IPASJ International Journal of Computer Science (IIJCS)
Web Site: http://www.ipasj.org/IIJCS/IIJCS.htm
A Publisher for Research Motivation ....
Email:editoriijcs@ipasj.org
Volume 6, Issue 6, June 2018
ISSN 2321-5992

INCORPORATING TESTING EFFORT INTO SOFTWARE RELIABILITY GROWTH MODEL WITH TIME VARYING LEARNING FACTORS

Dr. Shaheda Akthar[1] Shaik.Mohammad Rafi[2]

[1]Registrar I/c , Dr. Abdul Haq Urdu University, Kurnool, Andhra Pradesh

[1]Research Scholar, Department of Computer Science & Engineering, Acharya Nagarjuna University, Guntur.

ABSTRACT

Software testing is an important phase in software development. Many resources are consumed during the software testing. The goal of software testing is to identify the errors and then rectify the those errors. Error correction process can increase the quality of the software. During software testing testers uses resources like test case and testing time to detect the errors are termed as software effort. Software testing is continuous learning process where a testers are fully learn about error before removal. the learning and experience are two terms which influences the testing more. So in this process we integrated the software reliability growth model with testing effort and learning capacity of testers. parameters are estimated with standard procedure and result are compared with previous results and our proposed model fits good the real time datasets.

Keywords: Software Reliability, Testing Effort, Software Testing, Testers learning, Non-homogeneous Poisson Process (NHPP), Software Cost.

I. INTRODUCTION

Software reliability is defined as probability of working of the software before it fails. Testing is one important phase where errors are detected and corrected. Software Reliability growth models helping the industries to produce optimal quality software with minimum cost. Software Reliability growth models are mimics models of mathematics and statistics. Different authors were proposed different models in during course of time. Main motto behind these models to simulate the testing environment and process. Cost and Reliability are two different constraints for every software industry. Every industry wants to produce product in more reliable and cost effective manner. Two types of datasets which are mainly used to evaluate the proposed models, one failure count data and failure time data. Several models are integrated their models with these datasets. Software Reliability growth models also categorized based on perfect dubbing and imperfect debugging models. In perfect debugging models assumes total number of errors remain constant during testing and no new errors are introduced. Whereas imperfect debugging assumes that testing can introduce new errors into the product. Some authors proposed release time for software product. Once the testing is Complete product is ready to deliver to outside, these reliability and cost models can helps industries to archive their target. During the testing and debugging several resources are being consumed, like test cases, time of testing and testing effort. Testing effort is defined as it is the effort required to detect and correct the error during testing phase. Testing effort cannot be measured directly, but indirect way of measure is possible. [6]Several authors integrated testing efforts into their models. During testing testers' knowledge about errors and testing environment play an important role. As initially when the testing begins testers knowledge is limited about new environment, as it continuous their skills and knowledge will improve, which effect on testing. Chiu 2008[5] developed software reliability growth model integrating with testers experience and learning capacity. Chui 2012[2] and 2013[7] proposed time varying new learning factors into their models. Iqbal 2013 Proposed a model based on testers incapacity to detect faults. Iqbal 2017[4] Proposed a model based on integrating learning factors into imperfect debugging models. In this paper we proposed a new model of integrating testing effort into software reliability growth models with time varying learning factors. Proposed model is evaluated based on real time datasets, results are compared with previously proposed models.

II. SOFTWARE RELIABILITY GROWTH MODELS WITH LEARNING FACTORS

Assisted Work

A) Chiu, Huang and Lee learning model [5]

IPASJ International Journal of Computer Science (IIJCS)
Web Site: http://www.ipasj.org/IIJCS/IIJCS.htm
A Publisher for Research Motivation ....
Email:editoriijcs@ipasj.org
Volume 6, Issue 6, June 2018
ISSN 2321-5992

In this model authors proposed a imperfect debugging environment based software reliability model based on casual loop diagram. They incorporated learning and experience of software testers in their models. They feel that learning and experience of software during software testing can effect on software testing during defect identification in constant environment. They feel that learning and experience factors are constant.

