Stochastic Modelling of a Computer System with Hardware Redundancy

S.C. Malik
Dept. of Statistics
M.D. University, Rohtak-124001 (India)

V.J. Munday
Dept. of Statistics
M.D. University, Rohtak-124001 (India)

ABSTRACT
In this paper, an effort for the stochastic analysis of a computer system has been made considering the idea of hardware redundancy in cold standby. The hardware and software failures occur independently in the computer system with some probability. A single server is employed immediately to conduct hardware repair and software upgradation on need basis. The repair and upgradation activities performed by the server are perfect. The time to hardware and software failures follows negative exponential distribution, whereas the distributions of hardware repair and software upgradation times are taken as arbitrary with different probability density functions. The expressions for various reliability measures are derived in steady state using semi-Markov process and regenerative point technique. The graphs are drawn for arbitrary values of the parameters to depict the behaviour of some important performance measures of the system model.

Keywords
Computer System, Hardware Redundancy, Stochastic Model and Reliability Measures

1. INTRODUCTION
The demand for computer systems has increased many folds during last few years because of their wide applications in several sensitive areas like banking, communication, home appliances, automobiles and aerospace. Consequently, this leads to the need to specify and design computing systems which could fulfil the requirements of targeted applications at the lowest cost. But, designing a system to perform its intended job at least for a specific duration always has been a challenge to the reliability practitioners. The technique of redundancy has been adopted frequently as an effective strategy for enhancing system life span. Cao and Wu [1989], Lam [1997], Yadavalli et al. [2004] and Kumar et al. [2012] analyzed repairable system models using unit wise redundancy.

In spite of increasing demand of computer technology, a little work has been dedicated to the stochastic modelling of computer systems with independent failures of hardware and software components. And, most of the research work has been carried out either considering hardware or software alone. However, Malik and Anand [2010, 12] and Kumar et al. [2013] tried to establish computer system models with unit wise cold standby redundancy. It has been proved that component wise redundancy is better than unit wise redundancy in sense of reliability.

Thus, purpose of the present study is to analyze stochastically a computer system by providing hardware redundancy in cold standby. In computer system, hardware and software failures occur independently with some probability. A single server is called immediately to conduct hardware repair and software up-gradation when needed. The repair and up-gradation activities performed by the server are perfect. The time to hardware and software failures follows negative exponential distribution while the distributions of hardware repair and software up-gradation times are taken as arbitrary with different probability density functions. The expressions for various reliability measures such as transition probabilities and mean sojourn times, mean time to system failure (MTSF), availability, busy period of the server due to hardware repair and software up-gradation, expected number of hardware repairs and software up-gradations and profit function are derived in steady state using semi-Markov process and regenerative point technique. The cost-benefit analysis has been made using these measures. The graphs are drawn for arbitrary values of the parameters to depict the behaviour of MTSF, availability and profit function of the system model.

2. NOTATIONS
E : Set of regenerative states
E* : Set of non-regenerative states
O : Computer system is operative
Hcs : Hardware is in cold standby
a/b : Probability that the system has hardware / software failure
λ1/λ2 : Hardware/Software failure rate
HFUr/HFWr : The hardware is failed and under repair/waiting for repair
SFUg/SFWUg : The software is failed and under/waiting up-gradation
HFUR/HFWR : The hardware failed and continuously under repair / waiting for repair from previous state
SFUG/SFWUG : The software is failed and continuously under up-gradation /waiting for up-gradation from previous state
g(t)/G(t) : pdf/cdf of hardware repair time
f(t)/F(t) : pdf/cdf of software up-gradation time
q1(t)/ Q1(t) : pdf / cdf of first passage time from regenerative state Si to a regenerative state Sj or to a failed state Sj without visiting any other regenerative state in (0, t]
qik(t)/ Qik(t) : pdf/cdf of direct transition time from regenerative state Si to a regenerative state Sj or to a failed state Sj visiting state Sk once in (0, t]
Mj(t) : Probability that the system up initially in state Sj ∈ E is up at time t without visiting to any regenerative state
W(t) : Probability that the server is busy in the state Si up to time ‘t’ without making any transition to any other regenerative state or returning to the same state via one or more non-regenerative states.

