



SOFTWARE RELIABILITY GROWTH MODEL BASED ON PARETO TYPE III DISTRIBUTION

Y. Vamsidhar¹, Y.Srinivas², Achanta Brahmini³

Department of CSE, Swarnandra College of Engg & Technology, Narsapur, Andhrapradesh, India¹

Department of Information Technology, GITAM University, Visakhapatnam, Andhrapradesh, India²

Department of CSE, GITAM University, Visakhapatnam, Andhrapradesh, India³

ABSTRACT: In any business environment the ultimate objective is to derive a quality product. In order to deliver a quality product, the quality characteristics are to be mainly focused. Among the various Quality characteristics, Reliability is considered as the most predominant characteristics. Many models have been utilised for assessing the quality of a software using reliability but very little focus on Pareto Type III distribution. Hence this paper mainly projects in this direction.

KEYWORDS: Non-Homogeneous Poisson Process, Pareto Type II Growth model, Pareto Type III Growth Model, Software Reliability.

I.INTRODUCTION

Software Reliability is mainly focused on identifying the failures in a given software and helps to build a reliable model by which the identified failures can be overcome. Analysing the software helps to build an error-free system. Software growth model have gained importance since it can identify the probability of failure rate of a software in a given specified time and in particular to a specific environment. Any software reliability can be tested basing on different parameters such as lines of code (LOC), no. of loops within a program, based on the input and output data in these models are generally termed as defective density models. In any real time environment if a software occurs, we have to co-relate the failure rate and these failure rate is a co-relations of several factors. In order to identify these correlations statistical methodologies are well suited. Here the second types of models called reliability growth models are used.

Reliability Growth Models helps to identify the failure rate during the testing phase based on the input functions which are generally termed as exponential functions. These functions help to predict the behaviour of a software with the given functions. Much work has been reported in literature and most of the dominates the software Reliability Growth models are Crow and Basu(1988), Goel and Okumoto(1979,1984),Musa(1980),Pham(2005),Ramamurthy and Bastani (1982), Zhang,Teng and Pham(2003), Malaiya,Karunanithi and Verma (1982) and Wood(1996). Software Reliability models can be addressed using concave and S shaped Models, Gompertz curves etc. These models behave in a similar fashion where if probability of defects increases, the failure rate increases and if the no. of defects increases, the total no. of defects reaches a finite value. Software Reliability Models are probabilistic models where the failure of occurrences and fault removals are events in the model.

The Software reliability models are well estimated using Non-Homogeneous Poisson process(NHPPs)(R. Satya Prasad et al (2011)).Since its ability to determine the mean value function by which the expected no. of failures can be identified over a period of time. Among the Non-Homogeneous models Goel-Okumoto (1979, 1984) models proposed by R. Satya Prasad et al(2011) are mainly projects. The works of R. Satya Prasad are based on Pareto Type II models. But in order to estimate the failure rate most exactly pareto Type III distribution are more advantageous(Giuito Bottazi (2009)).Hence in this paper a Software Reliability Growth Model has been proposed using Pareto Type III distribution. The rest of the paper is organized as follows, Section II of the paper describe the Pareto model Development, Section III of the paper presents the parameter estimation of the pareto type III model, the experimentation based on NTD's Software failure data analysis is presented in Section IV and the conclusion derived or highlighted in Section V.



II. PARETO TYPE III MODEL DEVELOPMENT

Software Reliability Model is well estimated using Non-Homogenous Poisson process. Non-Homogenous is a counting process which is used to determine an appropriate mean value function $m(x)$.

Where $m(x)$ represents the expected no. of software failures at time in this paper, we consider Pareto type III distribution contains three parameters and $m(x)$ is given by

$$m(x) = 1 - \left(\frac{x}{s} + 1\right)^{-a} e^{-b\frac{x}{s}}$$

Where x represents the cumulative time,
 $s > 0$ is a scale parameter and

If $a > 0$ is a power coefficient then

$a \geq 0$ is an exponential coefficient

If the exponential coefficients say b is set to 0 then the above equation reduces to Pareto type II model. The equation as follows.

$$m(x) = 1 - \left(\frac{x}{s} + 1\right)^{-a} \quad [e^0 = 1]$$

We consider one new parameter in Pareto type III, (i.e.) scale parameter. The associated density function is

$$m(x) = \frac{1}{s} \left(\frac{x}{s} + 1\right)^{-a} e^{-b\frac{x}{s}} \left(b + \frac{a}{\frac{x}{s} + 1}\right)$$

The expressions of a , b , s can be estimated using logarithmic likelihood function and values can be obtained using Newton Raphson Model.

