Journal of Universal Computer Science, vol. 24, no. 12 (2018), 1717-1730
submitted: 3/2/17, accepted: 15/11/18, appeared: 28/12/18 © *J.UCS*

A Logistic Fault-Dependent Detection Software Reliability Model

Hoang Pham
(Department of Industrial and Systems Engineering
Rutgers University
Piscataway, New Jersey 08854, USA
hoang84pham@gmail.com)

Abstract: In this paper, we present a logistic fault-dependent detection model where the dependent-rate of detected faults in the software can grow much faster from the beginning but grow slowly as the testing progresses until it reaches the maximum number of faults in the software. The explicit function of the expected number of software failures detected by time t, called mean value function, of the proposed model is derived. Model analysis is discussed based on normalized-rank Euclidean distance (RED) and other criteria to illustrate the goodness-of-fit criteria of proposed model and compare it to several existing NHPP models using a set of software failure data. The confidence interval for the parameter estimates of the proposed model is also presented. A numerical analysis based on a real data set of the 7 or higher magnitude earthquake in the United States to illustrate the goodness-of-fit of the proposed model and a recent logistic growth model is also discussed. The results show that the proposed model fit significantly better than all the existing software reliability growth models.

Keywords: Non-homogeneous Poisson process (NHPP), software reliability growth model, logistic fault-dependent detection, predictive power, predictive-ratio risk, normalized-rank Euclidean distance.
Categories: D.4.5

1 Introduction

Many existing software reliability models [1-32] based on the non-homogeneous Poisson process have been studied in the past several decades to assess the reliability of software systems and other reliability measures including the number of residual faults and failure density. Pham et al developed a 4-parameter logistic growth model where the rate of change of quantity function is directly proportional to its remaining quantity for growth by a time-dependent logistic function per quantity per unit time.

The underlying common assumption of many existing models is that the rate of change of the number of software faults is proportional to the remaining fault contents. In many situations, this may not be the case because the dependent-rate of detected faults in the software will likely grow much faster from the beginning and grow slowly as the test goes toward the end of the testing process until it reaches the maximum number of faults in the software. In this paper, we address this problem.

In section 2, we discuss a new logistic fault-dependent detection software reliability model. The explicit solution of the mean value function is also presented in section 2. Recent proposed normalized-rank Euclidean distance (RED) and other common criteria such as mean square error (MSE), predictive-ratio risk (PRR), and predictive power (PP) using for determining the model performance are briefly discussed in Section 3. Model analysis and results are discussed in Section 4 to illustrate the goodness-of-fit criteria of proposed model and compare it to several common existing models based on existing software failure data.

2 A Logistic Fault-dependent Detection Model

Many existing reliability models [1-32] assume that the rate of change of the number of software faults is mostly proportional to the remaining fault contents. Let m(t) denote the expected number of software failures detected by time t also called the mean value function.

In this paper, we consider that, beside there is a finite number of faults in the software, the detected faults in the software grow much faster from the beginning with respect to the time-dependent fault-detection function and continue to detect them as the test progresses but grow slowly until it reaches the maximum number of faults in the software. Thus, a generalized mean value function m(t) can be obtained by solving the following proposed differential equation (with the initial condition $m(0) \neq 0$):

$$\frac{dm(t)}{dt} = b(t)\, m(t) \left(1 - \frac{m(t)}{a}\right) \quad (1)$$

where the initial condition $m(0) \neq 0$ and

m(t) expected number of software failures detected by time t

a Maximum number of faults in the software

b(t) time-dependent fault detection rate per fault per unit of time.

In this study, we consider the following time-dependent fault detection rate

$$b(t) = \frac{b}{1+\beta e^{-bt}} \quad (2)$$

The solution of the expected number of software failures detected by time t of the differential equation in (1) considering the function b(t) in (2), can be obtained as follows:

$$m(t) = \frac{a}{1+d\left(\frac{1+\beta}{\beta+e^{bt}}\right)} \quad (3)$$

To make it easy, we named it as logistic fault-detection model. From eq. (3), m(t) = a / (1+d) as t=0, and m(t) = a as t goes to infinity. It is worth to note that although this new model, function m(t), is just slightly different from some existing models such as delayed s-shaped [Yamada, 83], inflection s-shaped [Ohba, 84], error-dependent fault detection model [4, 20], the results of this new model have shown a very interesting aspect in term of the model performance based on some criteria for given data sets as shown in section 4. Table 1 summarizes the mean value functions of proposed model and several existing NHPP models.

