HLSRGM based SPRT: MMLE

Dr. R. Satya Prasad[#1], S. Janardhana Rao[#2], Dr. G. Krishna Mohan[#3]

[1]Associate Professor, Dept. of Computer Science & Engg., Acharya Nagrjuna University, Guntur, Andhrapradesh, India.

[2]Team Lead, JKC, APSFKNW, ieg, India.

[3]Reader, Dept. of Computer Science, P.B.Siddhartha College, Vijayawada, Andhrapradesh, India.

***Abstract*—*Sequential Analysis of Statistical science could be adopted in order to decide upon the reliability / unreliability of the developed software very quickly. The procedure adopted for this is, Sequential Probability Ratio Test (SPRT). It is designed for continuous monitoring. The likelihood based SPRT proposed by Wald is very general and it can be used for many different probability distributions. The parameters are estimated using Modified Maximum Likelihood Estimation (MMLE). In the present paper, the HLSRGM (Half Logistic Software Reliability Growth model) is used on five sets of existing software reliability data and analyzed the results.*

***Keywords*—** *HLSRGM, MMLE, Decision lines, Software testing, Software failure data.*

I. Introduction

Wald's procedure is particularly relevant if the data is collected sequentially. Sequential Analysis is different from Classical Hypothesis Testing were the number of cases tested or collected is fixed at the beginning of the experiment. In Classical Hypothesis Testing the data collection is executed without analysis and consideration of the data. After all data is collected the analysis is done and conclusions are drawn. However, in Sequential Analysis every case is analyzed directly after being collected, the data collected upto that moment is then compared with certain threshold values, incorporating the new information obtained from the freshly collected case. This approach allows one to draw conclusions during the data collection, and a final conclusion can possibly be reached at a much earlier stage as is the case in Classical Hypothesis Testing. The advantages of Sequential Analysis are easy to see. As data collection can be terminated after fewer cases and decisions taken earlier, the savings in terms of human life and misery, and financial savings, might be considerable.

In the analysis of software failure data it is often deal with either Time Between Failures or failure count in a given time interval. If it is further assumed that the average number of recorded failures in a given time interval is directly proportional to the length of the interval and the random number of failure occurrences in the interval is explained by a Poisson process then we know that the probability equation of the stochastic process representing the failure occurrences is given by a Homogeneous Poisson Process with the expression

$$P\left[N(t)=n\right]=\frac{e^{-\lambda t}\left(\lambda t\right)^{n}}{n!}$$

(1.1)

Stieber (1997) observes that if classical testing strategies are used, the application of software reliability growth models may be difficult and reliability predictions can be misleading. However, he observes that statistical methods can be successfully applied to the failure data. He demonstrated his observation by applying the well-known sequential probability ratio test (SPRT) of Wald (1947) for a software failure data to detect unreliable software components and compare the reliability of different software versions. In this paper a popular SRGM HLSRGM is considered and adopted the principle of Stieber (1997) in detecting unreliable software components in order to accept or reject the developed software. The theory proposed by Stieber (1997) is presented in Section 2 for a ready reference. Extension of this theory to the SRGM – HLSRGM is presented in Section 3. Application of the decision rule to detect unreliable software with respect to the proposed SRGM is given in Section 4. Analysis of the application of the SPRT on five data sets and conclusions drawn are given in Section 5 and 6 respectively.

II. Wald's Sequential Test for a Poisson Process

The sequential probability ratio test (SPRT) was developed by A.Wald at Columbia University in 1943. Due to its usefulness in development work on military and naval equipment it was classified as 'Restricted' by the Espionage Act (Wald, 1947). A big advantage of sequential tests is that they require fewer observations (time) on the average than fixed sample size tests. SPRTs are widely used for statistical quality control in manufacturing processes. An SPRT for homogeneous Poisson processes is described below.

