Available online at http://scik.org
Eng. Math. Lett. 2018, 2018:1
https://doi.org/10.28919/eml/3418
ISSN: 2049-9337

# AVAILABILITY MODELLING AND ENHANCEMENT OF A COMPUTER SYSTEM WITH REDUNDANT HETEREGENEOUD SOFTWARE

IBRAHIM YUSUF[1,*], BAFFA SANI[2], AND MANSUR BABAGANA[2]

[1]Department of Mathematical Sciences, Bayero University, Kano, Nigeria

[2]Department of Computer Science, Bayero University, Kano, Nigeria



**Abstract:** This paper studies the availability of one host system incorporating heterogeneous software in the host. The system has two types of heterogeneous software. Each type of software has an identical copy (homogeneous) on standby. Markov model of the system is derived through the system state transition probabilities and the corresponding differential difference equations are used to evaluate the system availability. Comparison of the availability for different values of hardware failure and repair rate are performed and found that availability is higher with low hardware failure rate and higher repair rates.

**Keywords:** Availability, hardware, software, heterogeneous, computer.

**2010 AMS Subject Classification:** 68N30.

## 1. Introduction

Reliability and availability assessment of a system provides insight into the probability that the system will be available to be committed to a specified requirement. Computers systems are

[*]Corresponding author
E-mail address: iyusuf.mth@buk.edu.ng
Received July 11, 2017

exposed various degree of hardware or software failure. On improving the reliability of the computer system through hardware and software redundancy, the system availability as well as the production output will increase. This can be achieved by maintaining reliability and availability at the highest order. Computer system reliability and availability are improved through a standby unit support which is capable of performing similar function with the operational unit but with different degree and desirability.

The unit wise redundancy technique has been considered as one of these in the development of stochastic models for computer systems. [10] Discussed the reliability modelling of a hardware/software system. The technique of unit wise redundancy in cold standby mode has also been used in computer systems. [1, 6, 7] analyzed different computer system models with unit wise cold standby redundancy and different repair policies. But, it is also proved that component wise redundancy is better than unit wise redundancy so far as reliability is concerned. [9] developed a stochastic model for a computer system with hardware component in cold standby redundancy. [2] studied a cold standby computer system by giving priority to hardware repair activities over software replacement. [8] analyzed computers systems with cold standby redundancy under different failures and repair policies. [3] have discussed modelling of a computer system with priority to preventive maintenance over software replacement and priority to hardware repair over replacement respectively. [4,5] have analyzed the performance of a computer system with fault detection of hardware.

Existing literatures ignores the reliability, availability and profit modelling of computer systems incorporating different software with similar task on a single host and the impact such heterogeneous software on computer system performance. Example of such heterogeneous software on computer system can be seen in operating systems (windows 7,8, 10, windows XP, Vista, ubuntu), application packages (latex and MS word), Mathematical software (Mathematica, Matlab, Maple), etc. Some of this software can be install on a single host. This heterogeneous software will assist in reducing operating costs and the risk of a catastrophic breakdown, extending the availability and working time, increasing the revenue generated for a system.

The problem considered in this paper is different from the work of discussed authors above. In this paper, a single host with two types of dissimilar cold standby heterogeneous software is considered and derived its corresponding mathematical models. The focus of our analysis is primarily to capture the effect of both failure and repair rates on the measures of system effectiveness like availability.

The organization of the paper is as follows. Section 2 contains a description of the system under study. Section 3 presents formulations of the models. The results of our numerical simulations are presented in section 4. Finally, we make some concluding remarks in Section 5.

## 2. Description of the System

In practical the system will consist of a primary storage formatted with NTFS in one partition and EXT in another partition. The failure of one partition is not necessarily the loss of data stored on all partitions. Data recovery can be invoked from the working partition. More so the system can still be used even with the failed partition. This is made possible by the heterogeneous characteristics exhibited by both file systems. In other words, each of the file systems fails independently of the state of the others and has an exponential failure distribution. In the same vein, a system that has Microsoft Office and Open Office installed will exhibit the same exponential failure distribution. The proposed system in this paper consists of two heterogeneous software running on one host as in Figure 1 which depicts the state transitions of the system according to Markov chain. Each of the type I and type II software fails independently of the state of the others and has an exponential failure distribution with parameter $\lambda_i, i = 1, 2$, exponential repair distribution with parameter $\mu_i$. When any of type $k, k = I, II$ copy I software fails, which occurs with failure rate $\lambda_i, i = 1, 2$, it is sent for repair with the service rate equal to $\mu_i$ and the corresponding type $k$ copy II then carries out the function of the failed software. When both types of software fail, the host is suspended. It is assumed that switching from standby to operation is perfect. System failure results from the failure of host or both types of software.