3 Model Selection Criteria

In this section we discuss briefly three common criteria such as MSE, PRR, and PP that will be used to compare the performance of those models as listed in Table 1. The mean square error (MSE) measures the deviation between the predicted values with the actual observation and is defined as:

$$MSE = \frac{\sum_{i=1}^{n}(\hat{m}(t_i)-y_i)^2}{n-k} \qquad (4)$$

where y_i is total number of failures observed at time t_i according to the actual data; $\hat{m}(t_i)$ is the estimated cumulative number of failures at time t_i for i =1, 2…, n; and n and k are the number of observations and number of parameters in the model, respectively.

The predictive-ratio risk (PRR) measures the distance of model estimates from the actual data against the model estimate, and is defined as [Pham, H. and Deng, C. 2003]:

$$PRR = \sum_{i=1}^{n}\left(\frac{\hat{m}(t_i)-y_i}{\hat{m}(t_i)}\right)^2 \qquad (5)$$

The predictive power (PP) measures the distance of model estimates from the actual data against the actual data, is as follows [Pham, 14]:

$$PP = \sum_{i=1}^{n}\left(\frac{\hat{m}(t_i)-y_i}{y_i}\right)^2 \qquad (6)$$

For all these three criteria – MSE, PRR, and PP – the smaller the value, the better the model fits, relative to other models run on the same data set.

It should be noted that there are more than a dozen of existing goodness-of-fit criteria. Obviously different criteria will likely lead to various, but different, impacts in measuring the software reliability and that no software reliability model is optimal for all contributing criteria. This makes the job of developers and practitioners much more difficult when they need to select the best model in order to use from among many existing software reliability growth models (SRGMs) based on a set of criteria. Pham [Pham, 14] recently proposed a *normalized-Rank Euclidean Distance criteria*, or RED criteria, to select the best model based on a set of contributing criteria. The RED criteria function is defined as follows:

Model	**m(t)**
Goel-Okumoto (G-O) [Goel, 79]	$m(t) = a(1 - e^{-bt})$
Delayed S-shaped [Yamada, 83]	$m(t) = a(1 - (1 + bt)e^{-bt})$
Inflection S-shaped [Ohba, 84]	$m(t) = \dfrac{a(1 - e^{-bt})}{1 + \beta e^{-bt}}$
Yamada Imperfect debugging [Yamada, 92]	$m(t) = a[1 - e^{-bt}]\left[1 - \frac{\alpha}{b}\right] + \alpha a t$
PNZ model [Pham, 99]	$m(t) = \frac{a}{1+\beta e^{-bt}}\left([1 - e^{-bt}]\left[1 - \frac{\alpha}{b}\right] + \alpha t\right)$
Pham-Zhang model [Pham, 03]	$m(t) = \dfrac{1}{1 + \beta e^{-bt}}\left((c + a)(1 - e^{-bt}) - \dfrac{a}{b - \alpha}(e^{-at} - e^{-bt})\right)$
Dependent-parameter model [Pham, 07]	$m(t) = \alpha(1 + \gamma t)(\gamma t + e^{-\gamma t} - 1)$
Dependent-parameter model with $m(t_0) \neq 0$ [Pham, 07]	$m(t) = m_0\left(\dfrac{\gamma t + 1}{\gamma t_0 + 1}\right)e^{-\gamma(t-t_0)} + \alpha(\gamma t + 1)\left[\gamma t - 1 + (1 - \gamma t_0)e^{-\gamma(t-t_0)}\right]$
Pham Inflexion model [Pham, 14]	$m(t) = N\left(1 - \dfrac{1}{\left(\frac{\beta + e^{bt}}{1 + \beta}\right)^{\frac{a}{b}}}\right)$
Vtub fault detection model [Pham, 14]	$m(t) = N\left(1 - \left(\dfrac{\beta}{\beta + a^{t^b} - 1}\right)^{\alpha}\right)$
Error-dependent fault detection model (new model) [4, 20]	$m(t) = \dfrac{Nb\left(1 - e^{-(b+aN)t}\right)}{\left(b + aNe^{-(b+aN)t}\right)}$
Logistic fault-detection model (new model)	$m(t) = \dfrac{a}{1 + d\left(\frac{1 + \beta}{\beta + e^{bt}}\right)}$

