Trade-off Assessment between Model Adequacy and Complexity in Nonhomogeneous Poisson Process Software Reliability Growth Models incorporating a Changepoint

Vidhyashree Nagaraju[1], Shadow Pritchard[2], Priscila Silva[3], and Lance Fiondella[3]
[1]Department of Computer Science, Stonehill College, MA USA
[2]Boeing, MO, USA
[3]Electrical and Computer Engineering, University of Massachusetts Dartmouth, MA, USA
Email: vidhyashreenagaraju@gmail.com, {psilva4,lfiondella}@umassd.edu

***Abstract*—Traditional Non-homogeneous Poisson process (NHPP) software reliability growth models (SRGM) enable quantitative assessment of software systems based on failure data collected during testing. However,traditional models assume failure data is characterized by a single continuous curve without considering several factors that could significantly impact the fault detection rate. To address this, many studies have developed models to incorporate changepoints or imperfect debugging or both, yet existing studies develop models without careful consideration of the relationships between models and their complexity. Therefore, this paper presents a sequence of progressively more complex software reliability growth models according to their nesting relationships, including imperfect debugging and error generation as well as changepoint models with imperfect debugging and error generation. Model selection based on a novel multi-criteria approach is demonstrated. Our results indicate that the most complex models are not recommended and that simpler models exhibit the desirable attributes of simplicity and visual goodness of fit as well as information theoretic and other measures of goodness of fit.**

***Index Terms*—Software reliability, software reliability growth model, non-homogeneous Poisson process, imperfect debugging, changepoint, error generation**

I. Introduction

Traditional non-homogeneous Poisson process (NHPP) software reliability growth models (SRGM) [1] offer methods to quantitatively assess software systems from failure data, including inferences [2] such as the number of remaining faults, failure intensity, mean time to failure, optimal release time as well as reliability. Despite these practical applications, traditional models assume failure data is best characterized by a single continuous smooth curve, yet there are many factors that affect the fault detection rate during testing, including change in testing environment or testing strategy, resource allocation, and integration testing. Traditional models also assume that a fault is removed as soon as it is discovered without introducing additional faults and without affecting the fault detection rate. Changepoint models seek to address these limitations, but also introduce additional parameters, increasing model complexity. Straightforward model selection techniques are needed to balance model goodness of fit with complexity.

Page [3] introduced Changepoint (CP) models in 1954 and were first applied to software reliability by Zhao [4] in 1993. Various extensions of changepoint models have been proposed since then by incorporating additional information such as imperfect debugging (ID) [5], [6], probability of error generation [7], and testing-effort functions [8], [9]. Subsequent extensions considered ID and error generation [10], [11] along with fault reduction factor [12], [13] and testing-coverage [14] as well as the impact of rates of fault detection and correction [15]. Fewer studies address parameter estimation [16] and changepoint identification [17] or perform modeling specific to factors underlying changepoints such as an environment function [18] and heterogeneous failure intensity [19].

The review of past studies described above suggests that the software reliability research community has proliferated models without careful consideration of the relationship between these models, their complexity, or verified claims regarding the interpretation of parameters in terms of software engineering activities an artifacts. This paper seeks to address the issue of model relationship while balancing model complexity and the pragmatic concerns of software reliability practitioners, namely the tradeoff between model adequacy and complexity. Toward this end, we consider a sequence of progressively more complex software reliability growth models according to their nesting relationships, including imperfect debugging and error generation as well as changepoint models with imperfect debugging and error generation. Specifically, six models are presented, including the Goel-Okumoto (GO) model [20], GO with imperfect debugging (GO-ID) [5], GO with changepoint (GO-CP) [4], GO-ID with error generation (GO-ID-E) [11], GO-CP with imperfect debugging (GO-CP-ID) [6], and GO-CP-ID with error generation (GO-CP-ID-E) [21]. Numerical values of model parameters are estimated through stable and efficient expectation conditional maximization (ECM) [22] algorithms. Model selection based on a novel multi-criteria method [23] is used to assess the suitability of the alternative models. The results indicate that the most complex models are not recommended and that simpler models exhibit desirable attributes of simplicity, visual goodness of fit as well as information theoretic and other measures of goodness of fit. These results align with the needs of practitioners, as simple models that characterize the data well often predict future failures better than excessively complex models that overfit the available data at the expense of predictive accuracy.

