



Original articles

N-policy for redundant machining system with double retrial orbits using soft computing techniques

Vijay Pratap Singh^a, Madhu Jain^{a,*}, Richa Sharma^b

^a Department of Mathematics, IIT Roorkee, Roorkee, Uttarakhand 247667, India

^b Department of Mathematics, JK Lakshmipat University, Jaipur 302026, India

ARTICLE INFO

Keywords:

Double retrial orbits
Redundant repairable system
ANFIS
N-policy
Imperfect service
Unreliable server
Metaheuristic optimization

ABSTRACT

The present study is concerned with the performance prediction of a double retrial orbit redundant repairable machining system. Both primary and secondary orbits are available as waiting/buffer space for the failed units. In these orbits, the failed units can reside and make re-attempts for the repair. As per N-policy, if there are no units in the orbits for the repairing job, the repairman goes on vacation and further starts the repair job when N-failed units are accumulated. The objective of this investigation is to evaluate the transient and steady-state distributions of the queue length of failed units under N-policy. The matrix analytic and matrix recursive methods are utilized for solution purpose while an adaptive neuro-fuzzy inference system (ANFIS) is employed for validating the feasibility of designing the AI-based controller. The harmonic search (HS) and particle swarm optimization (PSO) methods have been implemented for the cost optimization purpose so as to evaluate the optimal design parameters. The outputs of study provides critical insights into optimal system performance and improving the repair policy. Furthermore, a practical application of this investigation is demonstrated in a telecommunications network traffic system, where the proposed methods can be utilized to manage the maintenance issues of routers in the network traffic.

1. Introduction

The significance of machine repair model (MRM) lies in its ability to address the challenges and complexities, and performance issues of modern industrial systems operating in failure prone machining environment. By analyzing MRM redundancy, researchers can facilitate valuable insights into the system performance, maintenance strategies, and operational efficiency. Effective machine repair modeling facilitates the identification of optimal repair schedules, minimizing downtime and maximizing productivity. Additionally, analytical and numerical simulation results established via machine repair models enable the industries to proactively address maintenance planning, reliability, and efficiency upgradation, and enhance the fault tolerance measures and customer satisfaction. The motivation behind the study of machine repair models is to optimize the system performance and achieve sustainable production/service competitive advantages in today's dynamic industrial landscape operating in a machining environment.

Several fault tolerant machining systems (FTMS) require the effective control of the faults/failures and minimizing the waiting space, even in cases of imperfect repair. The retrial queueing modeling with vacationing server is a cost-effective way to enhance the FTMS's capacity and improvement in the future design. Under N-policy, the repairman starts the repair procedure once a certain

* Corresponding author.

E-mail addresses: vsingh3@ma.iitr.ac.in (V.P. Singh), madhu.jain@ma.iitr.ac.in (M. Jain), richasharma@jklu.edu.in (R. Sharma).

number of failed units (N) have been accumulated. This reduces the idle time by ensuring that the repairman only begins work when there are enough failed units available for the repair job in the system. To maintain the functionality of the system, the maintenance team first makes primary repairs. In case, if the primary repair is unsuccessful, the failed unit is immediately taken for the additional repair job. The failed units can also be sent to the secondary retrial orbit for the repair attempts.

The increasing complexity of machining system has created a need for the advanced techniques to develop the performance model in order to chalk out the maintainability of the system. The traditional methods often struggle with uncertainty, non-linearity, and imprecision present in the real-world machining environments. In this context, soft computing techniques such as Fuzzy Logic and Adaptive Neuro-Fuzzy Inference System (ANFIS) provide robust tools to handle these challenges effectively. Fuzzy Logic allows the model to deal with uncertainty and imprecision by mimicking human reasoning, making it highly suitable for the real-world problems. ANFIS, which combines the learning ability of neural networks with the reasoning capability of Fuzzy Logic, can be used for the efficient modeling of non-linear systems and improving the decision-making accuracy.

In this paper, we study the N -policy for the fault-tolerant machining system that incorporates many realistic features such as double retrial orbits, standbys, and unreliable server. The arrangement of the remaining contents of this article is as follows. Section 2 is dedicated to the literature review, research gap, and novelty of the research work. Section 3 provides preliminary overview of ANFIS and metaheuristic algorithms. Markov model along with requisite assumptions are described in Section 4. Section 5 is dedicated to the construction of Chapman Kolmogorov equations. In Section 6, various performance measures are established. The cost function is formulated in the next Section 7. Section 8 is devoted to the computational results, which are displayed in the graphs and tables for better visualization. Finally, Section 9 highlights the key features of the study. The significance and future scope of the present research work are also mentioned.

2. Literature review, research gap and contributions

The literature review can be helpful in identifying the gaps and relevance of the present work in many industries operating in the failure prone machining environment.

2.1. Literature review

This sub section presents important relevant existing literature on various topics related to the present study on MRM.

• MRM with redundancy

Redundant machining systems which are supported by standby units are used in various industries to regulate and maintain pre-specified availability. Standby support may be helpful to lower the downtime and ensure uninterrupted service due to component failure in the system. Fault-tolerant machining systems with redundancy are crucial for achieving the high performance and reliability in many sectors such as manufacturing, distribution, healthcare, telecommunications, distributed computing centers, and others. Wang & Sivazlian [1] analyzed the performance metrics of repairable machining system with spares. They incorporated the reboot and recovery mechanisms to ensure the system reliability and fault tolerance. A study on a finite Markov model for the machine repair system that integrates standby resources, and operational vacation was presented by Jain & Preeti [2]. An investigation on the standby machine repair problem with working breakdowns under Markov framework was studied by Liou [3]. Yang and Tsao [4] discussed the reliability and availability analysis of machine repair model involving standby support and working vacations. Castro and Dawabsha [5] investigated a Markovian machine repair problem (MRP) involving warm standby and preventive maintenance under multiple repairmen. The availability analysis of machine repair problem with warm standby in a shock environment was conducted by Juybari et al. [6].

• MRM with retrial orbit

Retrial queues are commonly noticed in various contexts including the banks, ATMs, call centers, telecommunication systems, railway ticket booking counters, and numerous other service-oriented environments. In the retrial queueing systems, when the customers come across a busy server, they have the option to avoid waiting in front of the service station by entering into the retrial orbit (RO). From the RO, the customers make re-attempts anticipating that the service facility will likely become available. Artalejo [7] provided a comprehensive overview and bibliography on the retrial queues. A retrial queue with vacation under server breakdown was examined by Jain et al. [8]. The investigation of the retrial queue with balking was carried out by Arivudainambi and Godhandaraman [9]. Jain and Sanga [10] studied fault-tolerant retrial systems under F-policy. They suggested a supplementary variable approach to investigate the concerned non-Markovian model. Markov model for retrial queue with vacationing server was studied by Ke et al. [11]. Yen et al. [12] facilitated a comprehensive study on the retrial machine repair model to investigate the F-policy. A retrial queueing model for machine repair system (MRS) with mixed spares subjected to breakdown pro-repairman was developed by Gao & Wang [13]. Liu et al. [14] considered the optimization issues related to the retrial model for MRS with unreliable server, reboot delay, and imperfect coverage. Kuo et al. [15] studied the performance and optimization issues of a retrial orbit non-Markovian queueing model and obtained the reliability indices using the supplementary variable technique. They addressed the inclusion of reboot delay and imperfect service, aiming to improve the system's overall performance and efficiency.

In a double-orbit retrial queueing system, when the server is busy, the customers have the facility to wait either in an ordinary orbit or a premium orbit. The customers can attempt for the service from the orbit; they make a subsequent effort to obtain the service till find the server free. Jain et al. [16] investigated double-orbit retrial queueing model under the assumption of the

unreliable server. Dimitriou [17] developed a Markovian queueing model that involves two types of retrial customers. Sanga & Jain [18] presented an investigation on the queueing and reliability indices of an FM/FM/1 queue with double orbits and facilitated optimal policy for the customers' joining strategy. The study of M/M/1 double orbit retrial queueing model involving the concepts of vacation and imperfect service was done by Dhibar & Jain [19]. Sanga & Jain [20] developed a queueing model with discouragement behavior by using the double orbit feature. They obtained both crisp and fuzzy reliability indices for the queueing system. Jain & Mehta [21] investigated a Markovian queueing model with double orbit and established the formulae for the queue length, throughput and profit function by using the probability generating function (PGF). A transient analysis of a fault-tolerant system operating in a machining environment and the provision of double orbits for keeping the failed machines while waiting for repair, was provided by Kumar et al. [22]. Recently, Li et al. [23] conducted a study on the reliability analysis and optimization design of a repairable K-out-of-N:G retrial system, addressing two failure modes and implementation of preventive maintenance.

- **MRM with imperfect repair**

Imperfect service in the machine repair problem illustrates how the primary repair job could not always return the failed machines to full operation. In certain machining systems, repair may not be successful in first attempt and further repair is required to get back the system into functional mode. Examining the system's availability is a crucial issue, especially when the machining system has undergone several imperfect repairs to make it again operating [24]. A double retrial queueing model having a vacationing server and imperfect service was studied by Dhibar & Jain [19]. The governing equations in this work were formulated and then analytically solved using a probability-generating function. Jain & Singh [25] examined a non-Markovian fault-tolerant degraded machine repair model emphasizing the imperfect repair and multi-type failures.

- **MRM with N-policy**

Optimal control strategies have been popular in the last several decades to regulate the machining system operating under a threshold-based vacation policy. N-policy can be used for the machine repair system functioning in a multi-component framework, facilitating efficient and economical system operation. N-policy states that the repair operations can begin after the accumulation of a certain threshold number (say N) of malfunctioning machines in the system. Yadin & Naor [26] presented an N-policy approach for the queueing system with a reliable server. Jain et al. [27] focused on the vacationing server queueing model operating under optimal control (N, F) policy wherein repair of failed units is done in multi-phases. A Markov queueing model including the vacationing server operating under the N-policy was investigated by Yang & Wu [28]. Chen & Wang [29] investigated the queue size distribution and performance indices of MRP wherein failed units wait in the retrial orbit in the case when repairman is busy. This system has provision of standby support, and the server operates under an N-policy. Meena et al. [30] studied an MRP with standby units and under the care of a server that operates under N-policy.

- **RMR with soft computing technique ANFIS**

Artificial neuro-fuzzy inference system (ANFIS) is an AI method that merges artificial neural networks (ANN) with fuzzy logic (FL). It can also be employed to analyze the queueing characteristics of FTMS by integrating the learning capabilities of ANN with the ability of fuzzy logic to handle imprecise information. Jang [31] made noteworthy contributions through his significant research efforts in the area of ANFIS. Bhargava & Jain [32] suggested an unreliable server queueing model with vacation using the application of ANFIS. This study aimed to examine the effectiveness and performance of the concerned queueing model of finite machine repair problem (MRP) via matrix approach and ANFIS. Jain & Meena [33] discussed a MRP having provision of mixed standbys. They employed Runge-Kutta method to acquire numerical outcomes for the system performance indices. Furthermore, ANFIS results have been provided to validate the feasibility of an ANFIS based controller. Sethi et al. [34] applied optimization and ANFIS approaches for the performance prediction of unreliable server queue operating under N-policy. Upadhyaya & Kushwaha [35] presented a bulk arrival retrial queueing model and obtained numerical results using the generating function and supplementary variable technique. Furthermore, they compared the numerical results obtained analytically with those obtained using ANIFS technique. ANFIS technique has been employed to study the M/M/1/N queueing model by Thakur et al. [36]. In this investigation, they used the matrix-geometric method to compute the queue length. The reliability prediction of automotive manufacturing system using ANFIS was done by Soltanali et al. [37]. The study of the queueing model with broken down server involving multistage services has been done by Ahuja et al. [38]. The performance metrics of the system were also computed using the ANFIS technique. Mabrook et al. [39] suggested an ANFIS algorithm to facilitate decision-making for the optimal allocation of occupied and available free channels in the communication network with precision. Rani et al. [40] investigated a double retrial Markov queue model with disaster and vacationing server. ANFIS-based computational results which are at par with analytical results are facilitated. Furthermore, golden section search was used for the cost optimization. Singh et al. [41] developed queueing model with two types of interruption in service and discouragement by using ANFIS approach.

- **Optimization using metaheuristic techniques**

Metaheuristic optimization techniques are powerful tools for solving the complex problems for which traditional optimization methods cannot be easily employed. The intelligent fault diagnosis and optimization issues for various rotating machines using a novel stacked transfer autoencoder was presented by Haidong et al [62] using the PSO algorithm. Manna and Bhunia [63] studied a queueing-based green production system with selling price and green level-sensitive demand, solving the interval-based optimization problem using various methods of measurement. Meena et al. [64] conducted a performance analysis of a fault-tolerant system with a

vacationing server and determined the optimal service rates using both GA and PSO algorithms. Jain et al [65] analyzed cost optimization and reliability of a machine repair model with service interruptions and reboot, utilizing the PSO algorithm for the optimization purpose. Singh et al. [66] conducted a fuzzy modeling and cost optimization study for the fault-tolerant system with service interruptions. They employed PSO and GA metaheuristics for optimization and compared the results with those obtained using the quasi-Newton method. Table 1 summarizes some notable contributions of the last 10 years by mentioning the key features, and methodologies used in relevant queueing models that utilized soft computing technique ANFIS and metaheuristics PSO, GA and HS techniques for the optimization purpose.

2.2. Research gap and motivation

The increasing complexity of modern machining systems demands more efficient and reliable repair strategies, motivating the need for an advanced modeling approach. Despite of extensive researches on queueing modeling on MRM, there is lack of study that integrates N-policy, double retrial orbits, standby components, imperfect repairs, and unreliable server. Notably, a very few researchers have studied the imperfect repair model and/or double orbit model for MRP, leaving a significant gap in the literature. Furthermore, previous models have utilized soft computing techniques like PSO and HS for the optimization. Only limited papers compared numerical results using analytical approach with those obtained via ANFIS for validation. To address these gaps, our proposed model incorporates the feature of imperfect repair of failed units in double orbit MRM involving spares, unreliable server, and employs metaheuristic optimization techniques, ensuring a more robust and optimized machine repair framework which can be used to resolve the maintenance and fault tolerance issues of many machining system.

2.3. Significance and key contributions

Several existing machine repair models proposed in the literature have incorporated a very limited features. Some models analyzed MRP with retrial orbits, and/or vacations. However, these models have certain limitations, which are effectively addressed by our proposed model. Our model offers an integrated, optimized, and holistic approach to MRP overcoming the limitations of existing

Table 1
The recent contributions of soft computing in the relevant area of the proposed model.

Soft Computing	Authors (year)	Model	Methodology	Features
Particle Swarm Optimization (PSO)	Yang & Wu [42]	Markovian queue model	MGM	Working vacation, N-policy, Unreliable server
	Zhang et al. [43]	Markovian retrial queue model	PGF	Social optimal strategy, State dependent service policy
	Wang et al. [44]	Markovian retrial queue model	PGF	N-policy, Equilibrium strategies
	Yang & Wu [45]	Markovian retrial queue model	MGM	Starting failure, Working vacation,
	Kumar & Jain [46]	Markovian queueing model	MGM	Unreliable server, Hybrid vacation
Genetic Algorithm (GA)	Sanga & Antala [47]	Non-Markovian MRM	SVT, LST	Multiple vacation, F-policy, Fuzzy
	Sanga & Jain [48]	Markovian double retrial queue model	PGF	Fuzzy, P-NLP approach
	Jain et al. [49]	Markovian double retrial queue model	PGF	Imperfect service, Balking, Vacation
	Oszczypala et al. [50]	Markovian MRM	Matrix	Mixed standby,
	Chahal & Kumar [51]	Markovian MRM	Recursive	Multiple working vacation, Generalized triadic policy
Harmonic Search (HS)	Meena et al. [52]	Non- Markovian MRM	SVT, LST, Recursive	Fuzzy, Vacationing server, Standby
	Jain et al. [53]	Non- Markovian MRM	SVT, LST, Recursive	Vacationing and unreliable server, Fuzzy, Standby
	Jain et al. [54]	Markovian MRM	LST, Recursive	Fuzzy, Imperfect coverage, reboot, standby
	Kumar et al. [55]	Machine repairable M/G/1 model	SVT, LST Recursive	N-policy, Vacationing server, Fuzzy logic
	Rani et al. [56]	Markovian MRM	Recursive	Reboot, Recovery, Vacationing server
Adaptive Neuro-Fuzzy Inference System (ANFIS)	Sanga & Charan [57]	Markovian Retrial MRM	Recursive	Retrial, Fuzzy, F-policy
	Jain & Meena [58]	Markovian MRM	SOR	Unreliable server, F-policy, Working vacation
	Sethi [59]	Markovian MRM	RKM	Start-up time, Threshold recovery
	Chahal et al. [60]	Markovian MRM	RKM	Multiple working vacation
	Nidhi & Sanga [61]	Non-Markovian MRM	SVT, LST Recursive	ANFIS, F-policy, startup

research works. The key contributions of the proposed model in comparison to existing MRM can be summarized as follows:

- Unlike prior models that focus on either retrial orbits or vacation mechanisms in MRP, our model integrates N-policy, standby units, double retrial orbits, ensuring a more realistic system representation.
- Most of the existing models have assumed the perfect repair and a fully reliable server, whereas our model deals with MRP with imperfect repair and server failures, making it closer to real-world scenarios.
- Existing studies primarily analyze either transient or steady-state behavior, whereas our model evaluates both, providing a more realistic performance assessment.
- Matrix spectral analytic and matrix recursive methods are used successfully for solving the transient and steady-state queue size distributions respectively, and other performance indices.
- Unlike traditional optimization approach for MRM that relies solely on mathematical analysis, we employ metaheuristic PSO and HS algorithms to optimize the design parameters and enhance the system efficiency at optimum cost.
- The feasibility of AI-based controllers using soft computing approach namely ANFIS technique has been also explored.