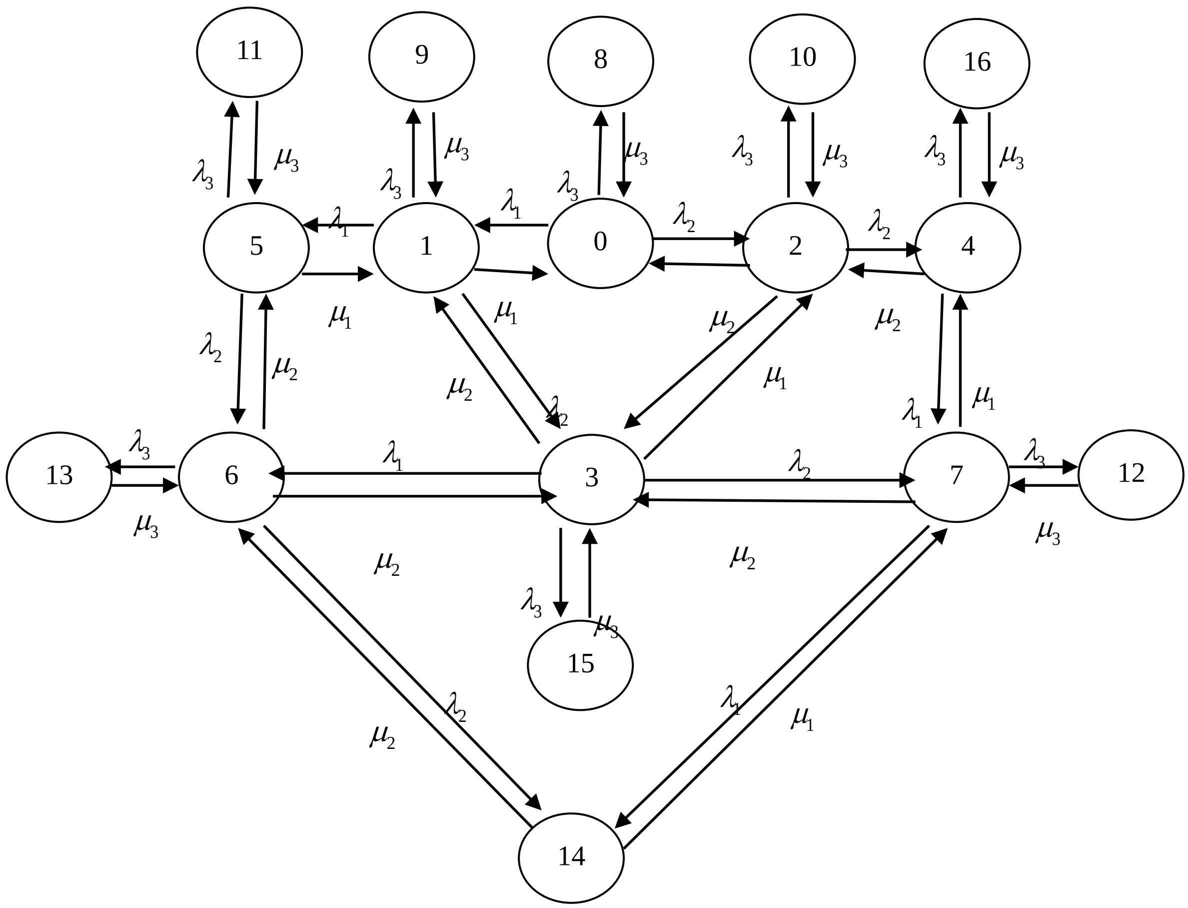


Figure 1: Transition diagram

$S_0$: Initial state, the host, type I copy I, type II copy I software are working, type I copy II and type II copy II software are on standby. The system is operative.

**$S_1$:** Type I copy I software has failed and is under repair, the host, type I copy II and type II copy I software are working; type II copy II software is on standby. The system is operative.

$S_2$: Type II copy I software has failed and is under repair, the host, type I copy I and type II copy II software are working; type I copy II software is on standby. The system is operative.

$S_3$: Type I copy I and type II copy I software have failed and are under repair, the host, type I copy II and type II copy II software are working. The system is operative.

$S_4$: Type II copy I software has failed and is waiting for repair, type II copy II software has failed and is under repair, the host and type I copy I software are working, type I copy II software is on standby. The system is operative.

$S_5$: Type I copy I software has failed and is waiting for repair, type I copy II software has failed and is under repair, the host and type II copy I software are working, type II copy II software is on standby. The system is operative.

$S_6$: Type I copy I software has failed and is waiting for repair, type I copy II and type II copy I software have failed and are under repair, the host and type II copy II software are working. The system is operative.

$S_7$: Type II copy I software has failed and is waiting for repair, type II copy II and type I copy I software have failed and are under repair, the host and type I copy II software are working. The system is operative.

$S_8$: The host has failed and is under repair, type I copy I and type II copy I software are suspended, copies II of both type I and type II are on standby. The system is inoperative.

$S_9$: The host and type I copy I software have failed and are under repair, type I copy II and type II copy I are suspended, type II copy II is on standby. The system is inoperative.

$S_{10}$: The host and type II copy I software have failed and are under repair, type I copy I is suspended, type I copy II and type II copy II are on standby. The system is inoperative.

$S_{11}$: The host and type I copy II software have failed and are under repair, type I copy I software has failed and is waiting for repair, type II copy I is suspended, type II copy II is on standby. The system is inoperative.

$S_{12}$: The host, type I copy I and type II copy II software have failed and are under repair, type II copy I software has failed and is waiting for repair, type I copy II is suspended. The system is inoperative.

$S_{13}$: The host, type I copy II and type II copy I software have failed and are under repair, type I copy I software has failed and is waiting for repair, type II copy II is suspended. The system is inoperative.

$S_{14}$: The host is suspended, type I copy II and type II copy II software have failed and are under repair, type I copy I and type II copy I software have failed and waiting for repair. . The system is inoperative.

$S_{15}$: The host, type I copy I and type II copy I software have failed and are under repair, type I copy II and type II copy II software are suspended. . The system is inoperative.