$$f(t) = (\alpha + \eta * F(t)) * (1 - F(t)) \tag{1}$$

above equation solved by assuming F(0)=0 then

$$F(t) = \frac{e^{(\alpha+\eta)*t} - 1}{\frac{\eta}{\alpha} + e^{(\alpha+\eta)*t}} \tag{2}$$

B) Kuei-Chen Chiu [2] and chiu , Kuei Chen 2013[7]

In this paper author proposed new model based on time varying learning phenomenon by introducing new learning factors. where they introduced two new time varying learning factors into their model. $\eta(t) = (1 + \xi * t)$ and $\eta(t) = e^{\xi * t}$. where ξ represents coefficient of accelerating factor.

$$F(t) = \frac{e^{1+\frac{\eta+\xi*t}{\alpha'}} - 1}{\frac{\eta}{\alpha} + e^{1+\frac{\eta+\xi*t}{\alpha'}}} \tag{3}$$

$$F(t) = \frac{e^{1+\frac{\eta+e^{\xi*t}}{\alpha'}} - 1}{\frac{\eta}{\alpha} + e^{1+\frac{\eta+e^{\xi*t}}{\alpha'}}} \tag{4}$$

Here learning and experience factors varies with time.

C) Javid Iqbal , N.Ahmad and S.M.K Quadri [1][3]

In this paper authors assumes that software testers ate little negligent during testing process where it has adverse effect on software testing . they incorporated an negligent factor into their model.

$$\alpha'(t) = \eta_1 * \alpha - \tau \tag{5}$$

now the

$$F(t) = \frac{e^{1+\frac{\eta_2}{\alpha'(t)}} - 1}{\frac{\eta}{\alpha} + e^{1+\frac{\eta_2}{\alpha'(t)}}} \tag{6}$$

D) Proposed Model

Present model we have integrated testing effort function into dynamic learning based software reliability growth models.

$$\frac{dF(t)}{dt} = (\alpha + \eta(t) * F(t)) * (1 - F(t)) \tag{7}$$

from equation (7) α testets capacity to detect the error during testing and $\eta(t)$ represents dynamic learning factors which represents testets learning capacity. [9] Proposed a Software reliability model in which he gave an expression in which shows the relation between instantaneous testing w(t) effort and learning function $\eta(t)$.

$$w(t) = \frac{k}{(\eta(t))^p} \tag{8}$$

Equation (8) is integrated into (7) we got

$$\frac{dF(t)}{dt} = [\alpha + \frac{k}{w(t)} * F(t)] * (1 - F(t)) \tag{9}$$

$$\int_0^t (1 - F(t)) dF(t) = k * \int_0^t \frac{F(t)}{w(t)} dt + \int_0^t \alpha dt \tag{10}$$

IPASJ International Journal of Computer Science (IIJCS)
Web Site: http://www.ipasj.org/IIJCS/IIJCS.htm
A Publisher for Research Motivation ....
Email:editoriijcs@ipasj.org
Volume 6, Issue 6, June 2018
ISSN 2321-5992

from above solution we have

$$F(t) = (1 - exp(-\{(\alpha * t) + k * \left\{\frac{\int_0^t F(t)\,dt}{w(t)} + \int_0^t \frac{(\int_0^t F(t)dt) * \frac{dw(t)}{dt}}{[w(t)]^2} dt\right\}\}) \quad (11)$$

$$m(t) = \theta * F(t) = \theta * \left\{(1 - exp(-\{(\alpha * t) + k * \left\{\frac{\int_0^t F(t)\,dt}{w(t)} + \int_0^t \frac{(\int_0^t F(t)dt) * \frac{dw(t)}{dt}}{[w(t)]^2} dt\right\}\})\right\} \quad (12)$$

$$m(t) = \theta * (1 - \exp(-[\alpha * t + \phi(t)])) \quad (13)$$

$$\phi(t) = k * \left\{\frac{\int_0^t F(t)\,dt}{w(t)} + \int_0^t \frac{(\int_0^t F(t)dt) * \frac{dw(t)}{dt}}{[w(t)]^2} dt\right\} \quad (14)$$

$$F(t) = (1 - (1 + b * t) * exp(-b * t)) \quad (15)$$

$$W(t) = a * (1 - exp(-\delta * t)) \quad (16)$$

We assumed two eqautions F(t) as distribution function an S shpaed function with b as parameter and W(t) as testing effort function an exponential function with parameter δ which are substituted in to equation (12) and solved for mean value function m(t).