The remainder of the paper is organized as follows: Section II describes a sequence of NHPP SRGM and their nesting relationships. Section III discusses model fitting algorithms, including the ECM algorithm and initial estimation strategy based on the EM algorithm. Section IV reviews measures to assess model fit and a multi-criteria model selection approach. Section V presents numerical examples, while Section VI provides a summary and identifies directions for future research.

II. Model Development

This section provides a clear development of progressively more complex models, including (i) NHPP software reliability growth models, (ii) NHPP SRGM with Imperfect Debugging (ID), and (iii) NHPP SRGM with Imperfect Debugging and Error Generation (ID-E). Next, (i) NHPP SRGM with Changepoint, (ii) NHPP SRGM with Changepoint and Imperfect Debugging (CP-ID), and (iii) NHPP SRGM with changepoint, imperfect debugging, and error generation (CP-ID-E) are described.

A. NHPP SRGM

The non-homogeneous Poisson process is a stochastic process [24] that counts the number of events observed as a function of time. In the context of software reliability, the NHPP counts the number of unique faults detected by time t. This counting process is characterized by a mean value function (MVF) $m(t)$, which characterizes the number of faults detected by time t and can assume a variety of forms. A general form of $m(t)$ can be obtained by solving [25],

$$\lambda(t) = \frac{\mathrm{d}m(t)}{\mathrm{d}t} = b(t)(a(t) - m(t)) \tag{1}$$

for $m(0) = 0$, where $b(t)$ is the time-dependent fault detection rate function and $a(t)$ is the time dependent fault introduction function. Specific forms of $b(t)$ and $a(t)$ produce different mean value functions.

Setting $b(t) = \frac{f(t)}{1-F(t)}$ and $a(t) = a$ in Equation (1) and solving produces the MVF

$$m(t) = a \times F(t), \tag{2}$$

where a denotes the expected number of unique faults that would be discovered with indefinite testing and $F(t)$ is the cumulative distribution function (CDF) of a continuous probability distribution characterizing the software fault detection process.

For example, the MVF of the Goel-Okumoto model (GO) [20]

$$m(t) = a(1 - e^{-bt}), \tag{3}$$

is obtained as the special case of Equation (1) where $b(t) = b$.

B. NHPP SRGM with Imperfect Debugging (GO-ID)

In general, a constant such as $a(t) = a$ in the NHPP model implies perfect debugging, which makes the unrealistic assumption that faults are identified and removed without introducing additional faults. To remove this assumption references [10], [25] explicitly modeled imperfect debugging, allowing for new faults to be introduced during the debugging phase at a constant rate β. If, $a(t)$ is a linear function of the expected number of faults detected by time t, then

$$a(t) = a + \beta(t)m(t) \tag{4}$$

Substituting Equation (4) into Equation (1) and solving under initial conditions $a(0) = a$, $\beta(t) = \beta$, and $m(0) = 0$ produces

$$m(t) = \frac{a}{1-\beta}\left(1 - (1 - F(t))^{(1-\beta)}\right) \tag{5}$$

For example, the GO model with imperfect debugging (GO-ID) is

$$m(t) = \frac{a}{1-\beta}\left(1 - e^{-(1-\beta)bt}\right) \tag{6}$$

1) NHPP SRGM with Imperfect Debugging and Error Generation (GO-ID-E) : To incorporate imperfect debugging [25] and the possibility of error generation [11] during fault removal, let

$$b(t) = p\frac{f(t)}{1-F(t)} \tag{7}$$

where p is the probability of perfect debugging. Substituting Equation (7) and Equation (4) into Equation (1) and solving under initial conditions $m(0) = 0$, the MVF is [10]

$$m(t) = \frac{a}{1-\beta}\left(1 - (1 - F(t))^{p(1-\beta)}\right) \tag{8}$$

For example, the GO model with imperfect debugging and error generation (GO-ID-E) is