$S_{16}$: The host and type II copy II software have failed and are under repair, type II copy I have failed and is waiting for repair, type I copy I software is suspended, type I copy II is on standby. The system is inoperative.

## 3. Formulation of the Model

Define $P_i(t)$ to be the probability that the system at time $t$ is in state $i$, $i = 0,1,2,3,...,16$ . The corresponding differential difference equations associated with the transition diagram in Figure 1 are:

$$p_0'(t) = -y_0 p_0(t) + \mu_1 p_1(t) + \mu_2 p_2(t) + \mu_3 p_8(t)$$

$$p_1'(t) = -y_1 p_1(t) + \lambda_1 p_0(t) + \mu_2 p_3(t) + \mu_1 p_5(t) + \mu_3 p_9(t)$$

$$p_2'(t) = -y_2 p_2(t) + \lambda_2 p_0(t) + \mu_1 p_3(t) + \mu_2 p_4(t) + \mu_3 p_{10}(t)$$

$$p_3'(t) = -y_3 p_3(t) + \lambda_2 p_1(t) + \lambda_1 p_2(t) + \mu_1 p_6(t) + \mu_2 p_7(t) + \mu_3 p_{15}(t)$$

$$p_4'(t) = -y_4 p_4(t) + \lambda_2 p_2(t) + \mu_1 p_7(t) + \mu_3 p_{16}(t)$$

$$p_5'(t) = -y_5 p_5(t) + \lambda_1 p_1(t) + \mu_2 p_6(t) + \mu_3 p_{11}(t)$$

$$p_6'(t) = -y_6 p_6(t) + \lambda_1 p_3(t) + \lambda_2 p_5(t) + \mu_3 p_{13}(t) + \mu_2 p_{14}(t)$$

$$p_7'(t) = -y_7 p_8(t) + \lambda_2 p_3(t) + \lambda_1 p_4(t) + \mu_3 p_{12}(t) + \mu_1 p_{14}(t)$$

$$p_8'(t) = -\mu_3 p_8(t) + \lambda_3 p_0(t)$$

$$p_9'(t) = -\mu_3 p_9(t) + \lambda_3 p_1(t)$$

$$p_{10}'(t) = -\mu_3 p_{10}(t) + \lambda_3 p_2(t)$$

$$p_{11}'(t) = -\mu_3 p_{11}(t) + \lambda_3 p_5(t)$$

$$p_{12}'(t) = -\mu_3 p_{12}(t) + \lambda_3 p_7(t)$$

$$p_{13}'(t) = -\mu_3 p_{13}(t) + \lambda_3 p_6(t)$$

$$p_{14}'(t) = -y_8 p_{14}(t) + \lambda_2 p_6(t) + \lambda_1 p_7(t)$$

$$p_{15}'(t) = -\mu_3 p_{15}(t) + \lambda_3 p_3(t)$$

$$p_{16}'(t) = -\mu_3 p_{16}(t) + \lambda_3 p_4(t) \tag{1}$$

This can be written in the matrix form as

$$\dot{P} = MP \tag{2}$$

The initial condition for this problem is:

$$p_i(0) = \begin{cases} 1, & i = 0 \\ 0, & i = 1, 2, 3, \ldots, 16 \end{cases} \tag{3}$$

$$M = \begin{bmatrix}
-y_0 & \mu_1 & \mu_2 & 0 & 0 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\lambda_1 & -y_1 & 0 & \mu_2 & 0 & \mu_1 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\lambda_2 & 0 & -y_2 & \mu_1 & \mu_2 & 0 & 0 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \lambda_2 & \lambda_1 & -y_3 & 0 & 0 & \mu_1 & \mu_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_3 & 0 \\
0 & 0 & \lambda_2 & 0 & -y_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_3 \\
0 & \lambda_1 & 0 & 0 & 0 & -y_5 & \mu_2 & 0 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \lambda_1 & 0 & \lambda_2 & -y_6 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_3 & \mu_2 & 0 & 0 \\
0 & 0 & 0 & \lambda_2 & \lambda_1 & 0 & 0 & -y_7 & 0 & 0 & 0 & 0 & \mu_3 & 0 & \mu_1 & 0 & 0 \\
\lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -y_8 & 0 & 0 \\
0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 \\
0 & 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3
\end{bmatrix}$$

where $y_0 = (\lambda_3 + \lambda_2 + \lambda_1)$, $y_1 = (\lambda_3 + \lambda_2 + \lambda_1 + \mu_1)$, $y_2 = (\lambda_3 + \lambda_2 + \lambda_1 + \mu_2)$,

$y_3 = (\lambda_3 + \lambda_2 + \lambda_2 + \mu_1 + \mu_2)$ , $y_4 = (\lambda_3 + \lambda_1 + \mu_2)$ , $y_5 = (\lambda_2 + \lambda_3 + \mu_1)$ , $y_6 = (\lambda_3 + \lambda_2 + \mu_1 + \mu_2)$ ,

