A Review on Software Realibility Growth Modelling

Suneeta Chaudhary
Deptt. Of Applied Sci.
M.I.E.T
Greater Noida, India

Diya Singh
Deptt. Of Mech. Engg.
M.I.E.T
Greater Noida, India

Ankit Sinha
Deptt. Of Mech. Engg.
M.I.E.T
Greater Noida, India

Sarvesh Mishra
Deptt. Of Applied Sci.
M.I.E.T
Greater Noida, India

Abstract*— Software reliability is the possibility of the failure free operation of software in a given period of time under some certain conditions. Due to demand of new features and highly reliable software system, the software industries are speeding their up-gradations/add ons in the software. The life of software is very short in the environment of perfect competition. Therefore the software developers have to come up with successive up gradations to survive. The reported bugs from the existing software and Features added to the software at frequent time intervals lead to complexity in the software system and add to the number of faults in the software. The developer of the software can lose on market share if it neglects the reported bugs and up gradation in the software and on the other hand a software company can lose its name and goodwill in the market if the reported bugs and functionalities added to the software lead to an increase in the number of faults in the software. To capture the effect of faults due to existing software and generated in the software due to add-ons at various points in time, we develop a multi up-gradation, multi release software reliability model. Due to complexity and incomplete understanding of the software, the testing team may not be able to remove/correct the fault perfectly on observation/detection of a failure and the original fault may remain resulting in the phenomenon known as imperfect debugging, or get replaced by another fault causing error generation. The testing team is unable to eradicate the errors thoroughly on the observation of a failure because of intricacy and partial knowledge of software system, consequently, original errors persist or get exchanged by distinct error thus the occurrence is termed as imperfect debugging and error generation, respectively.

Keywords*— *Up-gradation–NHPP–Laplace trend–Imperfect debugging , SRGM, Simple, Hard & Complex faults

I. INTRODUCTION

Computers become the imperative part of every company in the 21st century which amplifies the dependency on computer software consequently repeatedly facing the failures due to increment in size and complexity of software has grown up. During development of software, software reliability should be considered so as, to abstain the failure and faults which results in reliable software. There are several projects executed by NASA, and DRDO that deal with highly sophisticated software [1]. Due to change in the three lines of code in a single program me in 1991 the telephone system was collapsed in California and eastern parts. Software reliability is pivotal characteristic of software quality. Reliability is the property of referring how well software meet its requirement and also the probability of failure free operation for the specified period of time in a specified environment .Among all software reliability model SRGM is probably one of the most efficient mechanism in the literature with more than hundred models functioning in assorted forms. In application, SRGMS confront distinguished challenges. Firstly, software testers, scarcely pursue the operation profile to test the software, so, what is perceived during software testing is indirectly protractible to operation use. Secondly, when the chunk of failures accumulated in a project is circumscribed, it is tough to get numerically relevant reliable predictions. And lastly, the most of assumptions, of SRGMS are indiscreet e.g. assumptions that faults are independent of each other, each fault has the same chance to be spotted in one chance, and correction of a fault conceal new faults. The testing team may not be able to remove the fault perfectly on the detection of failure and the original fault may remain or replace by another fault. Because of incomplete understanding of the internal structure of the software while the first phenomenon is termed as imperfect fault removal the second is called error generation. In case of imperfect fault removal, the fault content increases as the testing progress and removal results in introduction of new faults while removing old ones. It was Goel [2] who first introduces the concept of imperfect debugging. He introduced the probability of imperfect debugging in Jenski and Moranda model, due to Chou and Ohbha [4] an error generation model is applied and is removed as complex fault in new release, also [6] introduced the imperfect fault thus the number of failures observed by time infinity is more than the initial fault content .Although these model describe two types of imperfect debugging phenomenon yet the software reliability growth curve of these models is always exponential. Moreover, they assume that probability of imperfect debugging is independent of the testing team.

Moreover, several continuous state space srgm based on stochastic differential equation of It^o type to measure the reliability growth of software have been developed corresponding to discrete state space NHPP based SRGM. Yamada et. all [13] proposed a simplified software reliability growth model to describe the fault detection during the testing phase by applying It^o type stochastic differential (SDE) and have compared the continuous state space srgm with the NHPP. Yamada et. all [14] has developed several SRGM based on stochastic differential equations of It^o type such as exponential delayed s-shaped and inflection s-shaped. lee et al.[7] used SDE to represents a per- fault detection rate that incorporates an irregular fluctuation instead of an NHPP , and consider a per- fault detection rate that depends on the testing time t . Yamada et al [15] have proposed a flexible stochastic differential equation model describing a fault detection process during the system testing team phase of the distributed development environment. Recently, Kapur et al [13] proposed, a composite model called generalized SRGM based on stochastic differential equation of It^o type which incorporates three different types of faults e.g. Simple, hard and complex.