μi : The mean sojourn time in state Si which is given by

\[ \mu_i = E(T) = \int_0^\infty p(T > t) \, dt = \sum_j m_{ij} \]  

where T denotes the time to system failure.

m_i : Contribution to mean sojourn time (μi) in state Si when system transits directly to state Sj so that

\[ \mu_i = \sum_j m_{ij} \]  

\[ m_i = \int_0^\infty \text{Id} Q_{ij}(t) = -q_i^* (0) \]

S/© : Symbol for Laplace-Stieltjes convolution/Laplace convolution

q/** : Symbol for Laplace Transformation (LT)/Laplace Stieltjes Transformation (LST)

3. TRANSITION PROBABILITIES AND MEAN SOJOURN TIMES

Simple probabilistic considerations yield the following recursive relations for the non-zero elements.

\[ p_{ij} = Q_{ij}(\infty) = \int_0^\infty q_i(t) \, dt \]

\[ p_{01} = \frac{1}{a_{11} + b_{12}} \]

\[ p_{10} = g^*(a_{11} + b_{12}) \]

\[ p_{13} = \frac{1}{a_{11} + b_{12}} \]  

\[ 1 - g^*(a_{11} + b_{12}) \]

\[ p_{14} = \frac{1}{a_{11} + b_{12}} \]  

\[ 1 - g^*(b_{11} a_{12} + b_{12}) \]

\[ p_{20} = f^*(0) \]

\[ p_{30} = p_{40} = g^*(0) \]

For f (t) = \( \theta e^{-\theta t} \) and g (t) = \( \alpha e^{-\alpha t} \) we have

\[ p_{11} = \frac{1}{a_{11} + b_{12}} \]  

\[ 1 - g^*(a_{11} + b_{12}) \]

\[ p_{12} = \frac{1}{a_{11} + b_{12}} \]  

\[ 1 - g^*(a_{11} + b_{12}) \]

But, \( f(0) = \theta \), \( g(0) = \alpha \) and \( a + b = 1 \)

\[ \mu_0 = \frac{1}{a_{11} + b_{12}} \]

\[ \mu_1 = \frac{1}{a_{11} + b_{12} + a} \]  

\[ \mu_2 = \frac{1}{\theta} \]

Also

\[ \mu_0 = m_{01} + m_{02} \]

\[ \mu_1 = m_{10} + m_{12} + m_{14} \]  

\[ \mu_2 = m_{20} \]

and

\[ \mu'_3 = m_{10} + m_{114} + m_{123} = \frac{1}{\alpha} \]  

(3)

4. RELIABILITY AND MEAN TIME TO SYSTEM FAILURE (MTSF)

Let \( \phi(t) \) be the cdf of first passage time from regenerative state S1 to a failed state. Regarding the failed state as absorbing state, we have the following recursive relations for \( \phi(t) \):

\[ \phi_0(t) = Q_{00}(t) \sum \phi_i(t) + Q_{02}(t) \]

\[ \phi_1(t) = Q_{01}(t) \sum \phi_i(t) + Q_{11}(t) + Q_{12}(t) \]  

(4)

Taking LST of equation (4) and solving for \( \phi^*(s) \), we have

\[ R^*(s) = \frac{1 - \phi_0^*(s)}{s} \]  

(5)

The reliability of the system model can be obtained by taking Laplace inverse transform of the equation (5).

The mean time to system failure (MTSF) is given by

\[ \text{MTSF} = \lim_{s \to 0} \frac{1 - \phi_0^*(s)}{s} = \frac{N_1}{D_1} \]  

(6)

Where

\[ N_1 = p_{00} \mu_1 + \mu_0 \]  

and

\[ D_1 = 1 - p_{00} \mu_0 \]  

(7)

5. STEADY STATE AVAILABILITY

Let \( A_i(t) \) be the probability that the system is in up-state at instant ‘t’ given that the system entered regenerative state \( S_i \) at t=0. The recursive relations for \( A_i(t) \) are given as:

\[ A_i(t) = M_i(t) + q_{i0}(t) \odot A_0(t) + q_{i1}(t) \odot A_1(t) + q_{i2}(t) \odot A_2(t) \]

\[ A_i(t) = M_i(t) + q_{i0}(t) \odot A_0(t) + q_{i14}(t) \odot A_1(t) + q_{i12}(t) \odot A_2(t) \]

\[ A_3(t) = q_{i0}(t) \odot A_0(t) \]  

(8)

where

\[ M_0(t) = -q_{i1} b_{12} k \]  

\[ M_1(t) = q_{i1} (a_{11} + b_{12}) k \]

Taking LT of equation (8) and solving for \( A_i^*(s) \), the steady state availability is given by

\[ A_i(\infty) = \lim_{s \to 0} s A_i(s) = \frac{N_i}{D_2} \]  

(9)