$$m(t) = \theta * (1 - e^{-[\alpha * t + k * \left\{\frac{t + \frac{b*t*e^{-b*t} + 2*e^{-b*t}}{b} - \frac{2}{b}}{\delta * a * e^{\delta * t}} + \frac{2*\delta^2*(b-\delta)*e^{-(b-\delta)*t} + b*\delta^2*(b*t - \delta*t + 1)*e^{(\delta-b)*t} - ((b*t-2)*\delta - b)*(b-\delta)^2*e^{\delta*t} - b^2}{a*\delta^2*(b-\delta)^2*b}\right\}]}) \quad (17)$$

III PARAMETER ESTIMATION

In this paper we used standard procedure as least squate estimation to validate our proposed. As the equation is little complex in nature we used numerical approximations [8]

$$SSE = \{(m_i - \theta * (1 - e^{-(\alpha * t_i + \phi(t_i))}))\}^2 \quad (18)$$

$$\frac{d(SSE)}{d\theta} = -2 * \sum_{i=1}^{n} \left\{(m_i - \theta * (1 - e^{-(\alpha * t_i + \phi(t_i))}) * \{(1 - e^{-(\alpha * t_i + \phi(t_i))})\}\right\} = 0 \quad (19)$$

$$\frac{d(SSE)}{d\alpha} = -2 * \sum_{i=1}^{n} \left\{(m_i - \theta * (1 - e^{-(\alpha * t_i + \phi(t_i))}) * \{((\theta * t_i)e^{-(\alpha * t_i + \phi(t_i))})\}\right\} = 0 \quad (20)$$

$$\frac{d(SSE)}{da} = -2 * \sum_{i=1}^{n} \left\{(m_i - \theta * (1 - e^{-(\alpha * t_i + \phi(t_i))}) * \left\{\theta * (e^{-(\alpha * t_i + \phi(t_i))} * \frac{d(-(\phi(t_i)))}{da})\right\}\right\} = 0 \quad (21)$$

$$\frac{d(SSE)}{db} = -2 * \sum_{i=1}^{n} \left\{(m_i - \theta * (1 - e^{-(\alpha * t_i + \phi(t_i))}) * \left\{\theta * (e^{-(\alpha * t_i + \phi(t_i))} * \frac{d(-(\phi(t_i)))}{db})\right\}\right\} = 0 \quad (22)$$

$$\frac{d(SSE)}{d\delta} = -2 * \sum_{i=1}^{n} \left\{(m_i - \theta * (1 - e^{-(\alpha * t_i + \phi(t_i))}) * \left\{\theta * (e^{-(\alpha * t_i + \phi(t_i))} * \frac{d(-(\phi(t_i)))}{d\delta})\right\}\right\} = 0 \quad (23)$$

$$\frac{d(SSE)}{dk} = -2 * \sum_{i=1}^{n} \left\{(m_i - \theta * (1 - e^{-(\alpha * t_i + \phi(t_i))}) * \left\{\theta * (e^{-(\alpha * t_i + \phi(t_i))} * \frac{d(-(\phi(t_i)))}{dk})\right\}\right\} = 0 \quad (24)$$

IV EVALUATIO**ltiple determinations (R^2)[8]** which measures the percentage of total variation about mean accounted for the fitted model and tells us how well a curve fits the data. It is frequently employed to compare model and access which model provides the best fit to the data. The best model is that which proves higher R^2. that is closer to 1.

$$R^2 = 1 - \frac{(Residual\ Sum\ of\ Squares)}{(Corrected\ Sum\ of\ Squares)} \quad (25)$$

$$R^2 = 1 - \sum_{i=1}^{n} \frac{(m_i - m(t_i))^2}{\sum_{i=1}^{n}(m_i - \sum_{j=1}^{n}\frac{m_j}{n})^2} \quad (26)$$

