IJCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper **Volume-4, Issue-7** **E-ISSN: 2347-2693**

Estimation of Software Reliability Using p – TEF Models

Suneet Saxena[1*], Ajay Gupta[2]

[1*]Department of Mathematics, SRMS College of Engineering & Technology, Bareilly, India
[2] Department of Mathematics, BIT, Muzaffarnagar, India.

Available online at: www.ijcseonline.org

Received:12/Jun/2016 Revised: 26/Jun/2016 Accepted: 22/Jul/2016 Published: 31/Jul/2016

***Abstract*— The paper present NHPP Software Reliability Growth model involving test effort function(TEF) and fault removal efficiency(TEF). TEF deals with the problem of limited time and resources available during software testing phase. FRE addresses the problem of multiple occurrences of fault before its final removal. In this paper we propose p-TEF models which incorporate both TEF and FRE. p is FRE which represent fraction of faults detected and corrected. If p is less than one, then debugging is imperfect whereas for p equals to one debugging is perfect. Existing TEF models compared with p-TEF models using statistical tools SSE , R2 and AIC. Results suggest that the p-TEF models fits and predict faults detection data better.**

***Keywords*— Fault removal efficiency, Test Effort Function, NHPP Models, Akaike's Information Criterion**

I. INTRODUCTION

Software testing is an important phase in Software Development Cycle. During testing phase an objective is to achieve high reliability of software subject to time and resource constraints. Time constraint is the limited time available for testing as software has to be released in market. Resource constraint (human power, cpu hours, etc..) is the limited resources available for testing. It is not possible to consume resources forever due to the expenditure incurred on their consumption. Test Effort Function (TEF) represents time and resource constraints. 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 1986 Ohtera, Narihisa and Yamada et al.[1] proposed SRGM's involving testing effort. Hishitani, Osaki and Yamada et al [2] developed model incorporating testing effort function given by Weibull distribution in year 1993. The blending of imperfect debugging with TEF done by Kapur and Younes[3] in 1994. In 1997 Shepperd and Schofield[4] estimated software project effort using analogies. Same year Logistic TEF was used by Huang, Kuo and Chen[5] and analysed reliability of software. In 1999 Huang, Kuo and Chen[6] incorporated TEF and efficiency while developing SRGM and estimated the cost. In 2002 Huang and Kuo [7] investigated a SRGM based on logistic TEF and predicted optimal software release policy involving cost-reliability criteria.

Change point concept with TEF was used by Huang [8] in 2005. Same year Kumar, Ahmad and Quadri [9] did data analysis with a Pareto TEF. In 2008 Ahmad, Bokhari, Quadri and khan [10] proposed SRGM based on exponential weibull distribution. They incorporated various TEF and estimated optimal release policy. Burr Type X TEF was used by Ahmad, Khan, Quadri and Kumar [11] in 2009 and determine release time. In 2011, log-logistic TEF with imperfect debugging used by Ahmad, Khan and Rafi [12]. They analyzed an inflection S- shaped SRGM. In 2012 Aggarwal, Kapur, Kaur and Kumar[13] 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 [14] in year 2012 incorporated generalized exponential TEF while developing SRGM.

Fault removal efficiency [FRE] is another factor which affects the reliability of software. Software would be more reliable if maximum number of faults detected and corrected at testing phase. Fault removal process is very complex process and time consuming. It involves detection and correction of faults. Fault removal efficiency is defined as the probability of perfectly removing a fault in first repair attempt [15]. Previously various authors developed SRGMs assuming value of Fault removal efficiency is unity, means fault detected and completely corrected and no occurrence of that fault in future. However, Experience shows that this is not the case. Software fault occur multiple times before it is being finally removed. Various authors [16-17] developed SRGM involving FRE.

The paper focuses on development of NHPP SRGM in which number of faults detected at any time t is function of test effort function and fault removal efficiency. After introduction there are four section in this paper. Section 2 present descriptions of various test effort functions. Section 3 discussed development of model and solution. New TEF involving fault removal efficiency proposed in this section. Section 4 provide estimation of the parameter and comparison using statistical tools. Finally conclusions have been highlighted in section 5.

II. TEST EFFORT FUNCTIONS

Notations

E Total testing effort consumed.
β Scale parameter in Exponential, Rayleigh, Weibull and log logistic distributions.
γ Shape parameter in Weibull, Generalized exponential and log logistic distribution.
α Consumption rate of testing effort expenditures in the Logistic TEF
λ Constant parameter in the logistic TEF.

Some test effort function are:

A. Exponential TEF

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

B. Rayleigh TEF

This TEF exhibits both increasing and decreasing phenomenon. PDF is represented by $w(t) = 2E\beta t exp(-\beta t^2)$. *CDF is given by* $W(t) = E[1 - exp(-\beta t^2)]$.

C. Weibull TEF

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

D. Logistic TEF

It is a smooth bell shaped function and represents TEF fairly accurate.PDF is given by $(t) = \frac{E\,\lambda\,\alpha\,\exp(-\alpha t)}{[1+\lambda\exp(-\alpha t)]^2}$. CDF is given by $(t) \;=\; \frac{E}{1+\lambda\exp(-\alpha t)}$.