Where

\[ N_2 = \mu_0 (1 - p_{11}) + p_{00} \mu_1 \]  

\[ D_2 = (1 - p_{114}) h_4 + p_{110} a_4 + p_{10} p_{02} a_2 + p_{123} a_5 a_2 \]  

(10)

6. BUSY PERIOD OF THE SERVER

(a). Due to Hardware Repair

Let \( B^H(t) \) be the probability that the server is busy in the unit due to hardware failure at an instant ‘t’ given that the system entered state \( S_i \) at t = 0. The recursive relations for \( B^H(t) \) are as follows:

\[ B_0^H(t) = q_{00}(t) \odot B_1^H(t) + q_{02}(t) \odot B_2^H(t) \]

\[ B_1^H(t) = W_1^H(t) + q_{10}(t) \odot B_0^H(t) + q_{114}(t) \odot B_1^H(t) + q_{123}(t) \odot B_2^H(t) \]

\[ B_2^H(t) = q_{20}(t) \odot B_0^H(t) \]  

(11)

where

\[ W_1^H(t) = e^{(a_{11} + b_{12}) k} G(t) + (a_{11} e^{(a_{11} + b_{12}) k} \odot 1) G(t) \]

\[ G(t) = e^{(b_{12} e^{(a_{11} + b_{12}) k} \odot 1)} \]

(b). Due to Software Up-Gradation

Let \( B^S(t) \) be the probability that the server is busy due to replacement of the software at an instant ‘t’ given that the system entered the regenerative state \( S_i \) at t = 0. We have the following recursive relations for \( B^S(t) \):

\[ B_0^S(t) = q_{00}(t) \odot B_1^S(t) + q_{02}(t) \odot B_2^S(t) \]

\[ B_1^S(t) = q_{10}(t) \odot B_0^S(t) + q_{114}(t) \odot B_1^S(t) + q_{123}(t) \odot B_2^S(t) \]
\[ B_2^S(t) = W^2_2(t) + q_{20}(t) \cap B_0^S(t) \] (12)

where

\[ W^2_2(t) = \frac{1}{\lambda} \int_0^t dt \]

Taking LT of equations (11) & (12), solving for \( B_0^{H+} (s) \) and \( B_0^{S+} (s) \), the time for which server is busy due to repair and replacements respectively is given by

\[ B_0^{H+} = \lim_{s \to 0} sB_0^{H+} (s) = \frac{N_1^H}{D_2} \] (13)

\[ B_0^{S+} = \lim_{s \to 0} sB_0^{S+} (s) = \frac{N_1^S}{D_2} \] (14)

where

\[ N_1^H = p_{01}w_{12}^H(0) \]

\[ N_1^S = (p_{01}p_{12.3} + p_{02}(1 - p_{11.4}))w_{12}^S(0) \]

and \( D_2 \) is already mentioned. (15)

7. EXPECTED NUMBER OF HARDWARE REPAIRS

Let \( NHR_i(t) \) be the expected number of hardware repairs by the server in \((0, t] \) given that the system entered the regenerative state \( S_i \) at \( t = 0 \). The recursive relations for \( NHR_i(t) \) are given as:

\[ NHR_0(t) = Q_{01}(t) \cap [1 + NHR_i(t)] + Q_{02}(t) \cap NHR_{i+1}(t) \]

\[ NHR_i(t) = Q_{01}(t) \cap NHR_{i-1}(t) + Q_{11}(t) \cap NHR_i(t) + Q_{12.3} \cap NHR_{i+1}(t) \] (16)

Taking LST of equation (16) and solving for \( R_0^*(s) \). The expected number of hardware repair is given by

\[ NHR_0 = \lim_{s \to 0} sNHR_0^*(s) = \frac{N_1}{D_2} \] (17)

where

\[ N_1 = p_{01}(1 - p_{11.4}) \] and \( D_2 \) is already mentioned. (18)

8. EXPECTED NUMBER OF SOFTWARE UP-GRADATIONS

Let \( NSU_i(t) \) be the expected number of software upgradations in \((0, t] \) given that the system entered the regenerative state \( S_i \) at \( t = 0 \). The recursive relations for \( NSU_i(t) \) are given as follows

\[ NSU_0(t) = Q_{01}(t) \cap NSU_0(t) + Q_{02}(t) \cap [1 + NSU_2(t)] \]

\[ NSU_i(t) = Q_{01}(t) \cap NSU_{i-1}(t) + Q_{11}(t) \cap NSU_i(t) + Q_{12.3} \cap NSU_{i+1}(t) \]

\[ NSU_2(t) = Q_{01}(t) \cap NSU_0(t) \] (19)

