British Journal of Mathematics & Computer Science
17(1): 1-15, 2016, Article no.BJMCS.24515
ISSN: 2231-0851

SCIENCEDOMAIN *international*
www.sciencedomain.org

Transient Solution for Single Server Machine Interference Problem with Additional Server for Long Queues under *N*-Policy Vacations

S. A. Ojobor[1*] and S. E. Omosigho[2]

[1]*Department of Mathematics and Computer Science, Faculty of Science, Delta State University, Abraka, Nigeria.*
[2]*Department of Mathematics, University of Benin, Benin City, Nigeria.*

Authors' contributions

This work was carried out in collaboration between both authors. Both authors read and approved the final manuscript.

Article Information

DOI: 10.9734/BJMCS/2016/24515
Editor(s):
(1) Dariusz Jacek Jakóbczak, Chair of Computer Science and Management in this Department, Technical University of Koszalin, Poland.
(2) Tian-Xiao He, Department of Mathematics and Computer Science, Illinois Wesleyan University, USA.
Reviewers:
(1) Radek Matusu, Tomas Bata University in Zlín, Czech Republic.
(2) Kek Sie Long, Universiti Tun Hussein Onn Malaysia, Malaysia.
(3) Hanifi Okan Isguder, Dokuz Eylul University, Turkey.
(4) Anonymous, Technical University of Ostrava, Czech Republic.
Complete Peer review History: http://sciencedomain.org/review-history/14759

Original Research Article

Received: 23rd January 2016
Accepted: 8th April 2016
Published: 24th May 2016

Abstract

We consider the transient state single server machine interference problem with additional server for long queues under *N*-policy vacations. There are M operating machines with two repairmen. The first repairman is always available for serving the failed machines but go on a single vacation when there are no failed machines in the system. The second repairman is always on vacation but only comes back from vacation to attend to broken down machines if there are more than or equal to N broken down machine in queue in the system (*N*-policy vacations). Otherwise he goes for another vacation. The number of servers available for service in this system is two. The service discipline is first in first out (FIFO). The Chapman-Kolmogorov differential equations obtained for the model is solved through ODE45 in MATLAB. The transient probabilities obtained for the model are used to compute the expected number of failed machines $E[F]$, expected number of operating machine $E[O]$, expected length of vacation the servers has $E[V]$, the machine availability at time t ($M.A.(t)$) and variance of the number of broken down machines $\sigma^2(t)$ for the systems. We investigate the effect of CPU time and different parameters on

*Corresponding author: E-mail: ojoborsun@yahoo.com;

the availability of the machine for the single server machine interference problem with additional server for long queues. We found that with the same service rate μ, failure rate λ and vacations length θ, as the number of failed machines that trigger repairman 2 in the system increases the variance is less than one. This is caused by the additional repairman. The additional repairman reduces the waiting time of failed machines in the system.

Keywords: Machine interference problem; MATLAB; N-policy vacations.

1 Introduction

Yue et al. [1] studied the machine repair vacation model with warm spares and two repairmen. The first repairman is always available for serving the failed units while the second repairman leaves for a vacation of random length when the number of failed units is less than N. The second repairman returns from vacation if there are N or more failed units accumulated in the system (N-policy); otherwise he goes for another vacation. Some performance measures for the queueing and the reliability of the system were obtained by them. Furthermore, they developed a cost model to determine the optimum N while the system availability is maintained at a certain level.

In similar note Yue et al. [2] considered the machine interference problem consisting of warm spares and two heterogeneous repairmen. One repairman is always available while the other repairman can proceed on vacation of random length when the failed units are less than N, a fixed number. Steady state measures of performance of the system were obtained and a cost model was used to determine the optimum value of N. A recursive method was used to obtain the steady state measures of performance while a heuristic method was adopted for the optimization problem to determine N.

Also Sharma [3] studied the machine interference problem consisting of M operating machines with S spare machines (cold standby or warm standby or hot standby machines) and R servers. The machines have two failure modes and the servers are unreliable, i.e. they are subjected to fail or breaks down. Sharma [3] developed the Chapman-Kolmogorov steady state equations for obtaining the probability of failed machines in the system and proposed that a recursive method can be used to obtain the results when $R = 1$ and that the solution will require a computer program for $R > 1$. Sharma [3] gave no indication of the behaviour of the results or the organization of the computer program. Sharma [3] studied machine repairable system with spares and two repairmen. One repairman is always available for serving the failed machine while the other server is always on vacations when the queue length is less than N. This type of vacation is called 'the partial server vacation'. At the end of vacation period the second repairman comes back from vacation if there are N or more failed machines in the system- N policy vacation, otherwise he goes for another vacation. The steady state measures of performance were derived and used to propose a procedure for obtaining optimal N. The system studied by Sharma [3] is similar to systems where additional servers are provided for long queues.

In another facet Maheshwari and Ali [4] studied a machine repair problem with warm and cold spares, balking and reneging. In the system, there are R permanent repairmen, r additional removable repairmen, M regular machines, $S1$ warm standby machines and $S2$ cold standby machines. The system works with at least m operating units where but for normal functioning $M>m$ units are required. If a regular machine fails, it is replaced by a cold standby machine if available; otherwise it is replaced by a warm standby machine. The additional repairmen are engaged when the number of failed machines is more than R. The recursive method was used to obtain steady state measures of performance. A cost minimization procedure was used to obtain the optimal number of spares and repairmen. Other authors that studied additional servers are Al-Seedy and Al-Ibraheem [5], Jain et al. [6].

Also Jain and Kumar [7] studied the machine repair problem consisting of two heterogeneous servers and mixed spares (warn and cold). Their two repairmen can go on vacation using two different N policies. Further, the two repairmen are used under different conditions. Failed machines are immediately replaced by spare machines (either a cold or a warm spare). A bi-level control policy was used to introduce the servers into the system. They applied Recursive method to derive steady state measures of performance.