$$m(t) = \frac{a}{1-\beta}\left(1 - e^{-bp(1-\beta)t}\right) \tag{9}$$

C. NHPP SRGM with Changepoint

This section presents a single changepoint model [19], where the fault detection process before and after the change point can be different. Let $b(t) = \frac{f(t)}{1-F(t)}$ denote the hazard function in Equation (1), where $f(t)$ is the probability density function and consider a vector of failure time data $\mathbf{T}$ with a single changepoint denoted as τ. Let $f_1(t)$, $F_1(t)$ and $f_2(t)$, $F_2(t)$ represent the PDF and CDF before and after the changepoint. These distributions can be the same (homogeneous) or different (heterogeneous). Let

$$b(t) = \begin{cases} \frac{f_1(t)}{1-F_1(t)}, & 0 \leq t \leq t_\tau \\ \frac{f_2(t)}{1-F_2(t)}, & t > t_\tau \end{cases} \tag{10}$$

Substituting this in Equation (1) and solving, the MVF is

$$m(t) = \begin{cases} m_1(t) = a \times F_1(t), & 0 \leq t \leq t_\tau \\ m_2(t - t_\tau) = (a - m_1(t_\tau))\left(1 - \frac{1-F_2(t)}{1-F_2(t_\tau)}\right), & t > t_\tau \end{cases} \tag{11}$$

where $m_1(t_\tau)$ and $m_2(t-t_\tau)$ respectively denote the mean value function before and after the changepoint and the term $(a-m_1(t_\tau))$ denotes the number of faults remaining at the changepoint.

For example, if both $F_1(t)$ and $F_2(t)$ follow exponential distributions [16], then

$$m(t) = \begin{cases} a(1-e^{-b_1 t}), & 0 \leq t \leq t_\tau \\ a\left(e^{-b_1 t_\tau}(1-e^{-b_2(t-t_\tau)})\right), & t > t_\tau \end{cases} \tag{12}$$

where b_1 and b_2 respectively represent a distinct fault detection rate according to the Goel-Okumoto model before and after the changepoint.

Hence, the mean value function of a homogeneous changepoint model (GO-CP) is

$$\begin{aligned} m(t) &= m_1(t) + m_2(t-t_\tau) \\ &= a(1-e^{-b_1 t_\tau - b_2(t-t_\tau)}) \end{aligned} \tag{13}$$

for $a > 0$, $b_1 > 0$, and $b_2 > 0$. Note that all faults detected in Equation (13) as t tends to infinity

$$\lim_{t\to\infty} m(t) = a \tag{14}$$

indicating perfect debugging.

1) NHPP SRGM with Changepoint and Imperfect Debugging (GO-CP-ID): To incorporate imperfect debugging into the changepoint model, assume the following fault introduction rate

$$\beta(t) = \begin{cases} \beta_1, & 0 \leq t \leq t_\tau \\ \beta_2, & t > t_\tau \end{cases} \tag{15}$$

where β_1 and β_2 are distinct fault detection rates before and after the changepoint.

Following steps similar to Sections II-B and II-B1, the MVF of the model incorporating changepoint and imperfect debugging is

$$m(t) = \begin{cases} \frac{a}{1-\beta_1} \times \left((F_1(t_\tau))^{(1-\beta_1)}\right), & 0 \leq t \leq t_\tau \\ \frac{a-m_1(t)}{1-\beta_2}\left(1-(\frac{1-F_2(t)}{1-F_2(t_\tau)})\right)^{1-\beta_2}, & t > t_\tau \end{cases} \tag{16}$$

For example, if the Goel-Okumoto model characterizes the data before and after the changepoint, then Goel-Okumoto model with changepoint and imperfect debugging is

$$m(t) = \begin{cases} \frac{a}{1-\beta_1} \times \left(1-e^{-(1-\beta_1)b_1 t}\right), & 0 \leq t \leq t_\tau \\ \frac{a}{1-\beta_2}\left(1-e^{(1-\beta_1)b_1 t_\tau - (1-\beta_2)b_2(t-t_\tau)}\right) & \\ +\frac{m_1(t_\tau)(\beta_1-\beta_2)}{1-\beta_2}, & t > t_\tau \end{cases} \tag{17}$$

2) NHPP SRGM with changepoint, imperfect debugging, and error generation (GO-CP-ID-E): To incorporate imperfect debugging and error generation into the changepoint model, let error generation be characterized as follows

$$p = \begin{cases} p_1, & 0 \leq t \leq t_\tau \\ p_2, & t > t_\tau \end{cases} \tag{18}$$

where p_1 and p_2 are the probability of perfect debugging before and after the changepoint.

