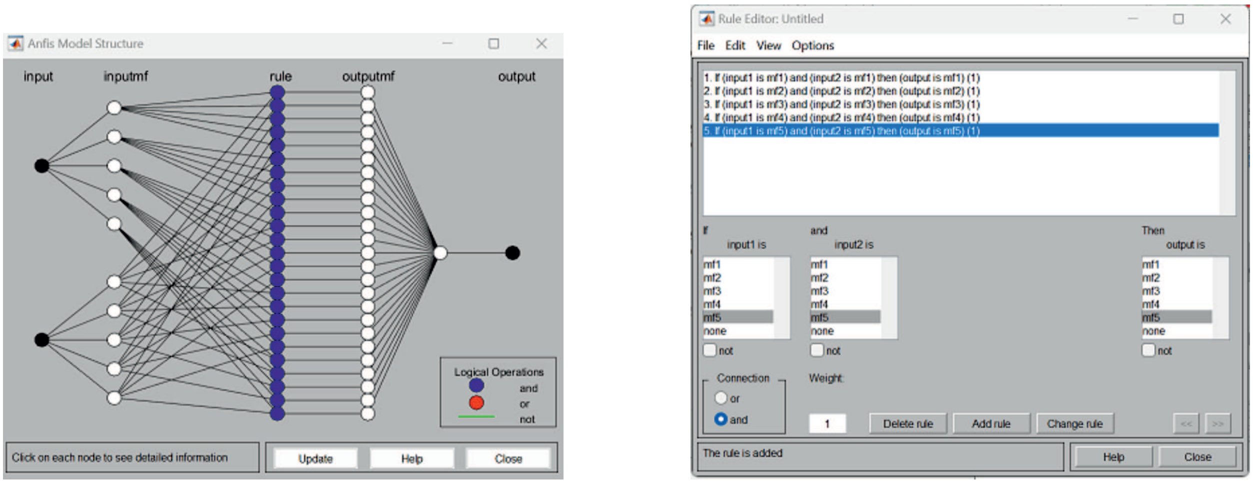


Figure 6. Membership function for ω.



Click on each node to see detailed information

1. If (input1 is mf1) and (input2 is mf1) then (output is mf1) (1)
2. If (input1 is mf2) and (input2 is mf2) then (output is mf2) (1)
3. If (input1 is mf3) and (input2 is mf3) then (output is mf3) (1)
4. If (input1 is mf4) and (input2 is mf4) then (output is mf4) (1)
5. If (input1 is mf5) and (input2 is mf5) then (output is mf5) (1)