Jain et al. [8] studied multi-component machine repair model consisting of two heterogenous server (primary and secondary). The failure of operating and standby units may occur individually or due to some common cause. The primary server may fail partially following full failure whereas secondary server faces complete failure only. The life times of servers and operating/standby units and their repair times is exponential distribution. They use the successive over relaxation (SOR) technique to obtain the steady state queue size distribution of the number of failed machines in the system.

Recently Ojobor [9] considered transient solution of machine interference problem with an unreliable server under multiple vacations policy. Their server is unreliable, that is when the server is active it can break down. Anytime the server breaks down it is immediately repaired. The server goes on multiple vacations.

Our work can be compared to the works of Yue et al. [1] and Sharma [3]. These articles assumed that their repairman 1 is always available for serving the failed machines. But here we assume that repairman 1 can go on vacation when there are no failed machines in the system. Also Yue et al. [1] and Sharma [3] only compute the optimum N in their steady state results for the system. But here we shall use transient state probabilities to compute the expected number of failed machine, the expected number of operating machines and the machine availability in the system.

2 Mathematical Formulations

We shall follow the treatment given by Yue et al. [1]

We describe the state of the system at epoch t by two variables namely: The number of failed machines in the system and the server rate. We assume that repairman 1 and 2 can go on vacation when there are no failed machines in the system.

2.1 Assumptions and notation

Throughout this section, we shall adopt the following assumptions and notation:

(i) Let the state of the system at epoch t be denoted by (i, n); $i=0, 1, 2$; $0\leq n\leq M$; where i is the state of the repairmen, and n is the number of failed machines in the system. M the number of operating machines in the system. When i=0, both repairman 1 and repairman 2 are on vacation, when i=1 repairman 1 is active, serving failed machines while repairman 2 is on vacation and when i=2 both repairmen are active.
(ii) The machines fail or arrive for service according to Poisson distribution with rate λ_n where n is the number of failed machine.
(iii) The failed machines are serviced (repaired) according to exponential distribution with rate μ_1 and μ_2, where μ_1 is the service rate of repairman 1 and μ_2 is the service rate of repairman 2.
(iv) When there are no failed machines queueing for service the servers go on vacations of random length. The vacation length is exponentially distributed with parameters θ_1 and θ_2, where θ_1 is vacation length for repairman 1 and θ_2 is the vacation length of repairman 2.
(v) The activation of the repairman 2 depends on the activation of the repairman 1. That is repairman 2 is active if and only if the repairman 1 is active.
(vi) The number of break down machines in the system is finite.

Consequently, the notations used are listed as follow:

M: number of operating machines
N: number of failed machines that trigger repairman 2

λ_n: Failure rate of the operating machine
μ_1: Service rate of repairman 1
μ_2: Service rate of repairman 2
θ_1: Vacation length of repairman 1
θ_2: Vacation length of repairman 2
$P_{0,n}$(t): The probability that there are *n* failed machines in the system when repairmen 1 and 2 are on vacation at time *t*
$P_{1,n}$(t): The probability that there are *n* failed machines in the system when repairman 1 is active serving failed machines at time *t*
$P_{2,n}$(t): The probability that there are *n* failed machines in the system when both repairmen are active serving failed machines at time *t*

Let *N (t)* be the number of exact failed machines in the system at time *t*, and $Y(t)$ the server state at time *t*, where

$$Y(t) = \begin{cases} 0 \text{ repairmen 1 and 2 are on vacation at time } t \\ 1 \text{ only repairman 1 is active at time } t \\ 2 \text{ repairmen 1 and 2 are active at time } t \end{cases}$$

The bivariate process $Y(t), N(t): t \geq 0\}$ is a continuous time

Markov process on a state space

$$s = \{(0,n): n = 0,1,2,\dots,N\} \cup \{(1,n): n = 0,1,2,\dots,M\} \cup \{(2,n): n = N, N+1,\dots,M\}.$$

We define the probabilities of the server state at time *t* for a certain number of exact failed machines as follow:

$$P_{0,n}(t) = prob\{Y(t) = 0, N(t) = n\}$$
$$P_{1,n}(t) = prob\{Y(t) = 1, N(t) = n\}$$
$$P_{2,n}(t) = prob\{Y(t) = 2, N(t) = n\}$$

2.2 Transient probability under *N*-policy vacation

Using elementary probability argument we shall derive transient probability for the system under *N*-policy vacation. We derive the number of broken down machines for the system between 1 and *N*-2.

The probability that there are no broken down machines when the server is on vacations in the interval [*t*, *t*+*h*] is obtained as follows: consider the state of the system between *t* and *t*+*h*, the first possibility is that at epoch *t* the servers are on single vacation, no failed machine arrive and no service completion during the interval *t* and *t*+*h*. This has probability $P_{0,0}(t)[1-\theta_1 h][1-\theta_2 h]$

The second possibility is that at epoch *t* repairman 1 is on single vacation, one failed machine arrive during the interval *t* and *t*+*h.* This has probability $P_{0,0}(t)[1-N\lambda_0 h]$

The third possibility is that at epoch *t* repairman 1 is active; one failed machine is serviced during the interval *t* and *t*+*h*. This has probability $P_{1,1}(t)[\mu_1 h]$.

Hence $P_{0,0}(t+h) = P_{0,0}(t)[[1-N\lambda_0 h][1-\theta_1 h][1-\theta_2 h]] + P_{1,1}(t)[\mu_1 h]$.

