Markovian analysis of unreliable multi-components redundant fault tolerant system with working vacation and F-policy

Madhu Jain¹ and Rakesh Kumar Meena¹*  

Abstract: This investigation presents admission control policy for the fault tolerant system comprising of multi-components operating machines and multiple types of warm standbys under the maintenance of single unreliable server. The concepts of F-policy which deals with the controlling of inflow of failed machines and imperfect coverage are incorporated to make Markov model formulation more realistic. The unreliable repairman takes some time before allowing the failed machines to wait in the system for the repair job. The successive over relaxation (SOR) method is used to obtain the system state probabilities at the steady state which are further used to evaluate other system indices including the mean queue length of failed machines, mean number of standby machines, throughput, etc. By constructing the cost function in terms of various performance indices and associated cost elements, the optimal repair rate is determined so that the maintenance of the concerned failure prone FTS can be done in an economic manner. The hybrid soft computing approach based on a neuro-fuzzy inference system is implemented to compare the results obtained by SOR method and neuro-fuzzy approach.

ABOUT THE AUTHORS

Madhu Jain is faculty in the Department of Mathematics, IIT Roorkee, Roorkee, India. She is a recipient of two gold medals of Agra University at MPhil level. There are more than 400 research publications in refereed international/national journals and more than 20 books to her credit. Thirty-five candidates have received their PhD under her supervision. Her current research interest includes the performance modeling, stochastic modeling, soft computing, bio-informatics, reliability engineering, and queueing theory. She has visited several reputed universities/institutes in the USA, Canada, UAE, Australia, UK, Germany, France, Holland, Belgium, the Netherlands, and Taiwan.

Rakesh Kumar Meena is a research scholar at the Department of Mathematics, Indian Institute of Technology Roorkee, Uttarakhand, India. He received his MSc degree in Applied Mathematics from Indian Institute of Technology Roorkee. His area of research interest includes queueing theory, reliability, stochastic modeling, performance analysis of fault tolerant systems, and soft computing.

PUBLIC INTEREST STATEMENT

Due to unavoidable failures of the machining parts not only the functioning of the organization/system is stopped but also result in the reputation loss and degradation in the quality of products. To ensure the pre-specified system efficiency and to achieve the production goal, the system designers have to pay much attention toward the fault tolerance strategies by designing the appropriate maintenance and redundancy policies. The features of recovery and reboot should be taken into account for the smooth functioning of the real-time machining systems in particular when there is requirement of high reliability. The indices established in this study to characterize the performance of the system can be helpful in the upgradation of the existing and for the future design of the new machining systems.
1. Introduction

The operation and capacity of fault tolerant systems involved in computer or communication networks, manufacturing or production systems and many other systems are highly affected by the failure of machining components. The occurrence of faults may cause not only a loss of desired output and efficiency but also an increase in the down time and cost. To avoid these adverse situations, the organizations or industries make provision of standbys and maintenance. In the fault tolerant systems (FTSs), some units may fail, but still the system remains operative and continues to perform its assigned job due to the provision of maintainability, optimal control and standbys. With the advancement in modern technology, performance modeling of fault tolerant systems plays a vital role to make the system more efficient and fault tolerable. For the smooth functioning and to achieve the desired reliability, the concept of fault tolerance via reboot and recovery processes has drawn the attention of practitioners as well as researchers engaged in the design and development of the machining system.

The finite population queueing models for the machining systems with standby support have been developed by many queue theorists due to its applicability in real-time systems having failure prone components. For example, a standby power equipment is required during the operation of a patient in any hospital due to random power breakdown. Many more instances of fault occurrence can be noticed in real-time systems, such as power stations, manufacturing and production units, nuclear and power plant systems, call centers, etc. Several authors have established notable results for the performance prediction of machine repair systems with standby provisioning (Jain, Kulshrestha, & Maheshwari, 2004; Sivazlian & Wang, 1989; Wang & Ke, 2003; Wang & Sivazlian, 1992). The profit model for the finite population M/M/R machining system with the provision of spares, reneging, and balking factors have been investigated by Wang, Ke, and Ke (2007) and Jain, Sharma, and Sharma (2008). The repairable system with the provision of cold standby units under the Poisson shocks has been analyzed by Wu and Wu (2011). Recently, Shree, Singh, and Sharma (2015) studied the machine repair problem with hot standbys and derived some queueing and reliability measures which can be further used to enhance the availability and throughput of the system.

The queueing modeling with vacationing server for the fault tolerant systems can also be done to deal with many realistic situations where the server may leave the system to go for vacation in case when the system becomes empty. Eminent research works have been done by a number of queue theorists on vacation queueing models (Doshi, 1986; Gupta, 1997). Under the assumption of vacation policy, Ke and Wang (2007) developed the machine repair model having two types of spares. In this study, they have used the matrix geometric method for the prediction of performance measures related to queueing characteristics. Ke, Wu, Liou, and Wang (2011) obtained the various system performance measures and presented the cost analysis for the machine repair problem (MRP) with standby support under the assumption of server vacation. In recent years, some variants of Markovian multi-server machine repair models with server vacations were considered by Ke and Wu (2012) and Wu, Tang, Yu, and Jiang (2014).