$y_7 = (\lambda_3 + \lambda_1 + \mu_1 + \mu_2)$, $y_8 = (\mu_1 + \mu_2)$

Equation (2) is expressed explicitly in the form

$$\begin{bmatrix} p_0' \\ p_1' \\ p_2' \\ p_3' \\ p_4' \\ p_5' \\ p_6' \\ p_7' \\ p_8' \\ p_9' \\ p_{10}' \\ p_{11}' \\ p_{12}' \\ p_{13}' \\ p_{14}' \\ p_{15}' \\ p_{16}' \end{bmatrix} = \begin{bmatrix} -y_0 & \mu_1 & \mu_2 & 0 & 0 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \lambda_1 & -y_1 & 0 & \mu_2 & 0 & \mu_1 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \lambda_2 & 0 & -y_2 & \mu_1 & \mu_2 & 0 & 0 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \lambda_2 & \lambda_1 & -y_3 & 0 & 0 & \mu_1 & \mu_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_3 & 0 \\ 0 & 0 & \lambda_2 & 0 & -y_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_3 \\ 0 & \lambda_1 & 0 & 0 & 0 & -y_5 & \mu_2 & 0 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \lambda_1 & 0 & \lambda_2 & -y_6 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_3 & \mu_2 & 0 & 0 \\ 0 & 0 & 0 & \lambda_2 & \lambda_1 & 0 & 0 & -y_7 & 0 & 0 & 0 & 0 & \mu_3 & 0 & \mu_1 & 0 & 0 \\ \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -y_8 & 0 & 0 \\ 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 \\ 0 & 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 \end{bmatrix} \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \\ p_4 \\ p_5 \\ p_6 \\ p_7 \\ p_8 \\ p_9 \\ p_{10} \\ p_{11} \\ p_{12} \\ p_{13} \\ p_{14} \\ p_{15} \\ p_{16} \end{bmatrix}$$

In the steady state, the derivatives of the state probabilities become zero and therefore equation (2) become

$$MP = 0 \tag{4}$$

which is in matrix form

$$
=\begin{bmatrix}
-y_0 & \mu_1 & \mu_2 & 0 & 0 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
\lambda_1 & -y_1 & 0 & \mu_2 & 0 & \mu_1 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
\lambda_2 & 0 & -y_2 & \mu_1 & \mu_2 & 0 & 0 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & \lambda_2 & \lambda_1 & -y_3 & 0 & 0 & \mu_1 & \mu_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_3 & 0\\
0 & 0 & \lambda_2 & 0 & -y_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_3\\
0 & \lambda_1 & 0 & 0 & 0 & -y_5 & \mu_2 & 0 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & \lambda_1 & 0 & \lambda_2 & -y_6 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_3 & \mu_2 & 0 & 0\\
0 & 0 & 0 & \lambda_2 & \lambda_1 & 0 & 0 & -y_7 & 0 & 0 & 0 & 0 & \mu_3 & 0 & \mu_1 & 0 & 0\\
\lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -y_8 & 0 & 0\\
0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0\\
0 & 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3
\end{bmatrix}
\begin{bmatrix}
p_0(\infty)\\ p_1(\infty)\\ p_2(\infty)\\ p_3(\infty)\\ p_4(\infty)\\ p_5(\infty)\\ p_6(\infty)\\ p_7(\infty)\\ p_8(\infty)\\ p_9(\infty)\\ p_{10}(\infty)\\ p_{11}(\infty)\\ p_{12}(\infty)\\ p_{13}(\infty)\\ p_{14}(\infty)\\ p_{15}(\infty)\\ p_{16}(\infty)
\end{bmatrix}
=\begin{bmatrix}
0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0
\end{bmatrix}
$$

The normalizing condition for this problem is:

$$
p_0(\infty)+p_1(\infty)+p_2(\infty)+p_3(\infty)+p_4(\infty)+\ldots+p_{16}(\infty)=1 \qquad (5)
$$

Following Wang and Kuo (2000) and Wang et al (2006) we substitute (5) in the last row of (4) to compute the steady-state probabilities.

$$
=\begin{bmatrix}
-y_0 & \mu_1 & \mu_2 & 0 & 0 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
\lambda_1 & -y_1 & 0 & \mu_2 & 0 & \mu_1 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
\lambda_2 & 0 & -y_2 & \mu_1 & \mu_2 & 0 & 0 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & \lambda_2 & \lambda_1 & -y_3 & 0 & 0 & \mu_1 & \mu_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_3 & 0\\
0 & 0 & \lambda_2 & 0 & -y_4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_3\\
0 & \lambda_1 & 0 & 0 & 0 & -y_5 & \mu_2 & 0 & 0 & 0 & 0 & \mu_3 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & \lambda_1 & 0 & \lambda_2 & -y_6 & 0 & 0 & 0 & 0 & 0 & 0 & \mu_3 & \mu_2 & 0 & 0\\
0 & 0 & 0 & \lambda_2 & \lambda_1 & 0 & 0 & -y_7 & 0 & 0 & 0 & 0 & \mu_3 & 0 & \mu_1 & 0 & 0\\
\lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -y_8 & 0 & 0\\
0 & 0 & 0 & \lambda_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\mu_3 & 0\\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1
\end{bmatrix}
\begin{bmatrix}
p_0(\infty)\\ p_1(\infty)\\ p_2(\infty)\\ p_3(\infty)\\ p_4(\infty)\\ p_5(\infty)\\ p_6(\infty)\\ p_7(\infty)\\ p_8(\infty)\\ p_9(\infty)\\ p_{10}(\infty)\\ p_{11}(\infty)\\ p_{12}(\infty)\\ p_{13}(\infty)\\ p_{14}(\infty)\\ p_{15}(\infty)\\ p_{16}(\infty)
\end{bmatrix}
=\begin{bmatrix}
0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\0\\1
\end{bmatrix}
$$

The system availability can be obtained from the solution for $p_i(\infty), i=0,1,2,\ldots,16.$ The steady-state availability (the proportion of time the system is in a functioning condition or equivalently, the sum of the probabilities of operational states) is given by

$$A_V(\infty) = p_0(\infty) + p_1(\infty) + p_2(\infty) + p_3(\infty) + p_4(\infty) + p_5(\infty) + p_6(\infty) + p_7(\infty) \qquad (6)$$

The expression for the $A_V(\infty)$ is too spacious to be shown here.