From which we obtain

$$P_{0,0}^{'}(t) = -(\theta_1 + \theta_2 + N\lambda_0)P_{0,0}(t) + \mu_1 P_{1,1}(t) \tag{1}$$

The probability that there are n failed machines when the servers are on vacations in the interval $[t, t+h]$ is obtained as follows: Consider the state of the system between t and $t+h$, the first possibility is that at epoch t the servers are on single vacation with n failed machine and no service completion during the interval t and $t+h$. This has probability $\left[P_{0,n}(t)[1-(N-n)\lambda_n h\right][1-\theta_1 h]][1-\theta_2 h]$.

The second possibility is that at epoch t the servers are on single vacation, one failed machine arrive and no service completion during the interval t and $t+h$. This has probability $P_{0,n-1}(t)[(N-n+1)\lambda_{n-1}h]$.

Hence

$$P_{o,n}(t+h) = P_{o,n}(t)[1-\theta_1 h-(N-n)\lambda_n h+\theta_1(N-n)h^2][1-\theta_2 h]+P_{0,n-1}(t)[[(N-n+1)\lambda_{n-1}h]]$$

From which we obtain

$$P'_{0,n}(t) = -(\theta_1+\theta_2+(N-n)\lambda_n)P_{0,n}(t)+(N-n+1)\lambda_{n-1}P_{0,n-1}(t) \qquad (2)$$
$$1 \le n \le N-2$$

The probability that there are N-1 failed machines when the server is on vacations in the interval $[t, t+h]$ is obtained as follows: consider the state of the system between t and $t+h$, the first possibility is that at epoch t the servers are on single vacation with N-1 failed machines and no service completion during the interval t and $t+h$. This has probability $P_{0,N-1}(t)[1-\theta_1 h][1-\theta_2 h]$.

The second possibility is that at epoch t the servers are on single vacation, one failed machine arrive and no service completion during the interval t and $t+h$. This has probability $P_{0,N-2}(t)\lambda_{N-2}h$.

Hence $P_{0,N-1}(t+h) = P_{0,N-1}(t)[1-\theta_1 h][1-\theta_2 h]+P_{0,N-2}(t)\lambda_{N-2}h$

From which we obtain

$$P'_{0,N-1}(t) = -(\theta_1+\theta_2)P_{0,N-1}(t)+\lambda_{N-2}P_{0,N-2}(t) \qquad (3)$$

The probability that there is no failed machine when the repairman 1 is active in the interval $[t, t+h]$ is obtained as follows: Consider the state of the system between t and $t+h$, the first possibility is that at epoch t the repairman 1 is active, no failed machine arrive and no service completion during the interval t and $t+h$. This has probability $P_{1,0}(t)[1-N\lambda h][1-\theta_2 h]$.

The second possibility is that at epoch t repairman 1 leaves single vacation to active, no failed machine arrive and no service completion during the interval t and $t+h$. This has probability $P_{0,0}(t)\theta_1 h$.

Hence $P_{1,0}(t+h) = P_{1,0}(t)[1-N\lambda h][1-\theta_2 h]+P_{0,0}(t)\theta_1 h$

From which we obtain

$$P'_{1,0}(t) = -(\theta_2+N\lambda)P_{1,0}(t)+\theta_1 P_{0,0}(t) \qquad (4)$$

The probability that there are n failed machines when repairman 1 is active in the interval $[t, t+h]$ is obtained as follows: Consider the state of the system at time t and $t+h$, the first possibility is that at epoch t repairman 1 is active with n failed machine, and repairman 2 is on single vacation and no service completion during the interval t and $t+h$. This has probability

$$P_{1,n}(t)[1-[(N-n)\lambda_n]h](1-\mu_1 h)(1-\theta_2 h).$$

The second possibility is that at epoch t repairman 1 is active, one failed machine arrives, and repairman 2 is on single vacation and no service completion during the interval t and $t+h$. This has probability $P_{1,n-1}(t)[N-n+1]\lambda_{n-1}h(1-\mu_1 h)$.

The third possibility is that at epoch t repairman 1 is active, one failed machine is serviced by repairman 1 and repairman 2 is on single vacation during the interval t and $t+h$. This has probability $P_{1,n+1}(t)\ \mu_1 h$.

The fourth possibility is that at epoch t repairman 1 leaves single vacation to active, no failed machine arrive, repairman 2 is on single vacation and no service completion during the interval t and $t+h$. This has probability $P_{0,n}(t)[(1-(N-n)\lambda_n h](1-\mu_1 h)\theta_1 h$.

Hence

$$P_{1,n}(t+h) = P_{1,n}(t)[1-[(N-n)\lambda_n+\mu_1+\theta_2]h] + P_{1,n-1}(t)[N-n+1]\lambda_{n-1}h + P_{1,n+1}(t)\mu_1 h + P_{0,n}(t)\theta_1 h$$

From which we obtain

$$P'_{1,n}(t) = P_{1,n}(t)\big[-[(N-n)\lambda_n+\mu_1+\theta_2]\big] + P_{1,n-1}(t)[N-n+1]\lambda_{n-1} + P_{1,n+1}(t)\mu_1 + P_{0,n}(t)\theta_1 \quad 1 \le n \le N-2 \tag{5}$$

The probability that there are N-1 failed machines when repairman 1 is active in the interval $[t, t+h]$ is obtained as follows: Consider the state of the system at time t and $t+h$, the first possibility is that at epoch t repairman 1 is active with N-1 failed machine, repairman 2 is on single vacation and no service completion during the interval t and $t+h$. This has probability

$$P_{1,N-1}(t)[1-(\mu_1)h][1-\theta_2 h].$$

The second possibility is that at epoch t repairman 1 is active, one failed machine arrives, and repairman 2 is on single vacation and no service completion during the interval t and $t+h$. This has probability $P_{1,N-2}(t)\lambda_{N-2}h$.

The third possibility is that at epoch t repairman 1 leaves single vacation to active with N-1 failed machine, repairman 2 is on single vacation and no service completion during the interval t and $t+h$. This has probability $P_{0,N-1}(t)\theta_1 h$.