The present article deals with a finite population queueing model for machining system with imperfect service, double retrial orbits, warm standby support, and N-policy. Now, we cite a real-life scenario wherein the considered model can be applied. Consider a telecommunications network having multiple routers that transmit data. The routers are critical components for maintaining the network connectivity. In this scenario, the double orbit retrial machine repair model with N-policy under a single repairman can be applicable to analyze the reliability and queueing indices of the network. A router may experience failure, disrupting the network connectivity; the failed router undergoes a retrial process, attempting to reconnect to the network after repair. If the repair is unsuccessful or the system is busy, it undergoes additional repair. There is only a network technician available for the repair of the failed routers. The technician can handle one failed router for the repair at a time and goes on vacation when there are no failed routers in the network. From vacation, the technician returns back when repair jobs of 'N' failed routers are accumulated. The present study may be helpful to the telecommunication companies to ensure fault-tolerant network by understanding the failure and repair dynamics of critical components like routers, etc. The performance indices evaluated can provide valuable insight to minimize the downtime and maintain a reliable and robust network infrastructure.

3. Preliminary of ANFIS, PSO and HS

It is important to provide a brief overview of the soft computing technique ANFIS, as well as metaheuristic PSO and HS techniques, which are employed in the present study to predict the optimal design parameters of the proposed Markovian queueing model for the redundant repairable machining system.

3.1. Artificial neuro-fuzzy inference system (ANFIS)

ANFIS leverages a multilayer adaptive network-based fuzzy inference system to efficiently handle complex computational challenges by integrating the strengths of both fuzzy logic and ANN. ANFIS employs a neuro-adaptive learning approach to update the membership function (MF) until it achieves the optimal solution. The iterative process is used to enable ANFIS to continuously enhance

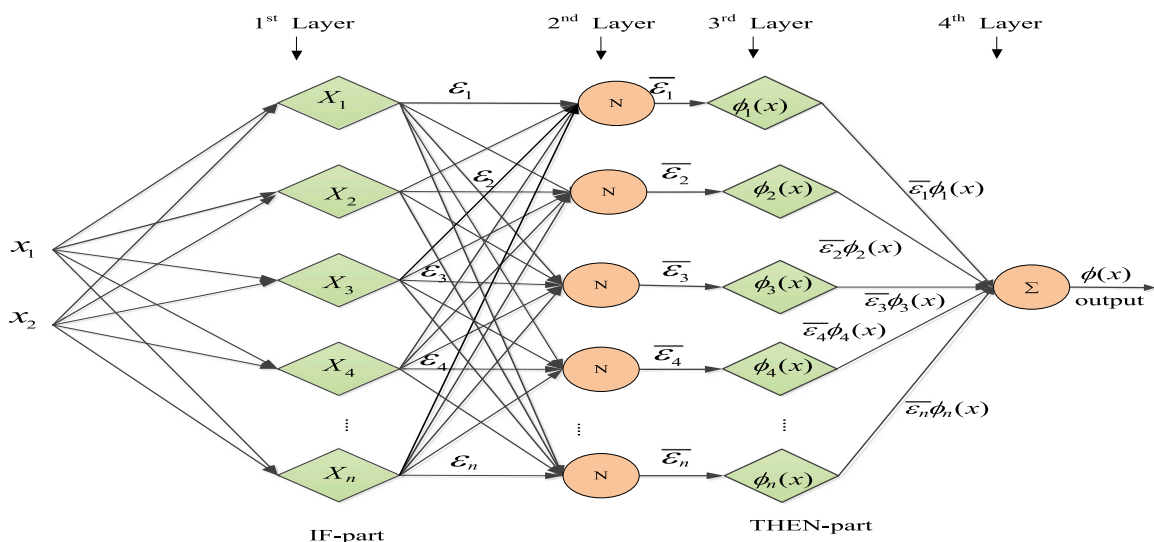


Fig. 1. ANFIS Structure.

and refine the MF, leading to the attainment of the best possible outcome. In the neuro-fuzzy approach, the fuzzy logic is used in accordance with Takagi-Sugeno (TS) approach using the least square method (LSM). By utilizing LSM, this methodology estimates and optimizes the parameters involved in the neuro-fuzzy system, facilitating effective modeling and inference based on the TS framework. The architecture of a two-input and one-output ANFIS model is displayed in Fig. 1. The ANFIS model with trapezoidal membership function employed for fuzzified parameters, are briefly described as follows:

The i^{th} rule of the first-order Takagi-Sugeno (TS) model, utilizing input $x = [x_1, x_2]$ and output $\phi(x)$, is given by-

IF input variable x_1 and x_2 are within the range of X_i ,

THEN the output function $\phi_i(x)$ is evaluated using

$$\phi_i(x) = u_i x_1 + v_i x_2 + w_i, \quad i = 1(1)n,$$

where u_i , v_i , and w_i represent the respective parameters known as resultant parameters.

First layer: The nodes within the ANFIS model perform fuzzification by generating the MF $\xi_i(x)$ for the i^{th} fuzzy set $\phi_i(x)$. This generation may be accomplished by utilizing the trapezoidal function, which is represented as

$$\xi_i(x) = \exp \left[- \left(\frac{x - y_i}{z_i} \right)^2 \right]$$

where y_i and z_i represent the specific parameters.

Second layer: In this layer, a normalization procedure occurs, yielding the normalized value through the calculation of the firing strength ratio as

$$\bar{\varepsilon}_i = \frac{\varepsilon_i}{\sum_{i=1}^n \varepsilon_i},$$

where parameter ε_i represents the degree of matching of the i^{th} rule.

Third layer: In this layer, all nodes are adaptive, and each node's calculation is determined based on the corresponding rule given by

$$\bar{\varepsilon}_i \phi_i(x) = \bar{\varepsilon}_i (u_i x_1 + v_i x_2 + w_i), \quad i = 1(1)n$$

Fourth rule: This layer is referred to as the output layer, and is computed by

$$\phi(x) = \sum_{i=1}^n \bar{\varepsilon}_i \phi(x_i)$$

Other soft computing techniques like artificial neural networks (ANN), extreme learning machines (ELMs), support vector machines (SVMs), and fuzzy logic could be applied to analyze the fault-tolerant machining system (Waseem et al. [67]). However, we prefer ANFIS because of its unique ability to integrate fuzzy logic and neural networks, making it particularly well-suited for developing the AI based controller for the automation of maintainability policy of FTMS. This combination of ANN and fuzzy allows ANFIS

Table 2

Pseudocode of PSO.

Fitness function= $TC(\mu_1, \mu_2)$, acceleration factor, inertia weight, max no. of iteration.

Initialize the population of 'N' particles.

do

for every individual entity within the particle population

calculate $TC(\mu_1, \mu_2)$

update p-best

update g-best

end

update the inertia weight

for every individual entity within the particle population

update particle's velocity and position

end

end

If the specified criterion for extinction is not achieved

Return The global optimum is referred to as the g-best.

to model the inherent uncertainties in the system, offer transparency in the decision-making through fuzzy rules, and adapt to changing system dynamics. It is capable of robustly handling nonlinearities and uncertainties involved in the system's behavior, especially when the system parameters are imprecisely defined, or incomplete data is available.

3.2. Particle swarm optimization (PSO)

The concept of PSO, conceptualized by Kennedy and Eberhart [68], is based on the behavior of flocks of birds. This method serves as a solution for the optimization problems that are highly non-linear. In PSO, we examine a collection of particles (representing potential solutions) and use certain predefined formulae to determine the best possible solution to a given problem. Each particle is assigned a fitness value, which assists in determining the personal best (p-best) and global best (g-best) solutions of the particles. In each iteration, the current g-best is compared to the p-best of the particle. If the personal best particle produces a better solution, then the global best value is updated to the personal best value. This cycle continues for the specified number of iterations. Table 2 provides the algorithmic steps of the PSO.

3.3. Harmonic search (HS) approach

A cutting-edge metaheuristic optimization method known as Harmony Search (HS) has been employed during the past ten years to handle a variety of difficult jobs. The HS method was proposed by Geem et al. [69] to solve the optimization problem based on the principle of musical improvisation. HS mimics the altering and improvising of the pitches of instruments and is influenced by jazz performers.

The musicians typically experiment with several possible pairings of the stored musical notes when composing the harmony. In fact, the pursuit of optimal solutions to engineering challenges resembles with the quest for the ideal harmony. Based on aesthetic criteria, a batch of musicians employed it to enhance the harmonies all around. The improvisation technique is employed in harmonic search to create better harmony, which is a unique interaction between a number of attractive sound waves. In HS, the pitch is altered utilizing the following relation:

$$w_j^{new} = w_j^{old} \pm F(0, 1) \times Bandwidth$$

Table 3
HS algorithm.

1.	Input parameters Fitness function as objective function $T(w)$, the input parameters used are HMCR, PAR, BW, NI, and HM size.
2.	Initialization of HM i Generate the vectors $HM = \{w_1, w_2, \dots, w_{HMS}\}$ ii Obtain the best & worst harmony in HM
3.	Harmony Generation For $i = 1, 2, \dots, k$ do If ($F(0,1) \leq HMCR$) then Begin $w_i \in \{w_{i,1}, w_{i,2}, \dots, w_{i,HMS}\}$ if ($F(0,1) \leq PAR$) then $w_j = w_j \pm F(0,1) \times BW$ end else $w_i \in W_i$ end if end for
4.	Update HM i. If $T(w_i) < T(w_{worst})$ then ii. update w_i to the HM, replacing w_{worst} by w_i to the HM
5.	Stopping criterion i. While (if termination criterion is not fulfilled) ii. Repeat steps 3 & 4 until the current best harmony is obtained.

where j^{th} component of the old (new) harmony is represented by $w_j^{\text{old}}(w_j^{\text{new}})$ and for uniformly distributed random numbers, $F [0,1]$ is utilized. Table 3. lists the pseudocode the HS algorithm.

We use metaheuristic optimization algorithms for optimization because they are well-suited for the complex, non-linear, and uncertain systems where traditional methods may not be practical or effective. Metaheuristic techniques offer the ability to find global optimal solutions efficiently in the dynamic environments, making them ideal for optimizing the repair parameters in the concerned with fault-tolerant machining system. The soft computing techniques PSO and HS are well-established and are implemented in our investigation to provide the efficient solution for optimizing the cost function. The optimization techniques used are capable of handling both continuous and discrete problems and have proven effective in optimization tasks. In some recent research papers, newer algorithms like Red Fox, Polar Bear, DE, and SGO are used and may offer some benefits in certain contexts. However, these techniques face issues like poor balance between exploration and exploitation, computational expense, and premature convergence. PSO and HS remain the most suitable choices for solving the complex non-linear problems due to their track record and computational efficiency. PSO and HS offer fast convergence making them ideal for the optimizing purpose.

4. Model description

This study examines the M/M/1 queueing model with double retrial orbits for the performance analysis of redundant machining system under the care of a failure-prone server. The system comprises of an unreliable repairman (i.e. server), L operational units, and S warm standby units. The machining system is operating following the N -policy for the repair of failed units of the system. The notations used for model formulation are listed in Table 4.

4.1. Assumptions

The assumptions to formulate the mathematical model for FTMS are as follows:

- Operating (standby) units are subject to failures, which occur in Poisson fashion with rate of $\lambda(\alpha)$. If a functioning unit malfunctions, it is substituted by an on-hand standby unit if available.
- Any failed unit does not join the repair queue in case if repairman is busy; but instead enters to retrial orbit, where it is stored in either the primary or secondary retrial orbit based on the care taker's choice,
- A failed unit first joins the primary retrial orbit and attempts to access repair job after a random retrial time. However, if the primary retrial orbit is full, the unit is redirected to the secondary retrial orbit.
- The provision of a skilled repairman for the repair job is also considered. The repairman repairs the failed units following exponential distribution (*Exp-D*) with rates μ_1 and μ_2 in case when failed units make re-attempts for the repair from the primary retrial orbit (PRO) and secondary retrial orbits (SRO), respectively.
- If the repair of the failed unit is not successful i.e. in case of imperfect repair, an additional repair is also preformed following *Exp-D* with rates $\mu_{1a}(\mu_{2a})$, when the repairman treating the failed units of primary (secondary) orbit.
- The free server repairs the failed units successfully with probability σ . If the service of the failed units is not successful, failed units require an additional repair with probability $\bar{\sigma} (= 1 - \sigma)$.
- Following N -policy, the repairman goes for vacation as soon as there is no failed unit and returns back to provide repair when N failed units are accumulated in the system. The server may go for a vacation with rate ε in case if there is no repair job. The server can initiate the repair of the failed units following N -policy with set up rate δ .
- The server may also breakdown in active state (i.e. normal busy state) following Poisson distribution with rate a . The broken down server is taken for repair immediately and may restore it functionality after repair and returns to normal busy state; the repair time of the failed repairman follows the *Exp-D* with rate b .
- The failed units are selected for the repair based on the First-Come-First-Serve (FCFS) discipline.

Table 4
Notations.

$\lambda(\alpha)$	:	Failure rate of active (standby) units
$L(S)$	:	Counts of operating machines (warm standby)
K	:	Total number of machines i.e. ($K = L + S$)
γ	:	Retrial rate from PRO (SRO) to NBS (WVS)
ε	:	Vacation rate
δ	:	Setup rate from vacation state to normal busy state
μ_1	:	Repair rate of failed machines when failed unit re-attempts for repair from PRO
μ_2	:	Repair rate of failed machines when failed unit re-attempts for repair from SRO
$\mu_{1a}(\mu_{2a})$	:	Additional service rate when the repairman treating the failed units of primary (secondary) orbit
λ_n	:	State dependent failure rate when there are n failed machines in the system
$a(b)$	:	Failure (repair) rate of the server
$\zeta(t)$	:	Random variable denoting the status of the server at t
σ	:	Probability of successful repair in first attempt

The status of the server at time t is characterized by a random variable $\zeta(t)$ which is defined as

$$\zeta(t) = \begin{cases} 0, & \text{Server is on vacation state} \\ 1, & \text{Server is free and units are waiting in the secondary retrial orbit} \\ 2, & \text{Additional service state when the failed units are waiting in secondary retrial orbit} \\ 3, & \text{The server is in normal busy state} \\ 4, & \text{Additional service state when the failed units are waiting in the primary retrial orbit} \\ 5, & \text{Server is free and failed units are waiting in the primary retrial orbit} \\ 6, & \text{Broken down state of the server} \end{cases}$$

The effective failure rate λ_n is defined as:

$$\lambda_n = \begin{cases} L\lambda + (S-n)\alpha, & 0 \leq n < S \\ (L+S-n)\lambda, & S \leq n \leq L+S \equiv K-1 \end{cases} \quad (1)$$

5. Governing equations and matrix method

The in-flow and out-flow rates for the system states of the concerned model are displayed in Fig. 2. Let $p_j(n, t)$ be the probability of n ($0 \leq n \leq K-1$) failed units in the system at the time 't' for sever status $\zeta(t) = j$, $0 \leq j \leq 6$. The probabilities related to various states of the system are determined by formulating Chapman-Kolmogorov equations as follows:

(i) $\zeta(t) = 0$: When the server on vacation state.

$$\frac{dp_0(0, t)}{dt} = -\lambda_0 p_0(0, t) + \varepsilon p_1(0, t) \quad (2)$$

$$\frac{dp_0(n, t)}{dt} = -\lambda_n p_0(n, t) + \lambda_{n-1} p_0(n-1, t); 1 \leq n \leq N-1 \quad (3)$$

$$\frac{dp_0(n, t)}{dt} = -(\lambda_n + \delta) p_0(n, t) + \lambda_{n-1} p_0(n-1, t); N \leq n \leq K-2 \quad (4)$$

$$\frac{dp_0(K-1, t)}{dt} = -\delta p_0(K-1, t) + \lambda_{K-2} p_0(K-2, t) \quad (5)$$

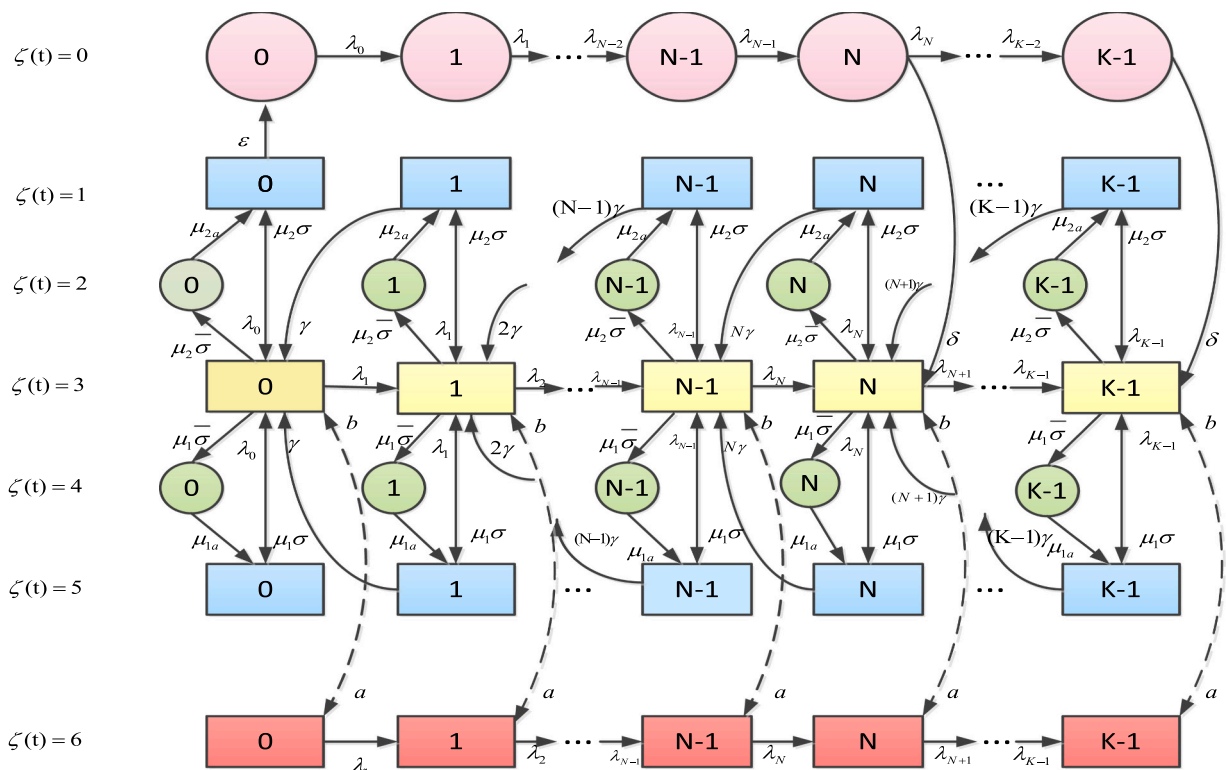


Fig. 2. Transition rates diagram.

