Volume 4, Issue 2, February 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

A Log Based Approach for Software Reliability Modeling

K.Venkata Subba Reddy[1]
Muffakham Jah College of Engineering and Technology
India

Dr.B.Raveendra Babu[2]
VNR Vignana Jyothi Institute of Engineering & Technology
India

Abstract— Most of the software reliability (SR) models are applicable to a single piece of software. In this paper a Log based approach is proposed for more than one software applications. The proposed method is compared against the conventional models and proved to be one of the best in the qualitative analysis. The method also proclaims the use of modular information and uses in software reliability assessment.

Keywords— Log Based Approach, Software Reliability, NHPP Models.

I. INTRODUCTION

Today, science and technology demand high-quality software for making improvements and breakthroughs [1]. Though a lot of SR models are proposed there is no single model which can be superior to all others [2]. Xie.et.al proposed several simple Non Homogeneous Poisson Model (NHPP) models which have some useful graphical interpretations. One among these models is the log-power NHPP model. If the log-power model is suitable for a given data set, the plot of the observed cumulative number of failures and the running time will tend to be on a straight line on a log-log-log scale. The slope of the fitted line and its intercept on the vertical axis will give us the estimates of the parameters.

In this paper, we analyze the log based NHPP software reliability model. The method will show that from various points of view, log-power model provides with a simple statistical inference procedure and is able to fit different data sets**.** Also, the log-based model has a relatively high predictive ability. The log power model is thus an applicable model in software reliability modeling.

II. BACKGROUND

There are three classes of Software Reliability models; they are Exponential NHPP models, Non-exponential NHPP models and Bayesian models.

A Poisson probability distribution function takes the form $f(t) = \lambda e^{-\lambda t}$ .The mean time function is $\mu(t) = \lambda$. Homogeneous Poisson Process models assume a constant mean time function while Non-Homogeneous Poisson Process models assume a mean tie function to be non-liner.

A) EXPONENTIAL NHPP MODELS:

Models in this type are based on shooman's model, musa's basic model, Jelinski and Moranda's model. Below is the probability distribution of these models

Shooman's model

$$f(t) = k[\frac{E_0}{I_T} - \epsilon_c(x)] \quad (1)$$

Where E_0 is the initial number of faults in the program that will leads to failures E_c is the number of faults in the program which have been found and corrected, K- is constant of proportionality

Musa's basic model

$$\mu(t) = \beta_0(1 - e^{-\beta_1 t}) \quad (2)$$

Where β is Negative of derivative of failure rate divided by failure rate

Jelinski and Moranda's Model

$$\mu(t) = N(1 - e^{-\phi t}) \quad (3)$$

Scheneidewind's models

$$\mu(t) = \frac{\alpha}{\beta}(1 - e^{-\beta i}) \quad (4)$$

B) NON-EXPONENTIAL NHPP MODELS:

Duane's Model

$$\mu(t) = \alpha t^{\beta}) \quad (5)$$

Brook and mohey's Poisson models

$$P(x = n_i) = \frac{(N_i \phi_i)^n \, e^{N_i \phi_i}}{n_i!} \quad (6)$$

Yamada's S-Model

$$\mu(t) = N \sum_{i=1}^{k} p_i [1 - e^{-\beta_i t}] \quad (7)$$

Musa and Okumoto model

$$\mu(t) = \lambda_0 e^{-\phi\mu} \quad (8)$$

C) BAYESIAN MODELS:

Little wood $\lambda_{linear}(t) = \frac{\alpha - 1}{\sqrt{\beta_0^2 + 2\beta_1 t(\alpha - 1)}}$ and

$$\lambda_{quadratic}(t) = \frac{v_1}{\sqrt{t^{2+v_2}}}\left(\left(t + (t^2 + v_2)^{\frac{1}{2}}\right)^{\frac{1}{3}} - (t - (t^2 + v_2)^{1/2})^{\frac{1}{3}}\right) \quad (9)$$

The accuracy of the models in the same class is generally the same, as the general reliability function of them is same. Hence, it is enough to argue about the accuracy if at least one model in each class is considered.

III. PROPOSED METHODOLOGY

A) Log Based Approach:

This model assumes that cumulative number of failures is an NHPP with the mean value function