Hence

$$P_{1,N-1}(t+h) = P_{1,N-1}(t)[1-(\mu_1)h][1-\theta_2 h] + P_{1,N-2}(t)\lambda_{N-2}h + P_{0,N-1}(t)\theta_1 h$$

From which we obtain

$$P'_{1,N-1}(t) = P_{1,N-1}(t)[-(\mu_1+\theta_2)] + P_{1,N-2}(t)\lambda_{N-2} + P_{0,N-1}(t)\theta_1 \tag{6}$$

The number of broken down machines for the system between N and M-1 is derived below. The probability that there are n failed machines when repairman 1 is active in the interval $[t, t+h]$ is obtained as follows: consider the state of the system at time t and $t+h$, the first possibility is that at epoch t repairman 1 is active with n failed machine, no arrival, repairman 2 is on single vacation and no service completion during the interval t and $t+h$. This has probability $P_{1,n}(t)(1-\lambda_n h)(1-\mu_1 h)$.

The second possibility is that at epoch t repairman 1 is active, one failed machine arrives, repairman 2 is on single vacation and no service completion during the interval t and $t+h$. This has probability $P_{1,n-1}(t)\lambda_{n-1}h$.

The third possibility is that at epoch t repairman 1 is active, one failed machine is serviced by repairman1 and repairman 2 is on single vacation during the interval t and t+h. This has probability $P_{1,n+1}(t)\mu_1 h$.

Hence $P_{1,n}(t+h) = P_{1,n}(t)(1-\lambda_n h)(1-\mu_1 h) + P_{1,n-1}(t)\lambda_{n-1}h + P_{1,n+1}(t)\mu_1 h$.

From which we obtain

$$P'_{1,n}(t) = P_{1,n}(t)\left[-(\lambda_n+\mu_1)\right] + P_{1,n-1}(t)\lambda_{n-1} + P_{1,n+1}(t)\mu_1 \qquad N \le n \le M-1 \tag{7}$$

The probability that there are M failed machines when repairman 1 is active in the interval [t, t+h] is obtained as follows: consider the state of the system between t and $t+h$, the first possibility is that at epoch t repairman 1 is active with M failed machines, repairman 2 is on single vacation, no arrival and no service completion during the interval t and $t+h$. This has probability $P_{1,M}(t)[1-\mu_1 h]$.

The second possibility is that at epoch t repairman 1 is active, one failed machine arrives, and repairman 2 is on single vacation and no service completion during the interval t and $t+h$. This has probability $P_{1,M-1}(t)\lambda_{M-1}h$.

Hence $P_{1,M}(t+h) = P_{1,M}(t)\left[1-(\mu_1)h\right] + P_{1,M-1}(t)\lambda_{M-1}h$.

From which we obtain

$$P'_{1,M}(t) = P_{1,M}(t)\left[-(\mu_1)\right] + P_{1,M-1}(t)\lambda_{M-1} \tag{8}$$

The probability that there are N failed machines when both repairmen are active in the interval [t, $t+h$] is obtained as follows: Consider the state of the system between t and $t+h$, the first possibility is that at epoch t repairman 2 is active with N failed machines and no service completion during the interval t and $t+h$. This has probability $P_{2,N}(t)[1-(M-N)\lambda_N h]\left[1-\mu_2 h\right]$.

The second possibility is that at epoch t repairman 2 leaves multiple vacations to active with N failed machines and no service completion during the interval t and $t+h$. This has probability $P_{0,N}(t)\theta_2 h$.

Hence $P_{2,N}(t+h) = P_{2,N}(t)\left[1-\left(\mu_2+(M-N)\lambda_N\right)h\right] + P_{0,N}(t)\theta_2 h$.

From which we obtain

$$P'_{2,N}(t) = P_{2,N}(t)\left[-\left(\mu_2+(M-N)\lambda_N\right)\right] + \theta_2 P_{0,N}(t) \tag{9}$$

The probability that there are N failed machines when repairman 2 is active in the interval [t, $t+h$] is obtained as follows: consider the state of the system between t and $t+h$, the first possibility is that at epoch t repairman 2 is active with n failed machine, no arrival and no service completion during the interval t and $t+h$. This has probability $P_{2,,n}(t)[1-\lambda_n h][1-(\mu_2)h]$,

The second possibility is that at epoch t repairman 2 is active, one failed machine arrives and no service completion during the interval t and $t+h$. This has probability $P_{2,n-1}(t)[\lambda_{n-1}h]$.

The third possibility is that at epoch t repairman 2 is active, one failed machine service during the interval t and $t+h$. This has probability $P_{2,,n+1}(t)\mu_2 h$.

The fourth possibility is that at epoch *t* repairman 2 leaves single vacation to active with *n* failed machines and no service completion during the interval *t* and *t+h*. This has probability $P_{0,n}(t)\theta_2 h$.

Hence

$$P_{2,n}(t+h) = P_{2,n}(t)[1-\lambda_n h][1-(\mu_2)h]+P_{2,n+1}(t)\mu_2 h + P_{2,n-1}(t)[\lambda_{n-1}h] + P_{0,n}(t)\theta_2 h$$

From which we obtain

$$P'_{2,n}(t) = P_{2,n}(t)[-(\lambda_n+\mu_2)] + P_{2,n-1}(t)[\lambda_{n-1} + P_{2,n+1}(t)\mu_2 + P_{0,n}(t)\theta_2 \quad N+1 \le n \le M-1 \qquad (10)$$

The probability that there are *M* failed machines when repairman 2 is active in the interval [*t*, *t+h*] is obtained as follows: Consider the state of the system between *t* and *t+h*, the first possibility is that at epoch *t* repairman 2 is active with *M* failed machines, no arrival and no service completion during the interval *t* and *t+h*. This has probability $P_{2,M}(t)(1-\mu_2 h)$.

