

## ANFIS COMPUTING FOR $MAP/PH/1$ QUEUEING SYSTEM WITH SERVER DETERIORATION AND RECOVERY UNDER VACATION AND BACKUP SERVER

BALARAMAN SOMASUNDARAM AND K. DIVYA

**ABSTRACT.** We examine a queueing system with vacation in which the service rate deteriorates during the service and recovers the service rate during the vacation. The backup server with less service rate takes the place of the main server during the recovery time. The matrix analytic method is used to analyse the system and some performance metrics are obtained. The Adaptive Neuro-Fuzzy Inference System (ANFIS) was utilized to construct various performance measures by developing rules from input datasets. The results and the sensitivity of the model parameters are demonstrated by numerical examples.

### 1. INTRODUCTION

Queueing systems play a crucial role in operations research and are widely used to evaluate the performance of various service-oriented processes, such as telecommunications, manufacturing, and computer networks. The  $MAP/PH/1$  queue, which features a Markovian arrival process (MAP) and a phase-type(PH) service time distribution controlled by a single server, is the type of queueing system that is examined in this paper. Further, servers do not always perform at their optimal efficiency in many real-world applications. In such scenarios, server degradation is a typical problem. To overcome this, systems often include recovery methods. In addition, to maintain service continuity, such systems may contain facilities for server vacations, in which the primary server is temporarily outside of work while a backup server handles the burden.

Thus, by modeling and analyzing one such  $MAP/PH/1$  queueing system with the aforementioned characteristics, this work aims to provide insights into the behaviour and performance of systems. The inclusion of server vacations and backup servers adds another layer of realism, reflecting common operational practices in service environments. This study not only enhances our theoretical understanding of queueing systems but also offers practical implications for designing more robust and efficient service systems.

We now provide a detailed literature review of the  $MAP/PH/1$  queueing model, focusing on studies addressing server deterioration and recovery mechanisms with backup server alongside vacation. This review highlights key findings and methodologies used to analyze and mitigate performance declines in such systems.

**MAP (Markovian Arrival Process):** Regarding the theory of point processes, the MAP stands out as one of the most versatile modeling tools, aiming to formulate incoming processes that may not necessarily be renewal processes, presenting a distinct approach. Neuts [32] introduced the MAP

---

Received by the editors 18 November 2024; accepted 14 March 2025; published online 28 March 2025.

2020 *Mathematics Subject Classification.* Primary 60K25, 60K20; Secondary 90B35, 90B36.

*Key words and phrases.* Markovian arrival process; phase-type; vacation; MAM; ANFIS..

by presenting a versatile class of point processes on the real line, which are closely associated with finite-state Markov processes.

A MAP is a stochastic process used to model arrivals in a queueing system. It generalizes the Poisson process by allowing for more complex, correlated arrival patterns. In a MAP, arrivals occur according to a Markov chain, which means the arrival rate can change depending on the state of the system.

Chakravarthy [9] further explained the Markovian Arrival Process (MAP), detailing its mathematical characteristics and practical applications in modeling complex and diverse arrival patterns in queueing systems. Bashari et al. [7] described the process of deriving properties of MAP from the general theory of Markov processes with a homogeneous second component. In addition, a *MAP/PH/1* queue with discretionary priority was studied by Zhao et al. [41]. Moreover, a *MAP/PH/1* queue was scrutinized recently by employing the matrix-analytic method by Sindhu and Krishnamoorthy [38].

**PH(Phase - Type) Distribution:** The phase-type (PH) distribution is a flexible distribution used to depict the time before an event occurs, represented by a succession of phases that a system must pass through. It is a probabilistic model that can approximate a variety of distributions. In this context, it refers to the time required to serve a customer.

Choudhary et al. [16] presented two models for *MAP/PH/1* queues. These stable models are analyzed and compared using standard matrix-analytic approaches to understand their differences. The numerical results highlight how important it is to integrate PH type service process and MAP arrival process when assessing call centre performance and planning capabilities, as highlighted by Dudina et al. [21]. Dudina et al. [22] also analysed a finite *MAP/PH/1* queue with feedback under a Markovian random environment. Ayyappan and Gowthami [2] presented a single server classical queue with MAP arrivals, PH-service, and various additional features. They establish the system's stability and explored its active period as well. Furthermore, Thakur et al. [39] examined a single-server queueing system with phase-type service and Markovian arrival process (*MAP/PH/1*), while also considering the impact of working vacations and breakdowns.

**Server Deterioration:** The phenomenon where the performance of the server degrades over time due to factors such as fatigue, wear and tear, or other forms of decline in efficiency is referred as server deterioration. This can lead to longer service times or increased failure rates.

In the real world, a number of factors, such as fatigue and equipment wear and tear, may lead to service degradation. Choudhary et al. [16] introduced deterioration via the use of a *MAP/PH/1* queue models to account for failures, repairs, and degradation. Ayyappan and Meena [3] proposed a queueing model: arrivals follow Markovian Arrival Processes (MAP), service rates degrade over  $K$  completions and are restored through repair. Moreover, Hu et al. [27] scrutinized an optimal scheduling of proactive care with patient deterioration and also with customer deterioration and improvement. Recently, Benvidi et al. [8] examined an emergency department in a queueing network setting with declining health in the aftermath of a tragedy.

**Recovery:** This describes the process where a deteriorated server regains its efficiency. This type of recovery can occur through maintenance, rest periods, or other interventions that restore the server's performance to its original or improved state. Rani et al. [34] provided insight into the integration of vacationing servers, setup, reboot, and recovery features to effectively address real-time scenarios in fault-tolerant systems. A queueing model with a non-reliable server and a threshold-based recovery mechanism was presented by Efrosinin and Semenova [23].

**Vacation:** In queueing theory, a vacation is a time period when the server is not accessible to provide service. This can be due to scheduled breaks, maintenance, or other reasons. During this time, no customers are served, and they may either wait or be directed to a backup server.

Queueing models that include interruptions in service delivery have been widely researched within the topic of queueing systems with vacations. Doshi [20] provided an overview of the decomposition findings and methodology for studying two vacation models in queueing systems. A single vacation with starting failure in a production industry was investigated by Karpagam et al. [29], utilizing a bulk queueing system. Chakravarthy [10] investigated a queueing model with multiple servers and non-renewal arrivals, wherein servers take breaks when they are idle.

Chakravarthy [11] addressed a of  $MAP/PH/1$  queue with discretionary secondary services and vacation. In addition, a simulation-based technique to comparing vacation models under different vacation regimes was presented by Chakravarthy [12]. Recently, a  $MAP/PH/1$  queue subject to under breakdown, instant feedback and server vacation was scrutinized by Ayyappan and Thilagavathy [4].

**Backup Server:** A secondary server that becomes active when the primary server is unavailable (e.g., during a vacation or a disaster). The backup server helps maintain service continuity and manage the queue when the primary server is not operational.

Recently, Chakravarthy et al. [13] investigated a queueing architecture with a secondary backup server to cover the primary server's absences (due to vacations and breakdown), which has several practical uses. A backup server's services might be considered as the primary server functioning during a vacation or breakdown time. Klimenok et al. [30] has introduced recent advancements in multi-server queueing systems. These leverage batch Markovian arrival processes (BMAP) and phase-type (PH) distributions to model stochastic behavior, while backup servers optimizing its performance.  $MAP/PH/1$  queueing system analysis under N-policy with catastrophic delay action, standby server subjected under working vacation, and vacation interruption was recently addressed by Ayyappan and Thilagavathy [5]. In addition, an examination of the  $MAP/PH_1, PH_2, PH_3/1$  queue with dual heterogeneous service, backup server, vacation, customer impatience was conducted lately by Ayyappan and Thilagavathy [6].