## 4. Numerical Example and Discussion

Numerical examples are presented to demonstrate the impact of repair and failure rates on steady-state availability and net profit of the system based on given values of the parameters. For the purpose of numerical example, the following sets of parameter values are used:

$\mu_1 = 0.12, \mu_2 = 0.1, \lambda_1 = 0.2, \lambda_2 = 0.2, \lambda_3 = 0.1, \mu_3(0.1, 0.2, 0.3)$ for figures 2-5 and $\lambda_3(0.1, 0.2, 0.3)$ for figures 6-9.

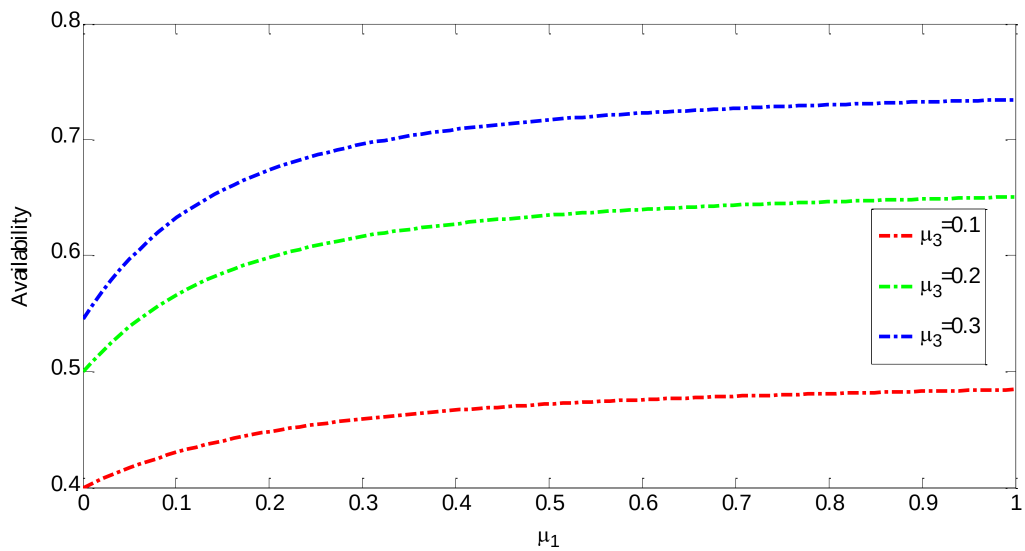


Figure 2: Availability against type I software repair rate $\mu_1$ for different values of $\mu_3(0.1, 0.2, 0.3)$

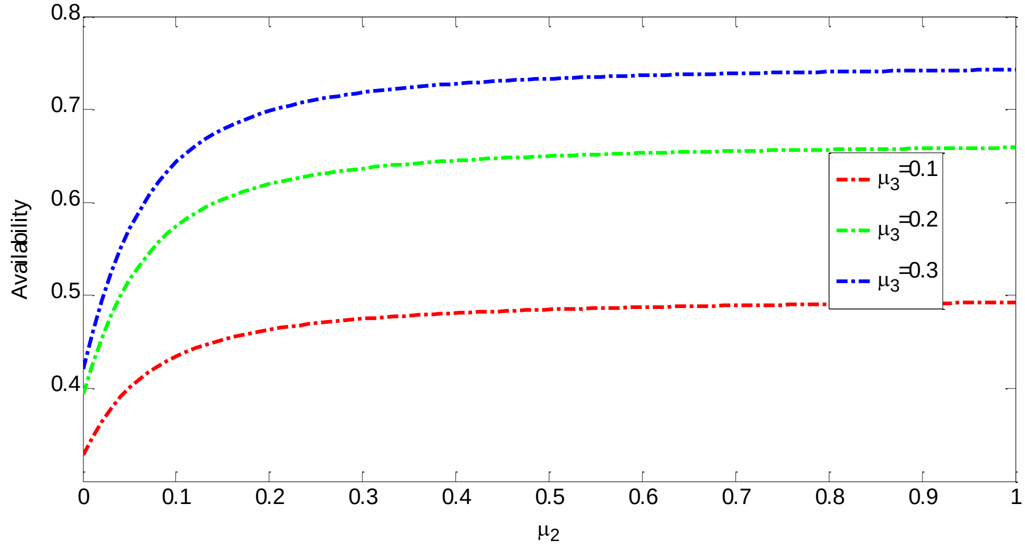


Figure 3: Availability against type II software repair rate $\mu_2$ for different values of $\mu_3(0.1, 0.2, 0.3)$