(ii) $\zeta(t) = 1$: When the server is free, and failed units are waiting in the SRO

$$\frac{dp_1(0, t)}{dt} = -(\varepsilon + \lambda_0)p_1(0, t) + \mu_{2a}p_2(0, t) + \mu_2\sigma p_3(0, t) \quad (6)$$

$$\frac{dp_1(n, t)}{dt} = -(\eta\gamma + \lambda_n)p_1(n, t) + \mu_{2a}p_2(n, t) + \mu_2\sigma p_3(n, t); 1 \leq n \leq K - 1 \quad (7)$$

(iii) $\zeta(t) = 2$: Additional service state when the failed units are waiting in SRO.

$$\frac{dp_2(n, t)}{dt} = -\mu_{2a}p_2(n, t) + \mu_2\bar{\sigma}p_3(n, t); 0 \leq n \leq K - 1 \quad (8)$$

(iv) $\zeta(t) = 3$: When the server is in normal busy state

$$\frac{dp_3(0, t)}{dt} = -(\mu_1 + \mu_2 + \lambda_1 + a)p_3(0, t) + \lambda_0p_1(0, t) + \lambda_0p_5(0, t) + bp_6(0, t) + \gamma p_1(1, t) + \gamma p_5(1, t) \quad (9)$$

$$\begin{aligned} \frac{dp_3(n, t)}{dt} = & -(\mu_1 + \mu_2 + \lambda_{n+1} + a)p_3(n, t) + \lambda_n p_3(n - 1, t) + \lambda_n p_1(n, t) + \lambda_n p_5(n, t) + bp_6(n, t) \\ & + (n + 1)\gamma p_1(n + 1, t) + (n + 1)\gamma p_5(n + 1, t); 1 \leq n \leq N - 1 \end{aligned} \quad (10)$$

$$\begin{aligned} \frac{dp_3(n, t)}{dt} = & -(\mu_1 + \mu_2 + \lambda_{n+1} + a)p_3(n, t) + \lambda_n p_3(n - 1, t) + \lambda_n p_1(n, t) + \lambda_n p_5(n, t) + bp_6(n, t) \\ & + (n + 1)\gamma p_1(n + 1, t) + (n + 1)\gamma p_5(n + 1, t) + \delta p_0(n, t); N \leq n \leq K - 2 \end{aligned} \quad (11)$$

$$\begin{aligned} \frac{dp_3(K - 1, t)}{dt} = & -(\mu_1 + \mu_2 + a)p_3(K - 1, t) + \lambda_{K-1}p_3(K - 2, t) + \lambda_{K-1}p_1(K - 1, t) + \lambda_{K-1}p_5(K - 1, t) \\ & + bp_6(K - 1, t) + \delta p_0(K - 1, t) \end{aligned} \quad (12)$$

(v) $\zeta(t) = 4$: Additional service state when failed units are waiting in the PRO.

$$\frac{dp_4(n, t)}{dt} = -\mu_{1a}p_4(n, t) + \mu_1\bar{\sigma}p_3(n, t); 0 \leq n \leq K - 1 \quad (13)$$

(vi) $\zeta(t) = 5$: When server is busy, and failed units are waiting in the PRO.

$$\frac{dp_5(0, t)}{dt} = -\lambda_0p_5(0, t) + \mu_{1a}p_4(0, t) + \mu_1\sigma p_3(0, t) \quad (14)$$

$$\frac{dp_5(n, t)}{dt} = -(\eta\gamma + \lambda_n)p_5(n, t) + \mu_{1a}p_4(n, t) + \mu_1\sigma p_3(n, t); 1 \leq n \leq K - 1 \quad (15)$$

(vii) $\zeta(t) = 6$: Broken down state of the server.

$$\frac{dp_6(0, t)}{dt} = -(\lambda_1 + b)p_6(0, t) + ap_3(0, t) \quad (16)$$

$$\frac{dp_6(n, t)}{dt} = -(\lambda_{n+1} + b)p_6(n, t) + ap_3(n, t) + \lambda_n p_6(n - 1, t); 1 \leq n \leq K - 2 \quad (17)$$

$$\frac{dp_6(K - 1, t)}{dt} = -bp_6(K - 1, t) + ap_3(K - 1, t) + \lambda_{K-1}p_6(K - 2, t) \quad (18)$$

The transient solution of Eqs. (2)-(18) is obtained by using matrix analytic method.

The initial conditions are provided in the following manner:

$$p_0(n, 0) = \begin{cases} 1, n = 0 \\ 0, 1 \leq n \leq K - 1 \end{cases};$$

$$(s + \lambda_0)\tilde{p}_0(0, s) - \varepsilon\tilde{p}_1(0, s) = \tilde{p}_0(0, 0)$$

5.1. Matrix method

We use the matrix method to solve the governing equations framed in the previous section. By taking Laplace transforms of Eqs. (2)-(18), we obtain

$$(s + \lambda_0)\tilde{p}_0(0, s) - \varepsilon\tilde{p}_1(0, s) = \tilde{p}_0(0, 0) \quad (19)$$

$$(s + \lambda_n)\tilde{p}_0(n, s) - \lambda_{n-1}\tilde{p}_0(n-1, s) = \tilde{p}_0(n, 0); 1 \leq n \leq N-1 \quad (20)$$

$$(s + \lambda_n + \delta)\tilde{p}_0(n, s) - \lambda_{n-1}\tilde{p}_0(n-1, s) = \tilde{p}_0(n, 0); N \leq n \leq K-2 \quad (21)$$

$$(s + \delta)\tilde{p}_0(K-1, s) - \lambda_{K-2}\tilde{p}_0(K-2, s) = \tilde{p}_0(K-1, 0) \quad (22)$$

$$(s + \varepsilon + \lambda_0)\tilde{p}_1(0, s) - \mu_{2a}\tilde{p}_2(0, s) - \mu_2\sigma\tilde{p}_3(0, s) = \tilde{p}_1(0, 0) \quad (23)$$

$$(s + \lambda_n + n\gamma)\tilde{p}_1(n, s) - \mu_{2a}\tilde{p}_2(n, s) - \mu_2\sigma\tilde{p}_3(n, s) = \tilde{p}_1(n, 0); 1 \leq n \leq K-1 \quad (24)$$

$$(s + \mu_{2a})\tilde{p}_2(n, s) - \mu_2\sigma\tilde{p}_3(n, s) = \tilde{p}_2(n, 0); 0 \leq n \leq K-1 \quad (25)$$

$$(s + \mu_1 + \mu_2 + \lambda_1 + a)\tilde{p}_3(0, s) - \lambda_0\tilde{p}_1(0, s) - \lambda_0\tilde{p}_5(0, s) - b\tilde{p}_6(0, s) - \gamma\tilde{p}_1(1, s) - \gamma\tilde{p}_5(1, s) = \tilde{p}_3(0, 0) \quad (26)$$

$$(s + \mu_1 + \mu_2 + \lambda_{n+1} + a)\tilde{p}_3(n, s) - \lambda_n\tilde{p}_3(n-1, s) - \lambda_n\tilde{p}_1(n, s) - \lambda_n\tilde{p}_5(n, s) - b\tilde{p}_6(n, s) - (n+1)\gamma\tilde{p}_1(n+1, s) - (n+1)\gamma\tilde{p}_5(n+1, s) = \tilde{p}_3(n, 0); 1 \leq n \leq N-1 \quad (27)$$

$$(s + \mu_1 + \mu_2 + \lambda_{n+1} + a)\tilde{p}_3(n, s) - \lambda_n\tilde{p}_3(n-1, s) - \lambda_n\tilde{p}_1(n, s) - \lambda_n\tilde{p}_5(n, s) - b\tilde{p}_6(n, s) - (n+1)\gamma\tilde{p}_1(n+1, s) - (n+1)\gamma\tilde{p}_5(n+1, s) - \delta p_0(n, s) = \tilde{p}_3(n, 0); N \leq n \leq K-1 \quad (28)$$

$$(s + \mu_1 + \mu_2 + a)\tilde{p}_3(K-1, s) - \lambda_{K-1}\tilde{p}_3(K-2, s) - \lambda_{K-1}\tilde{p}_1(K-1, s) - \lambda_{K-1}\tilde{p}_5(K-1, s) - b\tilde{p}_6(K-1, s) - \delta p_0(K-1, s) = \tilde{p}_3(K-1, 0) \quad (29)$$

$$(s + \mu_{1a})\tilde{p}_4(n, s) - \mu_1\sigma\tilde{p}_3(n, s) = \tilde{p}_4(n, 0); 0 \leq n \leq K-1 \quad (30)$$

$$(s + \lambda_0)\tilde{p}_5(0, s) - \mu_{1a}\tilde{p}_4(0, s) - \mu_1\sigma\tilde{p}_3(0, s) = \tilde{p}_5(0, 0) \quad (31)$$

$$(s + \lambda_n + n\gamma)\tilde{p}_5(n, s) - \mu_{1a}\tilde{p}_4(n, s) - \mu_1\sigma\tilde{p}_3(n, s) = \tilde{p}_5(n, 0); 1 \leq n \leq K-1 \quad (32)$$

$$(s + \lambda_1 + b)\tilde{p}_6(0, s) - a\tilde{p}_3(0, s) = \tilde{p}_6(0, 0) \quad (33)$$

$$(s + \lambda_{n+1} + b)\tilde{p}_6(n, s) - a\tilde{p}_3(n, s) - \lambda_n\tilde{p}_6(n-1, s) = \tilde{p}_6(n, 0); 1 \leq n \leq K-2 \quad (34)$$

$$(s + b)\tilde{p}_6(K-1, s) - a\tilde{p}_3(K-1, s) - \lambda_{K-1}\tilde{p}_6(K-2, s) = \tilde{p}_6(K-1, 0) \quad (35)$$

The matrix form of above Eqs. (19)-(35) is

$$\mathbf{Q}(s)\tilde{\mathbf{p}}(s) = \mathbf{p}(0) \quad (36)$$

where

$$\mathbf{Q}(s) = \begin{bmatrix} A_0 & C_0 & & & & & & & \\ B_0 & A_1 & C_1 & & & & & & \\ & B_1 & A_2 & C_2 & & & & & \\ & & B_2 & A_3 & C_3 & & & & \\ & & & \ddots & \ddots & \ddots & & & \\ & & & & \ddots & \ddots & \ddots & & \\ & & & & & B_{K-2} & A_{K-1} & C_{K-1} & \\ & & & & & & B_{K-1} & A_K & \end{bmatrix}_{7K \times 7K}$$

where

$$\mathbf{A}_0 = [-(s + \mu_{2a})]; \mathbf{C}_0 = [0 \quad \mu_{2a} \quad 0 \quad 0 \quad 0 \quad 0]; \mathbf{B}_0 = [0 \quad 0 \quad \mu_2 \bar{\sigma} \quad 0 \quad 0 \quad 0]^T$$

$$\mathbf{C}_1 = \begin{bmatrix} \lambda_0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \lambda_1 \end{bmatrix}; \mathbf{C}_r = \begin{bmatrix} \lambda_{r-1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda_r & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \lambda_r \end{bmatrix}; 2 \leq r \leq K-1$$

$$\mathbf{B}_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \gamma & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \gamma & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}; \mathbf{B}_r = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & r\gamma & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & r\gamma & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}; 2 \leq r \leq K-1$$

$$\mathbf{A}_1 = \begin{bmatrix} -(s + \lambda_0) & 0 & 0 & 0 & 0 & 0 \\ \varepsilon & -(s + \lambda_0 + \varepsilon) & \lambda_0 & 0 & 0 & 0 \\ 0 & \mu_2 \sigma & -(s + \mu_2 + \mu_1 + a + \lambda_1) & \mu_1 \bar{\sigma} & \mu_1 \sigma & a \\ 0 & 0 & 0 & -(s + \mu_{1a}) & \mu_{1a} & 0 \\ 0 & 0 & \lambda_0 & 0 & -(s + \lambda_0) & 0 \\ 0 & 0 & b & 0 & 0 & -(s + b + \lambda_1) \end{bmatrix}$$

$$\mathbf{A}_r = \begin{bmatrix} -(s + \lambda_{r-1}) & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -(s + \lambda_{r-1} + (r-1)\gamma) & 0 & \lambda_{r-1} & 0 & 0 & 0 \\ 0 & \mu_{2a} & -(s + \mu_{2a}) & 0 & 0 & 0 & 0 \\ 0 & \mu_2 \sigma & \mu_2 \bar{\sigma} & -(s + \mu_2 + \mu_1 + a + \lambda_r) & \mu_1 \bar{\sigma} & \mu_1 \sigma & a \\ 0 & 0 & 0 & 0 & -(s + \mu_{1a}) & \mu_{1a} & 0 \\ 0 & 0 & 0 & \lambda_{r-1} & 0 & -(s + \lambda_{r-1} + (r-1)\gamma) & 0 \\ 0 & 0 & 0 & b & 0 & 0 & -(s + b + \lambda_r) \end{bmatrix}$$

for $2 \leq r \leq N$

$$\mathbf{A}_r = \begin{bmatrix} -(s + \lambda_{r-1} + \delta) & 0 & 0 & \delta & 0 & 0 & 0 \\ 0 & -(s + \lambda_{r-1} + N\gamma) & 0 & \lambda_{r-1} & 0 & 0 & 0 \\ 0 & \mu_{2a} & -(s + \mu_{2a}) & 0 & 0 & 0 & 0 \\ 0 & \mu_2 \sigma & \mu_2 \bar{\sigma} & -(s + \mu_2 + \mu_1 + a + \lambda_r) & \mu_1 \bar{\sigma} & \mu_1 \sigma & a \\ 0 & 0 & 0 & 0 & -(s + \mu_{1a}) & \mu_{1a} & 0 \\ 0 & 0 & 0 & \lambda_{r-1} & 0 & -(s + \lambda_{r-1} + N\gamma) & 0 \\ 0 & 0 & 0 & b & 0 & 0 & -(s + b + \lambda_r) \end{bmatrix}$$

for $N+1 \leq r \leq K-1$

$$\mathbf{A}_K = \begin{bmatrix} -(s + \delta) & 0 & 0 & \delta & 0 & 0 & 0 \\ 0 & -(s + \lambda_{r-1} + (K-1)\gamma) & 0 & \lambda_{r-1} & 0 & 0 & 0 \\ 0 & \mu_{2a} & -(s + \mu_{2a}) & 0 & 0 & 0 & 0 \\ 0 & \mu_2 \sigma & \mu_2 \bar{\sigma} & -(s + \mu_2 + \mu_1 + a) & \mu_1 \bar{\sigma} & \mu_1 \sigma & a \\ 0 & 0 & 0 & 0 & -(s + \mu_{1a}) & \mu_{1a} & 0 \\ 0 & 0 & 0 & \lambda_{r-1} & 0 & -(s + \lambda_{r-1} + (K-1)\gamma) & 0 \\ 0 & 0 & 0 & b & 0 & 0 & -(s + b) \end{bmatrix}$$

It is noted that $\mathbf{A}_0$ is singleton matrix, $\mathbf{C}_0$ and $\mathbf{B}_0^T$ are row matrices of order six. Also $\mathbf{A}_1$ is a square matrix of order six, $\mathbf{C}_1$ is a rectangular matrix of order 6×7 , and $\mathbf{B}_1$ is a rectangular matrix of order 7×6 . Also $\mathbf{A}_r (2 \leq r \leq N)$, $\mathbf{A}_r (N+1 \leq r \leq K-1)$, $\mathbf{A}_K$, $\mathbf{B}_r (1 \leq r \leq K-1)$, $\mathbf{C}_r (2 \leq r \leq K-1)$ are square matrix of order seven.