$$m(t) = aln^b(1 + t) \ where \ a, b > 0, t \geq 0 \quad (10)$$

And it can be further decomposed as

$$lnm(t) = lna + blnln(1 + t) \quad (11)$$

Let $x=ln(1+t)$ then the mean value function becomes

$$m(x) = ax^b \quad , x > 0 \quad (12)$$

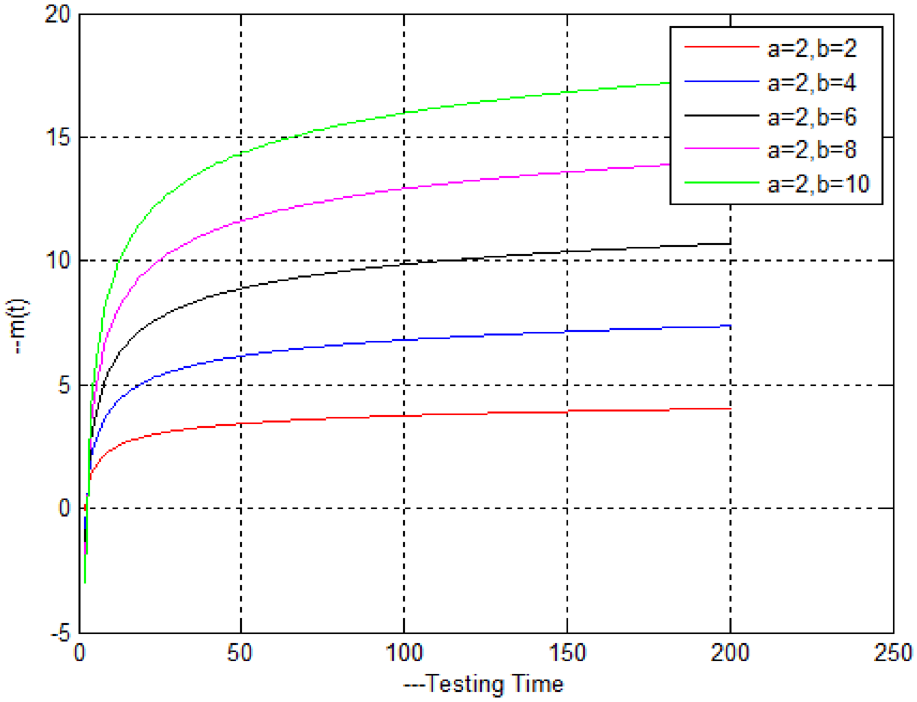

Fig. 1: Performance Analysis of the Proposed Method

This is the model proposed by Duane [3] and this widely applied in analyzing the data of repairable systems. The reason that the Duane model has not received much attention in software reliability is that the model usually overestimates the cumulative number of failures because its mean value function approaches infinity too fast. The log-power model which puts *ln(1 +t)* instead of t in the Duane model, seems to be a reasonable modification of the Duane model, since *ln(1 +t)* grows much slower than *t.*

The mean value function of the Musa-Okumoto model [4] is

$$m(t) = aln(1 + bt) \ where \ a, b > 0 \quad (13)$$

The model presented by Xie et .al [6] has a slight modification of the musa model and the mean function is

$$m(t) = aln(bt) \quad where \ a, b > 0 \ \& \quad t>1 \quad (14)$$

Based on the equations 8 to 12 a general log based mean function applicable to more software application is proposed as

$$m(t) = aln^c(1 + bt) \ where \ a, b, c > 0 \quad (15)$$

Table 1: Relative Error Performance of Different Models

n	Log Based Musa	Log Based Duane	Log Based Goel [5]
10	1.045	0.67	0.95
20	1.31	0.789	1.78
30	1.35	0.855	2.165
40	1.65	0.458	1.56
50	1.146	2.3	0.85

60	0.65	10.5	0.95
70	1.05	8.56	28.4
80	2.06	5.96	655.5
90	3.5	5.5	4735.3
100	1.25	31.65	2118.95

IV. CONCLUSIONS

Log based mean function is applied several data sets and compared against the well known models and the results are showing to be promising. The ML estimates of the parameters for the inter failure time data have very simple forms and no numerical techniques are needed for the calculation of the ML estimates. It has been observed that the log-power model has a strong prediction ability based on various criteria. The simple graphical interpretation and the easily calculated ML estimation indicate that this model seems to be useful and convenient for applications.

REFERENCES

[1] Michael R. Lyu, "Software Reliability Engineering: A road map", *IEEE conf on .future of software engineering (FOSE-2007).*

[2] M. Xie, "Software Reliability Modeling ", World Scientific Publisher, Singapore (1991) J.T. Duane," Learning curve approach to reliability Monitoring", *IEEE Trans. Aerospace,* AS-2, 563-566.

[3] J.T. Duane," Learning Curve Approach to Reliability Monitoring", *IEEE Trans. Aerospace,* AS-2, 563-566.

[4] J.D. Musa, and K. Okumoto," A logarithmic Poisson execution time model for software reliability measurement", Proc. *7th Inf. Conf. on Software Engineering,* 23@238 (1984).

[5] AL. Goel and K. Okumoto, "Time dependent error detection rate model for software reliability and other performance measures", *IEEE Trans Reliability,* R-28 206-21 1 (1979).

[6] M. Zhao and M. Xie, "Comparison of the log power NHPP model with some other two parametric Ones ", *Technical Report, LiTH-IKP-R-6.53,* Linkoping, Sweden (1991)

[7] H. Ascher. and H. Feingold, *Repairable Systems Reliability: Modeling, Inference, Misconceptions and their Causes.* Marcel Dekker, New York (1983).

[8] M.R. Lyu and A. Nikora, A heuristic approach for software reliability prediction: the equally-weighted linear combination model. *Proc. Inf. Symp. On Software Reliability Engineering,* May 18-19, Austin, Texas, 172-181 (1991).

[9] D.W. Noon, Practical software reliability. *IFAC Reliability of Instrumentation Systems,* The Hague, Netherlands, 89-96 (1991)

[10] M. Zhao and M. Xie, Applications of the log power NHPP software reliability model. *Technical Report, LiTH-IKP-R-652,* Linkoping University, Sweden (1991).

AUTHORS PROFILE

K.Venkata Subba Reddy obtained his B.Tech in Information Technology from University of Madras in 2002 and received the M.Tech in Software Engineering from Bharath University, Chennai in 2005; He is pursuing Ph.D., in Computer Science and Engineering, under the guidance of Dr.B.Raveendra Babu, at Acharya Nagarjuna University, Guntur, Andhra Pradesh, India. He is currently working as Assistant Professor in Computer Science & Engineering Department at Muffakham Jah College of Engineering & Technology, Banjara hills, Hyderabad. He has 9 years of experience in teaching. His research interests include Software Engineering, Software Quality Assurance and Testing and Cloud computing. He is a life member of ISTE and a member of CSI.

Dr B. Raveendra Babu obtained his Masters in Computer Science and Engineering from Anna University, Chennai. He received his Ph.D. at S.V University, Tirupati. He is currently working as Professor and Head, Department of CSE, VNRVJIET, Hyderabad. He has 26 years of teaching experience. He has more than 25 international & national publications to his credit. His research areas of interest include VLDB, Image Processing, and software engineering.