Let {N(t),t ≥0} be a homogeneous Poisson process with rate 'λ'. In our case, N(t)= number of failures up to time ' t' and ' λ' is the failure rate (failures per unit time). Suppose that a system is on test (for example a software system, where testing is done according to a usage profile and no faults are corrected) and that to estimate its failure rate ' λ'. We can not expect to estimate ' λ' precisely. But we want to reject the system with a high probability if our data suggest that the failure rate is larger than λ_1 and accept it with a high probability, if it's smaller than λ_0. As always with statistical tests, there is some risk to get the wrong answers. So we have to specify two (small) numbers 'α' and 'β', where 'α' is the probability of falsely rejecting the system. That is rejecting the system even if $\lambda \leq \lambda_0$. This is the "producer's" risk. β is the probability of falsely accepting the system .That is accepting the system even if $\lambda \geq \lambda_1$. This is the "consumer's" risk. With specified choices of λ_0 and λ_1 such that $0 < \lambda_0 < \lambda_1$, the probability of finding N(t) failures in the time span (0,t) with λ_1, λ_0 as the failure rates are respectively given by

$$Q_1 = \frac{e^{-\lambda_1 t}\left[\lambda_1 t\right]^{N(t)}}{N(t)!}$$

(2.1)

$$Q_0 = \frac{e^{-\lambda_0 t}\left[\lambda_0 t\right]^{N(t)}}{N(t)!}$$

(2.2)

The ratio $\frac{Q_1}{Q_0}$ at any time 't' is considered as a measure of deciding the truth towards λ_0 or λ_1, given a sequence of time instants say $t_1 < t_2 < t_3 < \ldots\ldots < t_K$ and the corresponding realizations $N(t_1), N(t_2), \ldots\ldots N(t_K)$ of N(t). Simplification of $\frac{Q_1}{Q_0}$ gives

$$\frac{Q_1}{Q_0} = \exp(\lambda_0 - \lambda_1)t + \left(\frac{\lambda_1}{\lambda_0}\right)^{N(t)}$$

The decision rule of SPRT is to decide in favor of λ_1, in favor of λ_0 or to continue by observing the number of failures at a later time than 't' according as $\frac{Q_1}{Q_0}$ is greater than or equal to a constant say A, less than or equal to a constant say B or in between the constants A and B. That is, we decide the given software product as unreliable, reliable or continue the test process with one more observation in failure data, according as

$$\frac{Q_1}{Q_0} \geq A$$

(2.3)

$$\frac{Q_1}{Q_0} \leq B$$

(2.4) $B < \frac{Q_1}{Q_0} < A$

(2.5)

The approximate values of the constants A and B are taken as $A \cong \frac{1-\beta}{\alpha}$, $B \cong \frac{\beta}{1-\alpha}$

Where 'α' and 'β' are the risk probabilities as defined earlier. A simplified version of the above decision processes is to reject the system as unreliable if N(t) falls for the first time above the line $N_U(t) = a.t + b_2$

(2.6)

To accept the system to be reliable if N(t) falls for the first time below the line

$$N_L(t) = a.t - b_1$$

(2.7)

To continue the test with one more observation on (t, N(t)) as the random graph of [t, N(t)] is between the two linear boundaries given by equations (2.6) and (2.7) where

$$a = \frac{\lambda_1 - \lambda_0}{\log\left(\frac{\lambda_1}{\lambda_0}\right)}$$

(2.8)

$$b_1 = \frac{\log\left[\frac{1-\alpha}{\beta}\right]}{\log\left(\frac{\lambda_1}{\lambda_0}\right)}$$

(2.9)

$$b_2 = \frac{\log\left[\frac{1-\beta}{\alpha}\right]}{\log\left(\frac{\lambda_1}{\lambda_0}\right)}$$

(2.10)

The parameters α, β, λ_0 and λ_1 can be chosen in several ways. One way suggested by Stieber (1997) is

$$\lambda_0 = \frac{\lambda.\log(q)}{q-1}, \qquad \lambda_1 = q\frac{\lambda.\log(q)}{q-1}$$

$$where\ q = \frac{\lambda_1}{\lambda_0}$$

If λ_0 and λ_1 are chosen in this way, the slope of NU (t) and NL (t) equals λ. The other two ways of choosing λ_0 and λ_1 are from past projects and from part of the data to compare the reliability of different functional areas.