Table 1: Software reliability models

$$D_i = \sum_{j=1}^{d} \left\{ \left(\sqrt{\left[\sum_{k=1}^{2} \left(\frac{C_{ijk}}{\sum_{i=i}^{s} C_{ijk}} \right)^2 \right]} \right) w_j \right\} \qquad (7)$$

Where s = total number of models

d = total number of criteria

w_j = the weight of the j^{th} criteria for $j = 1, 2, ..., d$

$$k = \begin{cases} 1 & \text{represents criteria j value} \\ 2 & \text{represents criteria j ranking} \end{cases}$$

C_{ij1} = the ranking based on specified criterion of model i with respect to (w.r.t.) criteria j

C_{ij2} = criteria value of model i w.r.t. criteria j where i = 1, 2..., s and j = 1, 2..., d

Thus, the smaller the RED value, D_i, it represents the better rank as compare to higher RED value. In Section 4, we illustrate the use of RED criteria based on three criteria above such as MSE, PRR and PP to select the best software reliability model.

4 Model Analysis and Results

4.1 Model Results and Comparison based on Software Failure Data

A set of system test data was provided in [9, page 149] which is referred to as Phase 2 data set and is given in Table 2. In this data set the number of faults detected in each week of testing is found and the cumulative number of faults since the start of testing is recorded for each week. This data set provides the cumulative number of faults by each week up to 21 weeks.

Table 3 summarizes the result of parameter estimates of all the models from Table 1 using the least square estimation (LSE) technique and their criteria (MSE, PRR and PP) values. Using a normal approximation, we can obtain the 95% confidence intervals of the parameter estimates of the new model as follows:

CI for a: [41.924, 45.677]
CI for b: [0.270, 0.373]
CI for d: [7.019, 18.822]
CI for β: [0, 3.967]

The coordinates X, Y and Z as shown in Figure 1 represent the model number, MSE values, and the RED criteria values, respectively. In Table 4, all the MSE, PRR and PP values of proposed model (model 12) are all the lowest compared to all the models in Table 4. Using eq. (3) and the criteria values given in Table 3, we can obtain the normalized-rank Euclidean distance (RED) values and their corresponding ranking as shown in Table 4 for criteria weight w_1=0.25, w_2 = 0.50 and w_3=0.25. In Figure 2, the coordinates X, Y and Z represent the model number and the corresponding RED value and ranking, respectively, of the model. For example, (X = 12, Y = 0.01469, and Z = 1) indicates that model 12 in Table 4 is ranked first (the best!) where the RED value is 0.01469.

Based on these results, we can draw a conclusion that the new model (model 12) provides the best fit as shown in Table 4 and Figures 3 - 5 based on the Phase 2 data set. Figure 6 shows the estimated number of failures of the new model versus the data set.

The result, in general, shows very encouraging since it indicates that a consideration of a more realistic logistic fault-dependent detection function can be more than compensated by the ability of the model to better describe the debugging process of reliability modeling. Obviously, further work in broader validation of this conclusion is needed using other data sets as well as other criteria.

Week Index	Exposure time (Cum. System test hours)	Fault	Cum. fault
1	416	3	3
2	832	1	4
3	1248	0	4
4	1664	3	7
5	2080	2	9
6	2496	0	9
7	2912	1	10
8	3328	3	13
9	3744	4	17
10	4160	2	19
11	4576	4	23
12	4992	2	25
13	5408	5	30
14	5824	2	32
15	6240	4	36
16	6656	1	37
17	7072	2	39
18	7488	0	39
19	7904	0	39
20	8320	3	42
21	8736	1	43

Table 2: Phase 2 system test data [Pham, 06]