B) Mean Square Estimation (MSE)[8] which gives the real measure of difference between actual value and predicted value

$$MSE = \sum_{i=1}^{k} [m(t_i) - m_i]^2 / k \quad (27)$$

IV MODEL PERFORMANCE ANALYSIS

A) DATA SETS

IPASJ International Journal of Computer Science (IIJCS)
Web Site: http://www.ipasj.org/IIJCS/IIJCS.htm
A Publisher for Research Motivation ....
Email:editoriijcs@ipasj.org
Volume 6, Issue 6, June 2018
ISSN 2321-5992

Dataset 1: Scaled and Truncated dataset for 30 inter failure data sets
Dataset 2 : It is from one of four major releases of software product of tandem Computers .
Dataset 3: PL/1 database application software system consisting of approximately 137000LOC. Over a course of 19 weeks. about 328 faults are removed.

Table 1: Original Datasets

S.No	Reference	Datasets
1	Kim 2015	Scaled and Truncated dataset for 30 inter failure data sets
2	Wood (1996)	Tandem Computers Software Project
3	Ohba (1984)	PL/1 database application software

Model comparisons are done through R^2 and MSE.

B) RESULTS

Following *Table 2* indicates parameters of our proposed models . Model parameters are estimated through least square estimation with numerical approximations. *Table 3* indicates all fitted results of comparisons of different models based on R^2 ans MSE values. table 4 shows the results of various models fitted on Kim 2015 model data set. As from the given table 4 it seems proposed models better predicts the software failures.

Table 2: Parameters of proposed models

S.No	Datasets	Proposed Model (t) from eq,13
1	Kim 2015	$\alpha = 0.05983, \delta = 0.4998, b = 0.08066, k = 0.04158, a = 1052, \theta = 36.18$
2	Wood (1996)	$\alpha = 0.1062, \delta = 0.1856, b = 0.01788, k = 0.7541, a = 2.877, \theta = 99.53$
3	Ohba (1984)	$\alpha = 0.07488, \delta = 0.1706, b = 0.02106, k = 0.8053, a = 4.203, \theta = 324.7$

Table 3: Comparison between different models

Models	Source of Datasets					
	Kim 2015		**Wood(1996)**		**Ohba(1984)PL/1**	
	MSE	R^2	MSE	R^2	MSE	R^2
Goel Okumoto	0.943	0.988	12.90	0.9857	156.2	0.987
Delayed S shaped Model	3.22	0.956	28.05	0.969	188.5	0.983
Inflection S shaped Model	0.943	0.988	10.56	0.989	98.20	0.922
Chiu 2008	0.979	0.988	10.56	0.989	98.20	0.992
Javaid Iqbal and N. Ahmad and S.M.K. Quadri 2012	1.016	0.988	11.97	0.989	112.14	0.992
Proposed Model	0.532	0.994	6.18	0.995	63.84	0.996

Table 4: Comparison between different models

S.No	actual	Proposed_model	Iqbal_Ahmad_Qadri_2012	Goel_Okumoto	Delayed_S_shape
1	1	1.022151	0.8752587	0.8757065	0.1217467
2	2	1.577253	1.3575004	1.3581930	0.2847856
3	3	2.145998	1.8568133	1.8577578	0.5175966
4	4	3.255607	2.8467186	2.8481582	1.1486265
5	5	5.230822	4.6636064	4.6659390	2.7737418
6	6	6.939398	6.2965773	6.2996948	4.5981649
7	7	8.127431	7.4685350	7.4722054	6.0429578
8	8	9.112408	8.4644818	8.4686151	7.3255766
9	9	9.301397	8.6582000	8.6624226	7.5789507

IPASJ International Journal of Computer Science (IIJCS)
Web Site: http://www.ipasj.org/IIJCS/IIJCS.htm
A Publisher for Research Motivation ....
Email:editoriijcs@ipasj.org
Volume 6, Issue 6, June 2018
ISSN 2321-5992

