2012 International Conference on Network and Computational Intelligence (ICNCI 2012)
IPCSIT vol. 46 (2012) © (2012) IACSIT Press, Singapore

A Duane Equivalent Software Reliability Growth Test

Jing HE[1] + and Yang LIU

[1] China Satellite Maritime TT&C Department

Abstract. Duane model is a hardware model for reliability growth test, whereas having no effective way to software growth test, this paper proposes a method applying Duane model as a equivalent substitute to software. In this paper, not only all of the equations involved Duane model equivalent test has been given, but also from a practical case of the S-band radio device for satellite tracking and controlling, we demonstrate the whole process of such a test. Drawing a Duane curve with the original reliability data of the S-band radio device, we use the least-squares-error means to estimate the parameters of Duane model and draw an amendable Duane curve. Finally we calculate the system software MTBF and A value and compare them with prediction value. Consequently it proves that Duane model is correct and reasonable for software reliability growth test.

Keywords: Duane model, Least squares estimation (LSE), Software reliability growth test, MTBF.

1. Introduction

Reliability growth test is a important step for product reliability to be improved efficiently. A lot of statistics show that a new sample product initially just has 10%～30% prediction of Mean Time Between Failures (MTBF) so there is no other way than reliability growth test to make product reliability increase to its prediction in the last figuration of product. With more and more large-scale complicated electro-mechanical products such as a certain satellite Tracking and Controlling S-band integrative system is made basically by software it is essential to do software reliability growth test. First work is to establish its distributed model. Presently there are hundreds of available models but effectiveness which can be achieved is so little on account of complicated software failures in principle that in the end of test the result is so bad as either errors between prediction and fact going beyond its limit and resulting the whole growth test failure or the test time is so long as to cost too much. A simpler and more effective way is supposed to use a more applied hardware growth model as a equivalent substitute to software. Generally Duane or AMSAA model is the best hardware growth model in use.

2. Duane model

Here a assumption, t means cumulative working time of repairable product, N(t) means cumulative invalid time on (0,t), $\lambda_\Sigma(t)$ means cumulative invalid rate which equals cumulative invalid time divided by means cumulative working time, namely:

$$\lambda_\Sigma(t) = N(t)/t \tag{1}$$

Duane model indicates a connection between cumulative invalid rate $\lambda_\Sigma(t)$ and cumulative working time t which can be drawed to a linear chart on a dual logarithmic graph paper which equation is:

$$\ln \lambda_\Sigma(t) = \ln \lambda_I - \alpha \ln t \quad . \tag{2}$$

+ Corresponding author. *E-mail address*: icejialinhe@sina.com.

$$\text{Namely } \lambda_{\Sigma}(t) = \lambda_I t^{-\alpha} \tag{3}$$

within this equation . $\lambda_{\Sigma}(t)$ equals cumulative invalid rate on the time of observation. λ_I equals initial invalid rate when cumulative working time t equals 1 which also calls scale parameter with a geometric meaning of Duane chart intercept on a dual logarithmic graph paper. α equals estimated reliability growth rate with a geometric meaning of Duane chart slope on a dual logarithmic graph paper which represents tendency of cumulative invalid rate to descend smoothly, and another equation can be established by Eq.(1) as:

$$N(t) = \lambda_{\Sigma}(t)t = \lambda_I t^{1-\alpha} \tag{4}$$

So by Eq.(4) it can establish a new equation of connection between instantaneous invalid rate and cumulative invalid rate as: .

$$\lambda(t) = dN(t)/dt = (1-\alpha)\lambda_I t^{-\alpha} = (1-\alpha)\lambda_{\Sigma}(t) \tag{5}$$

Thus evaluation of instantaneous invalid rate is as:

$$\hat{\lambda}(t) = (1-\alpha)\hat{\lambda}_I t^{-\hat{\alpha}}. \tag{6}$$

So cumulative value of Mean Time Between Failures (MTBF) $\theta_{\Sigma}(t)$ and its instantaneous value can be counted by a equation as follows:

$$\theta_{\Sigma}(t) = 1/\lambda_{\Sigma}(t) = t^{\alpha}/\lambda_I \tag{7}$$

$$\theta(t) = 1/\lambda(t) = t^{\alpha}/(1-\alpha)\lambda_I \tag{8}$$

3. Least squares estimation of Duane model

During software reliability growth test cumulative invalid rate $\hat{\lambda}_{\Sigma}(t)$ can be estimated by all kinds of software failures which occur successively and software reliability growth chart also can be drawed. So after this work instantaneous invalid rate $\hat{\lambda}(t)$ can be estimated with the first step to draw a chart of connection between $\lambda_{\Sigma}(t)$ and t on a dual logarithmic graph paper. Here is a example test of a certain satellite Tracking and Controlling S-band integrative system.