**Matrix Analytic Methods:** He [25] mentioned that matrix analytic methods(MAM), developed by Neuts in the 1970s, provide the basis for analyzing structured Markov processes. Over the last four decades, MAM has been widely used in fields such as computer science, telecommunications, and finance, allowing for exact modelling and performance assessment of complex systems, demonstrating its long-term importance.

The book by He [26] explored matrix analytic methods, phase type distributions, Markovian arrival process, and structured Markov chains, offering a probabilistic perspective crucial for engineering analysis, accompanied by comprehensive explanations and practical examples. Matrix analytic methods represent a success story, showcasing the enhancement of applied probability through the integration of digital computer technology, as extensively discussed in the research article by Latouche and Ramaswami [31]. Tian et al. [40] also demonstrated the effectiveness of the matrix analytic approach developed by Neuts for analyzing working vacation queues, while also establishing a theoretical framework based on these findings. Furthermore, books Chakaravarthy [14] and [15] thoroughly explain MAM concepts and elucidate the results through numerous worked-out examples in order to provide a comprehensive understanding of MAM and demonstrate its effectiveness.

**ANFIS:** Control systems, pattern forecasting, system identification, recognition, and time series analysis are applied across a range of fields that use a machine learning technique that combines neural networks and fuzzy logic. To address the intricacies of these systems, advanced methodologies such as the Adaptive Neuro Fuzzy Inference System (ANFIS) have been utilized. ANFIS effectively models nonlinear systems by integrating fuzzy logic with neural networks. It has been extensively utilized for classification, forecasting, regulation, and optimization applications, including transient analysis in

queueing systems. ANFIS, which was initially introduced by Jang [28], has made substantial contributions to queueing theory [[35], [17]]. This has allowed researchers to compare analytical formulas with ANFIS-generated numerical outcomes in order to improve their comprehension of the system.

ANFIS has several advantages, including its ability to learn from data adaptively, its interpretability due to fuzzy rule-based reasoning, and its capability to approximate complex nonlinear relationships. It effectively combines the learning power of neural networks with the transparency of fuzzy systems, making it useful for predictive modelling. However, ANFIS also has some limitations, such as high computational complexity when dealing with large datasets, sensitivity to parameter initialization, and potential overfitting if the number of fuzzy rules is not carefully controlled. ANFIS addresses uncertainty and imprecision by leveraging fuzzy logic principles. It maps input variables to fuzzy membership functions, which define linguistic rules to handle vagueness in data. Through training, the neural network component optimizes these rules to improve accuracy. This capability makes ANFIS particularly useful for datasets with noisy or incomplete information, as it provides a more robust approximation than purely deterministic models. While there are only a few studies addressing queueing issues with ANFIS [36], [37], [19], the single-server Markovian queue model with working vacation and multiple stage of service proposed by Ahuja and Jain [1] has been evaluated against ANFIS computational results. Additionally, Divya and Indhira [18] explored ANFIS computation for an unreliable Markovian queueing model under a hybrid vacation policy.

The above section provides a comprehensive literature review of *MAP/PH/1* queueing systems. From this review, it is evident that while significant research has been conducted on server deterioration, vacations, and backup servers with recovery mechanisms, there is a notable gap in the literature concerning the application of these concepts within the context of *MAP/PH/1* queueing model. In light of this, we extend the work of Hanukov and Shoval [24] by analyzing a *MAP/PH/1* queueing model with server deterioration and recovery under vacation and backup server to address this gap.

## 2. APPLICATION OF THE AFOREMENTIONED MODEL IN REAL LIFE:

A manufacturing plant assembles car engines, where various components are put together in sequential stages by specialized machines which can be modelled as a proposed queueing framework. The assembly station receives engine parts such cylinder heads, crankshafts, and pistons (customers). The time required to assemble the engine components (service) varies depending on the complexity of the tasks, the precision required, and potential issues that might arise (e.g., alignment, fitting parts). Here, parts are put together at a single workstation one at a time. The machine may get worn down with time (server deterioration), which might result in slower operating rates, less accuracy, or a higher chance of mistakes. In a similar vein, prolonged shifts may wear out human operators. Furthermore, the machine's efficiency is restored (recovery) via planned maintenance (such as lubrication and component replacement). In addition, planned breaks and shift changes enable human operators to recuperate from exhaustion and continue working efficiently. Additionally, scheduled pauses for the assembly line or human workers, such lunch breaks, regular maintenance, and shift changes acts as a vacation period. There is a backup assembly station available. When the operator is taking a break or when the primary machine needs repair, this backup may be turned on. When the principal operator is taking a break, a relief operator may also take over to guarantee uninterrupted operations. Therefore, by putting this queueing model into practice, the car manufacturing facility may optimize its engine assembly line, increasing productivity, maintaining high standards of quality, and cutting expenses associated with operations.

## 3. MODEL DESCRIPTION

We examine the  $MAP/PH/1$  queueing model with server recovery and deterioration, in which the efficiency of server's level deteriorates after each service. The server provides service to  $K$  customers and then server lose efficiency to provide the service. Server leaves the system and takes for recovery period (vacation) irrespective of the number of customers present in the system. The vacation time follows an exponential distribution with rate  $\alpha$ . The server's efficiency level improve during the vacation, meanwhile the primary server is swapped out for a backup server. The service time of backup server follows an exponentially distributed with rate  $\beta$ . If the system is empty when the primary server returns from vacation, it becomes idle. When the primary server returns from vacation and discovers that a customer is being served by the secondary server, it waits inside the system until the service is finished before switching out the backup server (Fig. 1).

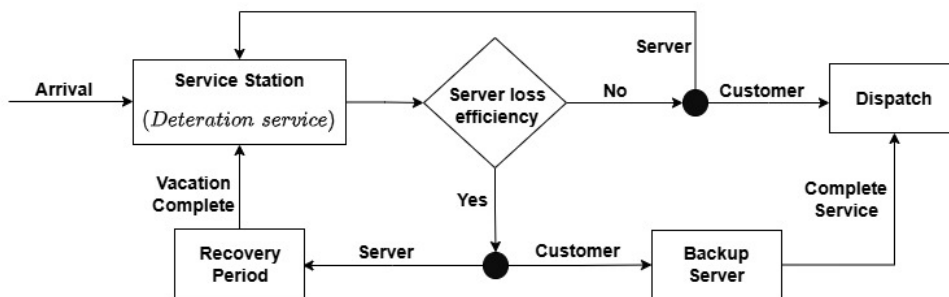


FIGURE 1. Schematic representation

- MAP is a broad category of point processes that includes processes that are known like as Poisson, Markov-modulated Poisson processes and PH-renewal processes.
- With irreducible generator  $D = D_0 + D_1$ , let  $\vartheta$  be the stationary probability vector of the Markov process. The unique (positive) probability vector that satisfies this condition is  $\vartheta$  with

$$\vartheta D = 0, \quad \vartheta e = 1.$$

- Demonstrate that the stationary form of a Markovian arrival process(MAP) has an arrival rate  $\lambda = \vartheta D_1 e$ , which is the expected number of arrivals per unit time.
- A phase type distribution with representation  $(\gamma, T_i)$  of order  $n_i$ ,  $i = 1, 2, \dots, K$  governs the service times. The average service rate,  $\mu_i$ , is given by  $\mu_i = [\gamma(-T_i)^{-1}e]^{-1}$ .
- The service rate degrades after completion of each service. Let the initial service rate be  $\mu_1$  and the service rate of the  $i^{th}$  service be  $\mu_i$  such that  $\mu_1 \geq \mu_2 \geq \mu_3 \geq \dots \geq \mu_K$ . The degraded service rate is restored to the original rate after completion of recovery period.