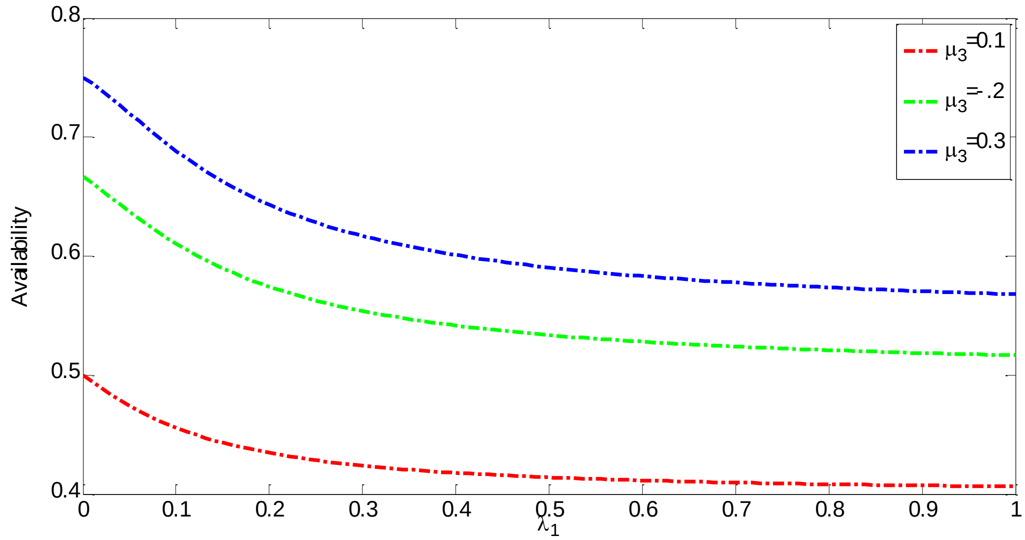


Figure 4: Availability against type I software failure rate $\lambda_1$ for different values of $\mu_3(0.1, 0.2, 0.3)$

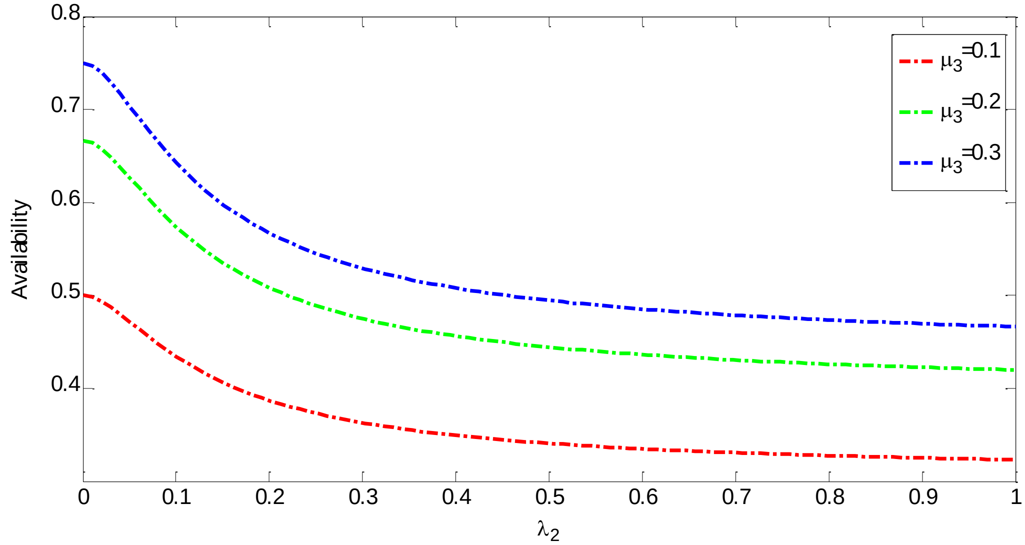


Figure 5: Availability against type II software failure rate $\lambda_2$ for different values of $\mu_3(0.1, 0.2, 0.3)$

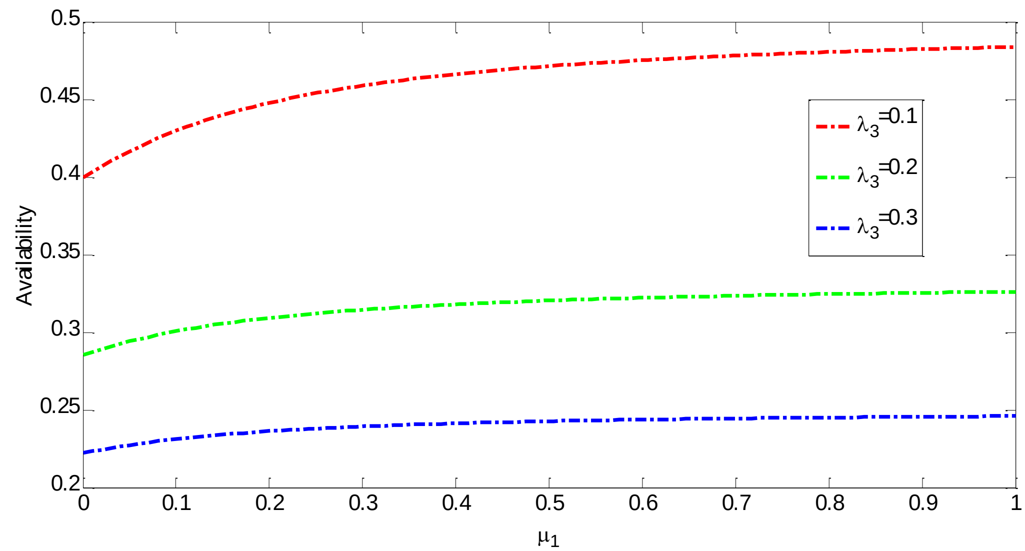


Figure 6: Availability against type I software repair rate $\mu_1$ for different values of $\lambda_3(0.1, 0.2, 0.3)$