Following steps similar to Sections II-B and II-B1, the MVF of model incorporating changepoint, imperfect debugging, and error generation is

$$m(t) = \begin{cases} \frac{a}{(1-\beta_1)} \times \left(1-(1-F_1(t_\tau))^{(1-\beta_1)p_1}\right), & 0 \leq t \leq t_\tau \\ \frac{a-m_1(t)}{(1-\beta_2)}\left(1-\left(\frac{1-F_2(t)}{1-F_2(t_\tau)}\right)^{(1-\beta_2)p_2}\right), & t > t_\tau \end{cases} \tag{19}$$

For example, if the Goel-Okumoto model to characterize the data before and after the changepoint, then Goel-Okumoto model with changepoint, imperfect debugging, and error generation (GO-CP-ID-E) is

$$m(t) = \begin{cases} \frac{a}{1-\beta_1} \times \left(1-e^{-p_1(1-\beta_1)b_1 t}\right), & 0 \leq t \leq t_\tau \\ \frac{a}{1-\beta_2}\left(1-e^{-p_1(1-\beta_1)b_1 t_\tau - (1-\beta_2)p_2 b_2(t-t_\tau)}\right) & \\ +\frac{m_1(t_\tau)(\beta_1-\beta_2)}{1-\beta_2}, & t > t_\tau \end{cases} \tag{20}$$

III. Parameter estimation

This section describes techniques to estimate the parameters of a model, including maximum likelihood estimation and the expectation conditional maximization algorithm as well as a method to estimate initial parameter values.

A. Maximum likelihood estimation

Maximum likelihood estimation maximizes the likelihood function, also known as the joint distribution of the failure data.

Let $\mathbf{T} = \langle t_1, t_2, \ldots, t_n \rangle$ denote a vector of individual failure times possessing density function $f(t_i; \Theta)$. The log-likelihood function is

$$LL(t_i; \Theta) = -m(t_n) + \sum_{i=1}^{n} \log\left[\lambda(t_i)\right], \tag{21}$$

where Θ is the vector of model parameters and $\lambda(t_i)$ is the instantaneous failure rate at time t_i. The MLE is found by numerically solving the following system of simultaneous equations

$$\frac{\partial LL(\Theta)}{\partial \Theta} = \mathbf{0} \tag{22}$$

with an algorithm such as the Newton–Raphson method [26]. To identify the location of the changepoint, Equation (21) must be solved to identify the failure τ that maximizes the likelihood. Since there is no closed form solution for t_τ, the changepoint is identified by maximizing the likelihood for each value of $\tau \in (2, (n-1))$ [17].

The reduced log-likelihood expectation conditional maximization [27] algorithm is used to identify the maximum likelihood estimates of the models proposed in this paper. The ECM algorithm [22] simplifies computation by dividing a single M-step of the EM algorithm [28] into ν conditional-maximization (CM) steps, where ν is number of model parameters. Expectation-Maximization [29] method is utilized to derive initial parameter estimates.

IV. Goodness of Fit Measures and Model Selection

This section summarizes goodness of fit measures to assess how well a model characterizes a failure data set as well as a model selection method based on these multiple goodness-of-fit measures.

A. Akaike Information Criterion

The Akaike information criterion [30] is a information theoretic measure of a model's goodness of fit. The AIC quantifies the tradeoff between model precision and complexity. The AIC of model i is a function of the maximized log-likelihood and the number of model parameters (ν).

$$AIC_i = 2\nu - 2LL(t_i;\hat{\Theta}) \tag{23}$$

The term 2ν in Equation (23) is a penalty function, which grows linearly with the number of parameters, while $LL(t_i;\hat{\Theta})$ evaluates the log-likelihood function of failure data t_i at the maximum likelihood estimate. Model j is preferred over model i if $AIC_{i,j} = AIC_i - AIC_j > 2.0$ [31].