We need to define a few notations:

- $N(t)$  is the number of customers in the system;
- $Y(t)$  is the type of service, if any;
- $S(t)$  is the phase of service, if any;
- $A(t)$  is the phase of arrival process.

Therefore  $\{N(t), Y(t), S(t), A(t); t \geq 0\}$  forms a Markov process on state space

$$\Omega = \{(0, j, a) : 1 \leq j \leq K + 1, 1 \leq a \leq m\} \\ \cup \{(i, j, s_j, a) : i \geq 1, 1 \leq j \leq K + 2, 1 \leq j \leq n_j; 1 \leq a \leq m\}.$$

The level dependent quasi birth-and-death (*LDQBD*) structure of the infinitesimal generator matrix  $Q$  is as follows.

$$Q = \begin{bmatrix} B_1 & B_0 & 0 & 0 & 0 & 0 & \dots \\ B_2 & A_0 & A_1 & 0 & 0 & 0 & \dots \\ 0 & A_2 & A_0 & A_1 & 0 & 0 & \dots \\ 0 & 0 & A_2 & A_0 & A_1 & 0 & \dots \\ \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \dots \end{bmatrix}.$$

The block matrices

$$B_1 = \begin{bmatrix} D_0 & 0 & \dots & 0 & 0 \\ 0 & D_0 & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & D_0 & 0 \\ \alpha I & 0 & \dots & 0 & D_0 - \alpha I \end{bmatrix},$$

$$B_0 = \begin{bmatrix} \gamma \otimes D_1 & 0 & \dots & 0 & 0 & 0 \\ 0 & \gamma \otimes D_1 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & \gamma \otimes D_1 & 0 & 0 \\ 0 & 0 & \dots & 0 & D_1 & 0 \end{bmatrix},$$

$$B_2 = \begin{bmatrix} 0 & T_1^0 \otimes I & 0 & \dots & 0 \\ 0 & 0 & T_2^0 \otimes I & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & T_n^0 \otimes I \\ 0 & 0 & 0 & \dots & \beta I \\ \beta I & 0 & 0 & \dots & 0 \end{bmatrix},$$

$$A_1 = \begin{bmatrix} T_1 \oplus D_0 & 0 & \dots & 0 & 0 & 0 \\ 0 & T_2 \oplus D_0 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & T_n \oplus D_0 & 0 & 0 \\ 0 & 0 & \dots & 0 & D_0 - \alpha I - \beta I & \alpha I \\ 0 & 0 & \dots & 0 & 0 & D_0 - \beta I \end{bmatrix},$$

$$A_0 = \begin{bmatrix} I \otimes D_1 & 0 & \dots & 0 & 0 & 0 \\ 0 & I \otimes D_1 & \dots & 0 & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & I \otimes D_1 & 0 & 0 \\ 0 & 0 & \dots & 0 & D_1 & 0 \\ 0 & 0 & \dots & 0 & 0 & D_1 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 0 & T_1^0 \gamma \otimes I & 0 & \dots & 0 & 0 \\ 0 & 0 & T_2^0 \gamma \otimes I & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \dots & T_n^0 \gamma \otimes I & 0 \\ 0 & 0 & 0 & \dots & \beta I & 0 \\ \beta(\gamma \otimes I) & 0 & 0 & \dots & 0 & 0 \end{bmatrix}.$$

#### 4. THE STEADY STATE PROBABILITY VECTOR

Defining  $A = A_0 + A_1 + A_2$  and  $\delta$  to represent the steady-state probability vector of matrix  $A$  (irreducible matrix) and to demonstrate that the vector  $\pi$ :

$$\pi A = 0, \quad \pi e = 1.$$

The stability criterion  $\pi A_0 e < \pi A_2 e$  follows by Neuts[33].

Let  $\mathbf{x}$  be the steady-state probability vector of the given infinitesimal generator matrix  $Q$ . That is  $\mathbf{x}$  satisfies

$$\mathbf{x}Q = 0, \quad \mathbf{x}e = 1.$$

Partitioning this vector  $\mathbf{x} = (\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots)$ . The steady-state probabilities of the queueing system are calculated by

$$\mathbf{x}_i = \mathbf{x}_1 R^{i-1}, \quad i \geq 1.$$

where  $R$  is the minimal non-negative solution to the matrix-quadratic equation:

$$R^2 A_2 + R A_1 + A_0 = 0.$$

The vectors  $\mathbf{x}_0, \mathbf{x}_1, \dots$  are obtained by solving

$$\mathbf{x}_0 B_1 + \mathbf{x}_1 B_2 = 0,$$

$$\mathbf{x}_0 B_0 + \mathbf{x}_1 (A_1 + R A_2) = 0,$$

subject to the normalizing condition

$$\mathbf{x}_0 + \mathbf{x}_1 (I - R)^{-1} e = 1.$$

The computation of the  $R$  matrix can be carried out using the method of "Logarithmic Reduction Algorithm" as stated below:

##### Logarithmic Reduction Algorithm

$$\text{Step 0: } H \leftarrow (-A_1)^{-1} A_0, \quad L \leftarrow (-A_1)^{-1} A_2$$

$$G \leftarrow L \quad T \leftarrow H$$

$$\text{Step 1: } U \leftarrow HL + LH$$

$$M \leftarrow H^2$$

$$H \leftarrow (I - U)^{-1} M$$

$$M \leftarrow L^2$$

$$L \leftarrow (I - U)^{-1} M$$

$$G \leftarrow G + TL$$

$$T \leftarrow TH$$

$$\text{Continue Step 1 until } \|e - Ge\|_\infty < \epsilon$$

$$\text{Step 2: } R = -A_0(A_1 + A_0 G)^{-1}.$$

## 5. PERFORMANCE MEASURES

- The probability of the primary server being on vacation:

$$P_{VAC} = \mathbf{x}_0 e_1 + \mathbf{x}_1 (I - R)^{-1} e_2.$$

- The probability of the main server being available in the system.:

$$P_{AV} = \mathbf{x}_1 (I - R)^{-1} e_3.$$

- The mean number of customers in the system:

$$\mu_{NS} = \mathbf{x}_1 (I - R)^{-2} e.$$

- The expected number of customers in the queue:

$$\mu_{NQ} = \mu_{NS} - (1 - \mathbf{x}_0 e).$$

- The expected waiting time of the customer in the system

$$\mu_{WS} = \frac{\mu_{NS}}{\lambda}.$$

## 6. NUMERICAL AND GRAPHICAL RESULT

In this section, we explore the performance of the proposed system through detailed numerical and graphical demonstrations. By analyzing a series of scenarios, we demonstrate the effectiveness and robustness of our approach. These examples provide insights into the behavior of the system under various conditions and highlight the key performance metrics that are central to our analysis.

Tables 1 through 4 illustrate the impact of the vacation rate across various system parameters. The following four different *MAP* representations, all of which have the same mean, say 1, are taken into consideration for the arrival process.

**Exponential:**

$$D_0 = [-1], \quad D_1 = [1].$$

**Erlang of order 2:**

$$D_0 = \begin{bmatrix} -2 & 2 \\ 0 & -2 \end{bmatrix}, \quad D_1 = \begin{bmatrix} 0 & 0 \\ 2 & 0 \end{bmatrix}.$$

**Hyper-exponential:**

$$D_0 = \begin{bmatrix} -1.90 & 0 \\ 0 & -0.19 \end{bmatrix}, \quad D_1 = \begin{bmatrix} 1.710 & 0.190 \\ 0.171 & 0.019 \end{bmatrix}.$$

**Negative Correlation:**

$$D_0 = \begin{bmatrix} -1.00222 & 1.00222 & 0 \\ 0 & -1.00222 & 0 \\ 0 & 0 & -225.75 \end{bmatrix}, \quad D_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0.01002 & 0 & 0.99220 \\ 223.4925 & 0 & 2.2575 \end{bmatrix}.$$

Since, all these three arrival process are renewal, their correlation is zero.