If input1 is mf5, and input2 is mf5, Then output is mf5

Connection: ○ or ● and

Weight: 1

The rule is added

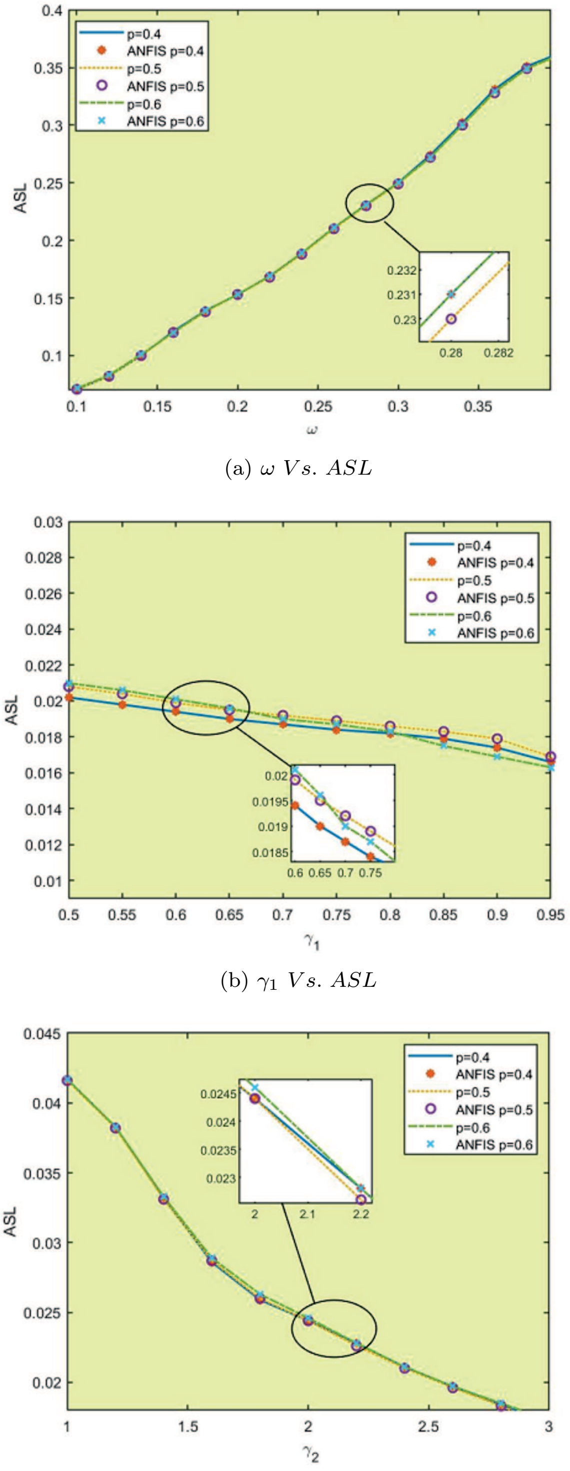
(a)

(b)

Figure 7. (a) ANFIS Structure and (b) Rules.

ANFIS was founded in the first few years of the 1980s by Professor Lotfi A. Zadeh [26]. The architecture of a two-input $(p, \omega, \gamma_1, \gamma_2)$, one-output (ASL) Adaptive Neuro-Fuzzy Inference System (ANFIS) model with five rules is illustrated in Fig. 7. Five fuzzy rules were created and the resulting Gaussian-shaped membership functions (MFs) of the inputs are displayed in Fig. 6. It is noteworthy to emphasize that each colored MF depicted in the curve represents a distinct cluster inside the input space. In ANFIS methodology, the parameters p, ω, γ_1, γ_2 and ASL are regarded as linguistic variables and subjected to training for a total of ten epochs. Three linguistic values have been utilized for the variables p, ω, γ_1, γ_2 and ASL, namely very low, low, medium, high, and very high. Gaussian membership functions were employed to represent the linguistic variables, as illustrated in Table 2.

Fig. 8 displays the analytical results using continuous lines, while the results acquired using ANFIS for ω, γ_1, and γ_2 are shown by a dotted marker point. Furthermore, it was observed that the analytical and ANFIS outcomes had a high degree of concurrence, displaying a significant overlap in their respective trends. Based on the data presented, it can be noted that there is a positive correlation between the system length (ASL) and the arrival rate (ω) while considering different values of p. As the arrival rate increases, the system length also increases. Conversely, the system



(a) ω *Vs. ASL*

(b) γ_1 *Vs. ASL*

(c) γ_2 *Vs. ASL*

Figure 8. Comparison of 2D numerical and ANFIS values for different values of p.

length lowers as the service rates (γ_1 and γ_2) decrease, while considering different values of p.

7. Cost model and optimization

The suggested queueing model has the potential to be implemented in both self-checkout systems and retail stores with human cashiers. In such scenarios, the primary objective of the manager is to minimize operational expenses. An essential consideration related to the self-checkout process pertains to the determination of the appropriate quantity of self-checkout machines for the service. Another crucial matter to consider is the necessity of upholding a satisfactory service rate in order to ensure the quality of service and customer satisfaction. It is feasible to improve the calibre of service rendered by the staff through comprehensive training. Moreover, it is not possible. To ascertain the customer's familiarity with the operation of a particular system. Self-checkout machines. Therefore, we proceed to formulate a cost function per unit of time in a manner that aligns with our expectations.