10	10	9.573680	8.9388193	8.9431708	7.9476025
11	11	9.759496	9.1313669	9.1358065	8.2014747
12	12	11.702543	11.1974875	11.2028573	10.9404691
13	13	12.382554	11.9442018	11.9499006	11.9220177
14	14	14.008514	13.7812658	13.7877579	14.2755634
15	15	14.921546	14.8444786	14.8514190	15.5816356
16	16	17.292159	17.6926279	17.7007290	18.8134803
17	17	17.453929	17.8898337	17.8980131	19.0212282
18	18	17.784310	18.2927148	18.3010529	19.4388240
19	19	18.490539	19.1516137	19.1602863	20.2980923
20	20	19.387756	20.2270597	20.2361432	21.3136401
21	21	19.868756	20.7898197	20.7991146	21.8181182
22	22	22.878850	23.8818814	23.8922937	24.2633475
23	23	23.570223	24.4613554	24.4719685	24.6617508
24	24	24.147468	24.9085201	24.9192863	24.9567704
25	25	24.886869	25.4369988	25.4479437	25.2917529
26	26	26.045804	26.1795858	26.1907781	25.7379686
27	27	26.653233	26.5341134	26.5454220	25.9411474
28	28	27.498249	26.9955822	27.0070407	26.1963008
29	29	28.803681	27.6530110	27.6646800	26.5420460
30	30	30.434754	28.4158528	28.4277612	26.9179177

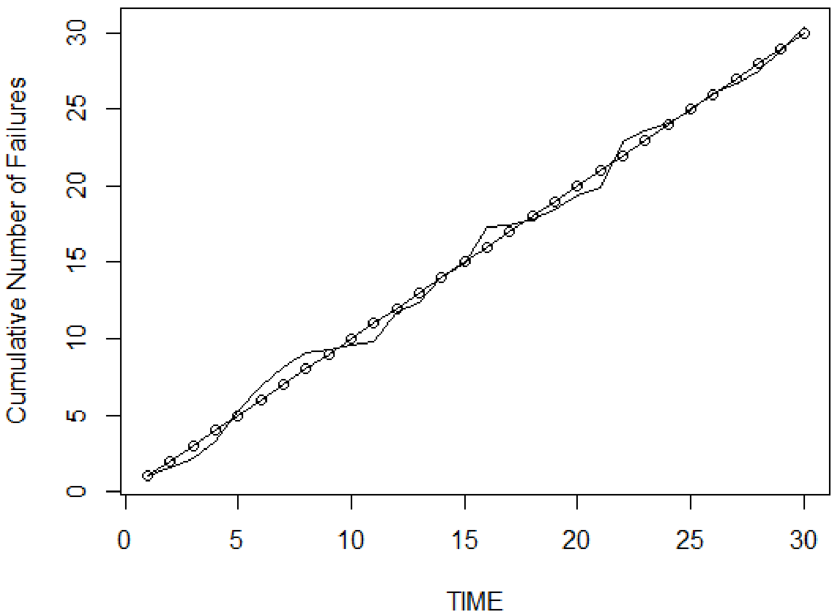

Figure 1 : Graph between cumulative number of failures and Time

Figure 1 indicates the estimated model data with original dataset 1 and ***Figure 2*** indicates the proposed model fitted with the original dataset 2