Now, we consider the probability vector.

$$\tilde{\mathbf{p}}(s) = [\tilde{p}_0(s), \tilde{p}_1(s), \tilde{p}_2(s), \tilde{p}_3(s), \tilde{p}_4(s), \tilde{p}_5(s), \tilde{p}_6(s)]^T$$

where

$$\tilde{p}_j(s) = [\tilde{p}_j(0, s), \tilde{p}_j(1, s), \dots, \tilde{p}_j(K-1, s)]^T : j = 0, 1, 2, 3, 4, 5, 6$$

and $\mathbf{p}(0) = [1, 0, 0, \dots, 0, 0, 0, 0, \dots, 0, 0, 0, \dots, 0]_{7K \times 1}$ is an initial vector.

We compute the probabilities $p_j(n, s) (0 \leq n \leq K-1, j = 0, 1, 2, 3, 4, 5, \text{ and } 6)$ as follows. For simplicity, we use $\pi_i(t) = p_j(n, t)$ where $i = 7n + j + 1$ for $j = 0(1)6$. Also $\tilde{\pi}_i(s) = \tilde{p}_j(n, s)$ where $i = 7n + j + 1$ for $j = 0(1)6$. Thus, we have

$$\tilde{\pi}_i(s) = \frac{|\mathbf{Q}_i(s)|}{|\mathbf{Q}(s)|}, 1 \leq i \leq 7K \quad (37)$$

where $|\mathbf{Q}(s)|$ denotes the determinant of matrix $\mathbf{Q}(s)$. Here matrix $\mathbf{Q}_i(s)$ is produced by substituting the initial vector $\mathbf{p}(0)$ for the i^{th} column of $\mathbf{Q}(s)$.

For $s = 0$, the characteristic equation of matrix $\mathbf{Q}(s)$ i.e. $|\mathbf{Q}(0)| = 0$, has only one root.

Let $s = (-d)$, so that we have

$$\mathbf{Q}(-d) = (\mathbf{Q} - d\mathbf{I}) \quad (38)$$

Now, Eq. (36) yields

$$\mathbf{Q}(-d)\tilde{\mathbf{p}}(s) = (\mathbf{Q} - d\mathbf{I})\tilde{\mathbf{p}}(s) = \mathbf{p}(0) \quad (39)$$

Let g and f be the real and complex characteristic roots of $|\mathbf{Q}(s)|$ which are denoted by $d_1, d_2, \dots, d_g$ and $(d_{g+1}, \bar{d}_{g+1}), (d_{g+2}, \bar{d}_{g+2}), \dots, (d_{g+f}, \bar{d}_{g+f})$, respectively. Here $\bar{d}$ represents the complex conjugate of d . Thus

$$|\mathbf{Q}(s)| = s \left[\prod_{i=1}^g (s + d_i) \right] \left[\prod_{i=1}^f (s + d_{g+i})(s + \bar{d}_{g+i}) \right] \quad (40)$$

Using Eq. (37) and applying Cramer's rule to Eq. (40), we obtain

$$\tilde{\pi}_i(s) = \frac{|\mathbf{Q}_i(s)|}{s \left[\prod_{i=1}^g (s + d_i) \right] \left[\prod_{i=1}^f (s + d_{g+i})(s + \bar{d}_{g+i}) \right]} ; 1 \leq i \leq 7K \quad (41)$$

The partial fractions of Eq. (41), give

$$\tilde{\pi}_i(s) = \frac{a_0}{s} + \left\{ \frac{a_1}{s + d_1} + \dots + \frac{a_g}{s + d_g} \right\} + \left\{ \frac{b_1 s + g_1}{(s + d_{g+1})(s + \bar{d}_{g+1})} + \dots + \frac{b_f s + g_f}{(s + d_{g+f})(s + \bar{d}_{g+f})} \right\} ; 1 \leq i \leq 7K \quad (42)$$

where, $a_0, a_q (q = 1, 2, \dots, g)$, $b_q (q = 1, 2, \dots, f)$ and $g_q (q = 1, 2, \dots, f)$ all real numbers. Assume that the real component u_q and the imaginary part v_q combine to form the complex characteristic root $d_{g+q} (q = 1, 2, \dots, f)$. Taking the inverse Laplace transform of Eq. (42), the transient probabilities of the system states $i (1 \leq i \leq 7K)$ are given by

$$\pi_i(t) = a_0 + \sum_{q=1}^g a_q e^{-d_q t} + \sum_{q=1}^f \left[b_q e^{-u_q t} \cos(v_q t) + \frac{g_q - b_q u_q}{v_q} e^{-u_q t} \sin(v_q t) \right] ; 1 \leq i \leq 7K, \quad (43)$$

5.2. Steady state solution

To analyze the concerned model of MRP at steady state, a matrix-geometric approach is used. We denote the probabilities in the steady state as follows: $p_{j,n} = \lim_{t \rightarrow \infty} p_j(n, t)$. In (2) - (18), the $\mathbf{Q}_0$ matrix associated with steady state exhibits a block-tridiagonal structure.

Thus

$$Q_0 = \begin{bmatrix} A_0^0 & C_0^0 & & & & & & & & & \\ B_0^0 & A_1^0 & C_1^0 & & & & & & & & \\ & B_1^0 & A_2^0 & C_2^0 & & & & & & & \\ & & B_2^0 & A_3^0 & C_3^0 & & & & & & \\ & & & \ddots & \ddots & \ddots & & & & & \\ & & & & \ddots & \ddots & \ddots & & & & \\ & & & & & B_{K-2}^0 & A_{K-1}^0 & C_{K-1}^0 & & & \\ & & & & & & B_{K-1}^0 & A_K^0 & & & \\ & & & & & & & & & & \end{bmatrix}_{7K \times 7K}$$

where $A_n^0, B_n^0, C_n^0 (0 \leq n \leq K-1)$ and A_K^0 are obtained by substituting $s = 0$ in $A_n, B_n, C_n (0 \leq n \leq K-1)$ and A_K , respectively.

We use the symbol P^0 to represent the steady-state probability row vector that corresponds to Q_0 i.e. $P^0 Q_0 = 0$. The vector P^0 can be partitioned as $P^0 = \{P_0^0, P_1^0, P_2^0, \dots, P_K^0\}$ and where $P_0^0 = p_{2,0}$ is non-negative real number. Also $P_1^0 = [p_{0,0}, p_{1,0}, p_{3,0}, p_{4,0}, p_{5,0}, p_{6,0}]$ is a row vector of order six, and $P_r^0 = [p_{0,r-1}, p_{1,r-1}, p_{2,r-1}, \dots, p_{6,r-1}]$; $2 \leq r \leq K$, are row vectors of order seven.

The solution to the steady-state equations $P^0 Q_0 = 0$, where 0 is a null vector of order $7K$, is obtained as follows:

$$P_0^0 A_0^0 + P_1^0 B_0^0 = 0 \quad (44)$$

$$P_0^0 C_0^0 + P_1^0 A_1^0 + P_2^0 B_1^0 = 0 \quad (45)$$

$$P_r^0 C_r^0 + P_{r+1}^0 A_{r+1}^0 + P_{r+2}^0 B_{r+1}^0 = 0; 1 \leq r \leq K-2 \quad (46)$$

$$P_{K-1}^0 C_{K-1}^0 + P_K^0 A_K^0 = 0 \quad (47)$$

From Eq. (47), we get

$$P_K^0 = -P_{K-1}^0 Y_K^0, \text{ where } Y_K^0 = C_{K-1}^0 (A_K^0)^{-1} \quad (48)$$

From Eqs. (45) and (46), we have

$$P_r^0 = -P_{r-1}^0 Y_r^0, \text{ where } Y_r^0 = C_{r-1}^0 [A_r^0 - Y_{r+1}^0 B_r^0]^{-1}; 1 \leq r \leq K-1 \quad (49)$$

For $r = 1$, from Eq. (49), we get

$$P_1^0 = -P_0^0 Y_1^0 \quad (50)$$

Using Eqs. (44) and (50), we get

$$P_0^0 (A_0^0 - Y_1^0 B_0^0) = 0 \text{ where } Y_1^0 = C_0^0 [A_1^0 - Y_2^0 B_1^0]^{-1} \quad (51)$$

By using Eqs. (48) and (49), we can determine $P_1^0, P_2^0, P_3^0, \dots, P_K^0$, in term of P_0^0 . We can obtain P_0^0 by applying the normalizing equation given by

$$\sum_{i=0}^K P_i^0 e = 1, \quad (52)$$

where e is a column vector in which every element is equal to one.

6. Performance measures

Assessment of transient performance indices of the concerned model can demonstrate its behavior in different scenarios which can be further utilized for the improvement of the system performance. These metrics may be beneficial to assess the accuracy, reliability, efficiency, and overall effectiveness of the concerned MRM

- Reliability of the system at time 't'

$$RY(t) = 1 - \sum_{j=0}^6 p_j(K-1, t) \quad (53)$$

- MTTF is determined by the following

$$MTTF = \lim_{s \rightarrow 0} \left(\int_0^{\infty} RY(t) e^{-st} dt \right) = \lim_{s \rightarrow 0} \left[\frac{1}{s} - \sum_{j=1}^6 \tilde{p}_j(K-1, s) \right] \quad (54)$$

- Expected number of broken-down units at the time ‘t’ in the system is

$$EN(t) = \sum_{j=0}^6 \sum_{n=0}^{K-1} p_j(n, t) \quad (55)$$

- The machine availability of the system at time ‘t’ is obtained using

$$MA(t) = 1 - \frac{EN(t)}{K} \quad (56)$$

- Probability of the busy state of the server at time ‘t’ is

$$f_B(t) = \sum_{n=0}^{K-1} p_3(n, t) \quad (57)$$

- Probability of the server being free and in primary and secondary retrial orbital states respectively, at time ‘t’ are

$$f_{PRO}(t) = \sum_{n=0}^{K-1} p_5(n, t) \text{ and } f_{SRO}(t) = \sum_{n=0}^{K-1} p_1(n, t) \quad (58)$$

- Probability of the retrial orbital states at time ‘t’ is

$$f_{ROS}(t) = \sum_{n=0}^{K-1} p_1(n, t) + \sum_{n=0}^{K-1} p_5(n, t) \quad (59)$$

- Probability of the server being in vacation state at time ‘t’ is

$$f_V(t) = \sum_{n=0}^{K-1} p_0(n, t) \quad (60)$$

- Probability that the server is rendering additional service at time ‘t’ is

$$f_{AS}(t) = \sum_{n=0}^{K-1} (p_2(n, t) + p_4(n, t)) \quad (61)$$

- Probability of the breakdown state of the server at time ‘t’ is

$$f_{BD}(t) = \sum_{n=0}^{K-1} p_6(n, t) \quad (62)$$

For the steady state, above performance indices (53)-(62) can be deduced by setting $t \rightarrow \infty$.

7. Cost function

The cost function can be designed to evaluate the effectiveness of the concerned MRM. The cost elements linked with different

activities are outlined as follows:

C_N	:	Holding cost/unit time of the failed units in the system.
$C_{PO}(C_{SO})$	:	Cost/unit time spent on the failed units residing in the primary (secondary) retrieval orbits.
C_V	:	Cost/ unit time in case when the server is on vacation.
C_B	:	Cost/unit time when the server is busy.
$C_{PU}(C_{SU})$	:	Cost/unit time of rendering repair with rate μ_1 (μ_2) by the server to the failed unit waiting in the primary (secondary) orbit.
C_{BD}	:	Cost/unit time when the server is in broken-down state.
C_{AS}	:	Cost/ unit time when the server is rendering additional repair.

The expected total cost per unit time is determined by the following equation:

$$TC(t) = C_N EN(t) + C_{PO}f_{PO}(t) + C_{SO}f_{SO}(t) + C_V f_V(t) + C_B f_B(t) + C_{BD}f_{BD}(t) + C_{AS}f_{AS}(t) + C_{PU}\mu_1 + C_{SU}\mu_2 \quad (63)$$

The cost function $TC(\mu_1, \mu_2)$ exhibits a highly nonlinear nature, making it a challenging job to analytically evaluate optimal values for the decision parameters (μ_1^*, μ_2^*) and the corresponding minimum cost. Using the soft computing-based PSO and HS techniques, the optimal values of μ_1^* and μ_2^* which are the decision variables, are determined.

8. Numerical results

The numerical simulation and sensitivity analysis of the concerned MRM with practical features namely double retrieval orbit, provision of standby support, imperfect repair, and N-policy have been presented in this section. The numerical findings using the matrix method and ANFIS for an illustration by setting $K = 7$ units have been presented. To ensure optimal performance of the MRM, the default parameter values used in our computational experiments are chosen at par with the relevant models in the literature. These values are selected with careful consideration to accurately represent a real-world machining system. The following default parameters are fixed to examine the impact of specific parameters on the system indices as follows:

$$L = 5, S = 2, \lambda = 0.5, \alpha = 0.2, \mu_1 = 4, \mu_2 = 3, \mu_{1a} = 2, \mu_{2a} = 2, a = 0.3, b = 4, \gamma = 2, \varepsilon = 3, \delta = 1, \sigma = 0.5.$$

Neuro-fuzzy tool of MATLAB is used to calculate the ANFIS outcomes by assuming the trapezoidal membership function for the fuzzified input parameter μ_1 .

8.1. ANFIS

The ANFIS outcomes can be comprehended by examining the graphs and tables generated through this process. These visual representations help to interpret the numerical results and gain insight into the system's behavior. The trapezoidal membership function allows a flexible and more realistic representation of the input parameters, enabling accurate modeling and analysis.

For the fuzzification of the input parameter μ_1 , we have chosen five membership functions, which are depicted in Fig. 3. These membership functions are categorized as very low, low, average, high, and very high.

8.2. Sensitivity analysis

Figs. 4–5 display the steady-state results obtained from two different approaches viz the matrix method and ANFIS. Fig. 4(i-iv) reveal the expected number of failed units by varying μ_1 for different values of $\lambda, \gamma, \varepsilon, \delta$, respectively. Fig. 5(i-iv) illustrate the machine availability for the same parameters setting. Numerical findings are compared with ANFIS results to validate the tractability and performance accuracy. The outcomes obtained through the matrix recursive method are graphically depicted using continuous lines, allowing for a smooth representation of the data. On the other hand, the numerical findings obtained by ANFIS are displayed using tick marks, providing a discrete representation of the results. This visual distinction facilitates the comparison outcomes which are almost coincided.

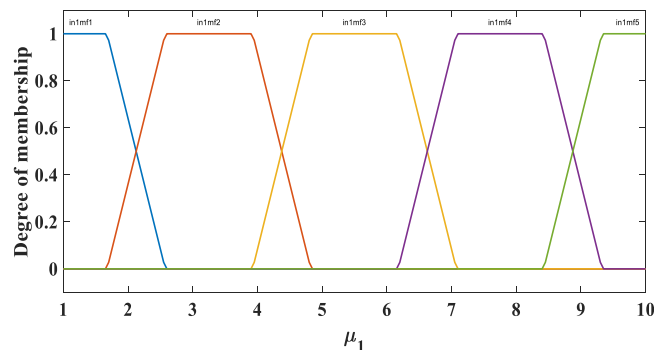
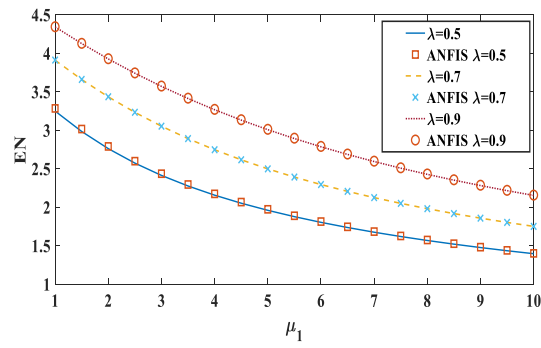
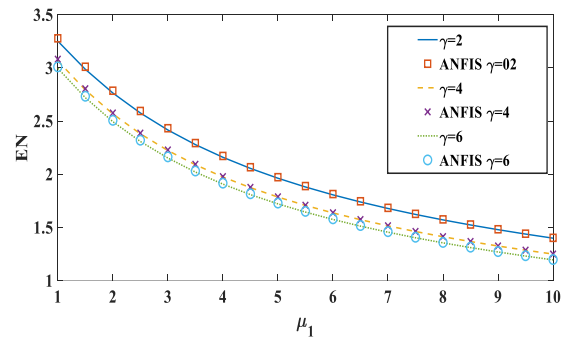


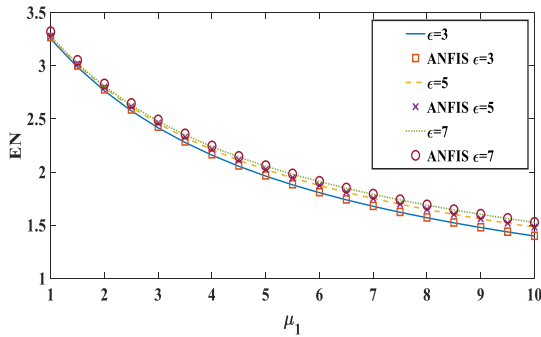
Fig. 3. Membership function for input variable μ_1 .