III. HLSRGM

One simple class of finite failure NHPP model is the HLSRGM, assuming that the failure intensity is proportional to the number of faults remaining in the software describing an exponential failure curve. It has two parameters. Where, 'a' is the expected total number of faults in the code and 'b' is the shape factor defined as, the rate at which the failure rate decreases. The cumulative distribution function of the model is: $F(t)=\frac{(1-e^{-bt})}{(1+e^{-bt})}$ .The expected number of faults at time 't' is denoted by $m(t)=\frac{a(1-e^{-bt})}{(1+e^{-bt})}$, $a>0, b>0, t\geq 0$.

The MML estimation of parameters 'a' and 'b' for the considered model is explained in Satyaprasad et. al., (2011).

IV. Sequential Test for SRGMS

In Section II, for the Poisson process we know that the expected value of N(t) = λt called the average number of failures experienced in time 't' .This is also called the mean value function of the Poisson process. On the other hand if we consider a Poisson process with a general function m(t) as its mean value function the probability equation of a such a process is

$$P[N(t)=Y]=\frac{[m(t)]^y}{y!}.e^{-m(t)}, y=0,1,2,----$$

Depending on the forms of m(t) we get various Poisson processes called NHPP.

We may write

$$Q_1=\frac{e^{-m_1(t)}.[m_1(t)]^{N(t)}}{N(t)!}$$

$$Q_0=\frac{e^{-m_0(t)}.[m_0(t)]^{N(t)}}{N(t)!}$$

Where, $m_1(t)$, $m_0(t)$ are values of the mean value function at specified sets of its parameters indicating reliable software and unreliable software respectively. Let P_0 , P_1 be values of the NHPP at two specifications of b say b_0, b_1 where $(b_0<b_1)$ respectively. It can be shown that for our models $m(t)$ at b_1 is greater than that at b_0 . Symbolically $m_0(t)<m_1(t)$. Then the SPRT procedure is as follows:

Accept the system to be reliable if $\frac{Q_1}{Q_0}\leq B$

i.e., $\frac{e^{-m_1(t)}.[m_1(t)]^{N(t)}}{e^{-m_0(t)}.[m_0(t)]^{N(t)}}\leq B$

i.e., $N(t)\leq\frac{\log\left(\frac{\beta}{1-\alpha}\right)+m_1(t)-m_0(t)}{\log m_1(t)-\log m_0(t)}$

(4.1)

Decide the system to be unreliable and reject if $\frac{Q_1}{Q_0}\geq A$

i.e., $N(t)\geq\frac{\log\left(\frac{1-\beta}{\alpha}\right)+m_1(t)-m_0(t)}{\log m_1(t)-\log m_0(t)}$

(4.2)

Continue the test procedure as long as

$$\frac{\log\left(\frac{\beta}{1-\alpha}\right)+m_1(t)-m_0(t)}{\log m_1(t)-\log m_0(t)}<N(t)<\frac{\log\left(\frac{1-\beta}{\alpha}\right)+m_1(t)-m_0(t)}{\log m_1(t)-\log m_0(t)}$$

(4.3)

Substituting the appropriate expressions of the respective mean value function – m(t) of HLSRGM, we get the respective decision rules and are given in followings lines

Acceptance region:

$$N(t)\leq\frac{\log\left(\frac{\beta}{1-\alpha}\right)+\left[\frac{2a\left(e^{-b_0t}-e^{-b_1t}\right)}{1+e^{-b_0t}+e^{-b_1t}+e^{-t(b_0-b_1)}}\right]}{\log\left[\left(\frac{1-e^{-b_1t}}{1+e^{-b_1t}}\right)\left(\frac{1+e^{-b_0t}}{1-e^{-b_0t}}\right)\right]}$$

(4.4)

Rejection region:

$$N(t)\geq\frac{\log\left(\frac{1-\beta}{\alpha}\right)+\left[\frac{2a\left(e^{-b_0t}-e^{-b_1t}\right)}{1+e^{-b_0t}+e^{-b_1t}+e^{-t(b_0-b_1)}}\right]}{\log\left[\left(\frac{1-e^{-b_1t}}{1+e^{-b_1t}}\right)\left(\frac{1+e^{-b_0t}}{1-e^{-b_0t}}\right)\right]}$$

(4.5)