In many service systems, the server while on vacation may not like to remain idle due to many reasons including the loss of profit in case when some jobs accumulate during the vacation period. The same is the case with the machine repair systems; in such case, when the failed machines join the system during vacation, the server rather than completely stopping the service provides repair to the failed machines at a different pace and is called on working vacation. The introductory work on the working vacation model in the queueing literature was due to Servi and Finn (2002) who studied the M/M/1 queueing system by incorporating the feature of working vacation. Due to the
enormous instances of MRPs with working vacation, the attention of queue theorists have diverted to this issue and a few papers on MRP with working vacations in different contexts have appeared during last decade (Jain & Upadhyaya, 2011; Lin & Ke, 2009; Wang, Chen, & Yang, 2009). Recently, Liu, Cui, and Wen (2015) studied a Markovian repairable system with cold standbys and having single repairman which is allowed to take working vacation and vacation interruptions after each repair according to Bernoulli rule.

In any machining system while providing service, the server may break down; the service interruption due to server failure for a long time directly affects the profit/goodwill as well as the hindrance in achieving desired output. In the queueing literature, the concept of server breakdown in variant scenarios has been studied by many researchers to analyze the performance metrics of machining system in different industries/organizations. Excellent works on the machine repair problem with server breakdowns in different contexts were presented by Jain and Bhargava (2009), Ke, Hsu, Liu, and Zhang (2013), and many more. The unreliable server Markovian MRP with multiple vacations and warm standbys was investigated by Wang, Liou, and Wang (2014). To determine the optimal system parameters, they have used the particle swarm optimization (PSO) which is well established soft computing based optimization method. The performance model of MRP with the provision of warm spares, common cause failure and unreliable server was investigated by Jain, Shekhar, and Shukla (2014). Yang and Wu (2015) analyzed a N-policy Markovian model with working vacation having an unreliable server. They have also used PSO approach to determine the optimal parameters for the system design to minimize the total cost incurred on different activities.

The past research works dealing with the controlling of queueing situations can be divided in two broad categories, first one to control the service and other one to control the arrival. By incorporating the concept of controlling the arrival based on F-policy, one can modified the machine repair model to portray the real-time FTSs. The concept of F-policy was first introduced by Gupta (1995). To have an idea on researches going on in the area of control policy of MRP, we refer some important researches which are also applicable to analyze the performance of FTS. Wang and Yang (2009) presented the steady-state results for the finite capacity Markovian queue with single unreliable server. To determine the queue length distribution, they have employed the matrix analytical approach. Furthermore, to determine the optimal parameters, they have also minimized the cost function using Newton-quasi method. Yang, Wang, and Wu (2010) extended this study by including the working vacation concept. The performance of machining system consisting of operating and warm standby machines was proposed by Kumar and Jain (2013a) for analyzing the control of both arrival and service based on threshold F-policy and N-policy, respectively. The transient analysis of F-policy Markovian retrial queue with finite capacity was done by Jain and Bhagat (2015).

In present scenario of modern technology, computer controlled fault tolerant machining system has become the necessity and it brought a tremendous change in the system design to control the risk of machine failure. Now-a-days, the machines are equipped with an inbuilt fault-handling mechanism which automatically detects the failure of a component and recovers the system by replacing the failed operating unit with a standby unit, if available. In many software embedded systems, in the case when the fault handling mechanism fails to detect and recover the faults, the machines can also reconfigure temporarily by reboot process. But in some practical situations, the fault-handling device may prove inadequate to recover a fault perfectly; this situation is known as imperfect coverage. In literature, a very few researchers have contributed toward the queueing and reliability analysis of machine repair problem with imperfect coverage. The cost–benefit analysis of MRP model with warm spares by including the imperfect coverage was carried out by Wang and Chiu (2006). Wang, Yen, and Jian (2013) proposed a Markovian model for MRP by incorporating the realistic assumptions of multiple types of imperfect coverage and state-dependent service rate using the pressure conditions. To determine the optimal control parameters, they have used quasi-Newton method and particle swarm optimization (PSO) algorithm by constructing a profit function. The provision of multiple vacation and imperfect coverage for the performance modeling of a repairable machining system was proposed by Jain and Gupta (2013). Later on, Jain, Shekhar, and Rani (2014) used matrix
method to explore the optimal N-policy for MRP by including some noble features such as unreliable server, imperfect coverage and reboot to make model more versatile and close to realistic situations. Ke and Liu (2014) studied a repairable system operating in failure prone environment with reboot delay, repair facility, and imperfect coverage.

The adaptive neuro-fuzzy inference system (ANFIS) presents a hybrid soft computing approach by using the features of both fuzzy logic and neural network. To explore the feasibility of the soft computing approach for the prediction of performance metrics of FTSs, ANFIS which employs the hybrid supervised learning algorithm based on gradient decent and least square methods has been implemented (Tong, 1979). For the modeling of FTS, the neuro-fuzzy inference model having the provision of neural network trained by using the available input/output data-sets, can be developed. In the context of automated machine repair system, the adaptive neuro-fuzzy controller can be easily designed for the prediction of optimal control parameters (Lin & Liu, 2001). As far as the applicability of ANFIS in queueing models is concerned, we cite some recent contributions related to investigation done in the present paper. Jain and Upadhyaya (2009) used ANFIS to match the soft computing based results with the analytical results obtained by matrix recursive method for the performance prediction of degraded multi-component machining system with switch over failure. They have developed the Markov model under more realistic assumptions such as N-policy and multiple vacations. K-heterogeneous servers and multiple vacations Markov model were proposed by Kumar and Jain (2013b) to analyze the machining system having operating as well as inventory of standby machines. Further, they have matched their results obtained by SOR with ANFIS generated results. Recently, the performance models of fault tolerant system incorporating some realistic features such as imperfect coverage, reboot, and server vacation have been developed by Jain and Meena (2016). In this investigation, they have presented the comparative study of results obtained by Runge-Kutta with hybrid soft computing technique ANFIS.