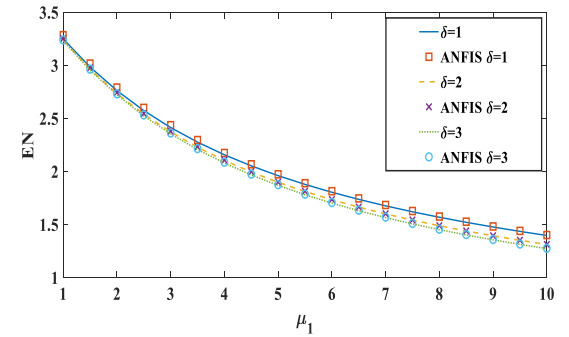
(i)



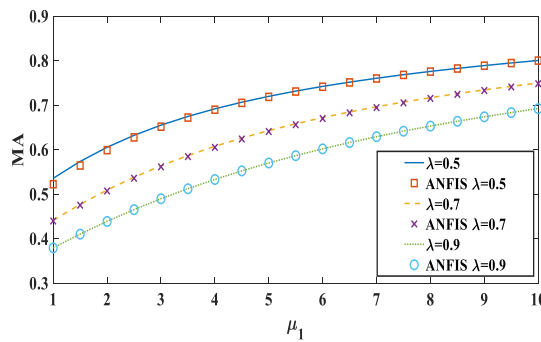
(ii)



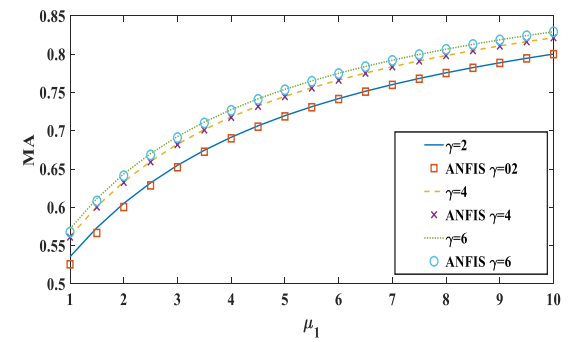
(iii)



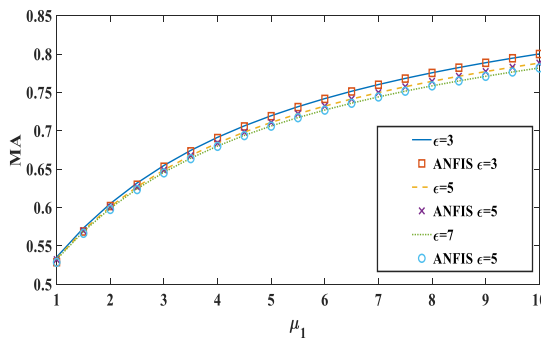
(iv)

Fig. 4. Expected number of failed units vs. μ_1 by varying (i) λ (ii) γ (iii) ϵ (iv) δ .

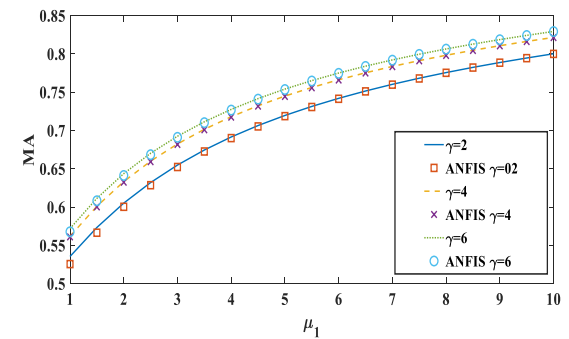
(i)



(ii)



(iii)



(iv)

Fig. 5. Machine availability vs. μ_1 by varying (i) λ (ii) γ (iii) ϵ (iv) δ .

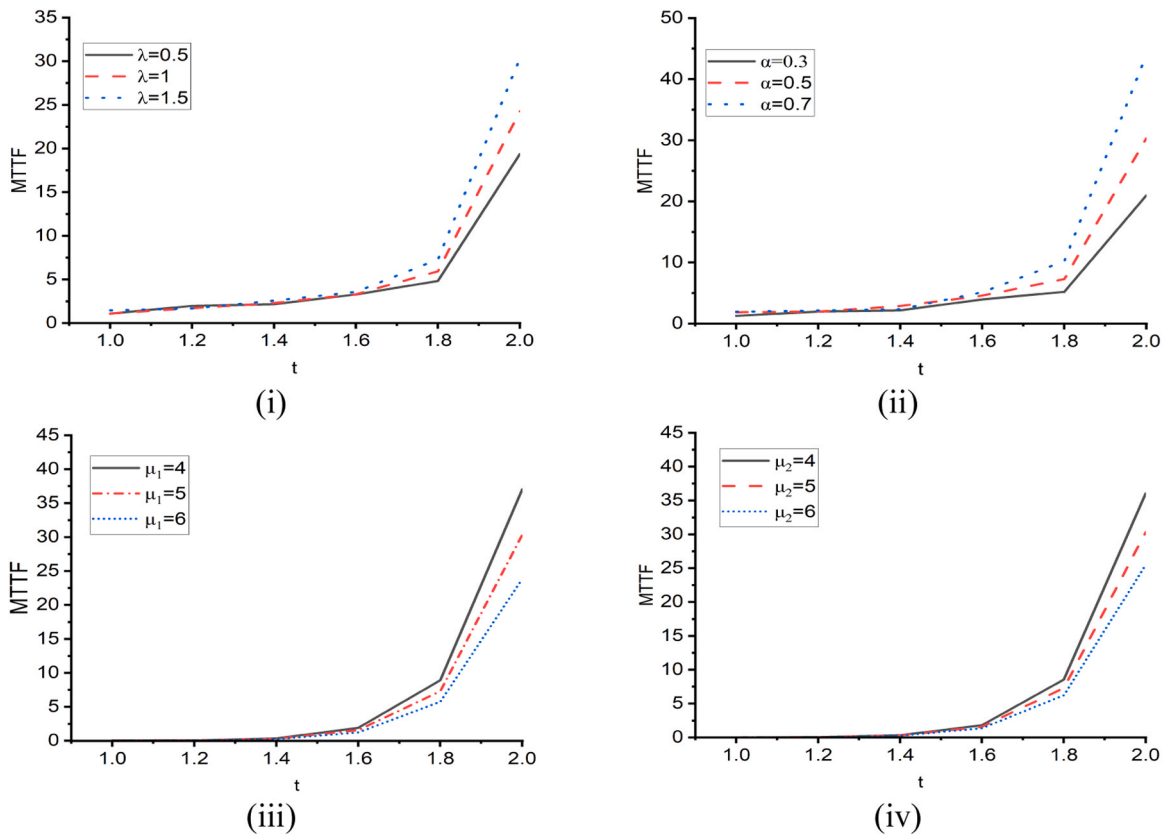


Fig. 6. MTTF for different values of (i) λ (ii) α (iii) μ_1 (iv) μ_2 .

Graphs and tables are commonly used to present numerical outcomes to understand the behavior of a system. Based on computational results displayed in Figs. 4–7 and Tables 5–8. Figs. 4 and 5 depict the trends in the expected number of failed units and machine availability by varying μ_1 for different values of $\lambda, \gamma, \varepsilon$, and δ . Figs. 6(i–iv) and 7(i–iv) present the graphs for the mean time to failure and reliability using transient probabilities for varying time for varying values of λ, α, μ_1 and μ_2 . Table 5 contains the numerical results of performance indices to examine the effect of parameters $\mu_1, \mu_2, \mu_{1a}, \mu_{2a}, b$ and σ on $f_B, f_{AS}, f_{ROS}, f_V$ and f_{BD} . Table 6 presents numerical results to explore the effect of $\lambda, \alpha, a, \varepsilon, \delta$ and γ on the same performance measures. Table 7 shows the effect of the number of standbys used on the performance indices. The errors between analytical and ANFIS results are presented in Table 8. The following observations for the numerical results of performance indices are worth noting:

- Expected number of failed units

The performance metric expected number of failed units refers to the queue size and can be helpful to evolve optimal maintenance strategies to minimize the delay in repair. Fig. 4(i–iv) display the graphs that represent the expected number of failed units (EN) vs μ_1 for different values of $\lambda, \gamma, \varepsilon, \delta$, respectively. Fig. 4(i) shows that EN decreases (increases) as μ_1 (λ) increases. From Fig. 4(ii), it is noticed that as γ increases EN seems to attain lower values. From Fig. 4(iii–iv), we see that EN increases (decreases) as ε and δ increase. From Fig. 4(i–iv), it is concluded that EN may be affected significantly by increasing retrial rate (γ), vacation rate (ε) and set up rate (δ). From Table 4, it is noticed that EN decreases as the number of standbys (operating) machines increases and (decreases). The effect of increment in the number of standbys demonstrates that EN may be controlled by enhancing the number of standbys which is same what we expect in real time system.

- Machine availability

Machine availability refers to the system's operational efficiency, indicating the proportion of time it remains functional without failures. Fig. 5(i–iv) depict the graphical representation of machine availability (MA) vs μ_1 for varying values of parameters $\lambda, \gamma, \varepsilon, \delta$. From Fig. 5(i), it is seen that MA ceases as λ increases. From Fig. 5(ii–iv), it is noticed that MA enhances as γ and δ increase but decreases as ε grows up. It is concluded that the machine availability can be improved by reducing the failure rate and vacation rate or increasing the retrial rate. Table 4 summarizes the results for the machine availability which seems to increase as the number of standbys increases.

- MTTF

MTTF, is used to determine the system's average operational duration in between two consecutive failures and plays a crucial role

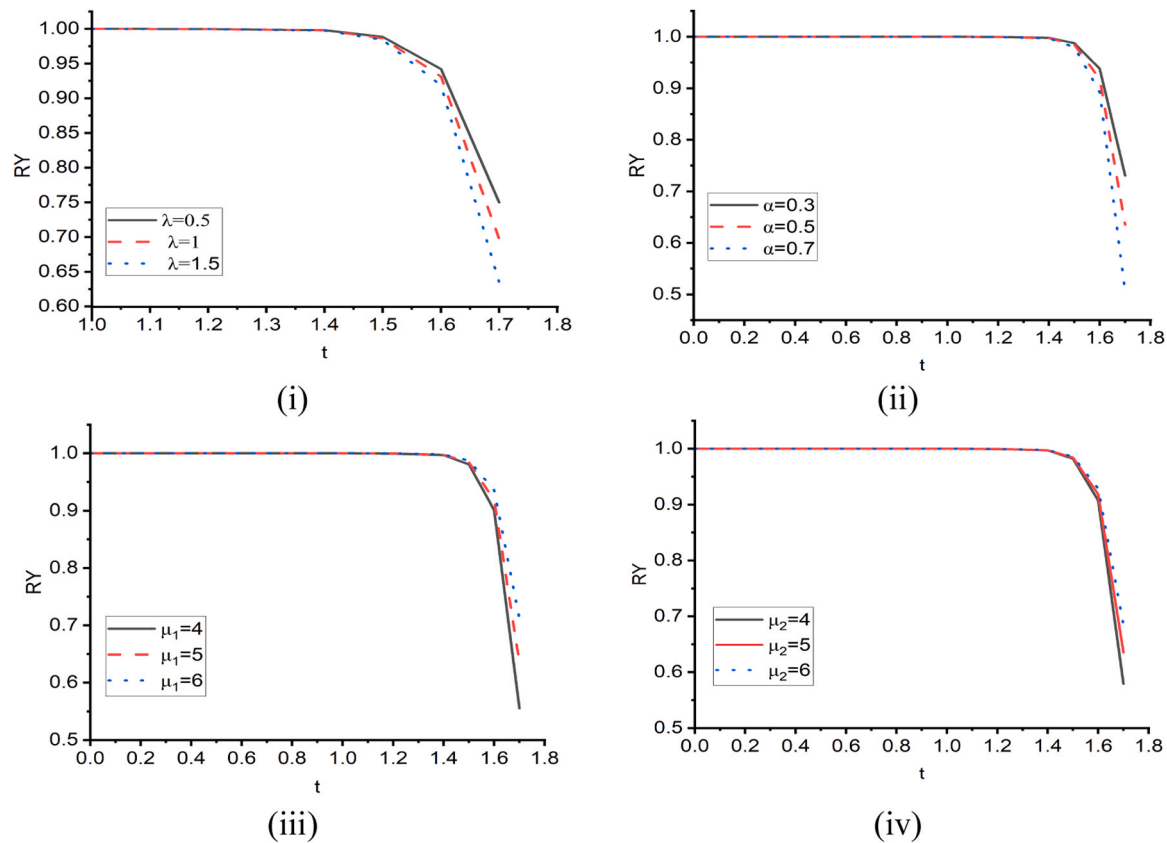


Fig. 7. Reliability for different values of (i) λ (ii) α (iii) μ_1 (iv) μ_2 .

Table 5
Effect of service rates ($\mu_1, \mu_2, \mu_{1a}, \mu_{2a}, b$) and σ on the performance indices.

Parameters		f_B	f_{AS}	f_{ROS}	f_V	f_{BD}
μ_1	4	0.2031	0.3255	0.2662	0.1597	0.0152
	5	0.1798	0.3284	0.2806	0.1662	0.0134
	6	0.1613	0.3311	0.2935	0.1699	0.0121
μ_2	4	0.2031	0.3255	0.2662	0.1597	0.0152
	5	0.1771	0.3182	0.2639	0.1913	0.0132
	5	0.1569	0.3123	0.2617	0.2165	0.0117
μ_{1a}	2	0.2031	0.3255	0.2662	0.1597	0.0152
	3	0.2179	0.2765	0.2855	0.1713	0.0163
	4	0.2261	0.2492	0.2963	0.1777	0.0169
μ_{2a}	2	0.2031	0.3255	0.2662	0.1597	0.0152
	3	0.2140	0.3000	0.2805	0.1682	0.0160
	4	0.2199	0.2861	0.2882	0.1729	0.0164
b	4	0.2031	0.3255	0.2662	0.1597	0.0152
	5	0.2034	0.3257	0.2671	0.1611	0.0122
	6	0.2035	0.3257	0.2677	0.1622	0.0101
σ	0.5	0.2031	0.3255	0.2662	0.1597	0.0152
	0.7	0.2368	0.2277	0.3104	0.1862	0.0177
	0.9	0.2839	0.0909	0.3721	0.2232	0.0212

for the maintenance and production planning. Fig. 6(i-iv) show the trends of *MTTF* for varying parameters λ, α, μ_1 and μ_2 . It is noticed that *MTTF* increases as time grows. From the observation in Fig. 6(i-ii), it is evident that the *MTTF* increases after a specific duration with increasing values of the failure rates λ and α . From Fig. 6(iii-iv), it is seen that the *MTTF* decreases as repair rates μ_1 and μ_2 increase.

- Reliability
Reliability in the context of MRM refers to the chance that the system will function without failure over a defined period, given

Table 6

Effect of failure rates (λ, α, a), and other parameters ($\varepsilon, \delta, \gamma$) on the performance indices.

Parameters		f_B	f_{AS}	f_{ROS}	f_V	f_{BD}
λ	0.5	0.2031	0.3255	0.2662	0.1597	0.0152
	0.7	0.2399	0.3989	0.2428	0.0792	0.0179
	0.9	0.2639	0.4495	0.2180	0.0361	0.0197
α	0.3	0.2031	0.3255	0.2662	0.1597	0.0152
	0.5	0.2091	0.3357	0.2626	0.1465	0.0156
	0.7	0.2144	0.3451	0.2595	0.1346	0.0160
a	0.3	0.2031	0.3255	0.2662	0.1597	0.0152
	0.5	0.2025	0.3254	0.2634	0.1542	0.0253
	0.7	0.2018	0.3252	0.2606	0.1490	0.0353
ε	3	0.2031	0.3255	0.2662	0.1597	0.0152
	5	0.2009	0.3247	0.2547	0.1776	0.0150
	7	0.1997	0.3244	0.2489	0.1866	0.0149
δ	1	0.2031	0.3255	0.2662	0.1597	0.0152
	2	0.2085	0.3336	0.2758	0.1349	0.0156
	3	0.2103	0.3361	0.2794	0.1263	0.0157
γ	2	0.2031	0.3255	0.2662	0.1597	0.0152
	4	0.2030	0.3154	0.2139	0.2124	0.0152
	6	0.2028	0.3106	0.1919	0.2351	0.0152

Table 7

Effect of number of standbys on the performance indices.