The second possibility is that at epoch *t* repairman 2 is active, one failed machine arrives and no service completion during the interval *t* and *t+h*. This has probability $P_{2,M-1}(t)\lambda_{M-1}h$.

The third possibility is that at epoch *t* repairman 2 leaves single vacation to active with *M* failed machines and no service completion during the interval *t* and *t+h*. This has probability $P_{0,M}(t)\theta_2 h$.

Hence $P_{2,M}(t+h) = P_{2,M}(t)(1-\mu_2 h) + P_{2,M-1}(t)\lambda_{M-1}h + P_{0,M}(t)\theta_2 h$

From which we obtain

$$P'_{2,M}(t) = P_{2,M}(t)[-(\mu_2)] + P_{2,M-1}(t)\lambda_{M-1} + P_{0,M}(t)\theta_2 \qquad (11)$$

For the single server machine interference problem with additional server for long queue the number of equations to be solved is 2-*N*+*2M*.

where

$$\lambda_n = \begin{cases}(M-n)\lambda, & 0 \le n \le M-1 \\ 0, & n = M\end{cases}$$

$$\mu = \begin{cases}\mu_{1,} & 0 \le n \le M \\ \mu_{2,} & N \le n \le M\end{cases}$$

The state transition diagram for the single server machine interference problem with additional server for long queue is given in Fig. 1.

3 Numerical Solutions

To determine the transient state results from the equations we use MATLAB programming language (version 7.5.0) to generate time dependent probabilities for the system under study.

Equations (1)-(11) representing the single server machine interference problem with additional server for long queues under *N* policy vacations are readily solved using the ODE45 (Runge-Kutta algorithm for solving ordinary differential equations) in MATLAB programming language.

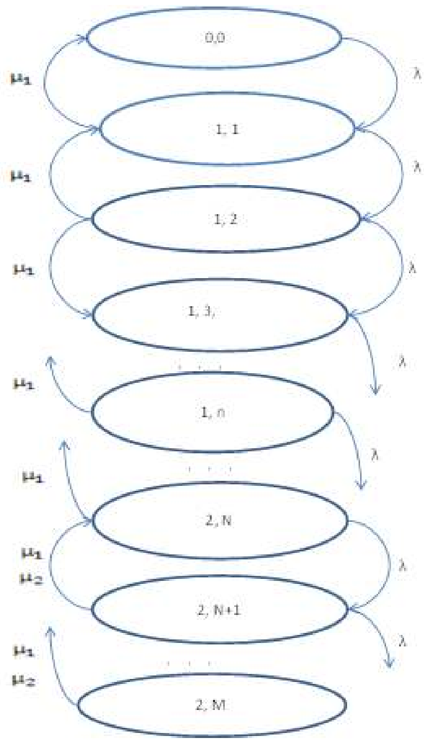

Fig. 1. The state transition diagram for the single server machine interference problem with additional server for long queue

The transient probabilities

$P_{(i,n)}$; where $i = 0, 1, 2$ and $0 \leq n \leq M$ for the system are computed for each time *t*. The system starts empty with $P_{(0,0)} = 1$ and $P_{(i,n)} = 0$ for all $i = 0, 1, 2$ and $n = 0, 1, 2, \ldots, M$ as initial conditions. We take various values of M and consider the effect of different parameters λ, μ, θ_1 and θ_2 on the machine availability in the system.

The expected number of failed machines in the system at time *t* is

$$E(F(t)) = \sum_{n=0}^{N-1} nP_{0,n}(t) + \sum_{n=1}^{N-1} nP_{1,n}(t) + \sum_{n=N}^{M} nP_{2,n}(t)$$

The expected number of operating machines at time t is

$$E(O(t)) = M - E(F(t))$$

Expected vacations the servers has is $E[V](t) = \sum_{n=0}^{M} nP_{0,n}(t)$.

The machine availability at time *t* ($M.A.(t)$)is given by the expression $M.A.(t) = 1 - \frac{E[F(t)]}{M}$

Variance: The variance of the number of broken down machines and the number of operating machines is calculated by using the expression.

$$\sigma^2(t) = \sum_{n=0}^{M} n^2 P_{0,n}(t) + \sum_{n=0}^{M} n^2 P_{1,n}(t) + \sum_{n=1}^{M} n^2 P_{2,n}(t) - [E(F(t))]^2$$

where

$$E(F(t)) = \sum_{n=0}^{M} n P_{0,n}(t) + \sum_{n=0}^{M} n P_{1,n}(t) + \sum_{n=1}^{M} n P_{2,n}(t)$$

Tables 1-2 shows the transient results for the MATLAB program for different values of *M*. We run the model for sufficient time *t*, after some time *t* the successive values of the expected number of failed and working machines no longer varies, this means that the transient results are close to the steady state results. This is shown in Table 1.