$$\mathcal{F} = F(\gamma_1, \gamma_2) = F_h \cdot ASL + F_b \cdot P_b + F_v \cdot P_{wv} + F_i \cdot P_{Io} + F_1 \cdot \gamma_1 + F_2 \cdot \gamma_2. \tag{7.1}$$

The variables in the equation are defined as follows: F_h represents the holding cost for each customer in the system, F_v represents the cost per unit of time when the server 2 in the working vacation service, F_b represents the cost per unit of time when server 2 is busy, F_i represents the cost per unit of time when server 2 is intermittently obtainable, F_1 represents the cost of providing a mean service rate γ_1 through server 1, and F_2 represents the cost of providing a mean service rate γ_2 through server 2. It is noteworthy to mention that (7.1) represents a mathematical function that depends on two continuous decision variables, denoted as γ_1 and γ_2. We set the cost elements as given in Table 3.

Table 3. Cost set values for various cost aspects.

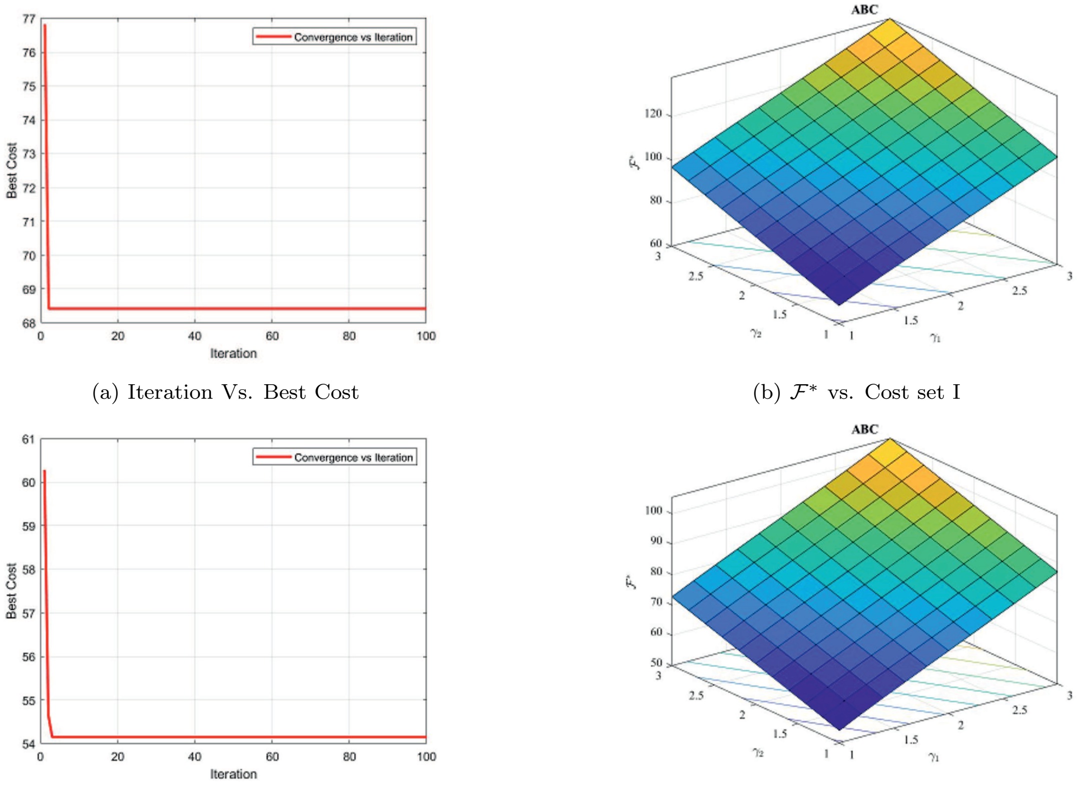
Cost set	F_h	F_b	F_v	F_i	F_1	F_2
I	60	50	40	30	25	15
II	50	45	35	20	20	10

7.1. Artificial bee colony optimization

ABC optimization, also known as Artificial Bee Colony optimization, is a meta-heuristic algorithm inspired by the foraging behavior of honey bees. It is a swarm-based optimization technique that can be applied to solve various optimization problems. The ABC optimization technique, initially introduced by Karaboga in 2005, has garnered significant recognition and acclaim in the field of optimization. The system employs worker bees, observer bees, and scout bees to explore the search space, exchange information, and discover improved solutions. The historical effect in ABC ensures that the algorithm strategically priorities regions of the search space that have demonstrated favorable outcomes, thus leveraging past successes. This enables ABC to efficiently converge towards optimal solutions and effectively address complex optimization problems.