III. PARAMETER ESTIMATION OF PARETO TYPE III MODEL

In this section we derive expression for estimating the parameters of Pareto type III distribution. The parameter estimation is of time primary importance in software Reliability prediction. A set of failure data can be commonly collected in one of the two ways, time domain and interval domain data. In this paper we consider time between software failures for estimating the parameters.

The mean value function of Pareto type III model is given by.

$$m(x) = \frac{1}{s} \left(\frac{x}{s} + 1\right)^{-a} e^{-b\frac{x}{s}} \left(b + \frac{a}{\frac{x}{s} + 1}\right)$$

Applying log on both L.H.S and R.H.S

$$\log m(x) = \log\left(\frac{1}{s}\right) - a \log\left(\frac{x}{s} + 1\right) - \frac{bx}{a} + \log\left(b + \frac{a(s+1)}{x}\right) \quad (1)$$

Parameter “ s ” can be estimated by Differentiate (1) w.r.t ‘ b ’

$$0 = 0 - 0 - \frac{x}{a} + \frac{1}{b + \frac{a(s+1)}{x}}$$

$$\Rightarrow \frac{x}{a} = \frac{x}{bx + a(s+1)}$$

$$\Rightarrow \frac{1}{a} = \frac{1}{bx + a(s+1)}$$

$$\Rightarrow a = bx + as + a$$



$$\Rightarrow s = \frac{-bx}{a}$$

Parameter “a” can be estimated by Diff (1) w.r.t. ‘s’

$$0 = s \left(-\frac{1}{s^2} \right) - \frac{a}{1 + \frac{x}{s}} \left(\frac{-x}{s^2} \right) + \frac{\frac{a}{x}}{b + \frac{a(s+1)}{x}}$$

$$\Rightarrow \frac{1}{s} - \frac{ax}{s(s+x)} = \frac{a}{bx+a(s+1)}$$

$$\Rightarrow \frac{(s+x)-ax}{s(s+x)} = \frac{a}{bx+a(s+1)}$$

$$\Rightarrow bx(s+x) - abx^2 + a(s+1)(s+x) - a^2x(s+1) = as(s+x)$$

$$\Rightarrow bx(s+x) - abx^2 + a(s+x) - a^2x(s+1) = 0$$

$$\Rightarrow a^2x(s+1) - a(s+x) + abx^2 - bx(s+x) = 0$$

$$\Rightarrow a^2x(s+1) - a[bx^2 - (s+x)] - bx(s+x) = 0$$

$$\Rightarrow a = \frac{(s+x)-bx^2 \pm \sqrt{[bx^2-(s+x)]^2 + 4bx^2(s+1)(s+x)}}{2x(s+1)}$$

Parameter “b” can be estimated by Diff w.r.t. ‘a’

$$0 = 0 - \log \left(1 + \frac{x}{s} \right) + \frac{bx}{a^2} + \frac{(s+1)}{x \left[b + \frac{a(s+1)}{x} \right]}$$

$$\Rightarrow \log \left(1 + \frac{x}{s} \right) = \frac{bx}{a^2} + \frac{s+1}{bx+a(s+1)}$$

$$\Rightarrow = \frac{bx[bx+a(s+1)]+a^2(s+1)}{a^2[bx+a(s+1)]}$$

$$b^2x^2 + abx(s+1) - a^2(s+1) = a^2[bx+a(s+1)] \log \left(1 + \frac{x}{s} \right)$$

$$\Rightarrow b^2x^2 + b \left[ax(s+1) - a^2x \log \left(1 + \frac{x}{s} \right) \right] + a^2(s+1) \left[1 - a \log \left(1 + \frac{x}{s} \right) \right] = 0$$

$$\Rightarrow b = \frac{a^2x \log \left(1 + \frac{x}{s} \right) - ax(s+1) \pm \sqrt{\{ax(s+1) - a^2x \log \left(1 + \frac{x}{s} \right)\}^2 - 4x^2(s+1) \{1 - a \log \left(1 + \frac{x}{s} \right)\}}}{2x^2}$$

The value of “s”, “b” in above equations can be obtained by using Newton Rapson Method. These equations are iteratively and the solutions are substituted in expression “a” to get the value of “a”.

IV. DATA ANALYSIS

In this section we present the NTDS software failure data, taken from Jelinski and Mornda (1972).the Naval tactical Data Systems (NTDS) software failure data are taken from U.S Navy Fleet Computer Programming Center. The NTDS software consists of an error which consists of 38 modules. Each module consists of three phases; the production phase, the test phase and user phase. The time between software failures is shown in below table.