From the literature survey, it is evident that a very few research articles have appeared on the performance analysis of machining system with spare provisioning and operating under vacation policy. It is noticed that there is a research gap in the area of MRP with the option of working vacation or complete vacation. In many real-time systems, whenever server becomes idle, it may have the option to go for either complete vacation or working vacation. This situation of choice of vacation and working vacation can also be realized in machine repair systems. From the literature review, it is noticed that there is no work on queueing models developed so far by taking combination of vacation and working vacation. In the present investigation, we are concerned with Markov analysis for the performance prediction of FTS by developing machine repair model with unreliable server and provision of standby machines. We have also incorporated the feature of server’s choice of either go for the complete vacation or opt for working vacation in case, when the system becomes empty i.e. there is no repair job of failed machines. In case of no line up repair job, the server can either take complete vacation and remain idle or go for working vacation after taking set up time. The modeling of MRP with reboot and recovery processes can be implemented for the performance improvement of FTSs. To explore the performance metrics of the unreliable server machining system with standby support by incorporating assumptions of imperfect coverage, reboot, and recovery along with the option of complete vacation or working vacation, a Markov model in general set up can be framed. Motivated by this fact, in the present article, we develop Markov model for the unreliable multi-component fault tolerant system by including the features of (i) multiple types of warm standbys, (ii) F-policy, (iii) optional working vacation, (iv) startup time, (v) imperfect coverage. The noble feature of the present investigation is to allow the server, either to take full vacation or to continue the repairs to failed machines with lower rate (i.e. working vacation) during the vacation also. For the maintainability of FTS at optimum cost, the optimal value of control repair parameter is suggested.

The successive over relaxation (SOR) method has been used to solve the set of equations governing the model in order to determine the steady-state probabilities associated with different system states. After solving the set of equations governing the concerned FT model, the impact of system descriptors on the performance metrics is examined by taking an illustration and conducting
numerical simulation. The hybrid soft computing technique known as adaptive neuro-fuzzy inference system (ANFIS) is implemented to compare the results obtained by SOR method. The remaining contents of the investigation are structured in different sections. In Section 2, we describe the model whereas in Section 3, difference equations are constructed on the basis of birth–death process. In Section 4, various system performance metrics are formulated in terms of the steady-state probabilities. The detailed explanation of neuro-fuzzy model is provided in Section 5. In Section 6, we present the numerical simulation results and sensitivity analysis. The final Section 7 is devoted to the conclusion wherein scope of the model for the future design of FTS is also discussed.

2. Model description
Consider a finite population Markov M/M/1/K/V+W model under admission control F-policy for the performance analysis of the multi-component fault tolerant system. The fault tolerant machining system consists of M identical operating machines and is supported with k types of warm standbys and an unreliable server. There are $S_i (1 \leq i \leq k)$ standby machines of type $i$ such that the total standby machines are $S = S^{(k)} = S_1 + S_2 + \ldots + S_k$. It is assumed that $i$th $(1 \leq i \leq k - 1)$ type standbys are used before $(i + 1)$th type standbys to replace the failed machines. The operating as well as standby machines are prone to failure. The life time of operating (standby) machines are assumed to be exponentially distributed with parameter $\lambda(a)$. Whenever an operating machine breaks down, it is immediately replaced by the $i$th type of standby machine, if available. If all the standby machines are used in replacing the failed machines and some more machines fail, then the system operates in short mode till there are $m(<M)$ operating machines in the system. The system fails with the failure of $(M + S - m)$th machine, i.e. as soon as the number of operating machines drops below $m$. The switchover of failed machines is not perfect i.e. the switch over of the failed machine takes place by standby machine with the coverage probability $c$. Whenever the switchover of failed machine by standby machine is unsuccessful with probability $(1 - c)$, the system goes to unsafe mode. The recovery as well reboot processes are governed by the exponential distribution. We assume that in the unsafe mode, the system is automatically cleared by a reboot process with rate $r$.

Once the system becomes empty, i.e. there is no job of repair, the server can take either complete vacation with probability $\bar{p} (= 1 - p)$ or working vacation with probability $p$. But before going for the vacation (working vacation), the system also needs some set up time which is exponentially distributed with rate $\epsilon_0(a)$. The repair time of failed machines during normal busy period (working vacation) is assumed to be governed by exponential distribution with mean $1/\mu_a (1/\mu)$. The duration of the working vacation period (vacation period) follows the exponential distribution with mean $\theta_w^{-1}$. The server from vacation returns to working vacation (normal busy) mode with rate $\theta(b)$ after completing a random duration which is exponential distributed. The life time and repair time of the sever are assumed to be exponentially distributed with mean rate $a$ and $b$, respectively.

The control of the arrivals of failed machines in the system is done according to the F-policy which states that when the system capacity becomes full, from the working vacation (normal busy period) the server moves to F-policy mode by taking set up time $\xi_w^{-1} (\xi_b^{-1})$. In F-policy mode, the system forbids any broken down machines from entering in the system until workload of repair jobs of failed machines ceases to a pre-specified threshold level $F(0 \leq F \leq K - 1)$. When the system again reaches to the threshold level “F” of the queue length, the server takes a startup time governed by exponential distribution with parameter $\gamma$ after completion of set up, the failed machines start to enter in the system. It is assumed that all the stochastic processes, associated with the set up and vacation (working vacation), reboot and recovery, and life time and repair times of machines, which are involved in the system, are independent and follow the Markovian property.

The bivariate Markov process $\chi(t) = \{(\xi(t), \eta(t)); t \geq 0\}$ is used to develop the Markov model. Here $\eta(t)$ and $\xi(t)$ denote the number of failed machines in the system and the state of the server at time $t$, respectively.