Model Name	**LSEs**	**MSE**	**PRR**	**PP**
G-O Model (model 1)	$\widehat{a} = 93.5106$ $\widehat{b} = 0.0278$	12.0275	0.4353	2.2451
Delayed S-shaped (model 2)	$\hat{a} = 62.3$ $\hat{b} = 2.8510^{-4}$	3.27	44.27	1.43
Inflection S-shaped (model 3)	$\hat{a} = 46.6$ $\hat{b} = 5.7810^{-4}$ $\hat{\beta} = 12.20$	1.87	5.94	0.90
Yamada imperfect debugging (model 4)	$\hat{a} = 1.5$ $\widehat{b} = 1.1110^{-3}$ $\hat{\alpha} = 3.810^{-3}$	4.98	4.30	0.81
PNZ model (model 5)	$\hat{a} = 45.99$ $\hat{b} = 6.010^{-4}$ $\hat{\alpha} = 0$ $\hat{\beta} = 13.24$	1.9964	5.2516	0.8971
Pham-Zhang model (model 6)	$\hat{a} = 0.06$ $\hat{b} = 6.010^{-4}$ $\hat{\alpha} = 1.010^{-4}$ $\hat{\beta} = 13.2$ $\hat{c} = 45.9$	2.12	6.79	0.95
Dependent parameter model (model 7)	$\hat{\alpha} = 3.010^{-6}$ $\hat{\gamma} = 0.49$	43.7343	435.2331	4.5431
Dependent parameter model with $m(t_0) \neq 0, t_0 \neq 0$ (model 8)	$\hat{a} = 890996$ $\hat{\gamma} = 1.210^{-6}$ $t_0 = 832$ $m_0 = 4$	24.79	1.14	0.73
Pham inflexion model (model 9)	$\widehat{N} = 45.8270$ $\hat{a} = 0.2961$ $\hat{b} = 0.2170$ $\hat{\beta} = 13.6298$	1.5108	3.1388	0.6800
Vtub uncertainty model (model 10)	$\widehat{N} = 43.25$ $\hat{a} = 2.662$ $\hat{b} = 0.196$ $\hat{\alpha} = 4.040$ $\hat{\beta} = 35.090$	1.80	2.06	0.77
Error-dependent detection model (model 11)	$\widehat{N} = 46.5534$ $\hat{a} = 0.0047$ $\hat{b} = 0.0185$	1.7718	1.0442	0.8813
Logistic fault-detection model (new model) (model 12)	$\hat{a} = 43.8005$ $\hat{b} = 0.32150$ $\hat{d} = 12.92043$ $\hat{\beta} = 1.55095$	0.9378	0.1713	0.1732

Table 3: Model Parameter Estimation and Comparison Criteria

Model / Criteria	MSE (Rank)	PRR (Rank)	PP (Rank)	RED Value (D_k)	Model Rank
1. G -O Model	12.0275 (10)	0.4353 (2)	2.2451 (11)	0.10801	8
2. Delayed S-shaped	3.27 (8)	44.27 (11)	1.43 (10)	0.14964	11
3. Inflection S-shaped	1.87 (5)	5.94 (9)	0.90 (8)	0.10437	7
4. Yamada imperfect debugging model	4.98 (9)	4.30 (7)	0.81 (5)	0.09740	5
5. PNZ model	1.9964 (6)	5.2516 (8)	0.8971 (7)	0.09835	6
6. Pham-Zhang model	2.12 (7)	6.79 (10)	0.95 (9)	0.12039	10
7. Dependent-parameter Model	43.7343 (12)	435.2331 (12)	4.5431 (12)	0.63372	12
8. Dependent-parameter model with $m(t_0) \neq 0$, $t_0 \neq 0$	24.79 (11)	1.14 (4)	0.73 (3)	0.11204	9
9. Pham inflexion model	1.5108 (2)	3.1388 (6)	0.68 (2)	0.05902	3
10.Vtub uncertainty model	1.80 (4)	2.06 (5)	0.77 (4)	0.06382	4
11. Error-dependent detection Model	1.7718 (3)	1.0442 (3)	0.8813 (6)	0.05402	2
12. Logistic fault-detection model (new model)	0.9378 (1)	0.1713 (1)	0.1732 (1)	0.01469	1