Consider the following three phase type distribution for the service times.

**Exponential:**

$$\alpha = [1], \quad T = [-1].$$

**Erlang of order 2:**

$$\alpha = [1, 0], \quad T = \begin{bmatrix} -2 & 2 \\ 0 & -2 \end{bmatrix}.$$

**Hyper-exponential:**

$$\alpha = [0.8, 0.2], \quad T = \begin{bmatrix} -2.8 & 0 \\ 0 & -0.28 \end{bmatrix}.$$

**Illustration 1.** In Table 1, the influence of the vacation rate is depicted for three distinct cases:  $M/M/1$ ,  $M/E/1$ , and  $M/H/1$ . These cases represent systems with Markovian arrival processes and three different service time distributions, allowing for a comprehensive comparison of the effects under different service dynamics. Here, we fix the values,  $\lambda = 0.1$ ,  $\mu_1 = 2$ ,  $\mu_2 = 1.5$ ,  $\mu_3 = 1$ ,  $\mu_4 = 0.5$ ,  $\beta = 0.4$ ,  $\alpha = 1, 3, 5, 7, 9$ .

As evident from Table 1, increasing the vacation rate leads to a decrease in both the probability that the main server is on vacation ( $P_{VAC}$ ) and the mean number of customers in the system ( $\mu_{NS}$ ). Conversely, the probability that the main server is available in the system ( $P_{AV}$ ) increases across all three cases:  $M/M/1$ ,  $M/E/1$ , and  $M/H/1$ .

Comparing the values across the  $M/M/1$ ,  $M/E/1$ , and  $M/H/1$  systems, it is observed that the  $M/H/1$  system has the highest mean number of customers in the system, indicating a higher load due to the variability in service times. The  $M/E/1$  system, however, exhibits the highest probability that the main server is on vacation, suggesting that this system experiences more frequent server downtime. Interestingly, the  $M/E/1$  system also has the highest probability that the main server is available in the system, highlighting its ability to quickly recover and serve customers despite the frequent vacations. This scenario implies that while the  $M/H/1$  system struggles with higher customer numbers due to service time variability, the  $M/E/1$  system, despite having more frequent server vacations, maintains a balance by ensuring the server's availability most of the time.

$\alpha$	$M/M/1$			$M/E/1$			$M/H/1$		
	$P_{VAC}$	$P_{AV}$	$\mu_{NS}$	$P_{VAC}$	$P_{AV}$	$\mu_{NS}$	$P_{VAC}$	$P_{AV}$	$\mu_{NS}$
1	0.04028	0.12280	0.14872	0.04030	0.12282	0.14813	0.04021	0.12266	0.15261
3	0.03175	0.12374	0.14744	0.03176	0.12376	0.14684	0.03171	0.12360	0.15136
5	0.02997	0.12393	0.14718	0.02997	0.12395	0.14659	0.02993	0.12379	0.15111
7	0.02919	0.12401	0.14707	0.02919	0.12403	0.14648	0.02915	0.12387	0.15100
9	0.02875	0.12405	0.14701	0.02876	0.12407	0.14641	0.02872	0.12392	0.15094

TABLE 1. Impact of vacation rate ( $\alpha$ ) with various performance measures.

**Illustration 2.** In Table 2, the influence of the vacation rate is depicted for three distinct cases:  $E/M/1$ ,  $E/E/1$ , and  $E/H/1$ . These cases represent systems with Markovian arrival processes and three different service time distributions, allowing for a comprehensive comparison of the effects under different service dynamics. Here, we fix the values,  $\lambda = 0.1$ ,  $\mu_1 = 2$ ,  $\mu_2 = 1.5$ ,  $\mu_3 = 1$ ,  $\mu_4 = 0.5$ ,  $\beta = 0.4$ ,  $\alpha = 1, 3, 5, 7, 9$ .

As shown in table 2, an increase in the vacation rate results in a reduction in both the probability that the main server is on vacation ( $P_{VAC}$ ) and the mean number of customers in the system ( $\mu_{NS}$ ). On

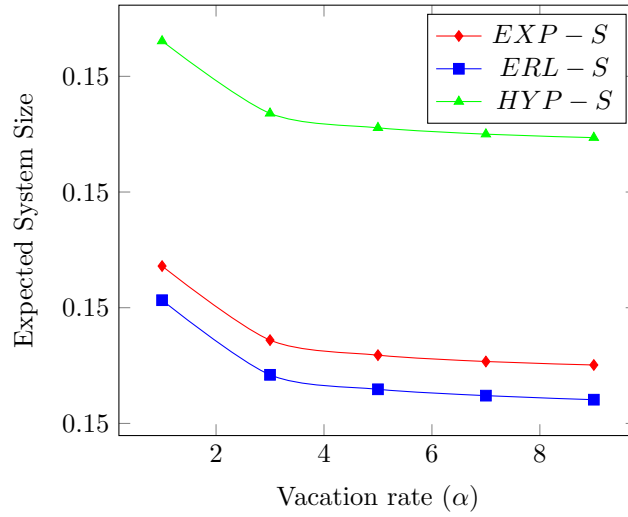


FIGURE 2. Poisson Arrival - Expected system size verses vacation rate  $\alpha$ .

the other hand, the probability that the main server is available ( $P_{AV}$ ) rises across all three scenarios:  $E/M/1$ ,  $E/E/1$ , and  $E/H/1$ .

$\alpha$	$E/M/1$			$E/E/1$			$E/H/1$		
	$P_{VAC}$	$P_{AV}$	$\mu_{NS}$	$P_{VAC}$	$P_{AV}$	$\mu_{NS}$	$P_{VAC}$	$P_{AV}$	$\mu_{NS}$
1	0.04672	0.14244	0.17835	0.04673	0.14245	0.17812	0.04667	0.14233	0.18088
3	0.03689	0.14370	0.17743	0.03690	0.14371	0.17720	0.03686	0.14359	0.17998
5	0.03482	0.14395	0.17725	0.03483	0.14396	0.17701	0.03479	0.14384	0.17980
7	0.03393	0.14406	0.17717	0.03393	0.14407	0.17693	0.03389	0.14395	0.17973
9	0.03342	0.14412	0.17712	0.03342	0.14413	0.17689	0.03339	0.14401	0.17968

TABLE 2. Impact of vacation rate ( $\alpha$ ) with various performance measures.

**Illustration 3.** In Table 3, the influence of the vacation rate is depicted for three distinct cases:  $H/M/1$ ,  $H/E/1$ , and  $H/H/1$ . These cases represent systems with Markovian arrival processes and three different service time distributions, allowing for a comprehensive comparison of the effects under different service dynamics. Here, we fix the values,  $\lambda = 0.1$ ,  $\mu_1 = 2$ ,  $\mu_2 = 1.5$ ,  $\mu_3 = 1$ ,  $\mu_4 = 0.5$ ,  $\beta = 0.4$ ,  $\alpha = 1,3,5,7,9$ .

As shown in Table 3, increasing the vacation rate results in a reduction in both the probability that the main server is on vacation ( $P_{VAC}$ ) and the mean number of customers in the system ( $\mu_{NS}$ ). In contrast, the probability that the main server is available ( $P_{AV}$ ) rises across all three scenarios:  $H/M/1$ ,  $H/E/1$ , and  $H/H/1$ .