B. Bayesian Information Criterion

The Bayesian information criterion of model i is a function of the maximized log-likelihood, number of model parameters ν, and the sample size n.

$$BIC_i = -2LL(t_i;\hat{\Theta}) + \nu\log(n) \tag{24}$$

The penalty term of the BIC is proportional to the number of parameters ν multiplied by the logarithm of the sample size n.

C. Sum of Squares Error (SSE)

The sum of squares error for failure times data is

$$SSE = \sum_{i=1}^{n}(\widehat{m}(t_i) - i)^2 \tag{25}$$

where $\widehat{m}(t_i)$ is estimated number of cumulative faults detected by time t_i according to the fitted model and i is the actual number of faults detected.

D. Root Mean Square Error (RMSE)

The root mean square error for failure times data is

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\widehat{m}(t_i) - i)^2} \tag{26}$$

E. Bias

The bias of model i is the sum of the deviations of the model estimates from the observed data. The bias for failure times data is

$$Bias_i = \frac{1}{n}\sum_{i=1}^{n}(\widehat{m}(t_i) - i) \tag{27}$$

F. Variance

The variance of model i for failure times data is

$$Variance_i = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\widehat{m}(t_i) - i - Bias)^2} \tag{28}$$

G. Root Mean Square Prediction Error (RMSPE)

The root mean square prediction error of model i for failure times data is

$$RMSPE_i = \sqrt{Variance + Bias^2} \tag{29}$$

H. Model Selection with Multiple Goodness of fit Measures

This section presents a simple model selection method [32] based on the critic method [33] to select a model given multiple goodness of fit measures.

Given n models and m measures, Let $f_{i,j}$ be the jth measure for the ith model. Each measure is assigned a normalized score according to

$$x_{i,j} = \frac{f_{i,j} - f_j^+}{f_j^- - f_j^+} \tag{30}$$

where f_j^+ and f_j^- respectively denote the best and worst values of a measure j across all models. Thus, $x_{i,j}$ indicates how close the jth measure of model i is to the ideal. One method to select a model is to compute the median and mean of each model's normalized scores and recommend the model with the highest value. Alternatively, model selection can be based on the average of each model's normalized scores.

V. Illustrations

This section applies the GO, GO-ID, GO-CP, GO-ID-E, GO-CP-ID, and GO-CP-ID-E models to the SYS1 data set [2], consisting of 136 failures over $88,682$ units of testing. The critic method is subsequently computed and the models recommended by the mean and median identified. The advantages and disadvantages of the mean and median are also discussed.

A. Visual and trend test assessment of alternative model fits

Figure 1 shows the fault detection process of the SYS1 dataset as well as plots of the mean value functions for each of the six models presented in Section II. The dashed vertical line at times $t_{16} = 1,056$ corresponds to the best location of the single changepoint in the GO-CP model, while the solid vertical line at time $t_{94} = 36,799$ indicates the best location of the single changepoint for the GO-CP-ID, and GO-CP-ID-E models. Of the five models considered, Figure 1 indicates that the GO-CP-ID model tracks the SYS1 data most closely. All other models underestimate the data prior to around $30,000$ units of testing and overestimate after that with the exception of the GO-CP-ID-E model, which grossly underestimates the number of faults beyond that point.

Figure 2 provides another view of the changepoints identified in Figure 1, showing the Laplace trend test [34] of the SYS1 data. Like Figure 1, the dashed vertical line corresponds to location of the changepoint in the GO-CP model, while the

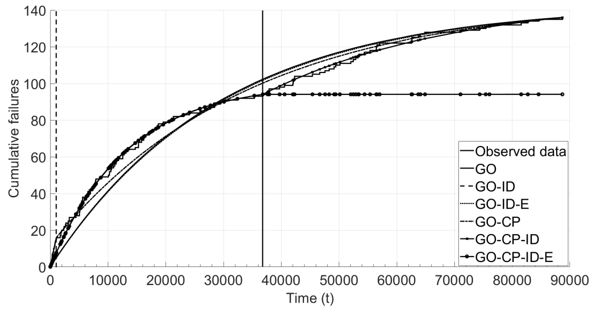

Fig. 1: Model fit

solid vertical line at time indicates the location of the change-point for the GO-CP-ID, and GO-CP-ID-E models. Since a decreasing trend indicates reliability growth, the location of the changepoints occur at times when reliability growth occurs and there is a decrease in the fault detection rate, which leads to the visible change point in Figure 1. The key observation from this example is that the most complex model may not necessarily perform best, even if simpler nested model exists.