In order to optimize ABC, the following default values are taken into account: $\omega = 3$, $\gamma_1 = 1.5$, $\gamma_2 = 0.6$, $\gamma_v = 0.5$, $\phi = 0.05$, $\tau = 0.5$, $\beta = 0.5$, $p = 0.4$, $p_1 = 0.6$ with a colony size of 100, a maximum of 100 iterations, an acceleration coefficient upper and lower bound are 1 and 5, and number of Onlooker bee 50, an abandonment limit parameter of 60.

Table 4 show the effect of cost elements F_h, F_b, F_v, F_i, F_1 and F_2 on the optimal service rates (γ_1^*, γ_2^*) and optimal total cost $(\mathcal{F}^*)$ for all two cost sets. The pseudo code of ABC algorithm is given in Table 1.



(a) Iteration Vs. Best Cost

(b) $\mathcal{F}^*$ vs. Cost set I

(c) Iteration Vs. Best Cost

(d) $\mathcal{F}^*$ vs. Cost set II

Figure 9. 2D and 3D visualization of ABC optimization.

7.2. Convergence

Convergence is a crucial aspect of meta-heuristic optimization algorithms. It signifies the process of gradually refining the candidate solutions toward an optimal or near-optimal solution. The convergence behavior of an algorithm is indicative of its ability to effectively search the solution space and approach the global optimum. In ABC, particles move towards the best-known solution, converging when their movements become limited and the best solution stabilizes. Fig. 9 (a) and (c) shows that ABC reach optimal cost convergence, Fig. 9 (b) and (d) shows the convexity and optimal of the cost function with respect to cost sets which are considered in the optimization analysis.

8. Conclusion

In this research has tackled the complexities of a retrial queueing system with heterogeneous servers, intermittent availability, feedback, and working vacation mechanisms. Employing a matrix geometric approach, we established the steady-state probability distribution and formulated

Table 4. The estimated optimal solutions (γ_1^*, γ_2^*) and their corresponding expected cost $\mathcal{F}^*$

$(\omega,\ \gamma_1,\ \gamma_2)$	γ_1^*	γ_2^*	$\mathcal{F}^*$	Iterations	CPU time(in Sec)
		Cost set I			
(3,1.5,0.6)	1.1619	1.6258	80.7970	11	$12.50e^{-6}$
(3,1.5,1)	1.1640	1.0003	72.0522	15	$16.90e^{-6}$
(3,3.5,0.6)	1.0349	1.0190	69.4773	24	$20.60e^{-6}$
(3,3.5,1)	1.0131	1.2647	72.4869	14	$15.60e^{-6}$
(5,1.5,0.6)	1.0119	1.0079	67.7305	20	$21.00e^{-6}$
(5,1.5,1)	1.0533	2.0517	84.8465	14	$15.60e^{-6}$
(5,3.5,0.6)	2.6079	1.0745	108.0219	18	$19.70e^{-6}$
(5,3.5,1)	1.0526	1.0593	69.7106	30	$30.40e^{-6}$
		Cost set II			
(3,1.5,0.6)	1.0116	1.4552	58.6088	13	$16.80e^{-6}$
(3,1.5,1)	1.0066	1.0265	54.5188	13	$17.60e^{-6}$
(3,3.5,0.6)	1.1555	1.1191	57.9891	12	$13.60e^{-6}$
(3,3.5,1)	1.0008	1.0089	54.2473	11	$13.50e^{-6}$
(5,1.5,0.6)	1.0102	1.0084	53.4592	10	$11.90e^{-6}$
(5,1.5,1)	1.3133	1.4153	64.3520	19	$22.30e^{-6}$
(5,3.5,0.6)	1.1714	1.0394	57.4689	12	$12.70e^{-6}$
(5,3.5,1)	1.0068	1.0198	53.5026	21	$23.40e^{-6}$