So according to the above way the first step is to draw a Duane chart of $\ln\hat{\lambda}_{\Sigma}(t) \sim \ln t$ on a dual logarithmic graph paper by MATLAB. Here is shown in Fig.1 with a collected data in Tab.1.

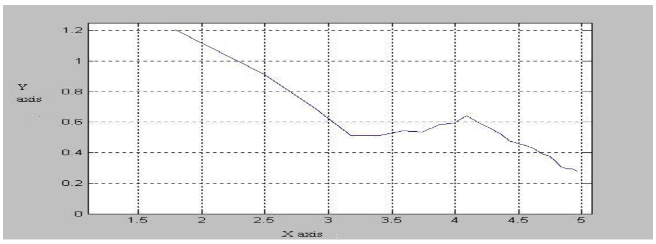

Fig. 1: a Duane chart on the original data

Therefore, scale parameter λ_I and reliability growth rate α can be gained by Least squares estimation or graphics estimation, so Duane model chart can be imitated and modified. Generally Least squares estimation is the best way to do this for its higher precision and easier operation in computer.

The dual logarithmic coordinate of Cumulative MTBF can be denoted by Eq.(7):

$$\ln\theta_{\Sigma}(t) = -\ln\lambda_I + \alpha\ln t \tag{9}$$

At the same time cumulative MTBF can be denoted by cumulative working time $t_1, t_2, ..., t_n$ and its corresponding Cumulative invalid numbers $N(t_1), N(t_2), ..., N(t_n)$ as:

$$\theta_{\Sigma}(t_i) = t_i/N(t_i) \quad (i = 1,2,...,n) \tag{10}$$

Then according to Duane model and Eq.(10) a new equation can be established as:

$$\ln\theta_\Sigma(t_i) = -\ln\lambda_I + \alpha\ln t_i + \varepsilon_i \quad (i = 1,2,...,n) \tag{11}$$

So remanent error squares sum equals:

$$\sum_{i=1}^{n}\varepsilon_i^2 = \sum_{i=1}^{n}(\ln\theta_\Sigma(t_i) + \ln\lambda_I - \alpha\ln t_i)^2 \tag{12}$$

Table. 1: Software invalid data grouped by time

serial	***cumulative working time***	***Invalid numbers per group***	***Cumulative Invalid numbers***	***Cumulative Invalid rate***
(num.)	t_i ***(days)***	$N(t_i) - N(t_{i-1})$ ***(num.)***	$N(t_i)$ ***(num.)***	$\lambda_\Sigma(t_i)$ ***(num./day)***
1	6	20	20	3.33
2	12	10	30	2.50
3	18	6	36	2.00
4	24	4	40	1.67
5	30	10	50	1.67
6	36	12	62	1.72
7	42	10	72	1.71
8	48	14	86	1.79
9	54	12	98	1.81
10	60	16	114	1.90
11	66	6	120	1.81
12	72	6	126	1.75
13	78	6	132	1.69
14	84	5	136	1.61
15	90	6	142	1.58
16	96	8	150	1.56
17	102	6	156	1.53
18	108	4	160	1.48
19	114	6	166	1.46
20	120	4	170	1.42
21	126	2	172	1.36
22	132	6	178	1.34
23	138	8	186	1.34
24	144	4	190	1.32

Under the circumstance of remanent error squares sum is minimum, Least squares estimation (LSE) on λ_I and α can be gained by a equation group as followed

$$\begin{cases} \dfrac{\partial}{\partial\ln\lambda_I}\sum_{i=1}^{n}\varepsilon_i^2 = 0 \Rightarrow \sum_{i=1}^{n}(\ln\theta_\Sigma(t_i) + \ln\lambda_I - \alpha\ln t_i) = 0 \\ \dfrac{\partial}{\partial\alpha}\sum_{i=1}^{n}\varepsilon_i^2 = 0 \Rightarrow \sum_{i=1}^{n}(\ln\theta_\Sigma(t_i) + \ln\lambda_I - \alpha\ln t_i)\ln t_i = 0 \end{cases} \tag{13}$$