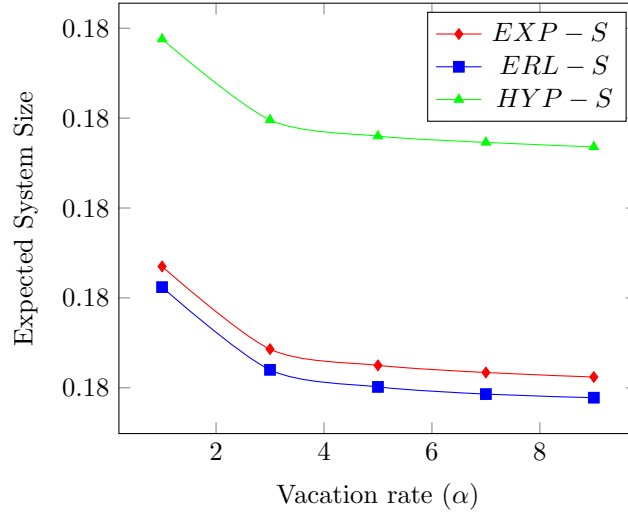


FIGURE 3. Erlang Arrival - Expected system size verses vacation rate  $\alpha$

$\alpha$	$H/M/1$			$H/E/1$			$H/H/1$		
	$P_{VAC}$	$P_{AV}$	$\mu_{NS}$	$P_{VAC}$	$P_{AV}$	$\mu_{NS}$	$P_{VAC}$	$P_{AV}$	$\mu_{NS}$
1	0.01859	0.05664	0.06425	0.01859	0.05665	0.06376	0.01856	0.05660	0.06738
3	0.01459	0.05684	0.06319	0.01459	0.05685	0.06270	0.01457	0.05680	0.06633
5	0.01375	0.05688	0.06298	0.01376	0.05689	0.06249	0.01374	0.05684	0.06613
7	0.01339	0.05690	0.06289	0.01340	0.05691	0.06240	0.01338	0.05686	0.06604
9	0.01319	0.05691	0.06284	0.01319	0.05692	0.06235	0.01318	0.05687	0.06599

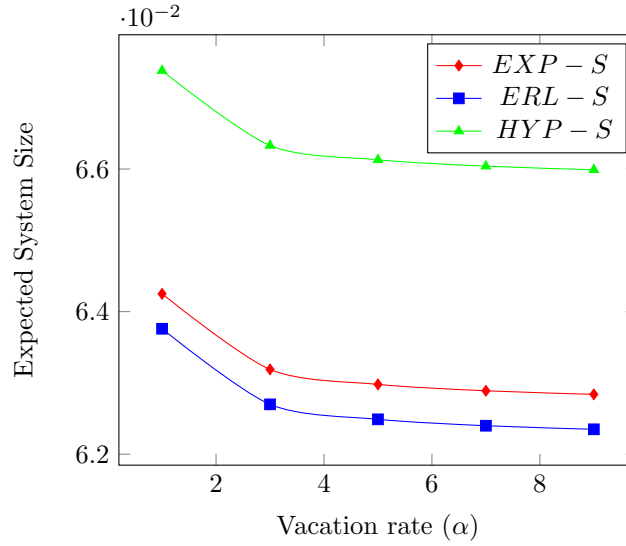
TABLE 3. Impact of vacation rate ( $\alpha$ ) with various performance measures.

**Illustration 4.** In Table 4, the influence of the vacation rate is depicted for three distinct cases:  $NC/M/1$ ,  $NC/E/1$ , and  $NC/H/1$ . These cases represent systems with Markovian arrival processes and three different service time distributions, allowing for a comprehensive comparison of the effects under different service dynamics. Here, we fix the values,  $\lambda = 0.1$ ,  $\mu_1 = 2$ ,  $\mu_2 = 1.5$ ,  $\mu_3 = 1$ ,  $\mu_4 = 0.5$ ,  $\beta = 0.4$ ,  $\alpha = 1,3,5,7,9$ .

As shown in Table 2, an increase in the vacation rate results in a reduction of both the probability that the main server is on vacation ( $P_{VAC}$ ) and the average number of customers in the system ( $\mu_{NS}$ ). On the other hand, the probability that the main server is available ( $P_{AV}$ ) increases across all three cases:  $NC/M/1$ ,  $NC/E/1$ , and  $NC/H/1$ .

### 7. ANFIS

ANFIS (Adaptive Neuro-Fuzzy Inference System) is a hybrid system that combines the strengths of fuzzy logic and neural networks. It leverages the human-like reasoning of fuzzy logic and the learning capabilities of neural networks to model complex systems with nonlinearities. ANFIS is based on a

FIGURE 4. Hyperexponential Arrival - Expected system size verses vacation rate  $\alpha$ .

$\alpha$	NC/M/1			NC/E/1			NC/H/1		
	$P_{VAC}$	$P_{AV}$	$\mu_{NS}$	$P_{VAC}$	$P_{AV}$	$\mu_{NS}$	$P_{VAC}$	$P_{AV}$	$\mu_{NS}$
1	0.02652	0.13836	0.25285	0.02652	0.13837	0.25254	0.02651	0.13830	0.25493
3	0.01433	0.13961	0.25142	0.01433	0.13962	0.25111	0.01433	0.13955	0.25353
5	0.01156	0.13987	0.25112	0.01156	0.13988	0.25081	0.01156	0.13981	0.25323
7	0.01032	0.13998	0.25099	0.01032	0.13999	0.25068	0.01032	0.13992	0.25310
9	0.00962	0.14005	0.25092	0.00961	0.14006	0.25061	0.00962	0.13998	0.25302

TABLE 4. Impact of Vacation rate ( $\alpha$ ) with various performance measures.

Sugeno-type fuzzy inference system. FIS uses a set of if-then rules with fuzzy logic to model complex systems by approximating nonlinear functions. The fuzzy rules allow for human-like reasoning, handling uncertainty and imprecision. The neural network aspect of ANFIS provides the system with learning capabilities. By adjusting membership functions through training, the system automatically fine-tunes the fuzzy rules based on input-output data. This process is akin to how neural networks adapt their weights during training.

ANFIS operates by taking inputs and passing them through fuzzy layers, where they are fuzzified into membership functions. These fuzzified inputs are then processed using fuzzy rules in a network-like structure shown in Figure 6. The system adjusts its parameters (like neural networks adjust weights) to minimize the difference between predicted and actual outputs during training, enhancing the model's accuracy.

Three main phases are involved in ANFIS modeling. In the process of building a model, fuzzy, inference engine, and defuzzification are essential steps. A nonlinear mapping between the inputs and

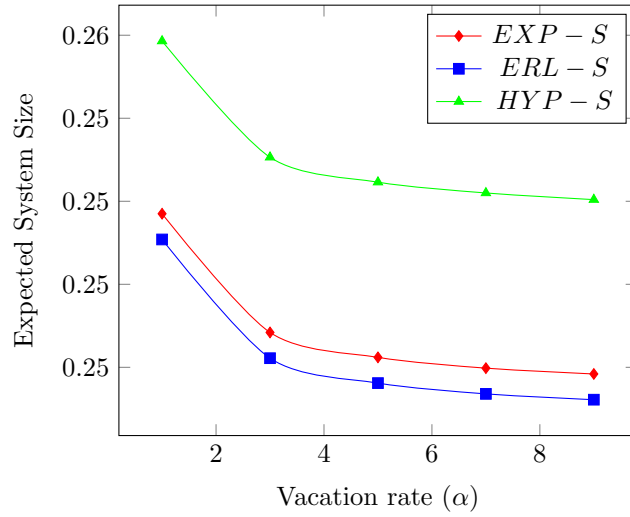


FIGURE 5. MNC - Arrival - Expected system size versus vacation rate  $\alpha$ .