Page 5 of 17
The steady-state probabilities associated with state space (see Figure 1)

\[
\chi = \{(i, n)i = 0, 1, 2, \ldots, 5; n = 0, 1, \ldots, K\} \cup \{(i, n)i = 6, 7, \ldots, 13; n = 1, 1, \ldots, K - 1\}
\]

are denoted by \(P_{i, n} = \lim_{t \to \infty} \text{Prob.}\{\xi(t) = i, \eta(t) = n\}\).

3. Governing equations

To evaluate the queue size distribution of the number of failed machines in the system, the governing equations are framed by setting the appropriate rates of underlying birth–death process. The law
of flow balance is used to construct these equations. Now, we define the failure rate depending upon the number of failed machines as follows:

\[
\lambda_n = \begin{cases} 
M \lambda + \left( \sum_{i=2}^{k} S_i \alpha_i + (S_1 - n) \alpha_1 \right); & 0 \leq n < S_1 \\
M \lambda + \left( S^{(j-1)} - n \right) \alpha_{j-1} + \sum_{i=j}^{k-1} S_i \alpha_i; & S^{(j-1)} \leq n < S^{(j)}; j = 2, 3, 4, \ldots, k - 1 \\
M \lambda + \left( S^{(k)} - n \right) \alpha_k; & S^{(k-1)} \leq n < S^{(k)} \\
(M + S^{(k)} - n) \lambda; & S^{(k)} \leq n \leq K = M + S^{(k)} - (m - 1) 
\end{cases}
\]

For the brevity, we have used the notation for failure rate of standby machines by

\[
a_n = \begin{cases} 
\left( \sum_{i=2}^{k} S_i \delta_i + (S_1 - n) \delta_1 \right); & 0 \leq n < S_1 \\
\left( S^{(j-1)} - n \right) \delta_{j-1} + \sum_{i=j}^{k-1} S_i \delta_i; & S^{(j-1)} \leq n < S^{(j)}; j = 2, 3, 4 \ldots, k - 1 \\
\left( S^{(k)} - n \right) \delta_k; & S^{(k-1)} \leq n < S^{(k)} 
\end{cases}
\]

(i) Server vacation (VAC) state when \( \xi(t) = 0 \):

As soon as the server becomes free i.e. when there is no job of repairing of the failed machines in the system, it reaches to vacation state. Now, we frame the Chapman-Kolmogorov equations for the states \( (0, n); n = 0, 1, \ldots, K \) as follows:

\[
(\lambda_0 + a_0 + \theta_b + \theta_v) P_{0,0} = \epsilon_z P_{1,0} + \rho \cdot P_{2,0} + \rho \cdot P_{4,0} 
\]

\[
(\lambda_n + a_n + \theta_b + \theta_v) P_{0,n} = a_{n-1} P_{0,n-1} + \sigma P_{0,n-1} + r P_{0,n-1}; \quad 1 \leq n \leq S - 1 
\]

\[
(\lambda_2 + \theta_v) P_{2,0} = a_{1} P_{2,0} + \sigma P_{1,0} + r P_{1,0} 
\]

\[
(\lambda_n + \theta_v) P_{0,0} = \sigma P_{0,n-1} + r P_{0,n-1}; \quad S + 1 \leq n \leq K - 1 
\]

\[
(\theta_v + \theta_b) P_{0,K} = \lambda_{K-1} P_{0,K-1} 
\]

(ii) Server working vacation (BWV) state when \( \xi(t) = 1 \):

In case of working vacation state of the server, the following steady-state equations hold:

\[
(\lambda_0 + \epsilon_v + \alpha + a_0 + \theta_w) P_{1,0} = \mu_v P_{2,0} + \beta P_{2,0} + \rho \cdot P_{3,0} + \rho \cdot P_{4,0} + \theta_v P_{0,0} 
\]

\[
(\lambda_n + a_n + \alpha + \theta_w + \mu_v) P_{1,n} = \mu_v P_{0,n} + \sigma P_{1,n-1} + r P_{1,n-1} + a_{n-1} P_{1,n-1} + \mu P_{1,n+1} + \beta P_{2,n}; \quad 1 \leq n \leq S - 1 
\]

\[
(\lambda_5 + a + \theta_w + \mu_v) P_{1,5} = \mu_v P_{1,5} + \sigma P_{1,5} + r P_{1,5} + a_{5} P_{1,5} + \mu P_{1,5} + \beta P_{2,5} 
\]

\[
(\lambda_n + \alpha + \theta_w + \mu_v) P_{1,n} = \mu_v P_{0,n} + \sigma P_{1,n-1} + r P_{1,n-1} + \mu P_{1,n+1} + \beta P_{2,n}; \quad S + 1 \leq n \leq K - 1 
\]

\[
(\alpha + \epsilon_v + \theta_w) P_{1,K} = \theta_v P_{0,K} + \lambda_{K-1} P_{1,K-1} + \beta P_{2,K} 
\]

(iii) Server broken down state (DBW) when \( \xi(t) = 2 \):

When the server is under repair while broken down during working vacation \( J(t) = 1 \) and normal busy state \( \xi(t) = 3 \), for the states \( (2, n); n = 0, 1, 2, \ldots, K' \), we frame the steady-state equations as follows:
\( (\lambda_0 + \alpha_0 + 2 \beta) P_{2,0} = aP_{1,0} + aP_{3,0} \)  

(11)