Continuation region:

$$\frac{\log\left(\frac{\beta}{1-\alpha}\right)+\left[\frac{2a\left(e^{-b_0t}-e^{-b_1t}\right)}{1+e^{-b_0t}+e^{-b_1t}+e^{-t(b_0-b_1)}}\right]}{\log\left[\left(\frac{1-e^{-b_1t}}{1+e^{-b_1t}}\right)\left(\frac{1+e^{-b_0t}}{1-e^{-b_0t}}\right)\right]}<N(t)<\frac{\log\left(\frac{1-\beta}{\alpha}\right)+\left[\frac{2a\left(e^{-b_0t}-e^{-b_1t}\right)}{1+e^{-b_0t}+e^{-b_1t}+e^{-t(b_0-b_1)}}\right]}{\log\left[\left(\frac{1-e^{-b_1t}}{1+e^{-b_1t}}\right)\left(\frac{1+e^{-b_0t}}{1-e^{-b_0t}}\right)\right]}$$

(4.6)

It may be noted that in the above model the decision rules are exclusively based on the strength of the sequential procedure (α,β) and the values of the respective mean value functions namely, $m_0(t)$, $m_1(t)$. If the mean value function is linear in 't' passing through origin, that is, m(t) = λt the decision rules become decision lines as described by Stieber (1997). In that sense equations (4.1), (4.2), (4.3) can be regarded as generalizations to the decision procedure of Stieber (1997). The applications of these results for live software failure data are presented with analysis in Section 5.

V. SPRT ANALYSIS OF LIVE DATA SETS

The developed SPRT methodology is for a software failure data which is of the form [t, N(t)]. Where, N(t) is the failure number of software system or its sub system in 't' units of time. In this section we evaluate the decision rules based on the considered mean value function for Five different data sets of the above form, borrowed from (Xie, 2002), (Pham, 2006) and (LYU,1996). The procedure adopted in estimating the parameters is a MMLE. Based on the estimates of the parameter 'b' in each mean value function, we have chosen the specifications of $b_0 = b - \delta$, $b_1 = b + \delta$ equidistant on either side of estimate of b obtained through a Data Set to apply SPRT such that b0 < b < b1. Assuming the value of $\delta = 0.001$, the choices are given in the following table.

TABLE 5.1: ESTIMATES OF A, B & SPECIFICATIONS OF B0, B1 FOR TIME DOMAIN

Data Set	Estimate of 'a'	Estimate of 'b'	b_0	b_1
XIE	31.62190	0.004021	0.003021	0.005021
NTDS	36.06863	0.005104	0.004104	0.006104
IBM	17.38641	0.006709	0.005709	0.007709
LYU	32.53707	0.018925	0.017925	0.019925
S2	38.04483	0.005231	0.004231	0.006231

Using the selected b_0 , b_1 and subsequently the $m_0(t), m_1(t)$ for the model, it is calculated the decision rules given by Equations 4.4 and 4.5, sequentially at each 't' of the data sets taking the strength (α, β) as (0.05, 0.2). These are presented for the model in Table 5.2. The following consolidated table reveals the iterations required to come to a decision about the software of each Data Set.