Taking LST of equation (19) and solving for \( N_5^*(s) \). The expected numbers of software up-gradation are given by

\[ NSU_0(\infty) = \lim_{s \to 0} sNSU_0^*(s) = \frac{N_5}{D_2} \] (20)

where

\[ N_5 = p_{02}(1 - p_{11.4}) \] and \( D_2 \) is already specified. (21)

9. COST-BENEFIT ANALYSIS

The profit incurred to the system model in steady state can be obtained as:

\[ P = K_1A_0 - K_2B_0^H - K_2B_0^S - K_3NHR_0 - K_4NSU_0 \]

where

\[ K_1 = \text{Revenue per unit up-time of the system} \]

\[ K_2 = \text{Cost per unit time for which server is busy due to hardware failure} \]

\[ K_3 = \text{Cost per unit repair of the failed hardware} \]

\[ K_4 = \text{Cost per unit up-gradation of the failed software and} \]

\( A_0, B_0^H, B_0^S, NHR_0, NSU_0 \) are already defined.

10. PARTICULAR CASES

Suppose \( g(t) = \alpha e^{-\alpha t} \) and \( f(t) = \theta e^{-\theta t} \)

We can obtain the following results:

\[ MTSF(T_o) = \frac{N_1}{D_2} \]

Availability \( (A_0) = \frac{N_5}{D_2} \)

Busy period due to hardware failure \( (B_0^H) = \frac{N_3^H}{D_2} \)

Busy period due to software failure \( (B_0^S) = \frac{N_3^S}{D_2} \)

Expected number of repair at hardware failure \( (NHR_0) = \frac{N_4}{D_2} \)

Expected number of up-gradation at software failure \( (NSU_0) = \frac{N_5}{D_2} \)

Where

\[ N_1 = \frac{2\alpha\lambda_1 + b\lambda_2 + \alpha}{(a\lambda_1 + b\lambda_2)(a\lambda_1 + b\lambda_2 + \alpha)} \]

\[ D_1 = \frac{(a\lambda_1 + b\lambda_2)(a\lambda_1 + b\lambda_2 + \alpha) - a\lambda_1\alpha}{a\lambda_1 + b\lambda_2} \]

\[ N_2 = \frac{1}{a\lambda_1 + b\lambda_2} \]

\[ D_2 = \frac{(a\lambda_1 + b\lambda_2 + \alpha)(\theta a\lambda_1 + \phi b\lambda_2) + \theta\alpha(b\lambda_2 + \alpha)}{\theta\alpha(a\lambda_1 + b\lambda_2)(a\lambda_1 + b\lambda_2 + \alpha)} \]

\[ N_3^H = \frac{a\lambda_1}{\alpha(a\lambda_1 + b\lambda_2)} \]

\[ N_3^S = \frac{b\lambda_2}{\theta(a\lambda_1 + b\lambda_2)} \]
\[ N_4 = \frac{a\lambda_1(b\lambda_2 + \alpha)}{(a\lambda_1 + b\lambda_2)(a\lambda_1 + b\lambda_2 + \alpha)} \]
\[ N_5 = \frac{b\lambda_2 + \alpha}{(a\lambda_1 + b\lambda_2 + \alpha)} \]

11. CONCLUSION

The effect of various parameters on performance measures of computer system model has been observed for a particular case as shown in figures 2, 3, and 4. It is analyzed that mean time to system failure (MTSF), availability and profit function go on decreasing with the increase of failure rates (\(\lambda_1\) and \(\lambda_2\)) while their values increase with the increase of hardware repair rate (\(\alpha\)) and software up-gradation rate (\(\theta\)) provided chances of hardware failure rate are more than that of software failure (\(a>b\)). However, the effect software failure rate is more on these measures. It is interesting to note that MTSF and availability decline in case software failure chance are high whereas system becomes more profitable may because of less cost for software up-gradation.

Hence, a computer system in which hardware redundancy is provided in cold standby can be made more profitable and reliable to use either by operating the software carefully or by increasing hardware repair rate and software up-gradation rate giving less cost for software up-gradation.

12. ACKNOWLEDGEMENT

The authors are grateful to the experts who have contributed towards development of the paper.

13. REFERENCES


---

**State Transition Diagram**

![State Transition Diagram](image)

○ Up-State □ Failed State • Regenerative Point

**Fig. 1**
MTSF Vs Hardware Failure Rate ($\lambda_1$)

Availability Vs Hardware Failure Rate ($\lambda_1$)

Profit Vs Hardware Failure Rate ($\lambda_1$)