Performance indices	(L, S)		
	(6, 1)	(5, 2)	(4, 3)
EN	1.8388	1.8306	1.8244
MA	0.7384	0.7396	0.7413
f_B	0.2097	0.2031	0.1924
f_{AS}	0.3376	0.3255	0.3054
f_{ROS}	0.2629	0.2662	0.2707
f_V	0.1445	0.1597	0.1854
f_{BD}	0.0157	0.0152	0.0144

Table 8

Analytical vs. ANFIS Results: Error Analysis.

Performance Indices	Parameters	Values	Absolute Error	Mean Squared Error	Root Mean Squared Error	R^2 Score	Accuracy (%)
EN	λ	0.5	1.26×10^{-4}	1.80×10^{-5}	4.22×10^{-3}	0.999930	99.83
		0.7	6.59×10^{-4}	1.30×10^{-5}	3.63×10^{-3}	0.999964	99.88
		0.9	6.76×10^{-4}	8.00×10^{-6}	2.78×10^{-3}	0.999980	99.92
	γ	2	1.26×10^{-4}	1.80×10^{-5}	4.22×10^{-3}	0.999930	99.83
		4	7.22×10^{-5}	2.00×10^{-5}	4.46×10^{-3}	0.999917	99.81
		6	1.55×10^{-4}	2.10×10^{-5}	4.54×10^{-3}	0.999912	99.80
	ε	3	1.26×10^{-4}	1.80×10^{-5}	4.22×10^{-3}	0.999930	99.83
		5	5.92×10^{-5}	1.70×10^{-5}	4.13×10^{-3}	0.999927	99.84
		7	2.46×10^{-5}	1.70×10^{-5}	4.08×10^{-3}	0.999925	99.85
	δ	1	1.26×10^{-4}	1.80×10^{-5}	4.22×10^{-3}	0.999930	99.83
		2	2.04×10^{-4}	1.90×10^{-5}	4.31×10^{-3}	0.999933	99.82
		3	2.40×10^{-4}	1.90×10^{-5}	4.35×10^{-3}	0.999934	99.82
	MA	0.5	1.67×10^{-5}	0.000000	6.03×10^{-4}	0.999930	99.93
		0.7	9.40×10^{-5}	0.000000	5.10×10^{-4}	0.999964	99.93
		0.9	9.52×10^{-5}	0.000000	3.97×10^{-4}	0.999980	99.94
	γ	2	1.67×10^{-5}	0.000000	6.03×10^{-4}	0.999930	99.93
		4	1.16×10^{-5}	0.000000	6.38×10^{-4}	0.999917	99.93
		6	1.36×10^{-5}	0.000000	6.49×10^{-4}	0.999912	99.93
	ε	3	1.67×10^{-5}	0.000000	6.03×10^{-4}	0.999930	99.93
		5	7.06×10^{-6}	0.000000	5.90×10^{-4}	0.999927	99.93
		7	2.12×10^{-6}	0.000000	5.83×10^{-4}	0.999925	99.93
	δ	1	1.67×10^{-5}	0.000000	6.03×10^{-4}	0.999930	99.93
		2	2.78×10^{-5}	0.000000	6.16×10^{-4}	0.999933	99.92
		3	3.30×10^{-5}	0.000000	6.22×10^{-4}	0.999934	99.92

specific conditions and assumptions. High reliability of machining system ensures minimal downtime and efficient operation. The effects of parameters λ, α, μ_1 and μ_2 on the reliability are examined in Fig. 7(i-iv), respectively. In these figures, the graphs of reliability with respect to time are displayed by using the transient properties of the system states. From Fig. 7(i-iv), it is noticed that the reliability ceases as time grows. Fig. 7(i-ii) reveal that the reliability diminishes as failure rates λ and α increase. From Fig. 7(iii-iv), it is observed that the reliability increases as repair rates μ_1 and μ_2 grow up. It is concluded that the reliability can be improved by enhancing the repair rates which is same as we experience in real time system.

- Long run probabilities

The numerical results for different rates $(\mu_1, \mu_2, \mu_{1a}, \mu_{2a}, b)$ and probability σ on the long run probabilities i.e. server busy state probability, retrial orbit states probability, vacation state probability, additional service state probability, and server breakdown state probability are presented in Table 5. The long run system state probabilities are presented in Table 6 for varying failure rates (λ, α, a) , and parameters ε, δ , and γ . Table 7 contains numerical results to exhibit the effect of counting standbys on some performance measures and system states probabilities. Overall, based on the trends of various indices by variation in parameters displayed in Tables 2–4, we have the following key observations:

- An increase in μ_1 and μ_2 can reduce the probabilities of the server's breakdown state and busy state. Consequently, higher values of μ_1 and μ_2 enhance the system efficiency by minimizing the breakdowns and congestion in the server busy state.
- An increase in μ_{1a} and μ_{2a} increases the probabilities of server's breakdown state and busy state. Therefore, to improve or enhance the system efficiency, the repair rates μ_{1a} and μ_{2a} should be set lower than μ_1 and μ_2 .
- As the b increases, the probability of the server breakdown state decreases. Therefore, by improving the value of b , one can lower down the server failure state. An increase in probability σ of successful repair results in higher probabilities of both the busy state and the breakdown state of the server.
- The high values of λ, α and δ lead to the higher probability of the server's busy state and breakdown state, indicating increased system load and failures. Therefore, to enhance the system efficiency, λ, α and δ should be controlled, as regulating these parameters results in the improvement in overall system performance. The effects of retrial rate (γ) and vacation rate (ε) seem to negligible on the performance indices.
- Use of more standby units reduces the expected number of failed units in the system, as well as the probabilities of the server's busy state, additional service state, and breakdown state. Additionally, machine availability and the probability of the retrial orbit state increase by increasing the number of standby units. It is concluded that the provision of standby units and an increase in their number can enhance the system efficiency

8.3. Cost analysis

The cost-benefit analysis can be utilized for upgrading the existing system and enhancing the design of future systems. In highly competitive industrial settings where machinery failures are inevitable, repair rates as decision variables become crucial for maintaining the system operations under techno-economic constraints. The cost function $TC(\mu_1, \mu_2)$ is intricate due to its nonlinearity, and use of various cost elements. The analytical optimization method seems to be challenging to apply to minimize the cost function of proposed model. Here, we evaluate the optimal repair rates μ_1^* and μ_2^* , respective total cost $TC(\mu_1, \mu_2)$ by employing the metaheuristic HS and PSO techniques.

To assess the cost function $TC(\mu_1, \mu_2)$ using Harmony Search method and PSO, the first dimension pertains to μ_1 , while the second dimension pertains to μ_2 . The cost function $TC(\mu_1, \mu_2)$ is evaluated for varying values of μ_1 and μ_2 in the range of 1–5. The HS and PSO techniques are employed to calculate the μ_1^* and μ_2^* for the four specified cost sets as mentioned in Table 9. For cost optimization, the same default parameters are selected as used for the sensitivity analysis.

At the outset, the initialization process involves setting up each element of Harmony within the system for range 1–5 for both μ_1 and μ_2 , respectively. To run HS, the values for HMCR, PAR, and HM are set as 0.8, 0.4, and 6, respectively, and the iteration process continues until it reaches 110 generations. To implement PSO for optimization, the number of iterations, population size, acceleration factor, max inertia weight, and min inertia weight are set at 100, 10, 2, 0.9, 0.4, respectively. The convexity of the cost function $TC(\mu_1, \mu_2)$ is visualized in Figs. 8–10 for $\varepsilon = 3, 5, 7$, respectively. The computational results of optimal repair rates (μ_1^*, μ_2^*) and respective minimal cost by varying λ and α using metaheuristics PSO and HS are displayed in Table 10. It is observed that as the failure rates of operating and standby units increase, the total system cost also rises. Therefore, to minimize costs, failure rates should be controlled.

Table 9
Cost sets.

Cost Sets	C_N	C_{PO}	C_{SO}	C_V	C_B	C_{BD}	C_{AS}	C_{PU}	C_{SU}
I	120	280	280	140	70	60	50	40	45
II	130	290	290	150	75	60	55	45	50
III	145	300	300	170	80	70	60	50	55
IV	150	320	320	180	85	75	60	60	65

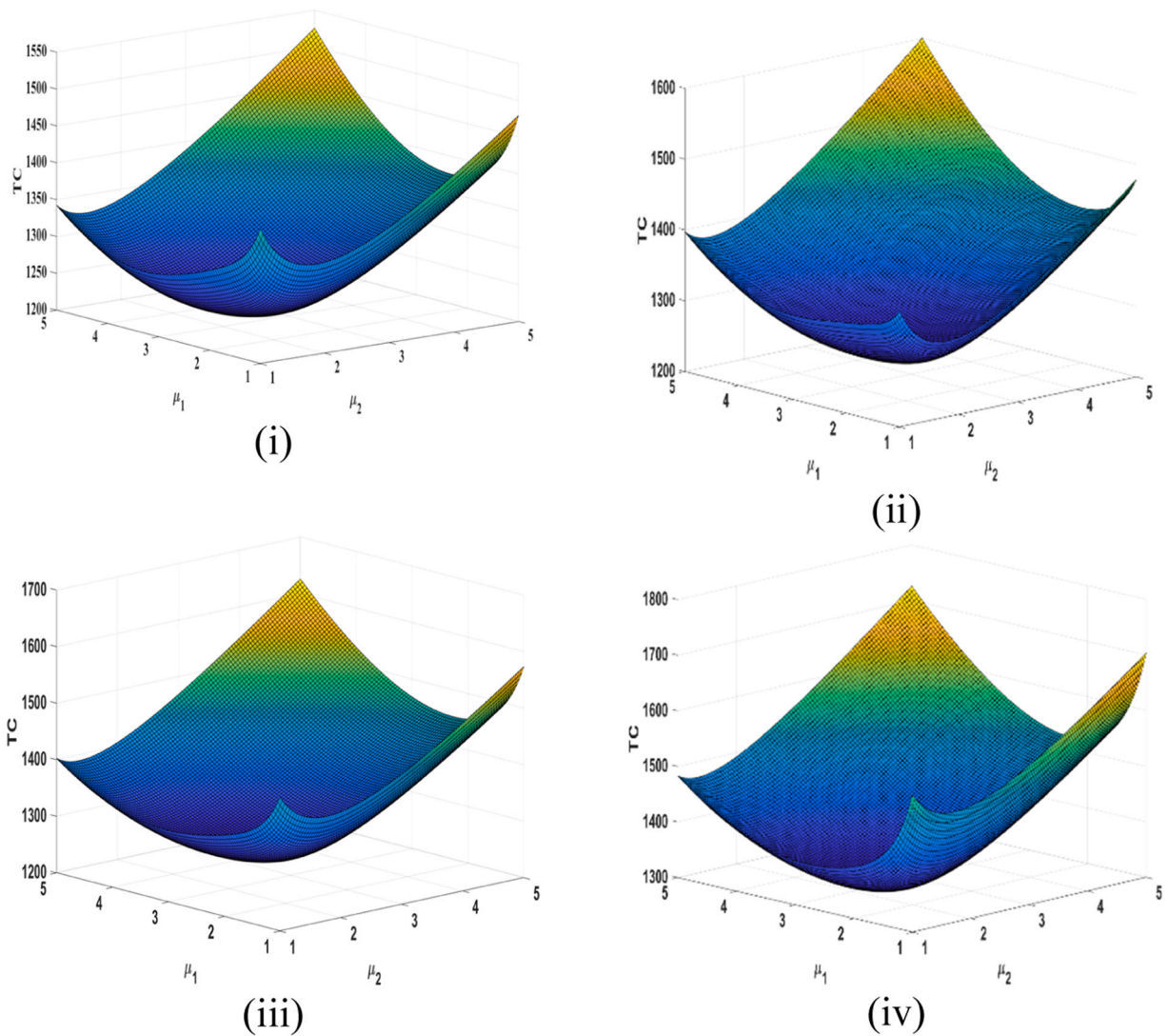


Fig. 8. Total cost vs. (μ_1, μ_2) for cost sets (i) I (ii) II (iii) III and (iv) IV, when $\varepsilon = 3$.

8.4. Statistical indices

The HS and PSO algorithms are evaluated based on the key statistical measures, including the mean function value and standard deviation. Furthermore, their computational run time and function evaluations are evaluated to assess the time and memory complexities. A comprehensive summary of these statistical indices is presented in Table 11. The convergence graphs of HS and PSO for four cost sets are displayed in Fig. 11. The worth-noting observation based on the cost optimization and trends of TC displayed in Tables 9–11 and Figs. 8–11 are as follows:

- For cost sets I and IV, the HS algorithm achieves a better optimal value compared to PSO. For cost set II, HS provides the same value as PSO. However, for cost set III, HS converges to a slightly higher value than PSO.
- Table 11 shows that PSO exhibits a lower standard deviation, indicating more consistent solutions, while HS shows higher variation in the numerical results. Fig. 11 illustrates the variation in PSO and HS, showing that PSO converges quickly, reaching near-optimal values within the first few iterations and achieving the optimum with only a negligible increase up to 105 iterations, whereas HS continues to improve gradually over all 105 iterations.
- HS performed significantly by fewer function evaluations compared to PSO, making it computationally more efficient. In the contrast, PSO performed more evaluations, leading to increased computations but ensuring more faster convergence.
- From Table 11, it is observed that HS has lower time and memory complexity because it requires fewer function evaluations, making it more efficient. In the contrast, PSO has higher time and memory complexity due to more evaluations, which ensures better stability. Thus, we conclude that HS is better from efficiency view point, while PSO is better for stability.

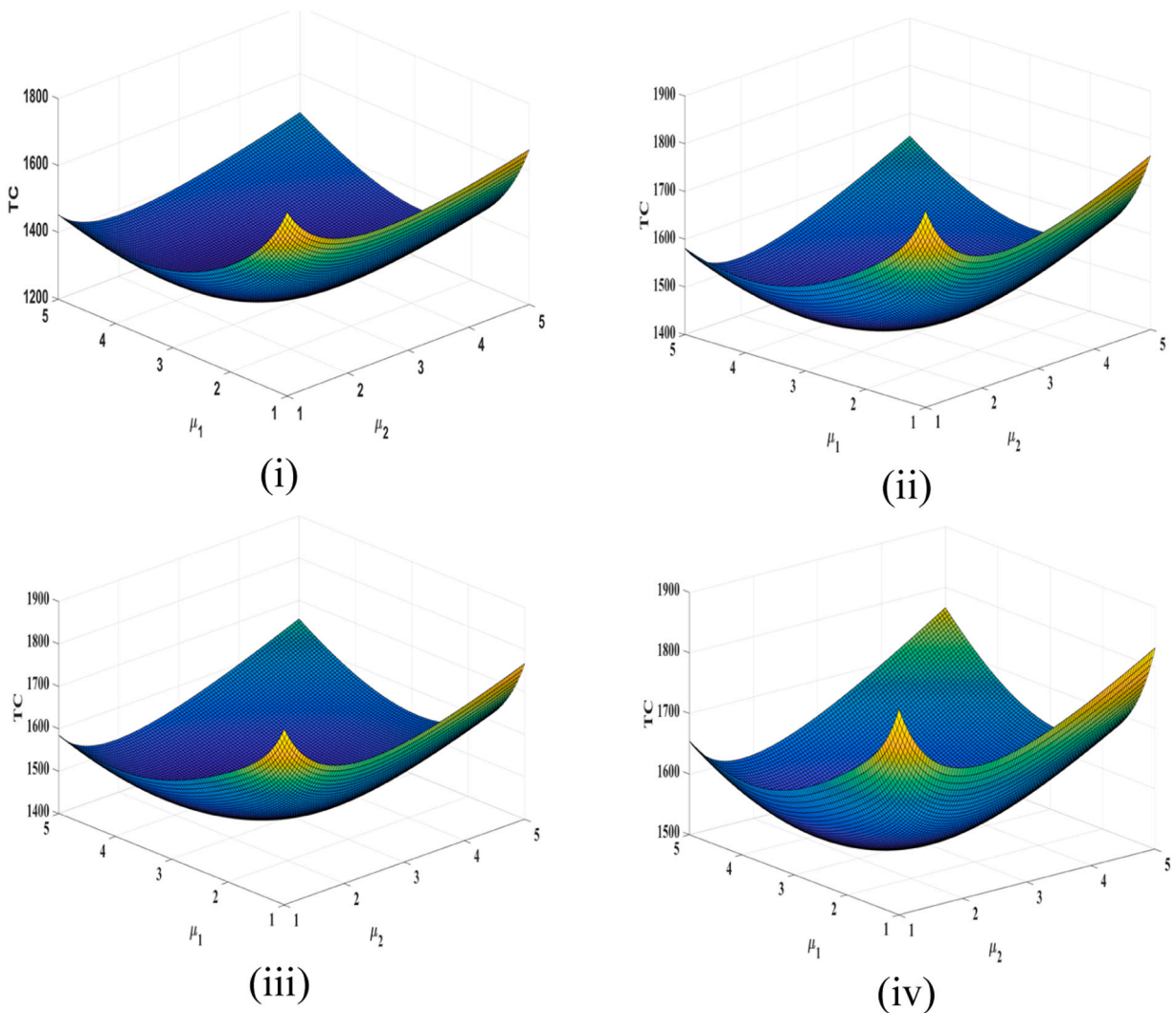


Fig. 9. Total cost vs. (μ_1, μ_2) for cost sets (i) I (ii) II (iii) III and (iv) IV, when $\varepsilon = 5$.