\( (\lambda_n + \alpha_n + 2 \beta) P_{2,n} = aP_{1,n} + aP_{3,n} + \sigma P_{2,n-1} + r P_{2,n-1} + a_{n-1} P_{2,n-1} \quad 1 \leq n \leq S - 1 \)  

(12)

\( (\lambda_S + 2 \beta) P_{2,S} = aP_{1,S} + aP_{3,S} + \sigma P_{2,S-1} + r P_{2,S-1} + a_{S-1} P_{2,S-1} \)  

(13)

\( (\lambda_n + 2 \beta) P_{2,n} = aP_{1,n} + aP_{3,n} + \sigma P_{2,n-1} + r P_{2,n-1} \quad S + 1 \leq n \leq K - 1 \)  

(14)

\( 2 \beta P_{2,K} = aP_{1,K} + aP_{3,K} + \dot{\lambda}_{K-1} P_{2,K-1} \)  

(15)

(iv) Server normal busy (NOB) state when \( \xi(t) = 3 \):

When the server is operating in busy mode to provide the repair of the failed machines, the steady-state equations are framed for the states \((3, n); n = 0, 1, \ldots, K\) as follows:

\( (\lambda_0 + \alpha_0 + \alpha + \beta \epsilon_0) P_{3,0} = \beta P_{2,0} + \mu_b P_{3,1} + \theta_{P,0} P_{1,0} + \theta_{w} P_{1,0} \)  

(16)

\( (\lambda_n + \alpha_n + \alpha + \mu_b) P_{3,n} = \theta_{P,0} P_{1,n} + \theta_{w} P_{1,n} + \beta P_{2,n} + \sigma P_{3,n-1} + r P_{3,n-1} + a_{n-1} P_{3,n-1} + \mu_b P_{3,n+1} + \gamma P_{3,n} ; \quad 1 \leq n \leq S - 1 \)  

(17)

\( (\lambda_S + \alpha + \mu_b) P_{3,S} = \theta_{P,0} P_{1,S} + \theta_{w} P_{1,S} + \beta P_{2,S} + \sigma P_{3,S-1} + r P_{3,S-1} + a_{S-1} P_{3,S-1} + \mu_b P_{3,S+1} + \gamma P_{3,S} \)  

(18)

\( (\lambda_n + \alpha + \mu_b) P_{3,n} = \theta_{P,0} P_{1,n} + \theta_{w} P_{1,n} + \beta P_{2,n} + \sigma P_{3,n-1} + r P_{3,n-1} + \mu_b P_{3,n+1} ; \quad S + 1 \leq n \leq F \)  

(19)

\( (\alpha + \xi_0) P_{3,K} = \beta P_{3,K} + \mu_b P_{3,K} + \dot{\lambda}_{K-1} P_{3,K-1} \)  

(20)

(v) Server is in normal busy state and the admission of failed machines is not allowed (BNF) when \( \xi(t) = 4 \):

Here steady-state equations are framed for the states \((4, n); n = 0, 1, \ldots, K\) as

\( (\alpha + \beta \epsilon_0) P_{4,0} = \mu_f P_{4,1} + \beta P_{5,0} \)  

(22)

\( (\alpha + \gamma + \mu_f) P_{4,n} = \mu_f P_{4,n+1} + \beta P_{5,n} \quad 1 \leq n \leq F \)  

(23)

\( (\alpha + \mu_f) P_{4,n} = \mu_f P_{4,n+1} + \beta P_{5,n} \quad F + 1 \leq n \leq K - 1 \)  

(24)

\( (\alpha + \mu_f) P_{4,K} = \beta P_{4,K} + \xi_b P_{4,K} + \beta P_{5,n} \)  

(25)

(vi) Server is in broken down state from busy state (DBF) when \( \xi(t) = 5 \):

In this case for states \( \xi(t) = 5 \), the steady-state equation is:

\( \beta P_{5,n} = aP_{4,n} \quad 0 \leq n \leq K \)  

(26)
(vii) Server is in recovery state when \( \xi(t) = 6, 8, 10, 12 \):

From \( \xi(t) = 0, 1, 2, 3 \) states, due to failure detection with probability \( c \), the system can go to recovery state \( \xi(t) = 6, 8, 10, 12 \). For the recovery states \( (i, n) \), the steady-state equations are framed as:

\[
\sigma P_{6+2i, n+1} = \lambda_n c P_{i, n}; \quad i = 0, 1, 2, 3; \quad 1 \leq n \leq K - 1
\]  

(viii) Server is in reboot state when \( \xi(t) = 7, 9, 11, 13 \):

From \( \xi(t) = 0, 1, 2, 3 \) state, due to unsuccessful failure detection with probability \( 1 - c \), the system can go to reboot state i.e. \( \xi(t) = 7, 9, 11, 13 \) states. Here, the steady-state equations are framed as:

\[
r P_{7+2i, n+1} = \lambda_n c P_{i, n}; \quad i = 0, 1, 2, 3; \quad 1 \leq n \leq K - 1
\]

Chapman-Kolmogorov equations framed for the continuous Markov chain of concerned model are difficult to solve analytically due to cumbersome algebraic manipulation involved in recursive approach. However, numerical method can be easily implemented to obtain the probabilities associated with a large state space for which steady-state equations are already constructed. In the present investigation, the steady-state Equations (1)–(28) are solved numerically by using well-known successive over relaxation (SOR) method, which is a powerful tool for the computation purpose of set of equations. In this method, the convergence rate is accelerated by selecting the appropriate relaxation parameter lying in interval [1, 1.25].