Table 1. Some performance measures for different values of *t* and *N* when λ=0.15, $\mu_1 = 1.1$, $\mu_2 = 1.2$, $\theta_1 = 1$, $\theta_2 = 2$, *M*=10

t	***N=3***			***N=4***		
	E(0)	**E(F)**	**M.A.**	**E(0)**	**E(F)**	**M.A.**
0	10.000	0.0000	1.0000	10.0000	0.0000	1.0000
1	6.8618	3.1382	0.6862	6.9481	3.0519	0.6948
2	6.1483	3.8517	0.6148	5.9387	4.0613	0.5939
3	6.2542	3.7458	0.6254	5.6993	4.3007	0.5699
4	6.6737	3.3263	0.6674	5.8382	4.1618	0.5838
5	7.1859	2.8141	0.7186	6.1376	3.8624	0.6138
6	7.6631	2.3369	0.7663	6.4862	3.5138	0.6486
7	8.0434	1.9566	0.8043	6.8265	3.1735	0.6827
8	8.3151	1.6849	0.8315	7.1324	2.8676	0.7132
9	8.4958	1.5042	0.8496	7.3945	2.6055	0.7394
10	8.6104	1.3896	0.8610	7.6129	2.3871	0.7613
11	8.6810	1.3190	0.8681	7.7913	2.2087	0.7791
12	8.7237	1.2763	0.8724	7.9354	2.0646	0.7935
13	8.7493	1.2507	0.8749	8.0504	1.9496	0.8050
14	8.7645	1.2355	0.8764	8.1417	1.8583	0.8142
15	8.7735	1.2265	0.8774	8.2138	1.7862	0.8214
16	8.7788	1.2212	0.8779	8.2706	1.7294	0.8271
17	8.7820	1.2180	0.8782	8.3151	1.6849	0.8315
18	8.7839	1.2161	0.8784	8.3500	1.6500	0.8350
19	8.7849	1.2151	0.8785	8.3772	1.6228	0.8377
20	8.7856	1.2144	0.8786	8.3985	1.6015	0.8398
Var	0.8718			0.9765		
CPU time	3.2173 secs			3.1628 secs		

4 Discussion

Tables 1-2 show some performance measures for different values of *t* and *N*. In Tables 1 and 2, we vary the values of *N* from 3 to 6 for fix values of λ=0.15, $\mu_1 = 1.1$, $\mu_2 = 1.2$, $\theta_1 = 1$, $\theta_2 = 2$, *M*=10. We found that the number of failed machines *N* that trigger repairman 2 affects the number of failed machines, the number of operating machine and the machine availability. We found that as the number of failed machines that trigger repairman 2 decreases the expected number of operating machines increases. While the expected number of failed machines decreases. Also with decrease in the number of failed machine that trigger repairman 2, the machine availability increases. We also found that as the number of failed machines that trigger repairman 2 decreases the CPU time to run the algorithm increases. This is true because as the number of failed machines that trigger repairman 2 decreases the number of equation to be solved also increases.

For the single server machine interference problem with additional server for long queues under n policy vacations there are 2-*N*+*2M* equations in the system, we also observe that for small *M*, say *M*=50, the CPU time is less than 20 seconds (Table 3). The actual CPU times observed for different number of machine in the system for the single server machine interference problem with additional server for long queues under n policy vacations is inputted into linear regression in EXCEL package to compute the predicted CPU time for the system. We found that the predicted:

Table 2. Some performance measures for different values of *t* and *N* when λ=0.15, $\mu_1 = 1.1$, $\mu_2 = 1.2$, $\theta_1 = 1$, $\theta_2 = 2$, *M*=10

t	*N=5*			*N=6*		
	E(0)	**E(F)**	**M.A.**	**E(0)**	**E(F)**	**M.A.**
0	10.000	0.0000	1.0000	10.0000	0 .0000	1.0000
1	8.4395	1.5605	0.8440	8.4395	1.5605	0.8440
2	8.4300	1.5700	0.8430	8.4300	1.5700	0.8430
3	8.4297	1.5703	0.8430	8.4297	1.5703	0.8430
4	8.4297	1.5703	0.8430	8.4297	1.5703	0.8430
5	8.4297	1.5703	0.8430	8.4297	1.5703	0.8430
6	8.4297	1.5703	0.8430	8.4297	1.5703	0.8430
7	8.4286	1.5714	0.8429	8.4286	1.5714	0.8429
8	8.3920	1.6080	0.8392	8.3920	1.6080	0.8392
9	8.5701	1.4299	0.8570	8.5701	1.4299	0.8570
10	7.9329	2.0671	0.7933	7.9329	2.0671	0.7933
11	6.6607	3.3393	0.6661	6.6607	3.3393	0.6661
12	7.2354	2.7646	0.7235	7.2354	2.7646	0.7235
13	7.1103	2.8897	0.7110	7.1103	2.8897	0.7110
14	7.0411	2.9589	0.7041	7.0411	2.9589	0.7041
15	7.0537	2.9463	0.7054	7.0537	2.9463	0.7054
16	7.0515	2.9485	0.7051	7.0515	2.9485	0.7051
17	7.0519	2.9481	0.7052	7.0519	2.9481	0.7052
18	7.0518	2.9482	0.7052	7.0518	2.9482	0.7052
19	7.0518	2.9482	0.7052	7.0518	2.9482	0.7052
20	7.0518	2.9482	0.7052	7.0518	2.9482	0.7052
Var	0.5600			0.5600		
CPU time	2.7124 secs			2.5087 secs		

$CPU\ time = a + bM$ where a and b are constants and M is the number of machines. We observe that the predicted CPU time is an indication of the actual CPU time. We also observe that the CPU time to solve this model is higher than that of Ojobor [7]. This is caused by the number of failed machines that trigger repairman 2 in the system. The number of equations to be solved here is also higher than that of Ojobor [7].

Table 3. Effect of M and *N* on the machine availability and CPU time for sufficient value of *t* for the single vacation policy. λ=0.15, $\mu = 1.1$, $\theta_1 = 1$, $\theta_2 = 2$.

	***N*=4, *M*=10**	***N*=8, *M*=20**	***N*=12, *M*=30**	***N*=16, *M*=40**	***N*=20, *M*=50**
E(F)	1.6015	2.5789	2.4450	2.3722	2.3253
E(0)	8.3985	17.4211	27.5550	37.6278	47.6747
E(V)	0.9340	0.0000	0.0000	0.0000	0.0000
M.A.	0.8398	0.8711	0.9185	0.9407	0.9535
VAR(E(0))	0.9765	0.2344	0.1394	0.1160	0.1418
CPU time (secs)	2.8780	4.1113	8.2357	10.6943	16.5528
Predicted Time (secs)	1.7128	5.1038	8.4948	11.8858	15.2768

The following are the findings from our work:

(i) In Tables 1-2 we obtained various value for the expected number of failed and operating machines and the machine availability for different value of N with respect to t. We found that as the number of failed machines N that trigger repairman 2 decreases the expected number of operating machine increases. While expected number of failed machine decreases.