Algorithm 1 Artificial Bee Colony

```
Input: Objective function $\mathcal{F}(\gamma_1, \gamma_2)$, Maximum number of iterations
Output: The best solution found
Initialization;
Initialize employed bees with random solutions;
Evaluate the fitness of each solution;
Set the best solution as the solution with the best fitness;
while Termination condition not met do
    for each employed bee do
        Select a solution randomly from the population;
        Generate a new solution by modifying the selected solution;
        Evaluate the fitness of the new solution;
        If the new solution is better, replace the old solution;
    end for
    Update the best solution if a better solution is found;
end while
return The bestsolution found;
```

performance metrics and a cost function. Leveraging the Artificial Bee Colony optimization algorithm, we optimized service rates effectively. Furthermore, we compared our numerical findings with ANFIS results, highlighting the potential synergy between traditional methods and advanced machine learning approaches in queueing theory research. In future, it is possible to expand the proposed model to include additional factors such as different server vacations, server breakdowns, and customer impatience.

REFERENCES

1. Agarwal N. N. *Some Problems in the Theory of Reliability and Queues.* Ph.D. Thesis. Kurukshetra, India: Kurukshetra University, 1965.
2. Ahuja A., Jain A., Jain M. Transient analysis and ANFIS computing of unreliable single server queueing model with multiple stage service and functioning vacation. *Math. Comput. Simulation*, 2022. Vol. 192. P. 464–490. DOI: 10.1016/j.matcom.2021.09.011
3. Bouchentouf A. A., Kadi M., Rabhi A. Analysis of two heterogeneous server queueing model with balking, reneging and feedback. *Math. Sci. Appl E-Notes*, 2014. Vol. 2, No. 2. P. 10–21.
4. Chakravarthy S. R. Analysis of a queueing model with MAP arrivals and heterogeneous phase-type group services. *Mathematics*, 2022. Vol. 10, No. 19. Art. no. 3575. DOI: 10.3390/math10193575
5. Divya K., Indhira K. Analysis of a heterogeneous queuing model with intermittently obtainable servers under a hybrid vacation schedule. *Symmetry*, 2023. Vol. 15, No. 7. Art. no. 1304. DOI: 10.3390/sym15071304
6. Divya K., Indhira K. Performance analysis and ANFIS computing of an unreliable Markovian feedback queuing model under a hybrid vacation policy. *Math. Comput. Simulation*, 2024. Vol. 218. P. 403–419. DOI: 10.1016/j.matcom.2023.12.004
7. Jain M., Jain A. Working vacations queueing model with multiple types of server breakdowns. *Appl. Math. Model.*, 2010. Vol. 34, No. 1. P. 1–13. DOI: 10.1016/j.apm.2009.03.019
8. Neuts M. F. Matrix-Geometric solutions in stochastic models. In: *DGOR. Operations Research Proceedings, vol. 1983. Steckhan H. and al. (eds.).* Berlin, Heidelberg: Springer, 1984. 425 p. DOI: 10.1007/978-3-642-69546-9_91
9. Krishnamoorthy A., Sreenivasan C. An $M/M/2$ queueing system with heterogeneous servers including one with working vacation. *Int. J. Stoch. Anal.*, 2012. Vol. 2012. Art. no. 145867. DOI: 10.1155/2012/145867
10. Kumar A., Jain M. Cost optimization of an unreliable server queue with two stage service process under hybrid vacation policy. *Math. Comput. Simulation*, 2023. Vol. 204. P. 259–281. DOI: 10.1016/j.matcom.2022.08.007
11. Kumar R., Sharma S. K. Two heterogeneous server Markovian queueing model with discouraged arrivals, reneging and retention of reneged customers. *Int. J. Oper. Res.*, 2014. Vol. 11, No. 2. P. 64–68.
12. Leemans H. E. *The Two-Class Two-Server Queueing Model with Nonpreemptive Heterogeneous Priority Structures.* Ph.D. Thesis. Leuven: Katholieke Universiteit Leuven, 1998.
13. Servi L. D., Finn S. G. $M/M/1$ queues with working vacations $(M/M/1/WV)$. *Perform. Eval.*, 2002. Vol. 50, No. 1. P. 41–52. DOI: 10.1016/S0166-5316(02)00057-3
14. Morse P. M. *Queues, Inventories and Maintenance: The Analysis of Operational Systems with Variable Demand and Supply.* Mineola, New York: Courier Corporation, 2004. 202 p.
15. Latouche G., Ramaswami V. *Introduction to Matrix Analytic Methods in Stochastic Modeling.* ASA-SIAM Series Stat. Appl. Math. Philadelphia, Pennsylvania: SIAM; Alexandria, Virginia: ASA, 1999. 334 p. DOI: 10.1137/1.9780898719734
16. Seenivasan M., Senthilkumar R., Subasri K. S. $M/M/2$ heterogeneous queueing system having unreliable server with catastrophes and restoration. *Mater. Today: Proc.*, 2022. Vol. 51, No. 8. P. 2332–2338. DOI: 10.1016/j.matpr.2021.11.567
17. Sethi R., Jain M., Meena R. K., Garg D. Cost optimization and ANFIS computing of an unreliable $M/M/1$ queueing system with customers' impatience under N-policy. *Int. J. Appl. Comput. Math.*, 2020. Vol. 6. Art. no. 51. P. 1–14. DOI: 10.1007/s40819-020-0802-0
18. Sanga S. S., Jain M. Cost optimization and ANFIS computing for admission control of $M/M/1/K$ queue with general retrial times and discouragement. *Appl. Math. Comput.*, 2019. Vol. 363. Art. no. 124624. DOI: 10.1016/j.amc.2019.124624
19. Sharda. A queuing problem with intermittently available server and arrivals and departures in batches of variable size. *ZAMM*, 1968. Vol. 48. P. 471–476.
20. Stojčić M., Pamučar D., Mahmutagić E., Stević Ž. Development of an ANFIS Model for the Optimization of a Queuing System in Warehouses. *Information*, 2018. Vol. 9, No. 10. Art. no. 240. 20 p. DOI: 10.3390/info9100240