4. Performance measures

The prime aim of determining probabilities in previous section is to formulate the various metrics to examine the performance of the concerned fault tolerant system. The expressions for the mean queue length of the failed machines in the system, effective joining rate of failed machines, throughput of the system, etc. are established as follows:

4.1. Queueing indices

(i) Mean queue size of failed machines \( E(N) \) is

\[
E[N] = \sum_{i=0}^{5} \sum_{n=0}^{K} n P_{i, n} + \sum_{i=6}^{13} \sum_{n=1}^{K-1} n P_{i, n}
\]  

(ii) The throughput \( \tau \) is

\[
\tau = \sum_{n=1}^{K-1} \mu_1 P_{1, n} + \sum_{n=1}^{K-1} \mu_2 P_{3, n} + \sum_{n=1}^{K} \mu_4 P_{4, n}
\]

(iii) Average number of available standby machines is

\[
E[S] = \sum_{i=0}^{5} \sum_{n=0}^{5} (S - n) P_{i, n} + \sum_{i=6}^{13} \sum_{n=0}^{5} (S - n) P_{i, n}
\]

(iv) The effective rate by which failed machines join the waiting queue, is given by

\[
\lambda_{eff} = \sum_{i=0}^{3} \sum_{n=0}^{S-1} (\lambda_n + g_n) P_{i, n} + \sum_{i=0}^{K} \sum_{n=5}^{K} \lambda_{n-S} P_{i, n}
\]
Machine availability is obtained using

\[ MA = 1 - \frac{E(N)}{M + \delta} \]  

### 4.2. Long run probabilities

Now we establish long run probabilities associated with different states of the server which may be busy (\(P_B\)), broken down and under repair (\(P_{BD}\)), on vacation (\(P_V\)) and on working vacation (\(P_{WV}\)), respectively. Thus

\[ P_B = \sum_{n=1}^{K-1} P_{1,n} + \sum_{n=1}^{K-1} P_{3,n} + \sum_{n=1}^{K} P_{4,n} \]  

\[ P_{BD} = \sum_{n=0}^{K} P_{2,n} + \sum_{n=1}^{K-1} P_{10,n} + \sum_{n=1}^{K-1} P_{11,n} + \sum_{n=0}^{K} P_{5,n} \]  

\[ P_V = \sum_{n=0}^{K} P_{0,n} + \sum_{n=1}^{K-1} P_{6,n} + \sum_{n=1}^{K-1} P_{7,n} \]  

\[ P_{WV} = \sum_{n=0}^{K} P_{1,n} + \sum_{n=1}^{K-1} P_{8,n} + \sum_{n=1}^{K-1} P_{9,n} \]  

### 4.3. Cost function

To quantify per unit time total cost \(TC(\mu)\) spent for the system, the various cost factors related to several system indices of Markovian model of fault tolerant system are taken into consideration. Now, we define per unit cost related to different activities as follows:

- \(C_H\) : Holding cost per unit time associated with each down machine
- \(C_B\) : Cost per unit time incurred when the server is in normal busy state
- \(C_{BD}\) : Cost per unit time incurred when the server is broken down and is under repair
- \(C_V\) : Cost per unit time incurred when the server is on vacation
- \(C_{WV}\) : Cost incurred when the server is in working vacation state
- \(C_f\) : Cost incurred for providing service to the failed machines when the admission of failed machines are not allowed
- \(C_a\) : Cost incurred for providing the repair to the failed machines when the admission of failed machines are allowed

The total cost per unit time incurred on the system is framed by summing different cost factors multiplied by respective system indices as follows:

\[ TC(\mu) = C_H E[N] + C_B P_B + C_{BD} P_{BD} + C_V P_V + C_{WV} P_{WV} + C_f \mu_f + C_a \mu_b \]  

### 5. Adaptive-Neuro fuzzy inference system (ANFIS) model

The hybrid soft computing approach ANFIS is a neural network based representation of fuzzy systems equipped with learning capabilities. In fuzzy rule-based ANFIS, the rules can be formulated as

IF \( u_1 \) is \( A_1 \) AND \( u_2 \) is \( A_2 \) … AND \( u_n \) is \( A_n \) THEN \( v = f(u_1, u_2, \ldots, u_n) \)

Here \( f \) is a linear combination of the input variables \((u_1, u_2, \ldots, u_n)\), and \(A_i\)’s are the associated fuzzy sets.
where \( w_0, w_1, \ldots, w_n \) are real constants. This is a particular case of the weighted average method of defuzzification.

The ANFIS has a number of layers where each layer has a number of nodes. For our FTS model, a fuzzy inference system with one input parameter (say \( \lambda \)) and one output \( E[N] \) can be described by the following \( n \) rules (Takagi & Sugeno, 1985):

\[
\text{IF } \lambda \text{ is } A_1 \text{ THEN } f_1 = p_1 \lambda + r_1 \\
\text{IF } \lambda \text{ is } A_2 \text{ THEN } f_2 = p_2 \lambda + r_2 \\
\vdots \\
\text{IF } \lambda \text{ is } A_n \text{ THEN } f_n = p_n \lambda + r_n
\]

Let \( O_{i,j} \) be the output of node \( i \) in layer \( l \). Thus, the functionalities of the layer architecture of ANFIS can be explained briefly as follows:

**Layer 1:** Each node in the 1st layer is an adaptive unit with output

\[
O_{1,i} = \mu_{A_{i}(\lambda)} = w_i \quad i = 1, 2, \ldots, n
\]

Here \( w_i \) is the firing strength of each node. In our model, the shape of membership function for each \( A_i \) is taken as Gaussian.