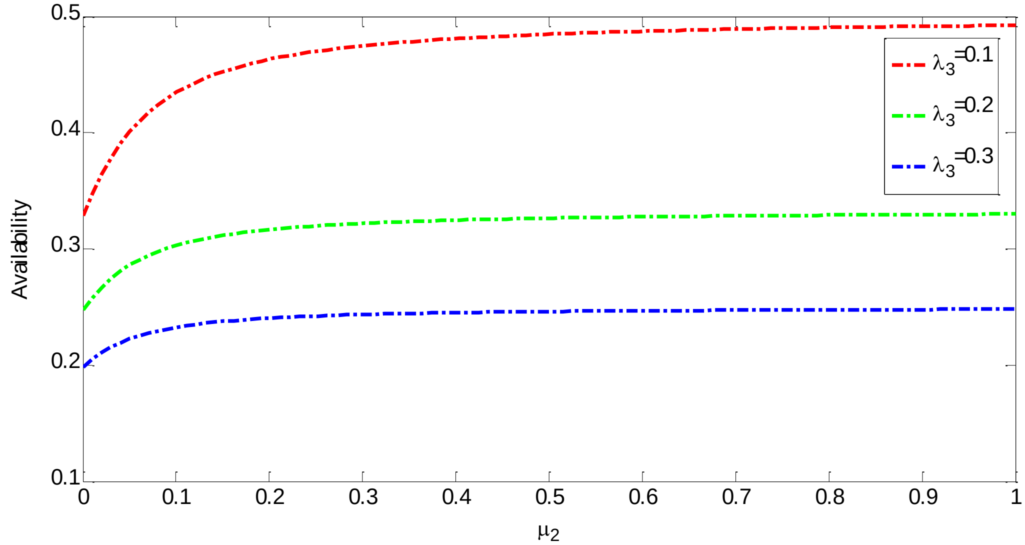


Figure 7: Availability against type II software repair rate $\mu_2$ for different values of $\lambda_3(0.1, 0.2, 0.3)$

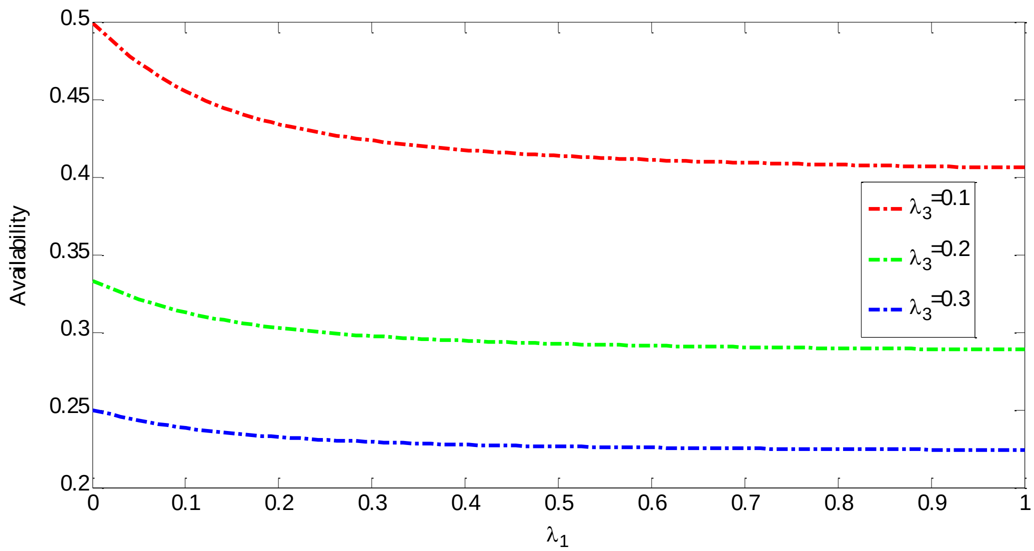


Figure 8: Availability against type I software failure rate $\lambda_1$ for different values of $\lambda_3(0.1, 0.2, 0.3)$

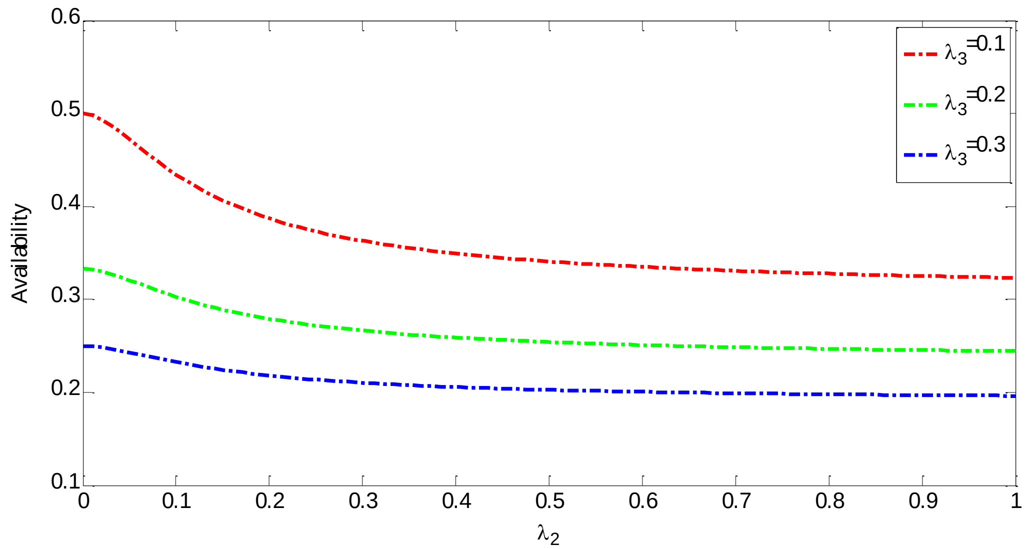


Figure 9: Availability against type II software failure rate $\lambda_2$ for different values of $\lambda_3 (0.1, 0.2, 0.3)$

