International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 07 Issue: 05 | May 2020 www.irjet.net p-ISSN: 2395-0072

Multi Generation Software Reliability Growth Model based on Log Logistic Test Effort Function

Suneet Saxena[1], Ajay kumar Gupta [2], Kumar Akhilesh Mohan [3]

[1]Department of Mathematics, SRMS College of Engineering & Technology, Bareilly (U.P), India
[2]Department of Mathematics, Bhagwant Institute of Technology, Muzaffarnagar (U.P.), India
[3]Department of Mathematics, Bhagwant Universtiy, Ajmer, India

---***---

Abstract - *Software Company has to upgrade its software system to meet the customer requirements which changes with time. Multi generation is also referred as multi-up gradation process in which new version of software is released in the market with some additional new features. Paper presents development of software reliability growth model under multi generation process using log logistic test effort function. Log logistic distribution describes increasing/decreasing phenomenon effectively as compared to weibull and logistic distribution. Three generation of model have been developed and compared using statistical tools R^2 and MSE. Results validates good fitting of third generation to given dataset.*

Key Words: *NHPP, Software Reliability, Multi Generation, log logistic Distribution.*

1. INTRODUCTION

Now a day's online services such as banking transaction, railway reservation and airline booking system, hotel booking, food ordering system etc are being used frequently by large number of users. With rapid growth in online services, complexity and size increases many times of software system running such services. Most important task of software developers is how to ensure that software perform its desired operation without failure. Reliability is the probability that system perform its desired function under specified conditions. To improve the reliability of software by analyzing failures, many mathematical models have been developed. Such models are called Software Reliability Growth Model (SRGM). SRGM predicts the number of faults detected with respect to testing time. Reliability depends on faults detected and corrected. Non Homogenous Poisson Process (NHPP) are extensively used to study failure distribution of software system.

Software reliability depends on number of test-cases, testing coverage, CPU hours, Human resource collectively known as Test Effort Functions (TEF). TEF involves two types of constrains namely, time and resource. Time constraint is the limited time available for testing as software has to be released in market. Resource constraint (human resource, CPU hours, etc.) is the limited resources available for testing.. Reliability of software depends on TEF and it is necessary to effectively consume TEF so as to achieve optimum reliability of software system.

Many authors have developed SRGM incorporating TEF. In 2002 Huang and Kuo investigated a SRGM based on logistic TEF and predicted optimal software release policy involving cost-reliability criteria. In 2008 Ahmad et al. proposed SRGM based on exponential and weibull distribution. They incorporated various TEF and estimated optimal release policy. In year 2011 log-logistic TEF with imperfect debugging used by Ahmad et al. They analyzed an inflection S- shaped SRGM. In year 2012 Aggarwal et al. incorporated various TEF in modular software. They categorized total faults as simple, hard and complex faults. These faults were considered as function of TEF described by Weibull type distribution. An optimization problem has been formulated of maximizing total faults removed subject to budgetary and reliability constrains. Genetic algorithm has been used to solve the problem. Reddy and Raveendrababu in year 2012 incorporated generalized exponential TEF while developing SRGM. In year 2013 Shinji Inoue and Shigeru Yamada proposed SRGM based on continuous time model such as lognormal process. They used Wiener process to represent fluctuations in fault detection and approximated fault detection rate with weibull based test effort function.

In this dynamic world, requirements of end user changes with time. To meet such requirements, working software has to be upgraded continuously. Railway system, Banking system, Health and Hospitality sector etc. requires continuous up gradation of software system. Multi up gradation is one of the various factors on which software reliability depends. If some new features are added to running software then possibility of failures increases. During testing phase faults detected and fixed. The process is termed as up gradation. In multi up gradation software upgraded continuously with time. Original software referred as first generation. The up graded first generation software referred as Second generation and so on. In 2010 Kapur, Tandon and kaur developed a multi up gradation SRGM considering that cumulative faults removed in a generation depend on all previous generation. Kapur et al. In year 2012 presented multi up gradation SRGM involving weibull testing effort function. Particular case discussed using two parameter exponential power distribution function. Recently Li and Yi [9] in year 2016 developed Multi generation SRGM involving power law function as TEF.