II. LITERATURE REVIEW

A. *Notations*

$m_f(t)$: The expected number of faults resulted in failure by time t.
$m_r(t)$: The expected number of faults removed by time t.
a(t): Time dependent total fault content in the software.
a: Initial number of faults.
γ: Error generation rate.
f(t): Time dependent fault removal rate.
p: Perfect debugging probability.
α,β: A constant of learning function.

B. *INTIAL SRGM'S*

1) *GOEL-OKUMOTO NHPP MODEL*

In this model Goel-Okumoto [2] assumed that a software system is subject to failure at random times caused by faults present in the system. The Non Homogeneous Poisson Process (NHPP) model is a Poisson type model that takes the number of faults per unit of time as independent Poisson random variables.

The basic assumptions of this model are:

1. Cumulative number of failures by time t follows a Poisson process.
2. Number of faults detected in each time interval is independent for any finite collection of time intervals.
3. Defects are repaired immediately when they are discovered.
4. Defect repair is perfect. That is, no new defect is introduced during test.
5. No new code is added to software during test.
6. Each unit of execution time during test is equally likely to find a defect.

The mean value function or the cumulative failure counts must be of the form

$\mu(t)=a(1- e^{-bt})$ for some constants b>0 and N>0. a is the expected total number of faults to be eventually detected. In this model a is the expected number of failures to be observed eventually and b is the fault detection rate per fault.

2) *YAMADA-S-SHAPED MODEL*

A modification of this model which takes care of different types of errors in the software depending on their severity is the modified exponential model developed by Yamada and Osaki [9]. This SRGM assumes two types of errors in the software; Type I errors are easy to detect, with b l as the error detection rate per Type I error, and Type II errors are difficult to detect, with b2 as the detection rate per Type II error. The mean value function is m(t) = ~p~a(1 - exp(-b,t)) (2) i where p~ is the error content proportion of type i error, 0<p~<-l, ~p~=l, 0<b2<bl where k is the number of clusters with similar characteristics, at is the number of errors to be eventually detected in cluster i and b~ is the error detection rate for cluster i.

3) *OHBA and CHOU MODEL (Exponential with Error Generation)*

Ohba and Chou (1989) proposed a SRGM with the assumption of error generation. The model is based on removal of detected faults. But, there is a possibility that new faults may be introduced. In this model, the rate of change of mr(t) with respect to time can be written as:

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

Where, $f(t) = f \text{ and } a(t) = a + \gamma m_r(t)$ (2)

On solving the above equations with mr(0)=0, we get

$$m_r(t) = \frac{a}{(1-\gamma)}(1 - e^{-f(1-\gamma)t}) \quad (3)$$

In this model, the probability of error generation is considered. In proposed models, we have considered imperfect fault removal and error generation with learning function.

C. *SRGM WITH IMPERFECT DEBUGGING*

1) *YAMADA IMPERFECT DEBUGGING MODEL-1*

This model, proposed by Yamada, is one of the most popular NHPP-based imperfect debugging models. This model assumes exponential a(t) function and constant b(t) function as follows:

$a(t) = ae^{\alpha t}$, (4)
$b(t) = b.$ (5)

The mean value function of this model is as follows:

$m(t) = ab\ \alpha + b\ (e^{\alpha t} - e^{-bt})$ (6)

2) *YAMADA IMPERFECT DEBUGGING MODEL-2*

This model is also proposed by Yamada[10]. The a(t) and b(t) functions of this model are given as follows:

$a(t) = a(1 + \alpha t),$ (7)

$b(t) = b.$ (8)

This model assumes constant fault introduction rate α. The b(t) function is also constant for this model. The mean value function of this model is as follows:

$$m(t) = a(1 - e^{-bt})\ 1 - \alpha b\ + a\alpha t. \quad (9)$$

3) PHAM-NORDMANN-ZHANG MODEL

In Pham–Nordmann–Zhang (PNZ) [7] model, the fault introduction rate is linear and b(t) function is S-shaped. The a(t) and b(t) functions of this model are as follows:

$$a(t) = a(1 + \alpha t), \quad (10)$$

$$b(t) = b\,(1 + \beta e^{-bt}). \quad (11)$$

The mean value function of this model is given as follows:

$$m(t) = a\,((1 + \beta e^{-bt}\,(1 - e^{-bt})\,)\,(1 - \alpha b)\,) + \alpha t. \quad (12)$$