21. Sudhesh R., Azhagappan A., Dharmaraja S. Transient analysis of $M/M/1$ queue with working vacation, heterogeneous service and customers' impatience. *RAIRO-Oper. Res.*, 2017. Vol. 51, No. 3. P. 591–606. DOI: 10.1051/ro/2016046
22. Thakur S., Jain A., Jain M. ANFIS and cost optimization for Markovian queue with operational vacation. *Int. J. Math. Eng. Management Sci.*, 2021. Vol. 6, No. 3. P. 894–910. DOI: 10.33889/IJMEMS.2021.6.3.053
23. Vijaya Laxmi P., Jyothsna K. Balking and reneging multiple working vacations queue with heterogeneous servers. *J. Math. Model. Algor.*, 2015. Vol. 14. P. 267–285. DOI: 10.1007/s10852-015-9271-6
24. Wu C.-H., Yang D.-Y. Bi-objective optimization of a queueing model with two-phase heterogeneous service. *Comput. Oper. Res.*, 2021. Vol. 130. Art. no. 105230. DOI: 10.1016/j.cor.2021.105230
25. Yohapriyadharsini R. S., Suvitha V. Multi-server Markovian heterogeneous arrivals queue with two kinds of working vacations and impatient customers. *Yugosl. J. Oper. Res.*, 2023. Vol. 33, No. 4. P. 643–666. DOI: 10.2298/YJOR221117011Y
26. Zadeh L. A. The concept of a linguistic variable and its application to approximate reasoning–I. *Inform. Sci.*, 1975. Vol. 8, No. 3. P. 199–249. DOI: 10.1016/0020-0255(75)90036-5