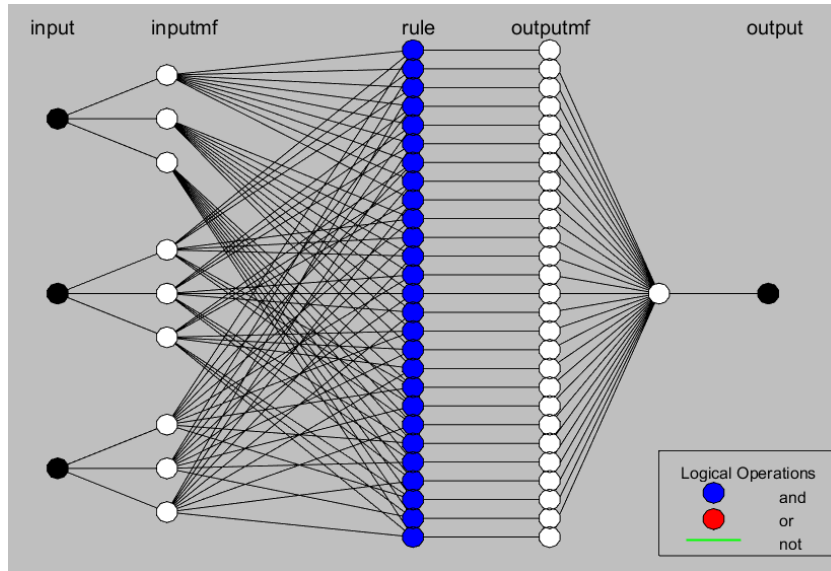


FIGURE 6. ANFIS model Structure

outputs is produced during the fuzzification process using membership functions. To produce the final output, fuzzy rules are created during the inference engine stage. The output is transformed into crisp output during the defuzzification process. ANFIS uses two methods to develop the rules: subtractive clustering and grid partitioning (see Table 5).

The model is trained using a three-input, single-output Sugeno model structure with three fuzzy rules. The tuned Gaussian-shaped membership function that results is taken into account. Figure 7 shows the adjusted Gaussian shape membership functions of the input variables that have been considered.

TABLE 5. Membership function values associated with linguistic terms for the input parameters.

Input parameters	No. of membership function	Linguistic Values
$\alpha, P_{VAC}, P_{AV}, \mu_{NS}$	3	Low Medium High

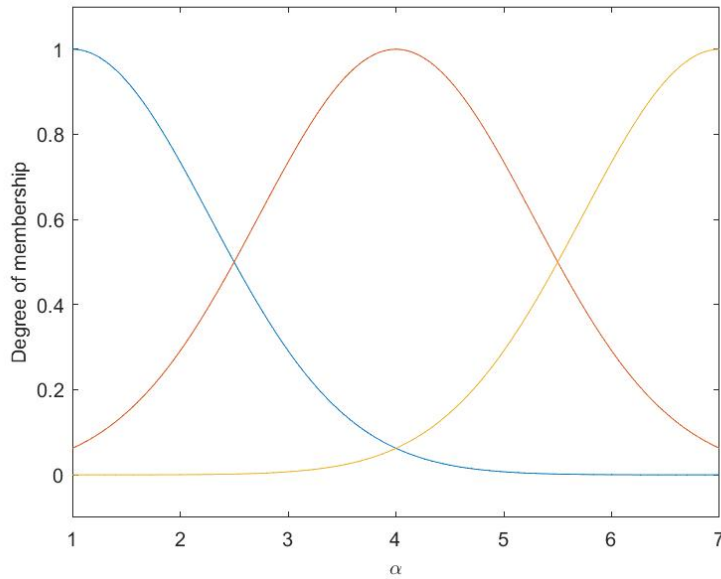


FIGURE 7. ANFIS membership function

**ANFIS based results.** In this paper, the Adaptive Neuro-Fuzzy Inference System (ANFIS) method and the numerical results derived from analytical equations are compared. MATLAB software's evalfis function is used to acquire the ANFIS findings for the system's default parameter settings, as explained in Section 6. The following formula determines the absolute percentage mistake ( $\Pi_e$ ):

$$\Pi_e = \frac{|\mu_{NS} - \mu_{NS}^*|}{\mu_{NS}} \times 100\%.$$

Tables 6-7 summarize the absolute percentage errors ( $\Pi_e$ ) and the accuracy of the estimated value in percentage of  $\Pi_e$ . The lower  $\Pi_e$  value implies that our ANFIS technique is closer to the analytical method's conclusions. The tick points in Fig. 8 (a-d) reflect the ANFIS findings for the  $\mu_{NS}$  estimate, while the continuous lines show the analytical results. In these photographs, check marks almost fully cover the curving lines. This shows that both results are successful.

## 8. CONCLUSION

We investigated *MAP/PH/1* queueing system with server deterioration and recovery under vacation and backup server. The server leaves for a vacation due to the loss of server efficiency. During the vacation, a backup server provides the service to the customers. Using MAM, We determined the

TABLE 6. Values of the  $\Pi_e$ ,  $\mu_{NS}$ ,  $\mu_{NS}^*$  by varying input parameters  $\alpha$ ,  $P_{VAC}$ ,  $P_{AV}$ 

$M/M/1$			$M/E/1$			$M/H/1$		
$\alpha$	$\mu_{NS}$	$\Pi_e$	$\alpha$	$\mu_{NS}$	$\Pi_e$	$\alpha$	$\mu_{NS}$	$\Pi_e$
1	0.14872	0.148719	1	0.14813	0.148129	1	0.15261	0.152609
3	0.14744	0.147438	3	0.14684	0.146839	3	0.15136	0.151359
5	0.14718	0.147180	5	0.14659	0.146589	5	0.15111	0.151109
7	0.14707	0.147069	7	0.14648	0.150922	7	0.151	0.15559
9	0.14701	0.152514	9	0.14641	0.153494	9	0.15094	0.158497
Average of $\Pi_e$			Average of $\Pi_e$			Average of $\Pi_e$		
Accuracy in predicted value			Accuracy in predicted value			Accuracy in predicted value		
99.962542			99.921277			99.919504		
$E/M/1$			$E/E/1$			$E/H/1$		
$\alpha$	$\mu_{NS}$	$\Pi_e$	$\alpha$	$\mu_{NS}$	$\Pi_e$	$\alpha$	$\mu_{NS}$	$\Pi_e$
1	0.17835	0.178349	1	0.17812	0.178119	1	0.18088	0.18087
3	0.17743	0.177429	3	0.1772	0.177199	3	0.17998	0.179979
5	0.17725	0.177249	5	0.17701	0.177009	5	0.1798	0.179799
7	0.17717	0.17719	7	0.17693	0.176887	7	0.17973	0.179667
9	0.17712	0.177173	9	0.17689	0.176746	9	0.17968	0.179550
Average of $\Pi_e$			Average of $\Pi_e$			Average of $\Pi_e$		
Accuracy in predicted value			Accuracy in predicted value			Accuracy in predicted value		
99.999549			99.998943			99.998927		

TABLE 7. Values of the  $\Pi_e$ ,  $\mu_{NS}$ ,  $\mu_{NS}^*$  by varying input parameters  $\alpha$ ,  $P_{VAC}$ ,  $P_{AV}$