(ii) We also found that as the number of failed machines that trigger repairman 2 decreases the CPU time to run the algorithm increases. This is true because as the number of failed machines that trigger repairman 2 decreases the number of equations to be solved also increases.

(iii) We found out in Table 3 that with the same service rate μ, failure rate λ and vacations length θ, as the number of operating machine and the number of failed machines that trigger repairman 2 in the system increases the variance is less than one. This is caused by the additional repairman. The additional repairman reduces the waiting time of failed machines in the system.

(iv) We found also that the CPU time to solve this model is higher than that of Ojobor [7]. This is caused by the number of failed machines that trigger repairman 2 in the system and the number of equations to be solved.

(v) We found that most research work on machine interference problem till date focused mainly on the average number of operating and failed machines in the system. In this work, apart from finding the average number of failed and operating machines, we also find the variance of the number of failed machines in the system. Haque and Armstrong (2007) stated that 'a system manager might prefer a service policy that provide smaller average number of operational machines if it is able to provide those machines more consistently'. Knowing the variance will help system managers to apply a particular service policy in a given queueing system. The variance and standard deviation are shown in Tables 1-2 above for the single server machine interference problem with additional server for longer queue.

(vi) Fig. 2 shows the effect of failure rate of operating machine on the expected number of failed machines in the system. We found that as the failure rate of operating machine increases the expected number of failed machines increases.

(vii) We also found that the additional server reduces the expected number of failed machines thereby reducing the waiting time of failed machines. This can be compared to the earlier two models considered in this thesis.

(viii) Fig. 3 below shows the effect of failure rate of operating machines on the expected number of operating units in the system. We found that the rate at which machines fail and are serviced affect the expected number of failed and operating machines in the system.

(ix) Figs. 4 and 5 below show the effect of service rate on the expected number of failed and operating machines in the system. We found that as the service rate increases the expected number of operating machines increases. Also as the service rate decreases the expected number of failed machines increases.

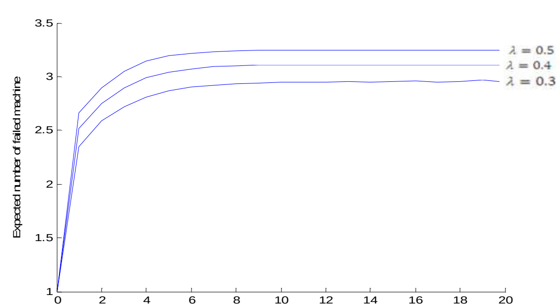

Fig. 2. Effect of failure rate of machines on the expected number of failed machines in the system at time t when $\theta_1 = 3, \theta_2 = 4, \mu_1 = 1.1, \mu_2 = 1.2$, N=6, M=10

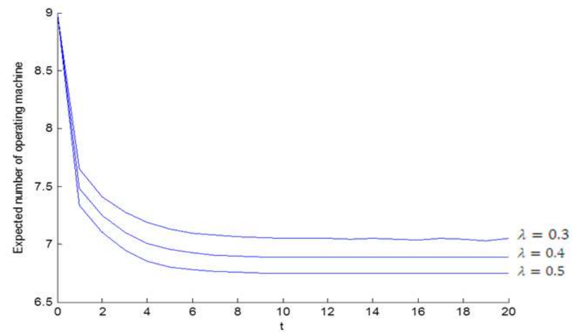

Fig. 3. Effect of failure rate of machines on the expected number of operating machines in the system at time *t* when $\theta_1 = 3$, $\theta_2 = 4$, $\mu_1 = 1.1$, $\mu_2 = 1.2$, *N*=6, *M*=10

In Figs. 2 and 3, we observe that when we run the model between 0 and 1 there is no variation in the expected number of failed and operating machines in the system with different failure rate, but as the model is run from 1 to 20 the expected number of operating machines increases with increase in the failure rate (Fig. 3). In a similar manner with decrease in the failure rate the expected number of failed machines decreases (Fig. 2).

In like manners Figs. 4 and 5 show the effect of service rate of broken down machines on the expected number of failed and operating machines in the system at time t, we observe that when we run the model between 0 and 1 there is no variation in the expected number of failed and operating machines in the system with different service rate, but as we run the model from 1 to 20 the expected number of operating machines increases with increase in service rate (Fig. 3). In a similar manner with decrease in service rate the expected number of failed machines increases (Fig. 2).

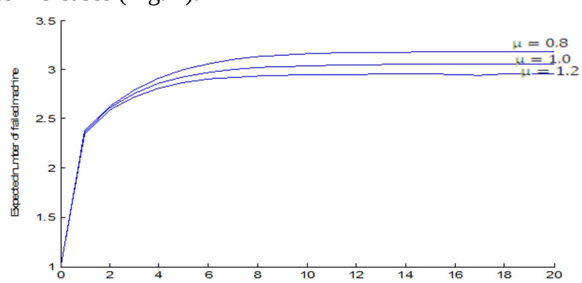

Fig. 4. Effect of service rate of failed machines on the expected number of failed machines in the system at time *t* when $\theta_1 = 3$, $\theta_2 = 4$, $\lambda = 0.3$, *N*=6, *M*=10