**Hidden layer “j” layer 2:** For each node in hidden layer output is obtained using\n
\[
O_{j,i} = \frac{w_i}{\sum_{i=1}^{n} w_i}, \quad i = 1, 2, \ldots, n; \quad j = 1, 2, \ldots, k - 1
\]

**Layer 3:** For each node in the layer 3 the output is obtained as

\[
O_{j+1,i} = \frac{w_i f_i}{\sum_{j=1}^{k-1} w_i f_j}, \quad i = 1, 2, \ldots, n; \quad j = 1, 2, \ldots, k - 1
\]

**Layer 4:** Considering the single node in output layer, the overall output is determined by

\[
O_{k+1,1} = \frac{\sum_{i=1}^{n} w_i f_i}{\sum_{i=1}^{n} w_i}
\]

### 6. Numerical results

To reveal the practical applicability of multi-component fault tolerant system operating in real-time machining environment, numerical illustration is taken for finite capacity model. To compute numerical results, we consider the constant failure rate of operating and standby units and fix the various parameters as

\[
K = 10, \ M = 8, \ S = 4, \ m = 2, \ c = 0.5, \ a = 0.2, \ \lambda = 0.5, \ \mu = \mu_b = \mu_f = 4, \ \alpha = 0.05, \ \beta = 2, \\
\sigma = 1, \ r = 1.5, \ \gamma = 1.5, \ \theta = \theta_b = \theta_w = 1.5, \ \xi_b = 0.5, \ \xi_w = 0.5, \ \varepsilon = 1.5, \ \epsilon = 1.0, \ \eta = 0.6.
\]

The sensitivity of parameters has been examined to reveal the impact of varying system descriptors on different system metrics. The numerical results displayed in the form of graphs can be easily interpreted to understand the behavior of FTS system. The optimal repair rate and associated minimum total cost are obtained for the two sets of cost factors and are displayed in Table 1.
The optimal repair rate \( \mu \) is obtained by computing the cost TC which is also depicted in Figures 2 and 3. Table 2 depicts the optimal repair rate and corresponding optimal total cost \( TC(\mu^*) \) for different sets of cost elements.

The impact of system descriptors on different indices are examined by displaying the numerical results in Tables 3 and 4 and Figures 5–8. The expected number of failed machines summarized in Tables 3 and 4 indicates that \( E[N] \) increases as \( \lambda \) grows up but decreases as \( \mu \) increases. The long run probabilities \( P_{BD}, P_B \) and total cost incurred on the system also increases as \( \lambda \) increases but lowers down as \( \mu \) increases. It is also found that the mean number of standby machines decreases (increases) as \( \lambda(\mu) \) increases. From Tables 3 and 4, we notice that the impact of \( c \) on various system indices is also significant.

Neuro-fuzzy technique is used to demonstrate the feasibility of soft computing approach for the quantitative assessment of various performance indices of the fault tolerant MRP in particular when input parameters are not crisp. The results by ANFIS approach have been computed by using

<table>
<thead>
<tr>
<th>Cost set</th>
<th>( C_a )</th>
<th>( C_b )</th>
<th>( C_{BD} )</th>
<th>( C_v )</th>
<th>( C_{wv} )</th>
<th>( C_r )</th>
<th>( C_s )</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>$80</td>
<td>$20</td>
<td>$30</td>
<td>$5</td>
<td>$20</td>
<td>$1</td>
<td>$6</td>
</tr>
<tr>
<td>II</td>
<td>$70</td>
<td>$10</td>
<td>$10</td>
<td>$5</td>
<td>$20</td>
<td>$1</td>
<td>$6</td>
</tr>
</tbody>
</table>

Figure 2. Variation of TC with respect to \( \mu \) for different values of \( \gamma \).

Figure 3. Variation of TC with respect to \( \mu \) for different values of \( \theta \).
The failure rate ($\lambda$) is treated as linguistic variable in the context of the fuzzy system. The membership function for the failure rate of operating machine ($\lambda$) is considered as Gaussian function.

### Table 2. Optimal repair rate $\mu^*$ and optimal cost

<table>
<thead>
<tr>
<th>$(\gamma, \theta)$</th>
<th>$\mu^*$</th>
<th>Optimal cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>(0.5, 1)</td>
<td>7.967</td>
<td>$300.7518$</td>
</tr>
<tr>
<td>(1, 1)</td>
<td>8.479</td>
<td>$295.1175$</td>
</tr>
<tr>
<td>(1.5, 1)</td>
<td>8.802</td>
<td>$330.6238$</td>
</tr>
<tr>
<td>(1.5, 0.5)</td>
<td>8.084</td>
<td>$330.6238$</td>
</tr>
<tr>
<td>(1.5, 1.5)</td>
<td>8.366</td>
<td>$337.9267$</td>
</tr>
<tr>
<td>(1.5, 2.5)</td>
<td>8.518</td>
<td>$343.4445$</td>
</tr>
</tbody>
</table>