$H/M/1$			$H/E/1$			$H/H/1$		
$\alpha$	$\mu_{NS}$	$\Pi_e$	$\alpha$	$\mu_{NS}$	$\Pi_e$	$\alpha$	$\mu_{NS}$	$\Pi_e$
1	0.06425	0.064249	1	0.06376	0.063759	1	0.06738	0.067379
3	0.06319	0.063189	3	0.0627	0.062699	3	0.06633	0.06632
5	0.06298	0.06298	5	0.06249	0.062489	5	0.06613	0.06613
7	0.06289	0.06288	7	0.0624	0.062531	7	0.06604	0.06603
9	0.06284	0.06284	9	0.06235	0.06565	9	0.06599	0.06599
Average of $\Pi_e$			Average of $\Pi_e$			Average of $\Pi_e$		
0.000023			0.055125			0.000002		
Accuracy in			Accuracy in			Accuracy in		
predicted value			predicted value			predicted value		
99.999976			99.944874			99.999999		
$NC/M/1$			$NC/E/1$			$NC/H/1$		
$\alpha$	$\mu_{NS}$	$\Pi_e$	$\alpha$	$\mu_{NS}$	$\Pi_e$	$\alpha$	$\mu_{NS}$	$\Pi_e$
1	0.25285	0.252849	1	0.25254	0.252539	1	0.25493	0.254929
3	0.25142	0.251418	3	0.25111	0.251108	3	0.25353	0.253528
5	0.25112	0.251119	5	0.25081	0.250809	5	0.25323	0.253229
7	0.25099	0.250989	7	0.25068	0.250678	7	0.2531	0.253099
9	0.25092	0.250919	9	0.25061	0.250609	9	0.25302	0.253019
Average of $\Pi_e$			Average of $\Pi_e$			Average of $\Pi_e$		
0.000016			0.000015			0.000015		
Accuracy in			Accuracy in			Accuracy in		
predicted value			predicted value			predicted value		
99.99998			99.99998			99.99998		

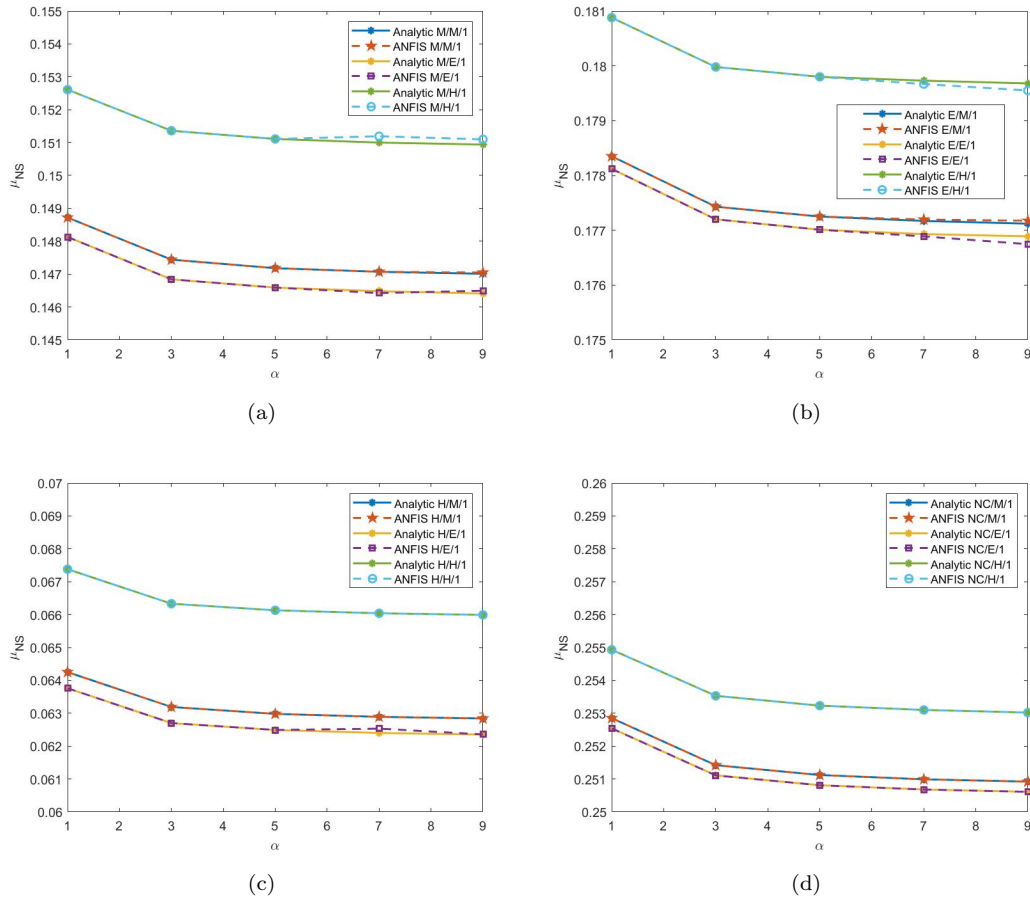


FIGURE 8. Effects of analytical and ANFIS results on  $\mu_{NS}$  with respect to different  $N$  values

steady state probability and derived several significant model performance metrics. The results achieved through ANFIS were compared to those obtained with the Matrix Geometric Method (MGM) to assess the Neuro-Fuzzy approach’s accuracy. This comparison was conducted to determine how effectively ANFIS can replicate or improve upon the accuracy of results typically provided by the established MGM technique. Numerical and graphical examples are provided. Further study of our, the busy period analysis and waiting time distribution of this model.