Numerical results of availability with respect to $\mu_1$ and $\mu_2$ are depicted in Figures 2 and 3 for different values of hardware repair rate. In these figures, system availability increases as $\mu_1$ and $\mu_2$ increases for different values of hardware repair rate $\mu_3$. Availability is higher when both $\mu_1$ and $\mu_2$ are higher. From Figures 4 and 5, system availability decreases as $\lambda_1$ and $\lambda_2$ increases for different values of hardware repair rate $\mu_3$. It is evident from these figures that availability is higher when the hardware repair $\mu_3$ equal to 0.1. This sensitivity analysis implies that major maintenance to the hardware should be invoked to lower the hardware failure rate, improve and maximize the system availability as well as production output. Similar observation is depicted in Figures 6 and 7 with respect to $\mu_1$ and $\mu_2$ for different values of hardware failure rates respectively. From these figures, system availability increases as $\mu_1$ and $\mu_2$ increases for different values of hardware failure rate $\lambda_3$. The gaps between the curves in the figures widen as $\lambda_3$ decreases. Availability is higher when $\lambda_3 = 0.1$. On the other hand, Figures 8 and 9 shows that the availability decreases as $\lambda_1$ and $\lambda_2$ increases for different values of hardware failure rate. It is clear from these Figures that system availability display decreasing pattern for different values of $\lambda_3$. The gaps between the curves in these figure become wider as $\lambda_3$ decreases from 0.3 to 0.1. This sensitivity analysis implies that preventive maintenance should be invoked to the hardware to minimize the failure of the system in order to maximize the system availability.

## 5. Conclusion

In this paper, we studied the one host computer system with two types of software in cold standby. Explicit expression for the steady-state availability is derived. The numerical simulations presented in Figures 2 – 9 provide a description of the effect of the failure rate and repair rate on steady-state availability for different values of hardware failure and repair rates. On the basis of the numerical results obtained for particular cases, it is suggested that the system availability can be improved significantly by:

(i) Adding more software and host in cold standby

(ii) Increasing the repair rate.

(iii) Reducing the failure rate of the system by hot or cold duplication method.

The system can further be developed into system with multiple hosts with heterogeneous software in solving reliability and availability problems. The present study will serve as a guide in relation to efficiency, reduction of system failure and operational costs, increase in production output and revenue mobilized.

## Conflict of Interests

The authors declare that there is no conflict of interests.

## REFERENCES

[1] A. Kumar, J. Anand, S.C. Malik. Stochastic Modeling of a Computer System with Priority to Up-gradation of Software over Hardware Repair Activities. Int. J. Agric. Stat. Sci., 9(1)(2013), 117-126.

[2] Kumar, S. C. Malik. Stochastic Modeling of a Computer System with Priority to PM over S/W Replacement Subject to Maximum Operation and Repair Times. Int. J. Comput. Appli., 43 (3)(2012), 27-34.

[3] Kumar, S.C. Malik. Reliability modelling of a computer system with priority to H/w repair over replacement of H/w and up-gradation of S/w subject to MOT and MRT, Jordan J. Mech. Ind. Eng., 8 (4)(2014), 233-241.

[4] Kumar, M. Saini, S.C. Malik. Performance Analysis of a Computer System with Imperfect Fault Detection of Hardware, Proc. Comput. Sci., 45(2015), 602-610.

[5] J. Anand, S.C. Malik. Analysis of a Computer System with Arbitrary Distributions for H/W and S/W Replacement Time and Priority to Repair Activities of H/W over Replacement of the S/W, Int. J. Syst. Assurance Eng. Manag., 3(3)(2012), 230-236.

[6] S.C. Malik. Reliability Modelling of a computer System with Preventive Maintenance and Priority Subject to Maximum Operation and Repair Times. Int. J. Syst. Assurance Eng. Manag., 4 (1)(2013), 94-100.

[7] S.C.Malik,J. Anand. Reliability and Economic Analysis of a Computer System with Independent Hardware and Software Failures, Bull. Pure Appl. Sci., 29 E (Math. & Stat.), No. 1, (2010), 141-153.

[8] S.C. Malik, J.K. Sureria. Profit Analysis of a Computer System with H/W Repair and S/W Replacement. Int J Comput. Appl, 47(1)(2012), 19-26.

[9] S.C. Malik, V.J. Munday. Stochastic Modeling of a Computer System with Hardware Redundancy, Int. J. Comput. Appl., 89 (7)(2014), 26-30.

[10] S.R.Welke, S.W. Labib, A.M. Ahmed. Reliability Modelling of Hardware/ Software System, IEEE Trans. Reliab., 44(3)(1995), 413-418.