Figure 2

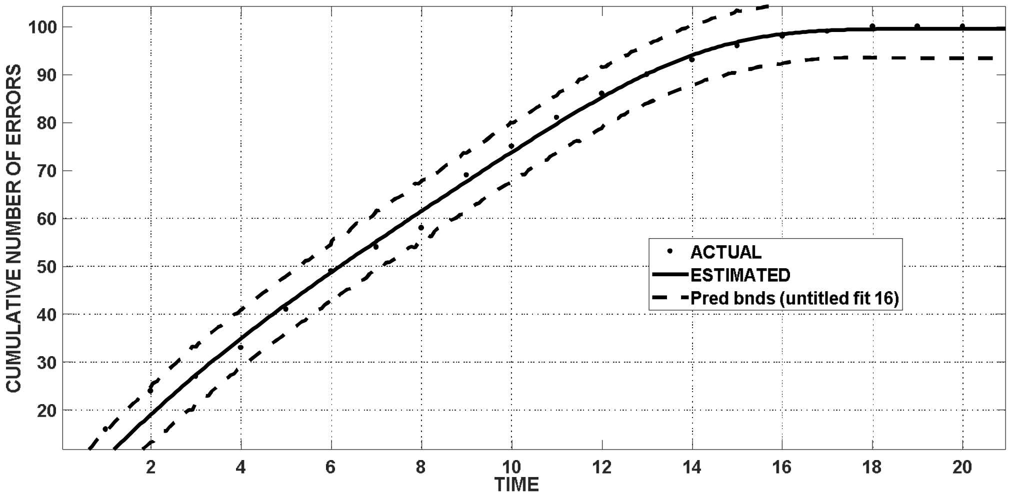

Figure 2: Graph between cumulative number of failures and Time

VI Conclusions

This paper mainly we integrated the testing effort into dynamic learning function into software reliability growth models. As testing is the one important phase where 50% of resources are being consumed. Testing phase itself is a dynamic environment where finding actual error are some difficult process same time testers need lots of experience and learning capacity to adopt the current environment fluctuations. By integrating Testing effort in to software reliability growth model can give a realistic evaluation of testing environment. Results have shown that our proposed model fits good compared with other models. In future we want to develop some more rigorous models which can capable to adopt the fluctuating environment.

References

[1] Javaid Iqbal and N. Ahmad and S.M.K. Quadri " A Software Reliability Growth Model with Two Types of Learning" Proceedings of the 2012 IEEE IEEM

[2] Kuei-Chen Chiu " A Study of Software Reliability Growth Model for Time-dependent Learning Effects" Proceedings of the 2012 IEEE IEEM

[3] Javid Iqbal "Analysis of Some Software Reliability Growth Models with Learning Effects" I.J.Mathematical Sciences and Computing 2016.

[4] Javid Iqbal " Software reliability growth models: A comparison of linear and exponential fault content functions for study of imperfect debugging situations" Cogent Engineering (2017).

[5] Kuei-ChenChiuYeu-ShiangHuang and Tzai-ZangLee"A study of software reliability growth from the perspective of learning effects" 2008.

[6] Huang, C.Y., Kuo, S.Y. and Lyu, M.R. (2000), “Effort-index based software reliability growth models and performance assessment”, in Proceedings of the 24th IEEE Annual International Computer Software and Applications Conference (COMPSAC’2000), pp. 454-9.

[7] Chiu ,Kuei Chen " A discussion of software reliability growth models with time varying learning effects" American Journal of Software engineering and Applications 2013

[8] Pham, H. Software Reliability, Springer-Verlag, New York, NY.2000

[9] S. CHATTERJEE , R. B. MISRA & S. S. ALAM "Joint effect of test effort and learning factor on software reliability and optimal release policy" Pages 391-396 | Received 31 Jul 1996, Accepted 30 Sep 1996, Published online: 05 Apr 2007.

AUTHOR

Shaik Mohammad Rafi completed B.Tech and M.Tech Computer Science and Engineering, pursuing Ph.D in Computer Science and Engineering from Acharya Nagarjuna University, He is having 15 years of teaching experience.. He has published more than 20 Research Papers in various International Journals. His area of Research includes Software Reliability, Software Architecture Recovery, Soft Computing, Data Mining,

Dr.Shaheda Akthar received Bachelor and Masters Degree in Computer Science from Acharya Nagarjuna University, M.S from Birla Institute of Technology and Science, Pilani, PhD Computer Science and Engineering from Acharya Nagarjuna University. , She is having 18 years of teaching experience. Presently working as Registrar I/c,, Dr.Abdul Haq Urdu University, and Kurnool. Her area of Research includes Software Architecture Recovery, Software Reliability, Soft Computing, Data Mining, and Security.