## REFERENCES

- [1] A. Ahuja, A. Jain and M. Jain, *Transient analysis and ANFIS computing of unreliable single server queueing model with multiple stage service and functioning vacation*, Mathematics and Computers in Simulation, **192** (2022), 464–490.
- [2] G. Ayyappan and R. Gowthami, *A MAP/PH/1 queue with setup time, Bernoulli vacation, reneging, balking, Bernoulli feedback, breakdown and repair*, Reliability: Theory and Applications, **16(2)**(2021), 191–221.
- [3] G. Ayyappan and S. Meena, *Phase type queueing model of server vacation, repair and degrading service with breakdown, starting failure and close-down*, Reliability: Theory & Applications, **18(1)**(2023), 464–483.
- [4] G. Ayyappan and G. Thilagavathy *Analysis of MAP/PH/1 queueing model with breakdown, instantaneous feedback and server vacation*, Applications and Applied Mathematics: An International Journal (AAM), **15(2)**(2020), 1–18.
- [5] G. Ayyappan and K. Thilagavathy *Analysis of MAP/PH/1 queueing system with catastrophic delay action, standby server, balking, working vacation and vacation interruption under N-policy*, International Journal of Mathematical Modelling and Numerical Optimisation, **13(3)**(2023), 223–243.
- [6] G. Ayyappan and K. Thilagavathy, K., *Analysis of MAP/PH(1), PH(2), PH(3)/1 queueing system with two modes of heterogeneous service, standby server, vacation, impatient behaviour of customers, additional service, start-up time, breakdown and phase type repairs*, International Journal of Operational Research, **49(1)**(2024), 85–130.
- [7] G. Basharin, V. Naumov and K. Samouylov, *On Markovian modelling of arrival processes*, Stat Papers, **59**(2018), 1533–1540.
- [8] S. Benvidi, T. Asghari, A. Aghsami and F. Jolai, *Analysing a queueing network of the emergency department with deteriorating health in post-disaster situations*, International Journal of Operational Research, **48(3)**(2023), 397–428.
- [9] S. R. Chakravorthy, *Markovian arrival process*, Wiley Encyclopedia of Operations Research and Management Science, John Wiley and Sons, 2011.
- [10] S. R. Chakravorthy, *Analysis of a multi-server queueing model with vacations and optional secondary services*, Mathematica Applicanda, **41**(2013), 127–151.
- [11] S. R. Chakravorthy, *Analysis of MAP/PH1, PH2/1 queue with vacations and optional secondary services*, Applied Mathematical Modelling, **37**(2013), 8886–8902.
- [12] S. R. Chakravorthy, *A comparative study of vacation models under various vacation policies—A simulation approach*. In Kulshrestha, R. et al. (Eds), *Mathematical Modeling and Computation of Real-time Problems: An Interdisciplinary Approach* (pp. 3–20). CRC Press, 2021.
- [13] S. R. Chakravorthy, G. Shruti and R. Kulshrestha, *A queueing model with server breakdowns, repairs, vacations and backup server*, Operations Research Perspectives, **7**(2020), 100–131.
- [14] S. R. Chakravorthy, *Introduction To Matrix-analytic Methods In Queues 1: Analytical And Simulation Approach - Basics*. Wiley and ISTE Ltd (2022).
- [15] S. R. Chakravorthy, *Introduction To Matrix-analytic Methods In Queues 2: Analytical And Simulation Approach - Queues and Simulation*. Wiley and ISTE Ltd. (2022).
- [16] A. Choudhary, S. R. Chakravorthy and D. C. Sharma, *Analysis of MAP/PH/1 queueing system with degrading service rate and phase type vacation*, Mathematics, **9(19)**(2021): 2387.
- [17] A. Dehmi, M. Boualem, S. Kahla and L. Berdjoudj, *ANFIS computing and cost optimization of an M/M/c/M queue with feedback and balking customers under a hybrid hiatus policy*, Croatian Operational Research Review, **15(2)**(2024), 159–170.
- [18] K. Divya and K. Indhira *Performance analysis and ANFIS computing of an unreliable Markovian feedback queueing model under a hybrid vacation policy*. Mathematics and Computers in Simulation, **218**(2024), 403–419.
- [19] K. Divya and K. Indhira, *Performance analysis and ANFIS computing of a Markovian queueing model with intermittently accessible server under a hybrid vacation policy*. RAIRO-Operations Research, **58(2)**(2024), 1257–1279.
- [20] B. T. Doshi, *Queueing systems with vacations—A survey*, Queueing Systems, **1**(1986), 29–66.
- [21] O. Dudina, C. Kim and S. Dudin, *Retrial queueing system with Markovian arrival flow and phase-type service time distribution*, Computers and Industrial Engineering, **66(2)**(2013), 360–373.
- [22] A. N. Dudin, A. V. Kazimirsky, V. I. Klimenok, L. Breuer and U. Krieger, *The queueing model MAP/PH/1/N with feedback operating in a Markovian random environment*, Austrian Journal of Statistics, **34(2)**(2005), 101–110.
- [23] D. Efrosinin and O. Semenova, *Queueing model with non-reliable server and threshold-based recovery*, In Conference Proceedings, MMR 2009, Moscow, pp. 546–550.
- [24] G. Hanukov and S. Shoval, *A Model for a Vacation Queueing Policy Considering Servers Deterioration and Recovery*, Mathematics, **11(12)**(2023), 2640–2651.

- [25] Q. M. He, *Matrix-Analytic Methods—An Algorithmic Approach to Stochastic Modelling and Analysis*, Optimization in Large Scale Problems: Industry 4.0 and Society 5.0 Applications (2019), pp. 41–45.
- [26] Q. M. He, *Fundamentals Of Matrix-Analytic Methods*, Springer, New York, 2014.
- [27] Y. Hu, C. W. Chan and J. Dong, *Optimal scheduling of proactive care with patient deterioration*, Management Science, **68**(4)(2019), 2533–2578.
- [28] J. S. R. Jang, *ANFIS: adaptive-network-based fuzzy inference system*, In IEEE Transactions on Systems, Man, and Cybernetics, **23**(3)(1993), 665–685.
- [29] S. Karpagam, G. Ayyappan and B. Somasundaram, *A bulk queueing system with rework in manufacturing industry with starting failure and single vacation*, International Journal of Applied and Computational Mathematics, **6**(6)(2020): 174.
- [30] V. I. Klimenok, A. N. Dudin and K. E. Samouylov, *Analysis of the BMAP/PH/N queueing system with backup servers*, Applied Mathematical Modelling, **57**(2018), 64–84.
- [31] G. Latouche and V. Ramaswami, *Introduction to matrix analytic methods in stochastic modeling*, ASA-SIAM Series on Statistics and Applied Probability, SIAM, Philadelphia, 1999.
- [32] M. F. Neuts, *A versatile Markovian point process*. Journal of Applied Probability, **16**(4)(1979), 764–779.
- [33] M. F. Neuts, *Matrix-Geometric Solutions in Stochastic Models*, Johns Hopkins series in the mathematical sciences, Johns Hopkins University Press, Baltimore, 1981.
- [34] S. Rani, M. Jain and R. K. Meena, *Queueing modeling and optimization of a fault-tolerant system with reboot, recovery, and vacationing server operating under admission control policy*, Mathematics and Computers in Simulation, **209**(2023), 408–425.
- [35] S. S. Sanga, *ANFIS computing and cost minimization of M/G/1/K fault-tolerant machining system with general startup under F-policy*, Physica A: Statistical Mechanics and its Applications, **657**(2025): 130219.
- [36] S. S. Sanga and M. Jain, *Cost optimization and ANFIS computing for admission control of M/M/1/K queue with general retrial times and discouragement*. Appl. Math. Comput., **363**(2020): 124624.
- [37] R. Sethi, R. Jain, R. K. Meena and D. Garg, *Cost optimization and ANFIS computing of an unreliable M/M/1 queueing system with customers impatience under N-policy*, Int. J. Appl. Comput. Math, **6**(2)(2020), 1-14.
- [38] S. Sindhu and A. Krishnamoorthy, *MAP/PH/1 queue with discarding customers having imperfect service*, 3c Empresa: Investigación Y Pensamiento Crítico, **11**(2)(2022), 116–137.
- [39] S. Thakur, A. Jain and A. Ahuja, *Analysis of MAP/PH/1 model with working vacation, working breakdown and two-phase repair*, Arabian Journal for Science and Engineering, **49**(3)(2023), 7431–7451.
- [40] N. Tian, J. Li and Z. G. Zhang, *Matrix-analytic method and working vacation queues-a survey*, International Journal of Information and Management Sciences, **20**(2009), 603–633.
- [41] N. Zhao, Y. Guo and Z. Lian and M. Wang, *Analysis of a MAP/PH/1 queue with discretionary priority*, In Proceedings of the 2015 Winter Simulation Conference (WSC), Huntington Beach, CA, USA, 6–9 December 2015, pp. 2872–2883.

B. SOMASUNDARAM, CORRESPONDING AUTHOR, DEPARTMENT OF MATHEMATICS, VEL TECH RANGARAJAN DR. SAGUNTHALA R&D INSTITUTE OF SCIENCE AND TECHNOLOGY, AVADI, CHENNAI, TAMIL NADU, INDIA

*Email address:* somu.b2@gmail.com

DEPARTMENT OF MATHEMATICS AND STATISTICS, SCHOOL OF APPLIED SCIENCES AND HUMANITIES, VIGNAN'S FOUNDATION FOR SCIENCE, TECHNOLOGY AND RESEARCH, VADLAMUDI, GUNTUR, ANDHRA PRADESH, INDIA

*Email address:* drdivyak.maths@gmail.com