So LSE value of λ_I and α equals:

$$\hat{\alpha} = \frac{n\sum_{i=1}^{n}\ln\theta_\Sigma(t_i)\ln t_i - (\sum_{i=1}^{n}\ln\theta_\Sigma(t_i))\cdot(\sum_{i=1}^{n}\ln t_i)}{n\sum_{i=1}^{n}(\ln t_i)^2 - (\sum_{i=1}^{n}\ln t_i)^2} \tag{14}$$

$$\hat{\lambda}_I = \exp\left\{\frac{1}{n}(\hat{\alpha}\sum_{i=1}^{n}\ln t_i - \sum_{i=1}^{n}\ln\theta_\Sigma(t_i))\right\} \tag{15}$$

The dual logarithmic coordinate of cumulative working time and cumulative MTBF is shown in Tab.2. Then it can be counted by table.2. as:

$$\sum_{i=1}^{24}\ln t_i = 97.7873 \;,\quad \sum_{i=1}^{24}(\ln t_i)^2 = 414.2938 \;,\quad \sum_{i=1}^{24}\ln\theta_\Sigma(t_i) = -12.5347 \;,\quad \sum_{i=1}^{24}\ln\theta_\Sigma(t_i)\ln t_i = -47.506$$

So LSE of α is : $\hat{\alpha} = \dfrac{-24 \times 47.506 + 12.5347 \times 97.7873}{24 \times 414.2938 - 97.7873^2} = 0.2248$

LSE of λ_I is : $\hat{\lambda}_I = \exp\left\{\dfrac{1}{24}(0.2248 \times 97.7873 + 12.5347)\right\} = 4.2132$

Table. 2: Data group of Duane model LSE

serial	$N(t_i)$	t_i	$\ln t_i$	$(\ln t_i)^2$	$\theta_\Sigma(t_i)$	$\ln\theta_\Sigma(t_i)$	$\ln\theta_\Sigma(t_i)\ln t_i$
1	20	6	1.7918	3.2105	0.3	-1.204	-2.15733
2	30	12	2.4850	6.1752	0.4	-0.9163	-2.27701
3	36	18	2.8904	8.3544	0.5	-0.6931	-2.00334
4	40	24	3.1781	10.1003	0.6	-0.5108	-1.62337
5	50	30	3.4012	11.5682	0.6	-0.5108	-1.73733
6	62	36	3.5835	12.8415	0.5806	-0.5437	-1.94835
7	72	42	3.7377	13.9704	0.5833	-0.5391	-2.01499
8	86	48	3.8712	14.9862	0.5581	-0.5832	-2.25768
9	98	54	3.9890	15.9121	0.551	-0.596	-2.37744
10	114	60	4.0943	16.7633	0.5263	-0.6419	-2.62813
11	120	66	4.1897	17.5536	0.55	-0.5978	-2.5046
12	126	72	4.2767	18.2902	0.5714	-0.5597	-2.39367
13	132	78	4.3567	18.9808	0.5909	-0.5261	-2.29206
14	136	84	4.4308	19.6320	0.6176	-0.4819	-2.1352
15	142	90	4.4999	20.2491	0.6338	-0.456	-2.05195
16	150	96	4.5643	20.8328	0.64	-0.4463	-2.03705
17	156	102	4.6250	21.3906	0.6538	-0.425	-1.96563
18	160	108	4.6821	21.9221	0.675	-0.393	-1.84007
19	166	114	4.7362	22.4316	0.6867	-0.3759	-1.78034
20	170	120	4.7875	22.9202	0.7059	-0.3483	-1.66749
21	172	126	4.8363	23.3898	0.7326	-0.3112	-1.50506
22	178	132	4.8828	23.8417	0.7416	-0.2989	-1.45947
23	186	138	4.9273	24.2783	0.7419	-0.2985	-1.4708
24	190	144	4.9698	24.6989	0.7579	-0.2772	-1.37763
sum			97.7873	414.2938		-12.5347	-47.506

LSE of instantaneous invalid rate can be estimated by Eq.(6):

$$\hat{\lambda}(t) = (1-\alpha)\hat{\lambda}_I t^{-\hat{\alpha}} = (1-0.2248) \times 4.2132 t^{-0.2248} = 3.2661 t^{-0.2248} \quad (0 \leqslant t \leqslant 144)$$