- There exists a trade-off between accuracy and computational cost, where HS provides superior solutions with lower computational expense; PSO ensures greater stability at a higher computational cost. Thus, HS is the better choice as it achieves better solutions with less computation.

In summary, for the proposed model, HS is preferable for the computational efficiency, while PSO ensures better stability. Parameter tuning can help in the trade-off between solution quality and efficiency. Overall, HS is the better choice for achieving the optimal solutions with less computation.

8.5. Managerial insight

The prediction of performance indices using ANFIS and metaheuristic optimization in a double retrieval machining system having many realistic features viz. standby support, breakdown pro-server, and N -policy. From a managerial viewpoint, we highlight our contributions as follows:

- The outcomes of the present article can be utilized in many real-time systems for significant enhancement of operational efficiency and reliability of the concerned model.
- The use of ANFIS demonstrates that it can be further employed to develop an AI-based controller for the concerned system to optimize the cost and evaluate the decision variables while the system is operating under imperfect recovery scenarios.
- Metaheuristic algorithms proposed may be helpful to handle the optimal control issues in order to schedule the maintenance plan, ensuring optimal resource utilization and minimal disruptions.

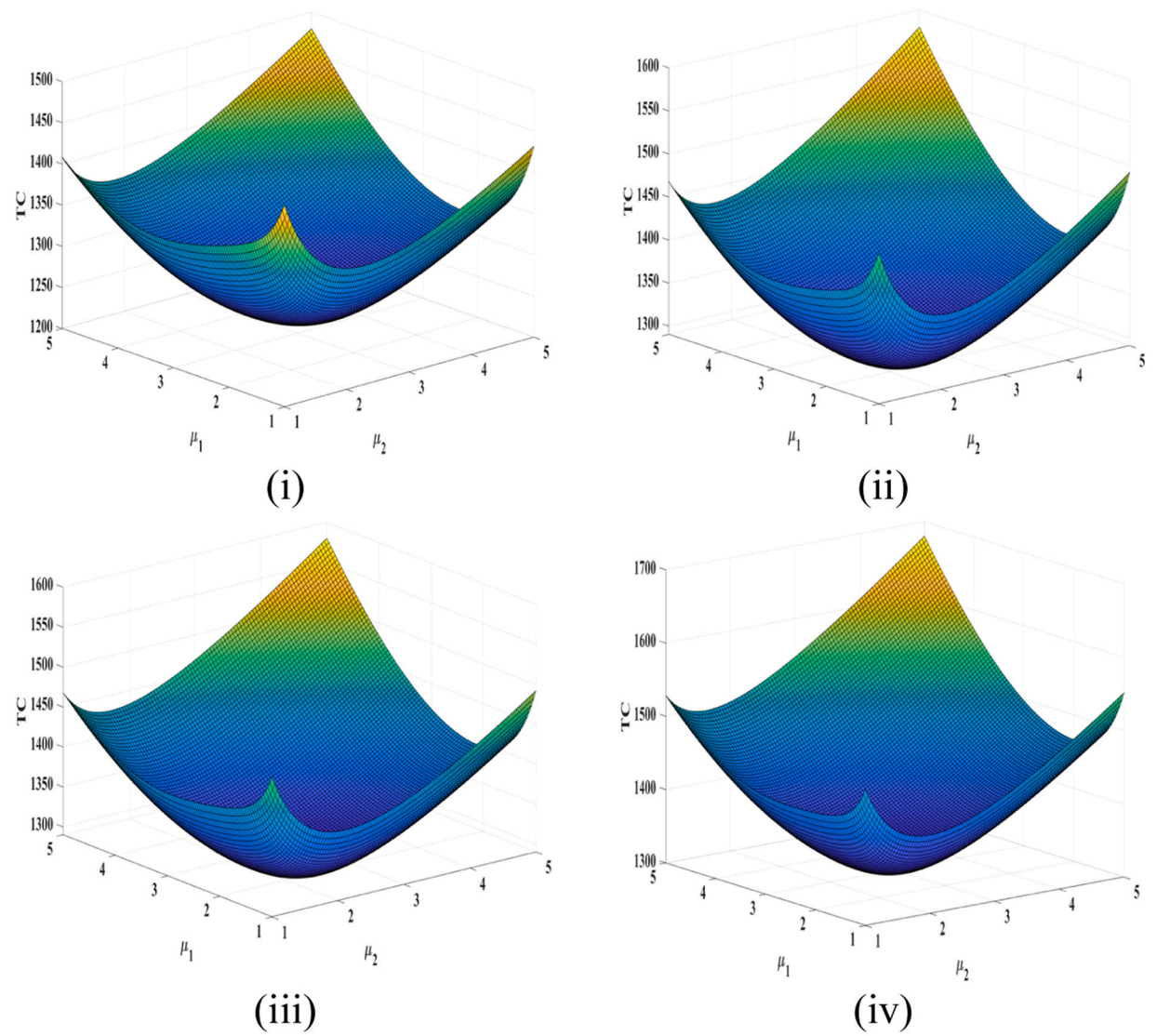


Fig. 10. Total cost vs. (μ_1, μ_2) for cost sets (i) I (ii) II (iii) III and (iv) IV, when $\varepsilon=7$.

Table 10
Effect of failure rates (λ, α) on the total optimum cost and optimal repair rates (μ_1^*, μ_2^*) .

Opt. technique	$(\mu_1^*, \mu_2^*, TC(\mu_1^*, \mu_2^*))$		Cost set I	Cost set II	Cost set III	Cost set IV
	Parameters					
HS	λ	0.5	(1.50,1.14,1235.94)	(1.34,1.07,1264.33)	(2.26,1.19,1293.69)	(2.27,1.07,1307.58)
		1	(1.44,1.14,1603.67)	(1.06,1.08,1631.64)	(1.52,1.14,1654.44)	(1.32,1.91,1679.86)
		1.5	(1.09,1.06,1834.77)	(1.19,1.18,1889.88)	(1.10,1.08,1938.88)	(1.08,1.16,1957.89)
	α	0.3	(1.50,1.14,1235.94)	(1.34,1.07,1264.33)	(2.26,1.19,1293.69)	(2.27,1.07,1307.58)
		0.5	(2.29,1.14,1268.76)	(2.26,1.16,1314.86)	(2.28,1.11,1381.62)	(2.26,1.21,1419.99)
		0.7	(2.35,1.08,1293.96)	(1.43,1.13,1334.87)	(1.44,1.23,1392.42)	(1.50,1.15,1425.94)
PSO	λ	0.5	(1.50,1.14,1238.54)	(1.35,1.08,1264.27)	(2.26,1.17,1289.91)	(2.28,1.07,1308.57)
		1	(1.43,1.13,1609.00)	(1.08,1.09,1635.01)	(1.50,1.12,1650.98)	(1.29,1.89,1687.42)
		1.5	(1.07,1.02,1837.90)	(1.16,1.13,1892.99)	(1.09,1.02,1934.90)	(1.01,1.09,1966.86)
	α	0.3	(1.50,1.14,1238.54)	(1.35,1.08,1264.27)	(2.26,1.17,1289.91)	(2.28,1.07,1308.57)
		0.5	(2.27,1.13,1271.99)	(2.26,1.18,1315.37)	(2.27,1.09,1377.59)	(2.25,1.17,1431.03)
		0.7	(2.33,1.08,1297.06)	(1.43,1.14,1334.07)	(1.43,1.24,1389.03)	(1.50,1.13,1427.98)

Table 11
Statistical Data for HS and PSO.

Cost Sets	HS				PSO			
	Mean Function value	Standard Deviation	Function Count	Run Time (Sec)	Mean Function value	Standard Deviation	Function Count	Run Time (Sec)
Set I	1249.14	7.8097	115	0.15	1238.53	1.79×10^{-13}	9450	136.31
Set II	1278.25	7.4968	115	0.13	1264.26	1.09×10^{-12}	9450	176.81
Set III	1305.97	7.5831	115	0.16	1289.91	5.18×10^{-13}	9450	168.88
Set IV	1325.53	9.2323	115	0.18	1308.57	6.80×10^{-13}	9450	157.60

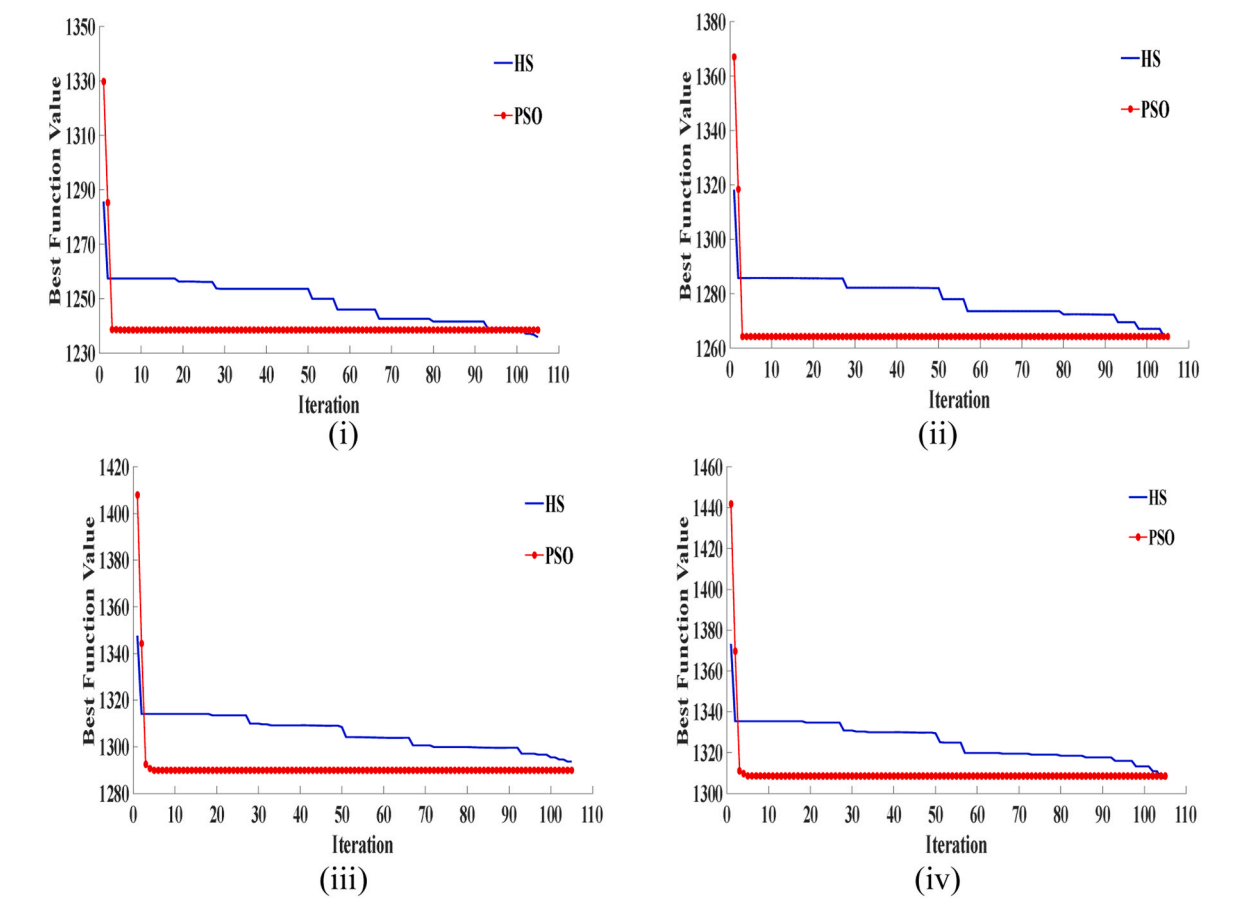


Fig. 11. The convergence graphs of HS and PSO for cost set (i) I (ii) II (iii) III and (iv) IV.

- The present study provides valuable insights into the higher system reliability, cost reduction through proactive maintenance, and improved customer satisfaction due to consistent product quality and timely deliveries.
- The proposed model may be helpful to the companies to enhance the resource utilization, minimize unexpected failures, and implement proactive maintenance strategies, leading to better productivity, higher product quality, and improved customer satisfaction.
- By implementing proposed soft computing techniques, the organization can achieve operational excellence. Managers should adopt our model to optimize maintenance, reduce downtime, and cut costs, ensuring better resource utilization and long-term reliability.
- Our model may pose challenges such as data availability, computational complexity, and the need for skilled personnel to fine-tune ANFIS, PSO, and HS parameters.
- Nevertheless, decision makers can use our model for the proper training, efficient data management, and use of scalable computational resources.

9. Conclusions

The performance assessment of a double retrial orbit machining system combined many realistic features such as a vacationing server operating under N -policy, standby support, and additional service, have done to enhance the operational efficiency and system reliability. The provision of waiting space as double retrial orbits offers an effective means of reducing the inconvenience to the caretakers of the machining system. Furthermore, increasing machine availability by incorporating the standbys is beneficial to the system organizers to cope with the machine failures. Several real-world applications of the present study can be noticed in industries such as electronic industry, security systems, power plants, etc., and helpful to design maintenance policies for the concerned fault-tolerant machining systems.

The N -policy-based vacation approach implemented can be beneficial to manage the backlog of failed units in an organized manner and ensure smoother repair operations. Additionally, use of N -policy offers flexibility to adjust repair process according to the system's size and workload, making it effective for both small and large-scale systems. The use of ANFIS can further enhance the system's performance by providing intelligent decision-making capabilities, optimizing resource allocation, and improving fault detection and prediction accuracy. Production engineers and system designers may find the current study helpful in enhancing the reliability of the machining system. The cost analysis reveals that the optimal parameters and system costs achieved using metaheuristic techniques like PSO and HS are almost coincide. Numerical findings emphasize the importance of redundancy in real-time FTMS for effectively handling the unpredictable breakdowns of the machining components and ensuring efficient performance of operators responsible for maintaining the workload of failed machines and enhancing the availability and reliability of the system. Additionally, the matrix method, including sensitivity and cost analysis carried out provide noble insights for the system organizers and decision-makers, helping them to address the economic challenges and minimize the operational disruptions more effectively.

The proposed model's drawback is that it may face challenges in scalability when applied to larger systems with more complex failure patterns, requiring further adjustments for practical implementation in diverse real-world scenarios. The study may be expanded to take working vacations and the reboot process, allowing the repairman to continue working while on break, thus reducing the system downtime. The reboot process would also facilitate quicker system recovery, minimizing interruptions and enhancing overall efficiency.

CRedit authorship contribution statement

Jain Madhu: Writing – review & editing, Supervision. **Sharma Richa:** Writing – review & editing. **Singh Vijay Pratap:** Writing – review & editing, Writing – original draft, Visualization, Methodology, Investigation, Formal analysis, Data curation, Conceptualization.

Acknowledgements

The first author (Vijay Pratap Singh) is grateful to the Ministry of Education, India, for providing JRF/SRF via Grant MHC01–23–200–428. The authors also sincerely appreciate the valuable comments and suggestions from the reviewers and Editor, which have significantly helped in improving the quality of this work.