Table 4: Parameter Estimation and Model Comparison when w1= 0.25, w2=0.50, w3= 0.25

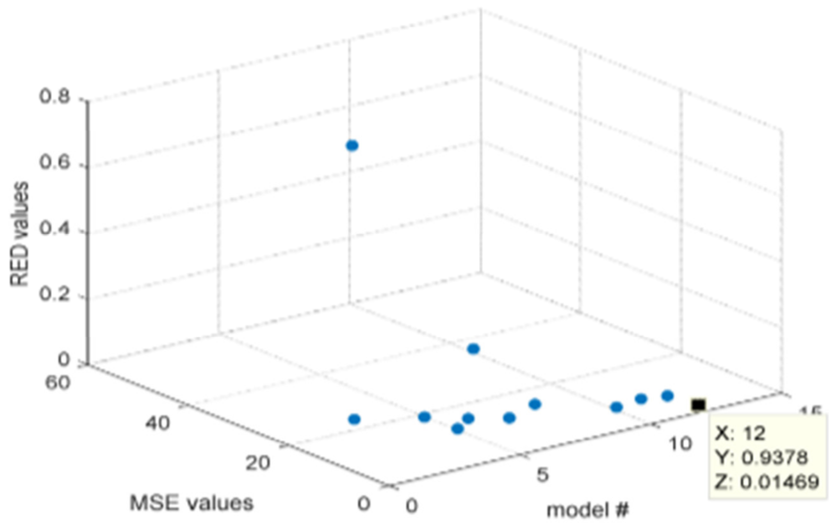

Figure 1: A plot (X,Y,Z) represents (model number, MSE value, normalized criteria value) based on Phase 2 Data set for w_1= 0.25, w_2=0.5, w_3= 0.25

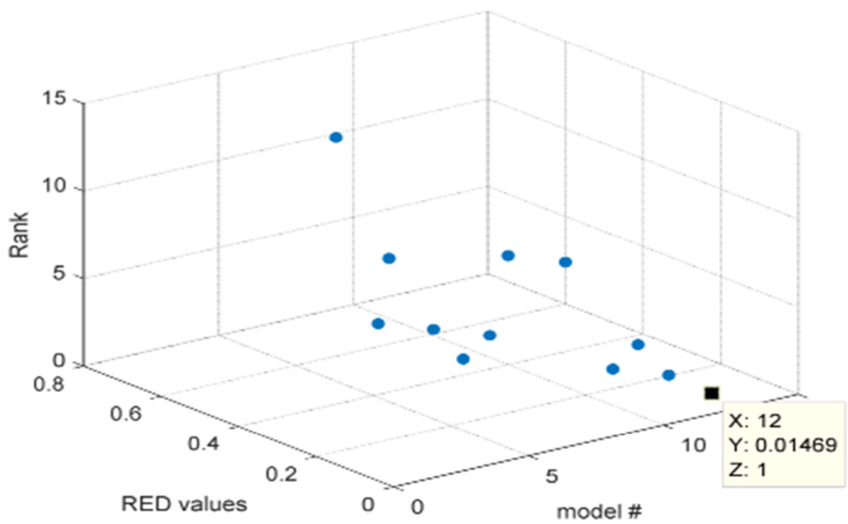

Figure 2: A plot (X,Y,Z) represents (model number, RED values, model ranking) based on Phase 2 Data set for w_1= 0.25, w_2=0.5, w_3= 0.25