Error Number N	Time between Errors Sk days	Cumulative Time xn = Days
Production (Checkout) Phase		
1	9	9
2	12	21
3	11	32
4	4	36
5	7	43
6	2	45
7	5	50
8	8	58
9	5	63
10	7	70
11	1	71
12	6	77
13	1	78
14	9	87
15	4	91
16	1	92
17	3	95
18	3	98
19	6	104
20	1	105
21	11	116
22	33	149
23	7	156
24	91	247
25	2	249
26	1	250
Test Phase		
27	87	337
28	47	384
29	12	396
30	9	405
31	135	540
User Phase		
32	258	798
Test Phase		
33	16	814
34	35	849

TABLE-1 NTDS Data

Table-1, showing the dataset containing 26 failures in 250 days, time of errors and cumulative errors.

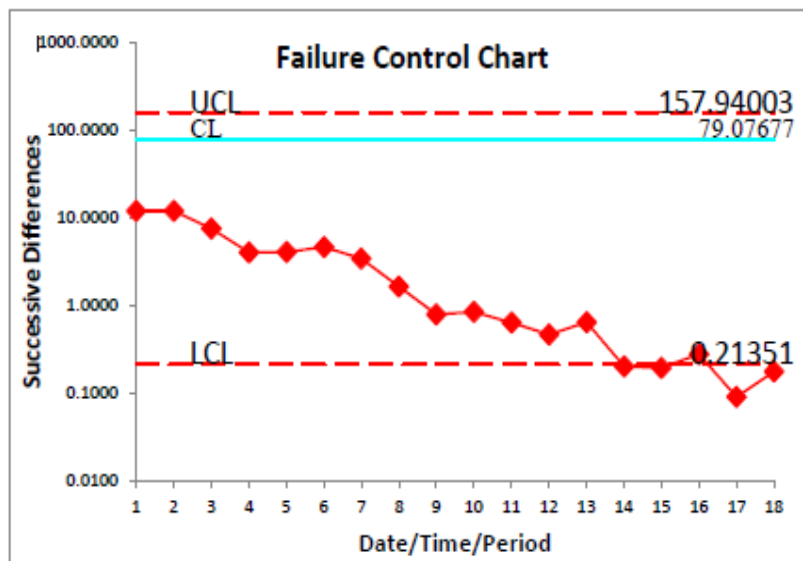


Figure 1. Mean Value chart

Fig-I, showing the control chart of the failure data.

VI.CONCLUSION

Software Reliability is an important quality measure that quantifies the operational profile of computer systems. In this paper we proposed Pareto type II software reliability growth model. This model is primarily useful in estimating and monitoring software reliability, which is viewed as a measure of software quality. Equations to obtain the maximum likelihood estimates of the parameters based on interval domain data are developed. This analysis shows that the Release 2 data (Figure 1) shows out of control signals i.e., below the LCL. We conclude that our method of estimation and the control chart are giving a +ve recommendation for their use in finding out preferable control process or desirable out of control signal. By observing the Mean Value Control chart (Figure 1) we identified that the failure situation is detected at 14th point of Table-2. Hence our proposed Mean Value Chart detects out of control situation. This is a simple method for model validation and is very convenient for practitioners of software reliability. The early detection of software failure will improve the software reliability. The methodology adopted in this paper is better than the methodology adopted by Xie et al [2002]. Therefore, we may conclude that this model is the best choice for an early detection of software failures.

REFERENCES

- [1] Goel, A.L., Okumoto, K., 1979. Time-dependent error detection rate model for software reliability and other performance measures. IEEE Trans. Reliab. R-28, 206-211.
- [2] Musa J.D, Software Reliability Engineering McGraw-Hill, 1998.
- [3] Pham .H (2005) "A Generalized logistic Software Reliability Growth Model", Opsearch, Vol. 42, No. 4, 332-331.MCGraw-Hill, New York.
- [4] Musa J.D., Iannino, A., Okumoto, K., 1987. Software Reliability: Measurement Prediction Application. MC Graw Hill, New York.
- [5] Wood, A(1996), "Predicting software Reliability", IEEE Computer,2253-2264.
- [6] Satya Prasad R and Geetha Rani N (2011), "Pareto type II Software Reliability Growth Model". International Journal of Software Engineering, Volume 2, Issue(4) 81-86.
- [7] Satya Prasad R (2007) "Half Logistic Software Reliability Growth Model", Ph.D Thesis of ANU, India.
- [8] Pham, H., "System Software Reliability", Springer 2006.
- [9] Satya Prasad R and Goutham V (2011), "Interval Domain Software Process Control-Goel Okumoto