TABLE 5.2: SPRT ANALYSIS FOR 5 DATA SETS OF TIME DOMAIN DATA

Data Set	T	N(t)	Acceptance region (≤)	Rejection Region (≥)	Decision
Xie	30.02	1	-1.342698	7.201955	*Reject*
	31.46	2	-1.267039	7.279608	
	53.93	3	-0.166009	8.423895	
	55.29	4	-0.104056	8.489201	
	58.72	5	0.049878	8.651966	
	71.92	6	0.611889	9.252971	
	77.07	7	0.818353	9.476814	
	80.9	8	0.967340	9.639516	
	101.9	9	1.717214	10.476656	
	114.87	10	2.125902	10.949528	
	115.34	11	2.139960	10.966060	
NTDS	9	1	-3.130824	7.782102	*Reject*
	21	2	-2.143404	8.786362	
	32	3	-1.307213	9.649855	
	36	4	-1.018950	9.950868	
	43	5	-0.534218	10.461547	
	45	6	-0.400259	10.603770	
	50	7	-0.074025	10.952326	
	58	8	0.422698	11.489711	
	63	9	0.717732	11.813266	
	70	10	1.111369	12.250869	
	71	11	1.165786	12.311952	
	77	12	1.482922	12.671122	
	78	13	1.534234	12.729779	
IBM	10	1	-4.634744	9.795210	*Continue*
	19	2	-4.179723	10.278280	
	32	3	-3.590548	10.938841	
	43	4	-3.151740	11.466755	
	58	5	-2.636489	12.146432	
	70	6	-2.289491	12.661193	
	88	7	-1.870631	13.393503	
	103	8	-1.608845	13.974797	
	125	9	-1.357710	14.794471	
	150	10	-1.248933	15.701716	
	169	11	-1.282180	16.391182	
	199	12	-1.523307	17.512828	
	231	13	-2.019126	18.802015	
	256	14	-2.570507	19.913057	
	296	15	-3.748940	21.957983	
LYU	0.5	1	-14.577397	26.364610	*Continue*
	1.7	2	-14.218605	26.729848	
	4.5	3	-13.424252	27.566605	
	7.2	4	-12.713367	28.354719	
	10	5	-12.031234	29.154758	
	13	6	-11.360371	29.994898	
	14.8	7	-10.986704	30.491586	
	15.7	8	-10.807797	30.738071	
	17.1	9	-10.539827	31.119236	
	20.6	10	-9.923674	32.061675	
	24	11	-9.396380	32.965828	
	25.2	12	-9.226543	33.282939	
	26.1	13	-9.104625	33.520241	
	27.8	14	-8.886935	33.967480	
	29.2	15	-8.719864	34.335069	
	31.9	16	-8.428185	35.043064	
	35.1	17	-8.133359	35.882564	
	37.6	18	-7.940449	36.540419	
	39.6	19	-7.809238	37.068938	
	44.1	20	-7.587367	38.269052	
	47.6	21	-7.483326	39.216853	
	52.8	22	-7.436374	40.656549	
	60	23	-7.577969	42.732025	
	70.7	24	-8.219828	46.055192	
S2	3.183	1	-3.715630	7.472559	*Reject*
	6.883	2	-3.363946	7.826120	
	11.55	3	-2.932672	8.261731	
	16.383	4	-2.500304	8.700905	
	21.217	5	-2.082085	9.128292	
	27.633	6	-1.548537	9.677657	
	37.133	7	-0.802457	10.454823	
	47.3	8	-0.060144	11.240584	
	53.383	9	0.357157	11.688938	
	59.883	10	0.781529	12.150697	
	64.467	11	1.067734	12.465896	
	70.467	12	1.426382	12.865790	
	83.8	13	2.160625	13.705180	
	103.967	14	3.115518	14.855007	
	110.75	15	3.396889	15.211822	
	111.583	16	3.430116	15.254662	

From the Table 5.2, a decision of either to accept, reject the system or continue is reached much in advance of the last time instant of the data.

VI. CONCLUSION.

The above consolidated table of HLSRGM as exemplified for five Data Sets indicates that the model is performing well in arriving at a decision. The model has given a decision of rejection for 3 Data Sets i.e. Xie, NTDS and S2 at 11^{th} , 13^{th} and 16^{th} instances respectively and a decision of continue for 2 Data Sets i.e. IBM and LYU. Therefore, we may conclude that, applying SPRT on data sets we can come to an early conclusion of reliability / unreliability of software.

REFERENCES

[1] Lyu, M.R. (1996). "The hand book of software reliability engineering", McGrawHill & IEEE Computer Society press.
[2] Pham. H., (2006). "System software reliability", Springer.
[3] Satyaprasad (2007)."Half logistic Software reliability growth model "Ph.D Thesis of ANU, India.
[4] Satyaprasad, R., Ramchand, H. R. and Kantam, R. R. L., (May, 2011). "Software Reliability Measuring using Modified Maximum Likelihood Estimation and SPC", International Journal of Computer Applications, Volume 21– No.7.
[5] Stieber, H.A. (1997). "Statistical Quality Control: How To Detect Unreliable Software Components", Proceedings the 8th International Symposium on Software Reliability Engineering, 8-12.
[6] Wald. A., 1947. "Sequential Analysis", John Wiley and Son, Inc, New York.
[7] Xie, M., Goh. T.N., Ranjan.P. (2002). "Some effective control chart procedures for reliability monitoring" - Reliability engineering and System Safety 77 143 -150.