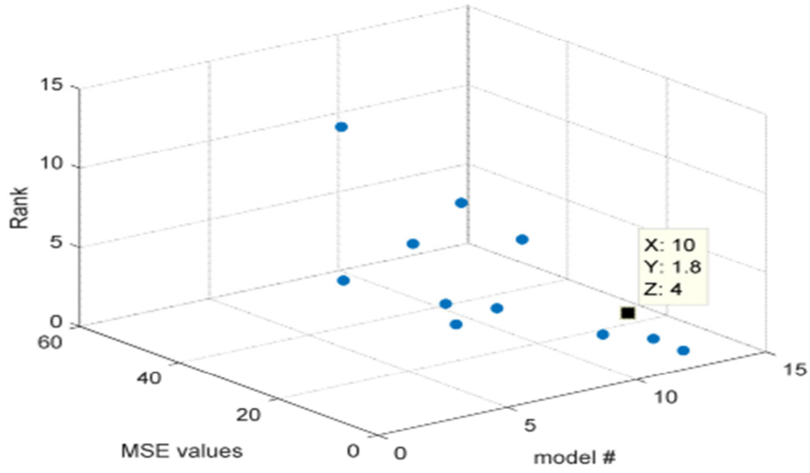

Figure 3: A plot (X,Y,Z) represents (model number, MSE value, model ranking) based on Phase 2 Data set for w_1= 0.25, w_2=0.5, w_3= 0.25

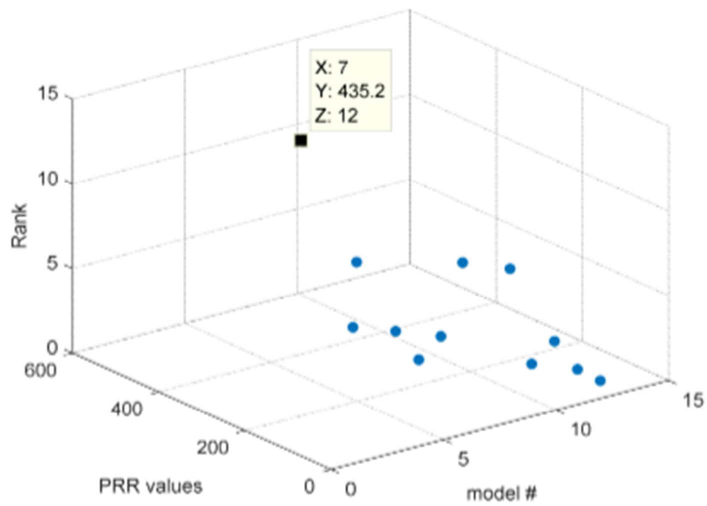

Figure 4: A plot (X,Y,Z) represents (model number, PRR value, model ranking) based on Phase 2 Data set for w_1= 0.25, w_2=0.5, w_3= 0.25

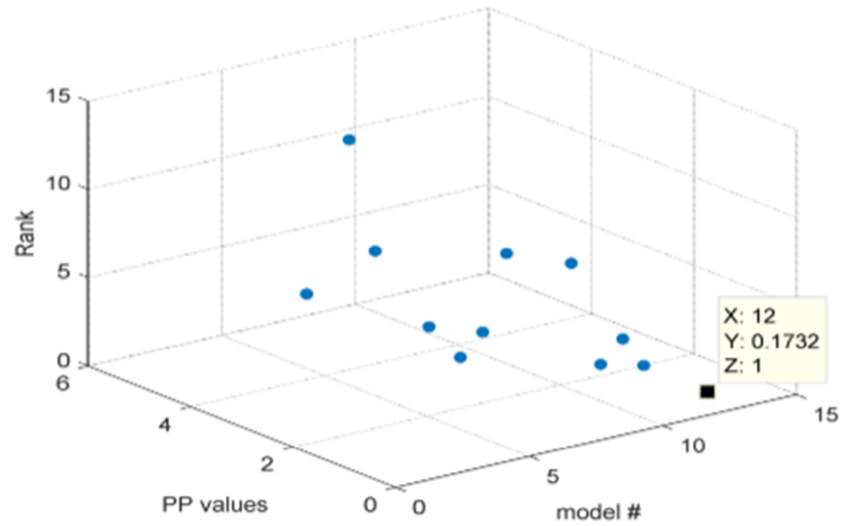

Figure 5: A plot (X,Y,Z) represents (model number, PP value, model ranking) based on Phase 2 Data set for w1= 0.25, w2=0.5, w3= 0.25