So by the end of test LSE of instantaneous invalid rate is:

$$\hat{\lambda}(t) = 3.2661 \times 144^{-0.2248} = 1.0686 \text{ (num./day)}$$

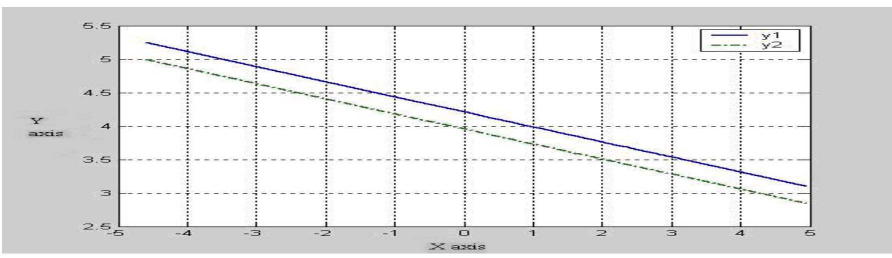

Fig. 2: An imitated and modified Duane chart

With intercept λ_I and slope α a modified Duane chart of $\ln\hat{\lambda}_\Sigma(t) \sim \ln t$ can be drawed on a dual logarithmic graph paper by Eq.(2) which is the same as the chart of $\ln\hat{\lambda}(t) \sim \ln t$ going straight up $(1-\alpha)$

unit. There is a chart of y_1 which sents $\ln \hat{\lambda}_{\Sigma}(t) \sim \ln t$ and a chart of y_2 which represents $\ln \hat{\lambda}(t) \sim \ln t$ in Fig.2.

4. Software reliability estimation

4.1. Estimation of Mean Time Between Failures (MTBF) $\hat{\theta}$

$$\hat{\theta} = 1/\hat{\lambda}(t) = 0.9358 day = 22.4592 \text{ hour}$$

4.2. Software reliability estimation $\hat{R}(t)$

When system software running all over the time within $(0, t]$ without changing in function , then software reliability estimation is: $\hat{R}(t) = \exp\{-\sum \lambda(t_i) t_i\}$ =99.86%

4.3. Estimation of software steady availability $\hat{A}$

When system software Mean Time To Repair (MTTR) is 5 minutes (abbr. M) , then:

$$\hat{A} = \hat{\theta}/(\hat{\theta} + M) \text{ [3] } = 99.63\%$$

For a certain satellite Tracking and Controlling S-band integrative system reliability prediction has been done at the stage of previous project argumentation, especially having a quantitative prediction of Mean Time Between Failures (MTBF) and software availability (A) using Goel-oknmoto NHPP model or MUSA model according to a software exploitation standard of GJB437-88. The prediction result is beyond the regulation building for this equipment manufacturing, so there is no need to introduce its details but give its result straightly:

$$\theta_{predict} = 20.12 hour$$

According to the regulation of system software Mean Time To Repair (MTTR) in software exploitation standard of GJB437-88, MTTR equals 5 minutes (0.083hour). So system software inherent availability can be estimated as: $A_{predict} = \theta_{predict} / (\theta_{predict} + M) = 20.12/(20.12 + 0.083) = 99.59\%$

That proves both estimation of MTBF and software availability are beyond their prediction in favor of software reliability growth test efficiency.

5. Conclusions

This test proves that using Duane model as a equivalent software reliability growth model has not only much effectiveness to engineering application but also can serve every software existence period. The model having a definite physical meaning is so intelligible that the control in the whole process of test from the start planning to middle tracing estimation till to the end is strict completely, so it has a great accuracy and creditability with a shorter test time and lower cost. Except this advantages Duane model also has some shortages such as that is unfit for agonic parameter estimation and MTBF term estimation or hypothetic check for model imitation and so on. Whereas Duane model still has a good perspective in engineering for its simpleness and clearness.

6. References

[1] Henley E J, Kumamoto H. Probability Risk Assessment, Reliability Engineering, *Design and Analysis.* New York: IEEE Press, Inc. ,1992

[2] Elsayed E A. *Reliability Engineering.* New York: Addison Wesley Longman, Inc. ,1996

[3] Kececioglu D. *Reliability Engineering Handbook*. Vol.1. New Jersey: PTR Prentice Hall, 1991.

[4] DODSON B, NOLAN D. *Reliability Engineering Handbook[M]*. New York: Marcel Dekker Inc. , 1996

[5] WASSERMAN. G S. *Reliability verification, Testingand Analysis in Engineering Design[M]*. New York: Marcel Dekker Inc. , 2003

[6] Chelson P. *Reliability Math Modeling Using the Digital Computer*. NASA/JPL TR32-1089, 1977.