D. TWO DIMENSION SRGM CONSIDERING TESTING TIME & TESTING EFFORT

1. TWO-DIMENSIONAL SRGM BY P.K KAPUR

The two dimensional flexible SRGM presented in this section was proposed by Kapur et al. [13]. It is based on the following assumptions. 1) The fault removal phenomenon is modeled by a NHPP. 2) The software is subject to failures during execution caused by faults remaining in the software. 3) The failure rate is equally affected by all the faults remaining in the software. 4) The fault detection rate is a non decreasing time and resource-dependent function. 5) Upon a failure, the fault causing that failure is immediately removed, and no new faults are introduced. 6) To cater the combined effect of testing time and resources, we use the Cobb-Douglas production function of the form (1) Under the above assumptions, the differential equation representing the rate of change of the cumulative number of faults detected w.r.t. to the combined effect of time and resources is given as

$$m'(\tau) = \frac{b}{1+\beta \exp(-b\tau)}(a - m(\tau)) \quad (13)$$

Integrating the above equation with initial condition , and using (1), we get

$$m(s,u) = \frac{a(1-\exp(-bs^{\alpha}u^{1-\alpha}))}{1+\beta\exp(-bs^{\alpha}u^{1-\alpha})} \quad (14)$$

The above model can also be written in the form

$$m(s,u) = a.F(s,u) \quad (15)$$

Where,

$$F(s,u) = \frac{(1-\exp(-bs^{\alpha}u^{1-\alpha}))}{1+\beta\exp(-bs^{\alpha}u^{1-\alpha})} \quad (16)$$

2. TWO –DIMENSIONAL S-SHAPED MODEL

In this paper we develop a two dimension S-shaped model determining the combined effect of testing time and testing coverage. We define some additional notations as follows:

$m_r(s,u)$: Mean Number of faults detected corresponding to Coverage u and time s

$m_f(s,u)$: Mean Number of failures corresponding to Coverage u and time s

The differential equation of the representing the rate of change of cumulative number of faults detected w.r.t. to the total testing resources is given as:

$$m_r(\tau) = a(N - m_r(\tau)) + \frac{b(N - m_r(\tau))}{N} \quad (17)$$

The mean value function of the number of faults detected with testing resources x using the initial condition x (0) = 0 is given as:

$$m_r(\tau) = \frac{Na(\exp((a+b)\tau)-1)}{b + a\exp((a+b)\tau)} \quad (18)$$

Now we extend the testing resource of one dimensional S shaped model to a two dimensional problem. Using the Cobbdouglas production the corresponding mean value function is given as:

$$m_r(\tau) = \frac{Na(\exp((a+b)s^{\alpha}u^{1-\alpha})-1)}{b + a\exp((a+b)s^{\alpha}u^{1-\alpha})} \quad (19)$$

In the above two-dimensional mean value function if $\alpha = 1$ the above mean value function can be regarded as a traditional one dimensional time dependent SRGM & if $\alpha = 0$ it becomes a testing coverage dependent SRGM.

To illustrate the estimation procedure and application of the SRGM (existing as well as proposed) we have carried out the data analysis of real software data set. The parameters of the models have been estimated using statistical package SPSS.

III. MODEL VALIDATION, DATA SET AND DATA ANALYSIS

To check the validity of the proposed model and to describe the software reliability growth, it has been tested on tandem computer four release data set. Also we have used non linear least square technique in SPSS software for estimation of parameters.

1) Criteria for comparisons

To give quantitative comparisons, some criteria were used to judge the performance of the proposed model. Here we let n represent the sample size of selected data set, y_i represent the actual number of faults by time t_i and m (t_i) represent the estimated number of faults by time t_i in all mentioned criteria the lower value indicate less fitting error.

i. The Bias is defined as:

$$Bias = \sum_{i=1}^{n}\left(\frac{(\hat{m}(t_i) - y_i)}{n}\right)$$

The difference between the observation and prediction of number of failures at any instant of time i is known as PE_i. (Prediction error). The average of PEs is known as bias. Lower the value of Bias better is the goodness of fit.

ii. VARIATION

$$Variation = \sqrt{\frac{\sum_{i=1}^{n}((\hat{m}(t_i) - y_i) - Bias)^2}{n-1}}$$

The average of the prediction errors is called the prediction Bias, and its standard deviation is often used as a measure of the variation in the predictions.