E. Log Logistic TEF

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

$$w(t) = \frac{E\left(\frac{t}{\lambda}\right)^{-\beta}\beta}{\left[1+\left(\frac{t}{\lambda}\right)^{-\beta}\right]^2 t} \qquad .$$

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

F. Generalized Exponential TEF

It is the generalized case of Exponential TEF exhibiting both increasing and decreasing phenomenon and reaches peak for shape parameter γ >0. Its PDF is $w(t) = E\,\beta\,\gamma\exp(-\beta t)\,(1-\exp(-\beta t))^{\gamma-1}$.
CDF is $W(t) = \int_0^t w(t)dt = E(1-\exp(-\beta t))^{\gamma}$

III. MODEL DEVELOPMENT

The rate of fault detected at time t is proportional to fault detection rate(b) and remaining fault $(\mathrm{a}-\mathrm{pm(t)})$ in software at time t. a is total number of faults. p is fault removal efficiency (FRE). $\mathrm{m(t)}$ is number of faults detected by time t. Incorporating TEF $\mathrm{w(t)}$, the failure rate differential equation is modified to

$$\frac{\mathrm{dm}}{\mathrm{dt}} = \mathrm{b}\big(\mathrm{a} - \mathrm{pm(t)}\big)\mathrm{w(t)}$$

Using marginal conditions $\mathrm{m(0)} = 0$, the solution is:

$$\mathrm{m(t)} = \frac{\mathrm{a}}{\mathrm{p}}[1 - \mathrm{pexp}(-\mathrm{pbW(t)})]$$

Where $\mathrm{W(t)} = \int_0^{\mathrm{t}}\mathrm{w(t)dt}$

For Various TEF we propose p-TEF models.

A. p-TEF(Exponential)

$$m(t) = \frac{a}{p}[1 - pexp(-pbE(1 - \exp(-\beta t)))$$

B. p- TEF(Rayleigh)

$$m(t) = \frac{a}{p}[1 - pexp(-pb\mathrm{E}(1 - \exp(-\mathrm{\beta t^2})))]$$

C. p- TEF(Weibull)

$$m(t) = \frac{a}{p}[1 - pexp(-pbE(1 - \exp(-\beta t^{\gamma})))]$$

D. p-TEF (Logistic)

$$m(t) = \frac{a}{p}[1 - pexp(-pb\;\frac{E}{1+\lambda\exp(-\alpha t)})]$$

E. p-TEF (Log Logistic)

$$m(t) = \frac{a}{p}[1 - pexp(-pb\,\frac{E}{1+\left(\frac{t}{\lambda}\right)^{-\beta}})]$$

F. p-TEF (generalized)

$$m(t) = \frac{a}{p}[1 - pexp(-pbE(1 - \exp(-\beta t))^{\gamma})]$$

IV. PARAMETER ESTIMATION AND COMPARISIONS

Existing TEF models compared with p-TEF models using following statistical tools Existing TEF models compared with p-TEF models using following statistical tools

A. Parameter Estimation

Parameters of models are estimated by Maximum likelihood method using Musa's [18] SS1a failure dataset . The likelihood function for unknown parameters is given by

$$L = \prod_{i=0}^{n} \frac{[m(t_i) - m(t_{i-1})]^{(y_i - y_{i-1})}}{(y_i - y_{i-1})!} \exp[-(m(t_i) - m(t_{i-1}))]$$

There are n observed data pairs (t_i, y_i) where y_i is observed cumulative faults at time t_i. The parameters are estimated by maximizing likelihood function L.

B. Comparisons

The models are compared using various tools of Goodness of Fit (GoF).Some of the measures used to determine GoF are given below:

(i) Sum of Square Error (SSE).

(ii) Akaike's Information Criterion(AIC).

(iii) Coefficient of Determination (R^2).

Sum of Square Error (SSE)

SSE is the sum of squares of residuals between observed value and estimated value. It can be expressed as

$$SSE = \sum_{j=1}^{n} \left(y_j - H_J\right)^2$$

where y_j is observed cumulative faults at time j and H_J estimated cumulative faults at time j. Model with lower SSE fits better to given dataset.

Akaike's Information Criterion(AIC)

AIC is used to compare the models. It can be evaluated as AIC=-2*log(likelihood function at its maximum value)+2*N. Where N is the number of parameters. The model with lower AIC value is chosen as the best model to fit the data. In AIC, the compromise takes place between the maximized log likelihood and the number of free parameters estimated within the model (the penalty component) which is a measure of complexity or the compensation for the bias in the lack of fit when the maximum likelihood estimators are used.

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. $R^2 = 1$ is perfect fitting, $R^2 = 0$ no fitting and R^2 close to 1is good fitting.

R^2 is defined as : $$R^2 = \frac{\sum_{j=1}^{n} (H_J - \bar{y})^2}{\sum_{j=1}^{n} (y_j - \bar{y})^2}$$

$$\bar{y} = \frac{1}{n} \sum_{j=1}^{n} y_j$$

where y_j is observed cumulative faults at time j and H_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.