### Table 3. Effect of $\lambda$ and $c$ on various system indices

<table>
<thead>
<tr>
<th>$c$</th>
<th>$\lambda$</th>
<th>$E[N]$</th>
<th>$E[S]$</th>
<th>$P_{BD}$</th>
<th>$P_\delta$</th>
<th>TC</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>0.5</td>
<td>3.21</td>
<td>2.712</td>
<td>0.00938</td>
<td>0.1732</td>
<td>277.4</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>4.11</td>
<td>2.042</td>
<td>0.01398</td>
<td>0.2241</td>
<td>322.5</td>
</tr>
<tr>
<td></td>
<td>1.5</td>
<td>4.51</td>
<td>1.773</td>
<td>0.01668</td>
<td>0.2475</td>
<td>342.3</td>
</tr>
<tr>
<td>0.6</td>
<td>0.5</td>
<td>3.21</td>
<td>2.710</td>
<td>0.00935</td>
<td>0.1722</td>
<td>277.4</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>4.10</td>
<td>2.044</td>
<td>0.01393</td>
<td>0.2226</td>
<td>322.2</td>
</tr>
<tr>
<td></td>
<td>1.5</td>
<td>4.49</td>
<td>1.779</td>
<td>0.01661</td>
<td>0.2457</td>
<td>341.6</td>
</tr>
<tr>
<td>0.8</td>
<td>0.5</td>
<td>3.21</td>
<td>2.707</td>
<td>0.00933</td>
<td>0.1713</td>
<td>277.4</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>4.10</td>
<td>2.046</td>
<td>0.01388</td>
<td>0.2211</td>
<td>321.8</td>
</tr>
<tr>
<td></td>
<td>1.5</td>
<td>4.48</td>
<td>1.784</td>
<td>0.01654</td>
<td>0.2440</td>
<td>341.0</td>
</tr>
</tbody>
</table>

### Table 4. Effect of $\mu$ and $c$ on various system indices

<table>
<thead>
<tr>
<th>$c$</th>
<th>$\mu$</th>
<th>$E[N]$</th>
<th>$E[S]$</th>
<th>$P_{BD}$</th>
<th>$P_\delta$</th>
<th>TC</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.3</td>
<td>2</td>
<td>5.082</td>
<td>0.868</td>
<td>0.0335</td>
<td>0.496</td>
<td>280.3</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>4.732</td>
<td>1.293</td>
<td>0.0284</td>
<td>0.412</td>
<td>269.3</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>4.631</td>
<td>1.496</td>
<td>0.0252</td>
<td>0.359</td>
<td>271.2</td>
</tr>
<tr>
<td>0.6</td>
<td>2</td>
<td>5.076</td>
<td>0.869</td>
<td>0.0334</td>
<td>0.491</td>
<td>279.9</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>4.722</td>
<td>1.296</td>
<td>0.0283</td>
<td>0.407</td>
<td>268.8</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>4.618</td>
<td>1.501</td>
<td>0.0251</td>
<td>0.355</td>
<td>270.5</td>
</tr>
<tr>
<td>0.8</td>
<td>2</td>
<td>5.070</td>
<td>0.871</td>
<td>0.0333</td>
<td>0.486</td>
<td>279.5</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>4.713</td>
<td>1.299</td>
<td>0.0282</td>
<td>0.403</td>
<td>268.2</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>4.606</td>
<td>1.506</td>
<td>0.0251</td>
<td>0.351</td>
<td>269.8</td>
</tr>
</tbody>
</table>

### Table 5. Linguistic values of the membership functions for input parameter $\lambda$

<table>
<thead>
<tr>
<th>Input variable</th>
<th>No. of membership functions</th>
<th>Linguistic values</th>
</tr>
</thead>
</table>
| Failure rate of operating unit $\lambda$ | 5 | • Very low
|                      |                      | • Low
|                      |                      | • Average
|                      |                      | • High
|                      |                      | • Very high

The failure rate ($\lambda$) is treated as linguistic variable in the context of the fuzzy system. The membership function for the failure rate of operating machine ($\lambda$) is considered as Gaussian function.
Table 5 provides the linguistic values of membership functions corresponding to the input parameter $\lambda$. The shape of the corresponding membership function treated as Gaussian functions is depicted in Figure 4. The numerical results corresponding to ANFIS are plotted by tick marks in Figures 5–8 whereas the continuous curves are drawn for the results computed by using SOR method.

From Figure 5, we see that as the rate of failed machine ($\lambda$) increases, $E[N]$ initially increases rapidly and then after becomes almost constant. The trend of $E[N]$ is plotted in Figure 5; a sharp increment is noticed up to $\lambda = 1$, and then after it becomes asymptotically stable as $\lambda$ grows. From Figures 6 and 7, it is clear that the machine availability ($MA$) and expected number of standby machines $E[S]$ decrease rapidly initially but as $\lambda$ grows, these indices become almost constant i.e. the higher value of $\lambda$ has negligible impact on $E[S]$ and machine availability ($MA$). From Figure 8, it is clearly seen that as failure rate of operating unit ($\lambda$) grows up, the throughput of the system ($TP$) increases rapidly initially and then after gradually becomes almost constant.

Figures 5–8 exhibit almost coincident values for both analytical and ANFIS results. Based on critical and comparative analysis of graphs, we conclude that the SOR results are very close to the results shown by neuro-fuzzy results as such neuro-fuzzy controller can be developed for the FTS to track the performance of many real-time embedded systems.
7. Conclusion

In this investigation, we have analyzed Markov model for F-policy based admission control of the M/M/1/K/V+WV multi-component fault tolerant system by incorporating several realistic features such as mixed standbys, server breakdown, options of complete vacation or working vacation, re-boots and imperfect coverage, set up time and start up times, etc. Numerical simulation carried out provides the valuable insights for the sensitivity of several descriptors on the different indices such as the queue length of failed machines, throughput, machine availability, etc. The minimum cost and optimal repair rate obtained using direct search method reveal the applicability of present model for the future design and improvement of the existing systems. Moreover, the successful implementation of ANFIS demonstrates the utility of hybrid soft computing approach to generate the performance results for the real-time FTS by developing ANFIS controller. It would be possible to extend the present investigation by incorporating the more realistic features of bulk failure and N-policy or provision of replacement along with repair of failed machines.