In this paper we propose Multi generation SRGM in which fault detection rate approximated by log logistic TEF. The log Logistic TEF captures increasing/decreasing failure occurrence phenomenon effectively. After introduction there are five section in this paper. Section 2 and 3 presents descriptions of Log Logistic model and some test effort functions. Section 4 discussed development of model and solution. Section 5 provides estimation of the parameters and comparison using statistical tools. Finally conclusions have been highlighted in section 6.

1.1 NOTATIONS

E	Total testing effort consumed.
β	Shape parameter of Log-Logistic, Logistic and Weibull distribution.
γ	Scale parameter of Log-Logistic, Exponential, Rayleigh and Weibull distribution.
α	Consumption rate of testing effort expenditures in the Logistic TEF.
a	Total number of faults
$m(t)$	Expected number of faults at time t
r	Fault detection rate constant

2. LOG-LOGISTIC SOFTWARE RELIABILITY GROWTH MODEL

Earlier NHPP models have constant, increasing or decreasing failure occurrence rates per fault. These models were inadequate to capture the failure processes underlying some of the failure data sets, which exhibit an random increasing/decreasing failure occurrence rates per fault. The Log-Logistic model was proposed to capture increasing/decreasing failure occurrence rates per fault.

Log-Logistic model is based on log-logistic distribution which is the distribution of a random variable whose logarithm has the logistic distribution. The distribution has applications in software reliability as it is used to model random failure rate. Logistic distribution has no shape parameter. It has only one shape which is the bell shape and depicts only increasing-decreasing failure rate. The shape of the logistic distribution is very similar to that of the normal distribution. Log-logistic distribution has scale as well as shape parameter which depicts either decreasing failure rate, or mixed decreasing-increasing failure rate, depending on the shape parameter.

3. TEST EFFORT FUNCTION

3.1 EXPONENTIAL TEF

It is non-increasing function. The PDF (current testing effort function at any time t) is given by $f(t) = E\exp(-\lambda t)$.
CDF [cumulative testing effort function consumed in (0,t)] is given by $F(t) = \int_0^t f(t)\,dt = E[1 - \exp(-\lambda t)]$.

3.2 RAYLEIGH TEF

This TEF exhibits both increasing and decreasing phenomenon. PDF is represented by $f(t) = 2E\lambda t\exp(-\lambda t^2)$ and CDF is $F(t) = E[1 - \exp(-\lambda t^2)]$.

3.3 WEIBULL TEF

It is generalized case of Exponential and Rayleigh TEF. Also exhibits peak phenomenon initially increasing and then decreasing. Its PDF is $f(t) = \lambda E\beta t^{\beta-1}\exp(-\lambda t^{\beta})$ CDF is given by $F(t) = E[1 - \exp\left(-\lambda t^{\beta}\right)]$.

3.4 LOGISTIC TEF

It is a smooth bell shaped function and represents TEF fairly accurate.PDF is given by

$$f(t) = \frac{E\lambda\alpha\exp(-\alpha t)}{[1+\lambda\exp(-\alpha t)]^2}.$$

CDF is given by $F(t) = \frac{E}{[1+\lambda\exp(-\alpha t)]}$.

3.5 LOG LOGISTIC TEF

It is similar to the log- normal distribution with elongated tails.

Its PDF is $f(t) = \frac{E\left(\frac{t}{\lambda}\right)^{-\beta}\beta}{\left[1+\left(\frac{t}{\lambda}\right)^{-\beta}\right]^2 t}$.

CDF is given by $F(t) = \frac{E}{1+\left(\frac{t}{\lambda}\right)^{-\beta}}$.

Parameter λ>0 is a scale parameter and is also the median of the distribution. The parameter β>0 is a shape parameter. The distribution is unimodal when β>1 and its dispersion decreases as βincreases.