References

- [1] K. Wang, B.D. Sivazlian, Cost analysis of the m/m/r machine repair problem with spares operating under variable service rates, *Microelectron Reliab* 32(8) 1171–1183. [https://doi.org/10.1016/0026-2714\(92\)90035-J](https://doi.org/10.1016/0026-2714(92)90035-J).
- [2] M. Jain, Preeti, Transient analysis of a machine repair system with standby, two modes of failure, discouragement and switching failure, *Int J. Oper. Res* 21 (3) (2014) 365–390. <https://doi.org/10.1504/IJOR.2014.065412>.
- [3] C.D. Liou, Optimization analysis of the machine repair problem with multiple vacations and working breakdowns, *J. Ind. Manag Optim.* 11 (1) (2015) 83–104. <https://doi.org/10.3934/jimo.2015.11.83>.
- [4] D.Y. Yang, C.L. Tsao, Reliability and availability analysis of standby systems with working vacations and retrial of failed components, *Reliab Eng. Syst. Saf.* 182 (2019) 46–55. <https://doi.org/10.1016/j.res.2018.09.020>.
- [5] J.E. Ruiz-Castro, M. Dawabsha, A multi-state warm standby system with preventive maintenance, loss of units and an indeterminate multiple number of repairpersons, *Comput. Ind. Eng.* 142 (2020). <https://doi.org/10.1016/j.cie.2020.106348>.
- [6] M.N. Juybari, A.Z. Hamadani, M.A. Ardakan, Availability analysis and cost optimization of a repairable system with a mix of active and warm-standby components in a shock environment, *Reliab Eng. Syst. Saf.* 237 (2023). <https://doi.org/10.1016/j.res.2023.109375>.
- [7] J.R. Artalejo, Accessible bibliography on retrial queues: Progress in 2000–2009, *Math. Comput. Model* 51 (9–10) (2010) 1071–1081. <https://doi.org/10.1016/j.mcm.2009.12.011>.
- [8] M. Jain, G.C. Sharma, R. Sharma, A batch arrival retrial queueing system for essential and optional services with server breakdown and Bernoulli vacation, *Int J. Internet Enterp. Manag* 8 (1) (2012) 16–45. <https://doi.org/10.1504/IJIE.2012.049868>.
- [9] D. Arivudainambi, P. Godhandaraman, Retrial queueing system with balking, optional service and vacation, *Ann. Oper. Res* 229 (1) (2015) 67–84. <https://doi.org/10.1007/s10479-014-1765-5>.
- [10] M. Jain, S.S. Sanga, Control F-policy for fault tolerance machining system with general retrial attempts, *Natl. Acad. Sci. Lett.* 40 (5) (2017) 359–364. <https://doi.org/10.1007/s40009-017-0573-2>.
- [11] T.-H. Chang, Liu, M/M/C balking retrial queue with vacation, *Qual. Technol. Quant. Manag* 16 (1) (2019) 54–66. <https://doi.org/10.1080/16843703.2017.1365280>.
- [12] T.C. Yen, K.H. Wang, C.H. Wu, Reliability-based measure of a retrial machine repair problem with working breakdowns under the F-policy, *Comput. Ind. Eng.* 150 (2020). <https://doi.org/10.1016/j.cie.2020.106885>.
- [13] S. Gao, J. Wang, Reliability and availability analysis of a retrial system with mixed standbys and an unreliable repair facility, *Reliab Eng. Syst. Saf.* 205 (2021). <https://doi.org/10.1016/j.res.2020.107240>.

- [14] T.H. Liu, F.M. Chang, J.C. Ke, S.H. Sheu, Optimization of retrial queue with unreliable servers subject to imperfect coverage and reboot delay, *Qual. Technol. Quant. Manag* 19 (4) (2022) 428–453, <https://doi.org/10.1080/16843703.2021.2020952>.
- [15] C.C. Kuo, K. Huang, J.C. Ke, T.H. Liu, S.H. Sheu, Modelling and optimisation for a retrial queue with imperfect coverage and reboot delay, *J. Control. Decis.* 12 (1) (2023) 123–130, <https://doi.org/10.1080/23307706.2023.2211082>.
- [16] M. Jain, A. Bhagat, C. Shekhar, Double orbit finite retrial queues with priority customers and service interruptions, *Appl. Math. Comput.* 253 (2015) 324–344, <https://doi.org/10.1016/j.amc.2014.12.066>.
- [17] I. Dimitriou, A two-class queueing system with constant retrial policy and general class dependent service times, *Eur. J. Oper. Res* 270 (3) (2018) 1063–1073, <https://doi.org/10.1016/j.ejor.2018.03.002>.
- [18] S.S. Sanga, M. Jain, FM/FM/1 double orbit retrial queue with customers' joining strategy: a parametric nonlinear programming approach, *Appl. Math. Comput.* 362 (2019), <https://doi.org/10.1016/j.amc.2019.06.056>.
- [19] S. Dhibar, M. Jain, Strategic behaviour for M/M/1 double orbit retrial queue with imperfect service and vacation, *Int J. Math. Oper. Res* 1 (1) (2022) 1, <https://doi.org/10.1504/ijmor.2022.10048415>.
- [20] S.S. Sanga, M. Jain, Fuzzy modeling of single server double orbit retrial queue, *J. Ambient Intell. Humaniz Comput.* 13 (9) (2022) 4223–4234, <https://doi.org/10.1007/s12652-022-03705-3>.
- [21] M. Jain, P. Mehta, Markovian unreliable server retrial queue with double orbit, imperfect repair and balking, *Natl. Acad. Sci. Lett.* 46 (5) (2023) 427–433, <https://doi.org/10.1007/s40009-023-01250-x>.
- [22] P. Kumar, M. Jain, R.K. Meena, Transient analysis and reliability modeling of fault-tolerant system operating under admission control policy with double retrial features and working vacation, *ISA Trans.* 134 (2023) 183–199, <https://doi.org/10.1016/j.isatra.2022.09.011>.
- [23] J. Li, L. Hu, Y. Wang, J. Kang, Reliability analysis and optimization design of a repairable K-out-of-N retrial system with two failure modes and preventive maintenance, *Commun. Stat. Theory Methods* 53 (15) (2024) 5524–5552, <https://doi.org/10.1080/03610926.2023.2222317>.
- [24] A. Biswas, J. Sarkar, Availability of a system maintained through several imperfect repairs before a replacement or a perfect repair, *Stat. Probab. Lett.* 50 (2) (2000) 105–114, [https://doi.org/10.1016/S0167-7152\(00\)00087-0](https://doi.org/10.1016/S0167-7152(00)00087-0).
- [25] M. Jain, V.P. Singh, Availability analysis of fault tolerant degraded machining system with imperfect repair and multi-type failures, *Int J. Appl. Comput. Math.* (2023), <https://doi.org/10.1007/s40819-023-01518-2>.
- [26] M. Yadin, P. Naor, Queueing systems with a removable service station, *J. Oper. Res. Soc.* 14 (4) (1963) 393–405, <https://doi.org/10.1057/jors.1963.63>.
- [27] M. Jain, G.C. Sharma, R. Sharma, Optimal control of (N, F) policy for unreliable server queue with multi optional phase repair and start up, *Int J. Math. Oper. Res* 4 (2) (2012) 152–174, <https://doi.org/10.1504/IJMOR.2012.046375>.
- [28] D.Y. Yang, C.H. Wu, Cost-minimization analysis of a working vacation queue with N-policy and server breakdowns, *Comput. Ind. Eng.* 82 (2015) 151–158, <https://doi.org/10.1016/j.cie.2015.01.017>.
- [29] W.L. Chen, K.H. Wang, Reliability analysis of a retrial machine repair problem with warm standbys and a single server with N-policy, *Reliab Eng. Syst. Saf.* 180 (2018) 476–486, <https://doi.org/10.1016/j.res.2018.08.011>.
- [30] R.K. Meena, M. Jain, A. Assad, R. Sethi, D. Garg, Performance and cost comparative analysis for M/G/1 repairable machining system with N-policy vacation, *Math. Comput. Simul.* 200 (2022) 315–328, <https://doi.org/10.1016/j.matcom.2022.04.012>.
- [31] J.-S. Roger Jang, ANFIS: adaptive-network-based fuzzy inference system, *IEEE Trans. Syst. Man Cyber* 23 (3) (1993) 665–685, <https://doi.org/10.1109/21.256541>.
- [32] C. Bhargava, M. Jain, Unreliable multiserver queueing system with modified vacation policy, *Opsearch* 51 (2) (2014) 159–182, <https://doi.org/10.1007/s12597-013-0138-1>.
- [33] M. Jain, R.K. Meena, Vacation model for Markov machine repair problem with two heterogeneous unreliable servers and threshold recovery, *J. Ind. Eng. Int* 14 (1) (2018) 143–152, <https://doi.org/10.1007/s40092-017-0214-x>.
- [34] R. Sethi, M. Jain, R.K. Meena, 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), <https://doi.org/10.1007/s40819-020-0802-0>.
- [35] S. Upadhyaya, C. Kushwaha, Performance prediction and ANFIS computing for unreliable retrial queue with delayed repair under modified vacation policy, *Int J. Math. Oper. Res* 17 (4) (2020) 437–466, <https://doi.org/10.1504/IJMOR.2020.110843>.
- [36] S. Thakur, A. Jain, M. Jain, ANFIS and cost optimization for Markovian queue with operational vacation, *Int J. Math. Eng. Manag Sci.* 6 (3) (2021), <https://doi.org/10.33889/ijmens.2021.6.3.053>.
- [37] H. Soltanali, A. Rohani, M.H. Abbaspour-Fard, J.T. Farinha, A comparative study of statistical and soft computing techniques for reliability prediction of automotive manufacturing, *Appl. Soft Comput.* 98 (2021), <https://doi.org/10.1016/j.asoc.2020.106738>.
- [38] A. Ahuja, A. Jain, M. Jain, Transient analysis and ANFIS computing of unreliable single server queueing model with multiple stage service and functioning vacation, *Math. Comput. Simul.* 192 (2022) 464–490, <https://doi.org/10.1016/j.matcom.2021.09.011>.
- [39] M.M. Mabrook, H.A. Taha, A.I. Hussein, Cooperative spectrum sensing optimization based adaptive neuro-fuzzy inference system (ANFIS) in cognitive radio networks, *J. Ambient Intell. Humaniz Comput.* 13 (7) (2022) 3643–3654, <https://doi.org/10.1007/s12652-020-02121-9>.
- [40] S. Rani, M. Jain, S. Dhibar, Analysis of Markovian retrial queue with double orbits, vacation, orbital search, and disaster using ANFIS approach, *Iran. J. Sci.* 47 (5–6) (2023) 1751–1764, <https://doi.org/10.1007/s40995-023-01541-4>.
- [41] N. Singh, M. Jain, S. Dhibar, ANFIS computing for M/M/∞ queue with two types of service interruption and balking, *Proc. Nat. Acad. Sci. India Sect. A* 94 (1) (2024) 63–73, <https://doi.org/10.1007/s40010-023-00821-3>.
- [42] D.Y. Yang, C.H. Wu, Cost-minimization analysis of a working vacation queue with N-policy and server breakdowns, *Comput. Ind. Eng.* 82 (2015) 151–158, <https://doi.org/10.1016/j.cie.2015.01.017>.
- [43] X. Zhang, J. Wang, Q. Ma, Optimal design for a retrial queueing system with state-dependent service rate, *J. Syst. Sci. Complex* 30 (4) (2017) 883–900, <https://doi.org/10.1007/s11424-017-5097-9>.
- [44] J. Wang, X. Zhang, P. Huang, Strategic behavior and social optimization in a constant retrial queue with the N-policy, *Eur. J. Oper. Res* 256 (3) (2017) 841–849, <https://doi.org/10.1016/j.ejor.2016.06.034>.
- [45] D.Y. Yang, C.H. Wu, Performance analysis and optimization of a retrial queue with working vacations and starting failures, *Math. Comput. Model Dyn. Syst.* 25 (5) (2019) 463–481, <https://doi.org/10.1080/13873954.2019.1660378>.
- [46] A. Kumar, M. Jain, Cost Optimization of an Unreliable server queue with two stage service process under hybrid vacation policy, *Math. Comput. Simul.* 204 (2023) 259–281, <https://doi.org/10.1016/j.matcom.2022.08.007>.
- [47] S.S. Sanga, K.S. Antala, Cost optimization and machine learning-based prediction of waiting time for fault-tolerance machining system with general vacation and F-policy, *Appl. Soft Comput.* 154 (2024), <https://doi.org/10.1016/j.asoc.2024.111404>.
- [48] S.S. Sanga, M. Jain, FM/FM/1 double orbit retrial queue with customers' joining strategy: A parametric nonlinear programming approach, *Appl. Math. Comput.* 362 (2019), <https://doi.org/10.1016/j.amc.2019.06.056>.
- [49] M. Jain, S. Dhibar, S.S. Sanga, Markovian working vacation queue with imperfect service, balking and retrial, *J. Ambient Intell. Humaniz Comput.* 13 (4) (2022) 1907–1923, <https://doi.org/10.1007/s12652-021-02954-y>.
- [50] M. Oszczypala, J. Ziolkowski, J. Malachowski, Redundancy allocation problem in repairable K-out-of-N systems with cold, warm, and hot standby: A genetic algorithm for availability optimization, *Appl. Soft Comput.* 165 (2024), <https://doi.org/10.1016/j.asoc.2024.112041>.
- [51] P.K. Chahal, K. Kumar, A genetic algorithm for cost optimization in queueing models of machining systems with multiple working vacation and generalized triadic policy, *Decis. Anal.* 10 (2024), <https://doi.org/10.1016/j.dajour.2023.100395>.
- [52] R.K. Meena, M. Jain, S.S. Sanga, A. Assad, Fuzzy modeling and harmony search optimization for machining system with general repair, standby support and vacation, *Appl. Math. Comput.* 361 (2019) 858–873, <https://doi.org/10.1016/j.amc.2019.05.053>.
- [53] M. Jain, P. Kumar, R.K. Meena, Fuzzy metrics and cost optimization of a fault-tolerant system with vacationing and unreliable server, *J. Ambient Intell. Humaniz Comput.* 11 (11) (2020) 5755–5770, <https://doi.org/10.1007/s12652-020-01951-x>.

- [54] M. Jain, P. Kumar, S.S. Sanga, Fuzzy Markovian modeling of machining system with imperfect coverage, spare provisioning and reboot, *J. Ambient Intell. Humaniz Comput.* 12 (7) (2021) 7935–7947, <https://doi.org/10.1007/s12652-020-02523-9>.
- [55] P. Kumar, M. Jain, R.K. Meena, Optimal control of fault tolerant machining system with reboot and recovery in fuzzy environment using harmony search algorithm, *ISA Trans.* 119 (2022) 52–64, <https://doi.org/10.1016/j.isatra.2021.02.027>.
- [56] S. Rani, M. Jain, R.K. Meena, Queueing modeling and optimization of a fault-tolerant system with reboot, recovery, and vacationing server operating under admission control policy, *Math. Comput. Simul.* 209 (2023) 408–425, <https://doi.org/10.1016/j.matcom.2023.02.015>.
- [57] S.S. Sanga, G.S. Charan, Fuzzy modeling and cost optimization for machine repair problem with retrial under admission control F-policy and feedback, *Math. Comput. Simul.* 211 (2023) 214–240, <https://doi.org/10.1016/j.matcom.2023.03.036>.
- [58] M. Jain, R.K. Meena, Markovian analysis of unreliable multi-components redundant fault tolerant system with working vacation and F-policy, *Cogent Math.* 4 (1) (2017) 1306961, <https://doi.org/10.1080/23311835.2017.1306961>.
- [59] R. Sethi, A. Bhagat, D. Garg, ANFIS based machine repair model with control policies and working vacation, *Int J. Math. Eng. Manag. Sci.* 4 (6) (2019) 1522–1533, <https://doi.org/10.33889/IJMEMS.2019.4.6-120>.
- [60] P. Kaur Chahal, K. Kumar, B.S. Soodan, reliability analysis and anfis computation for multi-server redundant machining system with the generalized triadic policy, *Yugosl. J. Oper. Res.* 4 (2) (2024) 307–336, <https://doi.org/10.2298/YJOR230915030C>.
- [61] S.S. Nidhi, Sanga, ANFIS computing and cost minimization of M/G/1/K fault-tolerant machining system with general startup under F-policy, *Phys. A: Stat. Mech. Appl.* 657 (2025), <https://doi.org/10.1016/j.physa.2024.130219>.
- [62] S. Haidong, D. Ziyang, C. Junsheng, J. Hongkai, Intelligent fault diagnosis among different rotating machines using novel stacked transfer auto-encoder optimized by PSO, *ISA Trans.* 105 (2020) 308–319, <https://doi.org/10.1016/j.isatra.2020.05.041>.
- [63] A.K. Manna, A.K. Bhunia, Investigation of green production inventory problem with selling price and green level sensitive interval-valued demand via different metaheuristic algorithms, *Soft Comput.* 26 (19) (2022) 10409–10421, <https://doi.org/10.1007/s00500-022-06856-9>.
- [64] R.K. Meena, M. Jain, A. Assad, R. Sethi, D. Garg, Performance and cost comparative analysis for M/G/1 repairable machining system with N-policy vacation, *Math. Comput. Simul.* 200 (2022) 315–328, <https://doi.org/10.1016/j.matcom.2022.04.012>.
- [65] M. Jain, P. Kumar, M. Singh, R. Gupta, Cost optimization and reliability analysis of fault tolerant system with service interruption and reboot, *Reliab Eng. Syst. Saf.* 249 (2024), <https://doi.org/10.1016/j.ress.2024.110229>.
- [66] V.P. Singh, M. Jain, R.K. Meena, P. Kumar, Fuzzy modelling and cost optimization of fault-tolerant system with service interruption, 89-06, *ISA Trans.* 157 (2025), <https://doi.org/10.1016/j.isatra.2024.12.006>.
- [67] W. Waseem, M. Sulaiman, P. Kumam, M. Shoaib, M.Asif Zahoor Raja, S. Islam, Investigation of singular ordinary differential equations by a neuroevolutionary approach, *Plos One* 15 (7) (2020), <https://doi.org/10.1371/journal.pone.0235829>.
- [68] J. Kennedy, R. Eberhart, A discrete binary version of the particle swarm algorithm. In: *Proceedings of the IEEE international conference on systems, man, and cybernetics. computational cybernetics and simulation*, 1997. IEEE; 1997. p. 4104–4108. <https://doi.org/10.1109/ICSMC.1997.637339>.
- [69] Z. Woo Geem, J. Hoon Kim, G.V. Loganathan, A new heuristic optimization algorithm: harmony search, *Simulation* 76 (2) (2001) 60–68, <https://doi.org/10.1177/003754970107600201>.