iii. *The Root Mean Square Prediction Error (RMSPE) is defined as:*

$$RMPS = \sqrt{((Bias)^2 + (Variation)^2)}$$

RMSPE is a measure of the closeness with which the model predicts the observation.

iv. *MEAN SQUARE ERROR*

The difference between the expected values, $\hat{m}(t_i)$ and the observed data y_i is measured by:

$$MSE = \sum_{i=1}^{k} \frac{(\hat{m}(t_i) - y_i)}{k}$$

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

IV. CONCLUSION

There are many SRGM'S available in literature in spite of that currently industries are focusing on two dimensions SRGM'S as its assumptions are more close to real situations. However, research papers on two dimension SRGM'S with multi-up gradation with inculcation of stochastic differential equations in order to inculcate the uncertainties involved looks more promising.

REFRENCES

[1] Brooks WD, Motley RW," Analysis of discrete software reliability models- Technical report (RADC-TR-80-84" 1980; New York: Rome Air development center.
[2] Goel AL, Okumoto K. (1979); "Time dependent error detection rate model for software reliability and other performance measures" IEEE Transactions on Reliability R-28(3): 206-211.
[3] Inoue S, Yamada S "Testing-Coverage Dependent Software Reliability Growth Modeling" International Journal of Reliability, Quality and Safety Engineering, Vol. 11, No. 4, 303-312, 2004.
[4] Ohba, M. (1984), Software reliability analysis models, IBM Journal of research and Development 28, 428-443.
[5] Kapur PK and R.B. Garg (1992), A software reliability growth model for an error removal phenomenon, Software Engineering Journal 7, 291-294.
[6] Musa, J.D., A. Iannino and K. Okumoto (1987), Software reliability: Measurement, prediction, Applications, Mc Graw Hill, New York.
[7] Pham H, Zhang X. "NHPP Software reliability and Cost models with Testing Coverage" European Journal of Operational Research; 145: 443-454, 2003.
[8] Tomotaka Ishii and Tadashi Dohi "Two-Dimensional Software Reliability Models and Their Application" 12th Pacific Rim International Symposium on Dependable Computing (PRDC'06).
[9] Yamada S., Osaki S., "Optimal software release policies with simultaneous cost and reliability requirements", European Journal of Operational Research 31(1987) 46-51.
[10] Yamada S, Ohba M, and Osaki S. (1984) S-shaped software reliability growth models and their applications, IEEE Transactions on Reliability, 1984; R-33: 169-175.
[11] Yashwant K. Malaiya, Michael Naixin, James M. Bieman and Rick Karcich; Software Reliability Growth with Testing Coverage, IEEE Transactions on Reliability,Vol. 51(4), pp.420-426, 2002.
[12] Goel, A. L., & Okumoto, K. (1979). Time-dependent error-detection rate model for software reliability and other performance measures. IEEE Transactions on Reliability, 28(3), 206-211.
[13] Kapur, P. K., Anand. S., Yadavalli, V. S. S., & Beichelt. F. (2006b). A generalized software growth model using SDE. In International Conference on Quality, Reliability and Infocom Technology, ICQRIT, (pp. 451-461).
[14] Kapur, P. K., Basirzadeh, M., Inoue, S., & Yamada, S. (2010b). Stochastic differential equation based SRGM for errors of different severity with testing-effort. International Journal of Reliability, Quality and Safety Engineering, 17(03), 179-197.
[15] Kapur, P. K., Gupta, A., & Jha, P. C. (2006a). On modeling failure phenomenon of multiple release of a software in operational phase for product and project type software: a theoretical approach. In Proceedings of International Conference on Quality, Reliability and Infocom Technology, 55-164.
[16] Kapur, P. K., Tandon, A., & Kaur, G. (2010a, December). Multi up-gradation software reliability model. In Reliability, Safety and Hazard (ICRESH), 2010 2nd International Conference on (pp. 468-474). IEEE.
[17] Kapur, P. K., Younes, S., & Agarwala, S. (1995). Generalised Erlang model with n types of faults. ASOR Bulletin, 14(1), 5-11. Pham, H. (2006). System software reliability. Springer Science & Business Media. Pham, H., & Zhang, X. (2003). NHPP software reliability and cost models with testing coverage. European Journal of Operational Research, 145(2), 443-454.
[18] Singh O., Kapur P. K., & Anand, A. (2011). A stochastic formulation of successive software releases with faults severity, In International Conference on Industrial Engineering and Engineering Management (IEEM), (pp. 136-140),