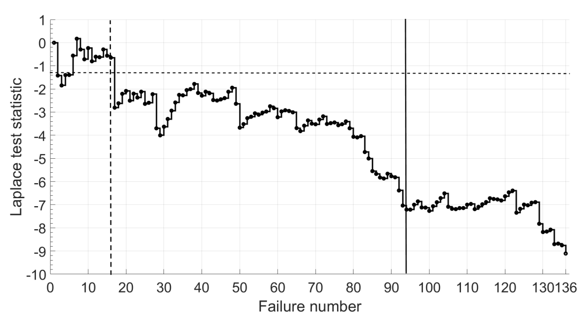

Fig. 2: Laplace trend test of SYS1 data (vertical lines denote best location of a single changepoint for alternative models)

B. Model comparison using multiple goodness-of-fit measures

This section compares models according to multiple goodness-of-fit measures and then applies the critic method to inform model selection.

Table I lists each model, number of unique parameters (ν), maximum log-likelihood (LLF), and corresponding goodness of fit measures described in Section IV. The values of models that achieve the best performance with respect to each measure are indicated in bold for clarity. Table I indicates that no model performs best with respect to all measures of goodness of fit. Thus, model selection is not straight forward. For example, both AIC and BIC recommend the simpler GO-CP model, while the LLF suggests the most complex GO-CP-ID-E. The remaining four of eight measures recommend the GO-CP-ID model with a LLF value very close to the GO-CP-ID-E. Given the disagreement between goodness of fit measures, no model is the clear winner and a more detailed approach that does not simply rank models according to the ranks of the measures is needed in order to preserve objectivity.

Table II reports the normalized goodness-of-fit values for each model presented according to Equation 30 with the values reported in Table I as well as the resulting mean and median for each model. The mean of the critic values recommends the GO-CP and GO-CP-ID models as two best all around fits, whereas the median of the critic values recommends the GO-CP-ID and GO-CP models as the top two choices. The median is robust to outliers, in which case the GO-CP-ID model may be preferred. However, the mean incorporates all of the goodness of fit measures into a single value, in which case the simpler GO-CP model may be preferred. Ultimately, the burden of model selection rests with the user. Tools that allow a user to specify which measures to include in the critic method as well as their weights [35] will simplify the model selection process based on that user's needs.

VI. Conclusions and Future Research

This paper presented a sequence of progressively more complex software reliability growth models, including imperfect debugging and error generation as well as changepoint models with imperfect debugging and error generation. Model selection based on a multi-criteria approach was applied to the SYS1 data set. The results indicated that the most complex models were not recommended and that simpler models exhibited desirable attributes of simplicity, visual goodness of fit as well as information theoretic and other measures of goodness of fit. These results align with the needs of practitioners, as simple models that characterize the data well often predict future failures better than complex models that overfit the available data at the expense of predictive accuracy.

Possible directions for future research include: (i) incorporating additional measures into the critic method and (ii) machine learning approaches to select models that predict well.

Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant Number 1749635. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

[1] W. Farr and O. Smith, "Statistical modeling and estimation of reliability functions for software (SMERFS) users guide," Naval Surface Warfare Center, Dahlgren, VA, Tech. Rep. NAVSWC TR-84-373, Rev. 2, 1984.

[2] M. Lyu, Ed., *Handbook of Software Reliability Engineering*. New York, NY: McGraw-Hill, 1996.

[3] E. Page, "Continuous inspection schemes," *Biometrika*, vol. 41, no. 1/2, pp. 100–115, 1954.

[4] M. Zhao, "Change-point problems in software and hardware reliability," *Communications in Statistics-Theory and Methods*, vol. 22, no. 3, pp. 757–768, 1993.

[5] M. Ohba and X. Chou, "Does imperfect debugging affect software reliability growth?" in *International Conference on Software engineering*, 1989, pp. 237–244.