4. MODEL DEVELOPMENT

The rate of fault detected at time t depends on instantaneous TEF and remaining fault in software at time t.

$$\frac{dm}{dt} = f(t)[a - m(t)] \quad (1)$$

Using condition m(0) = 0 and integrating we get

$$m(t) = a\left[1-\exp\left(-\int_0^t f(t)\,dt\right)\right] = a[1-\exp(-(F(t)-F(0)))] \quad (2)$$

Using log logistic TEF, we get

$$m(t) = a\left[1-\exp\left(-\mathrm{E}\left(\left(1+\left(\frac{t}{\lambda}\right)^{-\beta}\right)^{-1}-1\right)\right)\right] \quad (3)$$

$m(t) = aT(t)$ where

$$T(t) = \left[1-\exp\left(-\mathrm{E}\left(\left(1+\left(\frac{t}{\lambda}\right)^{-\beta}\right)^{-1}-1\right)\right)\right] \quad (4)$$

$T(t)$ is Test Effort Probability Distribution function.

4.1 FIRST GENERATION

Let a_1 be the total number of faults and $m_1(t)$ number of faults at time t. Software released at time t_1, equation (4) becomes

$$m_1(t) = a_1\left[1-\exp\left(-\mathrm{E}\left(\left(1+\left(\frac{t}{\lambda}\right)^{-\beta}\right)^{-1}-1\right)\right)\right]$$

$$0 \le t \le t_1 \quad (5)$$

4.2 SECOND GENERATION

Let a_2 be the number of faults correspond to second generation and remaining faults of first generation are $a_1 - m_1(t_1)$. Total faults are $a_2 + a_1 - m_1(t_1)$. If software released at time t_2 then faults $m_2(t)$ detected at any time t given by

$$m_2(t) = [a_2 + a_1 - m_1(t_1)]T(t-t_1) \quad (6)$$
$$t_1 \le t \le t_2$$

Where $[a_2 + a_1 - m_1(t_1)] = \left[a_2 + a_1 - a_1\left[1-\exp\left(-\mathrm{E}\left(\left(1+\left(\frac{t_1}{\lambda}\right)^{-\beta}\right)^{-1}-1\right)\right)\right]\right]$ and

$$T(t-t_1) = \left[1-\exp\left(-\mathrm{E}\left(\left(1+\left(\frac{t-t_1}{\lambda}\right)^{-\beta}\right)^{-1}-1\right)\right)\right]$$

4.3 THIRD GENERATION

Let a_3 be the number of faults correspond to third generation and remaining faults of second generation are $a_2 - m_2(t_2 - t_1)$. Total faults are $a_3 + a_2 - m_2(t_2 - t_1)$. If software released at time t_3 then faults $m_3(t)$ detected at any time t given by

$$m_3(t) = [a_3 + a_2 - m_2(t_2 - t_1)]T(t-t_3) \quad (7)$$
$$t_2 \le t \le t_3$$

Where $[a_3 + a_2 - m_2(t_2 - t_1)] = [a_3 + a_2 - [a_2 + a_1 - m_1(t_1)]T(t_2 - 2t_1)] = \left[a_3 + a_2 - \left[a_2 + a_1 - a_1\left[1-\exp\left(-\mathrm{E}\left(\left(1+\left(\frac{t_1}{\lambda}\right)^{-\beta}\right)^{-1}-1\right)\right)\right]\right]T(t_2 - 2t_1)\right]$

Also $T(t-t_3) = \left[1-\exp\left(-\mathrm{E}\left(\left(1+\left(\frac{t-t_3}{\lambda}\right)^{-\beta}\right)^{-1}-1\right)\right)\right]$

5. PARAMETER ESTIMATION AND MODEL COMPARISIONS

5.1 PARAMETER ESTIMATION

Parameters of models are estimated by using Non Linear Least Square Method. Data set used is Apache 2.0.39.