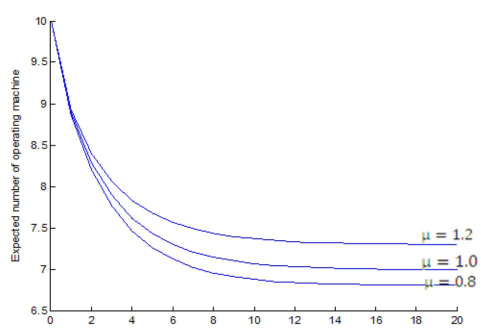

Fig. 5. Effect of service rate of failed machines on the expected number of operating machines in the system at time *t* when $\theta_1 = 3$, $\theta_2 = 4$, $\lambda = 0.3$, *N*=6, *M*=10

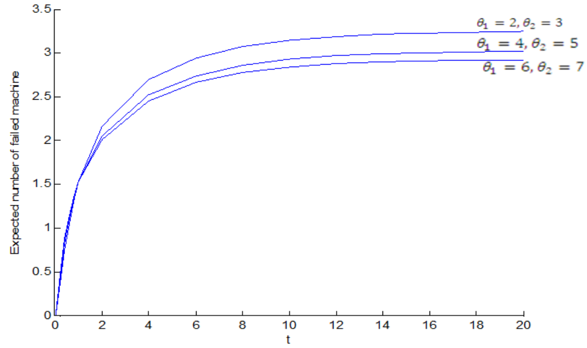

Fig. 6. The effect of vacation length of server on expected number of failed machines in the system at time *t* when $\mu_1 = 1.1, \mu_2 = 1.2$,, $\lambda = 0.3$, *N*=6, *M*=10

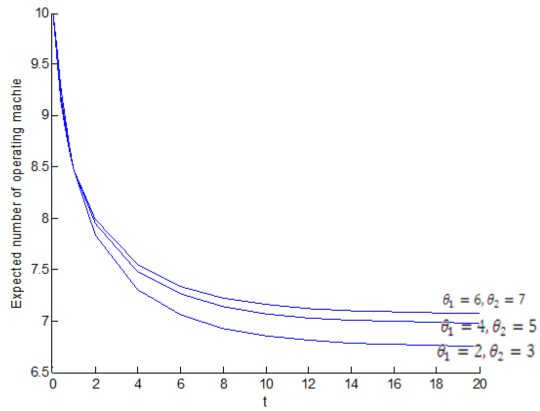

Fig. 7. The effect of vacation length of server on the expected number of operating machines in the system at time *t* when $\mu_1 = 1.1, \mu_2 = 1.2$,, $\lambda = 0.3$, *N*=6, *M*=10

Fig. 7 shows the effect of vacation length of server on the expected number of operating machines in the system.

Figs. 6 and 7 shows the effect of vacation length on the expected number of failed and operating machine in the system. We found out that as vacation length increase the expected number of operating machine increases. While as vacation length decreases the expected number of failed machines in the system also increases.

5 Conclusion

In conclusion, we developed a single server machine interference problem with additional server for long queues under *N* policy vacations. The single server was always available for attending to broken down machines, but go on vacation when there are no broken down machines. The additional server is always on vacation but only come back from vacation to attend to broken down machines if there were more than or equal to *N* broken down machines in queue in the system (*N*-policy vacation). Otherwise he goes for another vacation. We assumed that our repairmen 1 and 2 could go on vacation. We obtained various values for the expected number of failed, the expected number operating and the machine availability for differet value of *N* with respect to *t*. We found that as the number of failed machines *N* that triggers repairman 2 decreased the expected number of operating machine increased. While expected number of failed machines decreased.

We also found that as the number of failed machines that triggers repairman 2 decreased the CPU time to run the algorithm increased. This is true because as the number of failed machines that triggers repairman 2 decreased the number of equations to be solved also increased.

Disclaimer

This manuscript was presented in the conference.
Conference name: "27th European Conference on Operational Research".
Conference link is "https://www.euro-online.org/conf/euro27/treat_abstract?paperid=818"
Date: 12-15 July 2015.

Competing Interests

Authors have declared that no competing interests exist.

References

[1] Yue D, Yue W, Qi H. Analysis of a machine repair system with warm spares and N-policy vacations. The 7th International Symposium on Operations Research and its Applications. China. 2008;190-198.

[2] Yue D, Yue W, Qi H. Performance analysis and optimization of a machine repair problem with warm spares and two heterogeneous repairmen. Optim Eng. 2012;13:545–562.

[3] Sharma DC. Machine repair problem with spares and N-policy vacation. Research Journal of Recent Sciences. 2012;1(4):72-78.

[4] Maheshwari S, Ali S. Machine repair problem with mixed spares, balking and reneging. International Journal of Theoretical & Applied Sciences. 2013;5(1):75-83.

[5] Al-Seedy RO, Al-Ibraheem FM. An interaarival hyperexponential machine interference with balking, reneging, state dependent, spares and an additional server for longer queues. International Journal of Mathematics and Mathematical Sciences. 2001;27(12):737-747.

[6] Jain M, Mittal R, Kumari R. (m, M) machining system with two unreliable servers, mixed spares and common cause failure. Journal of Industrial Engineering International. 2015;11:171-178.

[7] Jain M, Kumar S. Bi-level control policy for redundant repairable multi-component system with reneging. Set up and vacation. IJE Transactions A: Basics. 2009;22(2):159-168.

[8] Jain M, Sharma GC, Singh M. Queueing modelling of machining system with balking, reneging, additional repairman and two modes of failure. Journal of King Abdul-Aziz University, Engineering Science. 2002;14(2):53-71.

[9] Ojobor SA. Transient solution of machine interference problem with an unreliable server under multiple vacations policy. British Journal of Mathematics and Computer Science. 2015;6(4):351-369.

Peer-review history:
The peer review history for this paper can be accessed here (Please copy paste the total link in your browser address bar)
http://sciencedomain.org/review-history/14759