[6] H. Shyur, "A stochastic software reliability model with imperfect-debugging and change-point," *Journal of Systems and Software*, vol. 66, no. 2, pp. 135–141, 2003.

TABLE I: Goodness of fit measures on SYS1 data

Model Name	ν	LLF	AIC	BIC	SSE	RMSE	Bias	Variance	RMSPE
GO	2	-974.807	1953.61	1959.44	9346.13	8.29	-4.34	7.06	5.09
GO-ID	3	-974.807	1955.61	1964.35	9346.13	8.29	-4.34	7.06	5.09
GO-ID-E	4	-974.807	1957.61	1969.26	9645.56	8.42	-4.70	6.99	5.39
GO-CP	3	-966.157	**1938.31**	**1947.05**	3169.07	4.83	-1.89	4.44	2.83
GO-CP-ID	5	-966.098	1942.20	1956.76	**951.57**	**2.65**	-0.96	**2.46**	**1.84**
GO-CP-ID-E	7	$\mathbf{-966.097}$	1946.19	1966.58	26320.90	13.91	$\mathbf{-7.55}$	11.69	8.28

TABLE II: Normalized measures on SYS1 data with mean and median critic values

	GO	GO-ID	GO-ID-E	GO-CP	GO-CP-ID	GO-CP-ID-E
LLF	0.000	0.000	0.000	0.993	0.999	1.000
AIC	0.207	0.104	0.000	1.000	0.799	0.592
BIC	0.442	0.221	0.000	1.000	0.563	0.121
SSE	0.669	0.669	0.657	0.913	1.000	0.000
RMSE	0.499	0.499	0.488	0.806	1.000	0.000
Bias	0.513	0.513	0.568	0.141	0.000	1.000
Variance	0.502	0.502	0.509	0.786	1.000	0.000
RMSPE	0.495	0.495	0.449	0.846	1.000	0.000
Mean	0.481	0.445	0.408	**0.832**	0.803	0.301
Median	0.499	0.499	0.488	0.913	**0.999**	0.000

[7] S. Yamada, K. Tokuno, and S. Osaki, "Imperfect debugging models with fault introduction rate for software reliability assessment," *International Journal of Systems Science*, vol. 23, no. 12, pp. 2241–2252, 1992.

[8] C. Huang, "Performance analysis of software reliability growth models with testing-effort and change-point," *Journal of Systems and Software*, vol. 76, no. 2, pp. 181–194, 2005.

[9] C. Lin, C. Huang, and J. Chang, "Integrating generalized Weibull-type testing-effort function and multiple change-points into software reliability growth models," in *IEEE Asia-Pacific Software Engineering Conference*, 2005.

[10] P. Zeephongsekul, "Reliability growth of a software model under imperfect debugging and generation of errors," *Microelectronics Reliability*, vol. 36, no. 10, pp. 1475–1482, 1996.

[11] P. Kapur, H. Pham, S. Anand, and K. Yadav, "A unified approach for developing software reliability growth models in the presence of imperfect debugging and error generation," *IEEE Transactions on Reliability*, vol. 60, no. 1, pp. 331–340, 2011.

[12] S. Chatterjee and A. Shukla, "Modeling and analysis of software fault detection and correction process through weibull-type fault reduction factor, change point and imperfect debugging," *Springer Arabian Journal for Science and Engineering*, vol. 41, no. 12, pp. 5009–5025, 2016.

[13] ——, "An ideal software release policy for an improved software reliability growth model incorporating imperfect debugging with fault removal efficiency and change point," *Asia-Pacific Journal of Operational Research*, vol. 34, no. 03, p. 1740017, 2017.

[14] ——, "A unified approach of testing coverage-based software reliability growth modelling with fault detection probability, imperfect debugging, and change point," *Journal of Software: Evolution and Process*, vol. 31, no. 3, p. e2150, 2019.

[15] R. Peng, Y. Li, W. Zhang, and Q. Hu, "Testing effort dependent software reliability model for imperfect debugging process considering both detection and correction," *Elsevier Reliability Engineering & System Safety*, vol. 126, pp. 37–43, 2014.