5.2 MODEL COMPARISONS

Three generation of models are compared using following tools of Goodness of Fit (GoF).

5.2.1 COEFFICIENT OF DETERMINATION R^2

Coefficient of Determination is also known as multiple correlation coefficient. It measures the correlation between the dependent and independent variables. Value of R^2 vary from 0 to 1. If $R^2 = 1$ then perfect fitting, $R^2 = 0$ no fitting, and R^2 close to 1 good fitting. R^2 is defined as

$$\mathrm{R}^2 = \frac{\sum_{j=1}^{n}(Y_j - y_j)^2}{\sum_{j=1}^{n}(y_j - \bar{y})^2} \qquad \bar{y} = \frac{1}{n}\sum_{j=1}^{n} y_j$$

where y_j is observed cumulative faults at time j and Y_j estimated cumulative faults at time j. n is number of data points. Model fits better to given dataset if R^2 close to 1.

Table 1. Goodness of Fit

	R^2	MSE
First Generation	0.688	1.11
Second Generation	0.875	0.67
Third Generation	0.922	0.54

6. CONCLUSION

Paper discussed about multi up gradation software reliability growth model. Model has been developed using log logistic test effort function. Distribution described increasing/decreasing phenomenon fairly. Three generation of model have been developed and compared using statistical tools R^2 and MSE. Table of goodness shows value of R^2 is higher for third generation up gradation and also MSE is more closes to zero. Results validates good fitting of third generation to given dataset.

REFERENCES

1. C.Y. Huang, and S.Y. Kuo, "Analysis of Incorporating Logistic Testing-Effort Function into Software Reliability Modeling", IEEE Transactions On Reliability, 2002, 51, 3, pp 261-270.
2. N. Ahmad, M.U. Bokhari, S.M.K. Quadri, and M.G.M. Khan, "The Exponentiated Weibull Software Reliability Growth Model with Various Testing-Efforts and Optimal Release Policy: A Performance Analysis", International Journal of Quality and Reliability Management, 2008, 25 (2), pp 211-235.
3. N. Ahmad, M.G.M Khan, and L.S. Rafi, "Analysis of an Inflection S-Shaped Software Reliability Model Considering Log-Logistic Testing-Effort and Imperfect Debugging", International Journal of Computer Science and Network Security,2011, 11, 1, pp 161-171.
4. A.G. Agarwal, P.K. Kapur, G. Kaur, and R. Kumar, "Genetic Algorithm Based Optimal Testing Effort Allocation Problem for Modular Software", BIJIT-BVICAM's International Journal of Information Technology, 2012, 4,1, pp 445-451.
5. K.V.S Reddy and B. Raveendrababu, "Software Reliability Growth Model with Testing-Effort by Failure Free Software", International Journal of Engineering and Innovative Technology, 2012, Vol 2, 6, pp 103-107.
6. S. Inoue and S. Yamada,"Lognormal Process Software Reliability Modeling with Testing Effort", Journal of Software Engineering and Applications, 2013, Vol 6, pp 8-14.
7. P.K. Kapur, A.Tandon and G.Kaur, "Multi Up Gradation Software Reliability Model", 2nd International Conference on Reliability, Safety and Hazard, 2010, IEEE, Mumbai. Pp 468-474.
8. O. Singh, P.K. Kapur and Jyotish N.P. Singh, "Testing Effort Based Multi Up Gradation Software Reliability Gowth Model", Communications in Dependability and Management, 2012, Vol 15, pp 88-100.
9. F. Li and Z.L.Yi,"A New Software Reliability Growth Model: Multigeneration Faults and a Power-Law Testing Effort Function", Mathematical Problems in Engineering, Volume 2016, pp 1-13.
10. Balakrishna, K. Harishchandra, "On a Software Reliability Growth Model with Log Logistic failure time distribution", International Journal of Engineering And Computer Science, vol. 5, Issue. 12 , pp. 19552-19560, 2016.