Some Proposed models compared with respective previous models. Goodness of fit table is given below.

Table 1. Goodness of Fit for data set SS1a

Models Compared	**SSE**	**R^2**	**AIC**
Log Logistic	4456.28	1.45	232.03
p-TEF (Log Logistic)	3243.56	0.92	205.45
Generalized	3786.23	1.22	223.75
p-TEF(Generalized)	2864.57	0.89	154.22

V. CONCLUSION

The paper investigates various existing TEF model. Involving FRE, Models have been modified and tested for existing dataset of software faults. TEF addresses the time and resource constraint aspect in software reliability testing whereas FRE deals with the problem of occurrence of faults more than one time before its final removal. Values of Statistical tools SSE and AIC are lower for **p-TEF** models. Values of R^2 close to 1 for **p-TEF** models. Comparison tools suggest that **p-TEF** models fits better than existing TEF models for given dataset.

REFERENCES

[1] S. Yamada, H. Ohtera, and H. Narihisa, "Software Reliability Growth Models with Testing-effort", IEEE Trans. Reliability,1986, R-35,1, pp 19–23.

[2] S. Yamada, J. Hishitani, and S. Osaki, "Software Reliability Growth Model with Weibull Testing Effort: A Model And Application", IEEE Trans. Reliability,1993, 42, pp 100–105.

[3] P.K. Kapur, and S. Younes, "Modeling an Imperfect

Debugging Phenomenon with Testing Effort", In: Proceedings of 5th International Symposium on Software Reliability Engineering, 1994, pp 178-183.

[4] M. Shepperd, and C. Schofield, "Estimating Software Project Effort using Analogies", IEEE Trans. Software Engineering,1997, 23, pp 736–743.

[5] C.Y. Huang, S.Y. Kuo, and I.Y. Chen, "Analysis of a Software Reliability Growth Model with Logistic Testing-Effort Function", In Proceedings of 8th International Symposium on Software Reliability Engineering (ISSRE'1997),1997, pp 378–388.

[6] C.Y. Huang, J.H. Lo, S.Y. Kuo, and M.R. Lyu, "Software Reliability Modeling and Cost Estimation Incorporating Testing-Effort And Efficiency", In Proceedings of 8th International Symposium on Software Reliability Engineering (ISSRE'1999), 1999, pp 62–72.

[7] 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.

[8] C.Y. Huang, "Performance Analysis Of Software Reliability Growth Models With Testing-Effort And Change Point", Journal of Systems and Software, 2005 76, pp 181- 194.

[9] M. Kumar, N. Ahmad, and S.M.K. Quadri, S.M.K. "Software Reliability Growth Models and Data Analysis with a Pareto Test-Effort", RAU Journal of Research, 2005, 15 (1-2), pp 124-128.

[10] 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.

[11] N. Ahmad, M.G.M Khan, S.M.K Quadri, and M. Kumar, M. "Modeling And Analysis of Software Reliability With Burr Type X Testing-Effort and Release-Time Determination", Journal of Modeling in Management, 2009, 4(1), pp 28-54.

[12] 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.

[13] 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.

[14] 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.

[15] D.R. Jeske, X.M. Zhang and L. Pham "Accounting for Realities when Estimating the Field Failure Rate of Software", In Proceedings of 8th International Symposium on Software Reliability Engineering (ISSRE'01), 2001 pp 332-339.

[16] X.M. Zhang, X.L. Teng and H. Pham, "Considering Fault Removal Efficiency in Software Reliability Assessment", IEEE Trans. Systems, Man and Cybernetics-Part A: System and Humans, 2003, 33, 1, pp 114-120.

[17] B. Purnaiah, K.V. Rama and V.K.G. Bala, "Fault Removal Efficiency in Software Reliability Growth Model", Advances in Computational Research, 2012 4, 1, pp 74-77.

[18] Musa. "DACS software reliability dataset", Data & Analysis Center for Software, J.D. 980. www.dacs.dtic.mil/databases/sled/swrel.shtml.

Authors Profile

Dr. Suneet Saxena pursed Bachelor of Science from University of Allahabad, Allahabad in 1997 and Master of Science from M.J.P Rohilkhand University in year 2000. He did Ph.D. from M.J.P Rohilkhand University, Bareilly in year 2010 and currently working as Assistant Professor in Department of Mathematics, SRMS College of Engineering & Technology, Bareilly, India.. He has published various research papers in reputed national and international journals.. His main research work focuses Software reliability. He has 14 years of teaching experience and 10 years of Research Experience.

Dr. Ajay Gupta did Bachelor of Science and Master of Science in Mathematics. He has been awarded Ph.D. from M.J.P Rohilkhand University, Bareilly in year 2006 and currently working as Associate Professor in Department of Mathematics, BIT, Muzaffarnagar, Bareilly, India.. He has published various research papers in reputed national and international journals.. His main research work focuses Software reliability. He has 16 years of teaching experience and 10 years of Research Experience.