[16] Y. Chang, "Estimation of parameters for nonhomogeneous Poisson process: Software reliability with change-point model," *Communications in Statistics-Simulation and Comp.*, vol. 30, no. 3, pp. 623–635, 2001.

[17] F. Zou, "A change-point perspective on the software failure process," *Software Testing, Verification and Reliability*, vol. 13, no. 2, pp. 85–93, 2003.

[18] J. Zhao, H.-W. Liu, G. Cui, and X.-Z. Yang, "Software reliability growth model with change-point and environmental function," *Journal of Systems and Software*, vol. 79, no. 11, pp. 1578–1587, 2006.

[19] V. Nagaraju, L. Fiondella, and T. Wandji, "A heterogeneous single changepoint software reliability growth model framework," *Software Testing, Verification and Reliability*, vol. 29, no. 8, p. e1717, 2019.

[20] A. Goel and K. Okumoto, "Time-dependent error-detection rate model for software reliability and other performance measures," *IEEE Transactions on Reliability,*, vol. 28, no. 3, pp. 206–211, 1979.

[21] S. Khurshid, A. Shrivastava, and J. Iqbal, "Effort based software reliability model with fault reduction factor, change point and imperfect debugging," *International Journal of Information Technology*, vol. 13, no. 1, pp. 331–340, 2021.

[22] P. Zeephongsekul, C. Jayasinghe, L. Fiondella, and V. Nagaraju, "Maximum-likelihood estimation of parameters of NHPP software reliability models using expectation conditional maximization algorithm," *IEEE Transactions on Reliability*, vol. 65, no. 3, pp. 1571–1583, 2016.

[23] V. Nagaraju, C. Jayasinghe, and L. Fiondella, "Optimal test activity allocation for covariate software reliability and security models," *Journal of Systems and Software*, p. 110643, 2020.

[24] S. Ross, *Introduction to Probability Models*, 8th ed. New York, NY: Academic Press, 2003.

[25] H. Pham, L. Nordmann, and Z. Zhang, "A general imperfect-software-debugging model with s-shaped fault-detection rate," *IEEE Transactions on reliability*, vol. 48, no. 2, pp. 169–175, 1999.

[26] R. Burden and J. Faires, *Numerical Analysis*, 8th ed. Belmont, CA: Brooks/Cole, 2004.

[27] V. Nagaraju, L. Fiondella, P. Zeephongsekul, C. Jayasinghe, and T. Wandji, "Performance optimized expectation conditional maximization algorithms for nonhomogeneous Poisson process software reliability models," *IEEE Transactions on Reliability*, vol. 66, no. 3, pp. 722–734, 2017.

[28] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society: Series B*, vol. 39, no. 1, pp. 1–38, 1977.

[29] H. Okamura, Y. Watanabe, and T. Dohi, "An iterative scheme for maximum likelihood estimation in software reliability modeling," in *IEEE International Symposium on Software Reliability Engineering*, Nov 2003, pp. 246–256.

[30] L. Fiondella and S. Gokhale, "Software reliability model with bathtub-shaped fault detection rate," in *Annual Reliability and Maintainability Symposium*, 2011, pp. Session 9D–2.

[31] Y. Sakamoto, M. Ishiguro, and G. Kitagawa, "Akaike information criterion statistics," *Dordrecht, The Netherlands: D. Reidel*, p. 81, 1986.

[32] V. Nagaraju, C. Jayasinghe, and L. Fiondella, "Optimal test activity allocation for covariate software reliability and security models," *Journal of Systems and Software*, p. 110643, 2020.

[33] D. Diakoulaki, G. Mavrotas, and L. Papayannakis, "Determining objective weights in multiple criteria problems: The critic method," *Computers and Operations Research*, vol. 22, no. 7, pp. 763–770, 1995.

[34] O. Gaudoin, "Optimal properties of the laplace trend test for soft-reliability models," *IEEE Transactions on Reliability*, vol. 41, no. 4, pp. 525–532, 1992.

[35] J. Aubertine, V. Nagaraju, and L. Fiondella, "The covariate software failure and reliability assessment tool (c-sfrat)," *International Conference on Reliability and Quality in Design*, pp. 21–26, 2021.