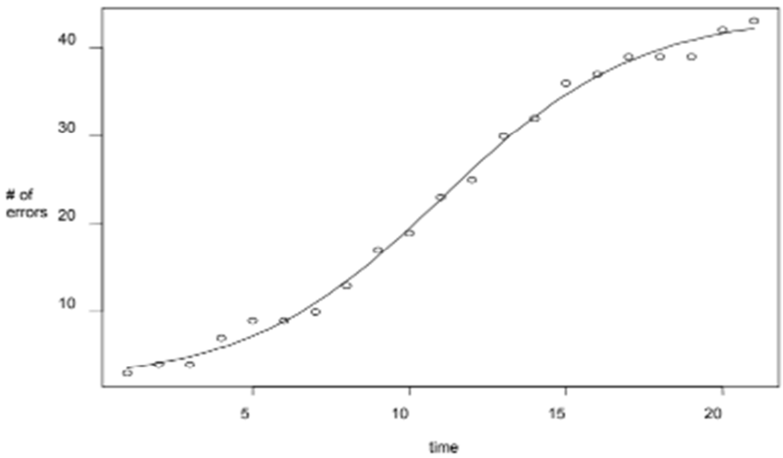

Figure 6: The expected number of failures of new model (model 12) versus time based on Phase 2 Data

4.2 Model Results and Comparison based on Earthquake Data

As we all know the cost of damages of any magnitude-7 or larger earthquake is definitely very significant and difficult to predict. [Pham, 14] recently use their model to assess and predict the number of earthquakes in the coming years based on the magnitude range 7.0-7.9 earthquake data in the United States for a period of 43 years from 1970 - 2012. We now use this same data set to illustrate the goodness-of-fit of our proposed model and compare the results to the logistic growth model in [Pham, 14]. The parameter estimates of the proposed model in equation (3) based on this earthquake data set [Pham, 14] are as follows:

$a = 39.9774$ $b = 0.0795$ $d = 68.3223$ $\beta = -0.9118$

Figure 7 shows the estimated number of earthquakes of the new model (model 12) versus the year index based on earthquake data set. The MSE and PRR values of the new model are 0.4702 and 1.1504, respectively. Although the PRR value of the new model is larger than the logistic model in [Pham, 14], the MSE value of the new model shows significantly smaller compared to logistic model [Pham, 14]. This is again also encouraging! It would be of interest to do further analysis of this new model applying to other applications for a broader validation of this study.

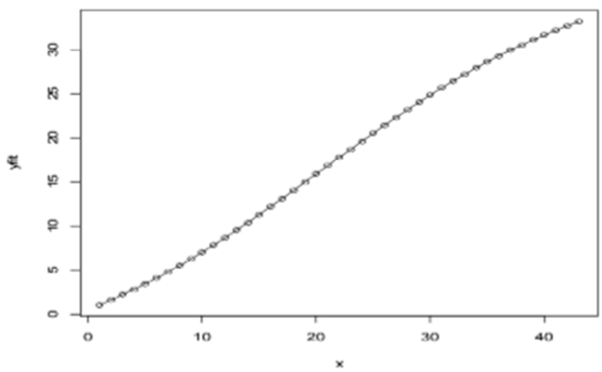

Figure 7: The expected number of earthquakes of the new model (model 12) versus the year index based on earthquake data

5 Conclusions

In this study, we discuss a new logistic fault-dependent detection model with considerations of time dependent-rate of detected faults in the software. The proposed expected number of software detected failures function is discussed. Numerical examples to illustrate the goodness-of-fit criteria of the new model and compare it to several existing NHPP models based on failure data sets are discussed. The confidence interval for the parameter estimates of the proposed model is also presented. The results show that the proposed model fit significantly better than all the existing software reliability growth models based on given criteria.

References

[Goel, 79] Goel, A. L., Okumoto, K.: Time-dependent fault-detection rate model for software and other performance measures, IEEE Transactions on Reliability, Vol. 28, 206-211, 1979.

[Inoue, 14] Inoue, S., Yamada, S.: A nonparametric Bootstrap method for software reliability assessment and release planning, Advances in Reliability, Vol. 1, 51-59, 2014.

[Kapur, 12] Kapur, P. K., Pham, H., Aggarwal, A. G. and Kaur, G.: Two-dimensional multi-release software reliability modeling and optimal release planning, IEEE Trans. on Reliability, Vol. 61, Number 3, 758-768, 2012.

[Kapur, 92] Kapur, P. K., Garg, R. B.: A software reliability growth model for an error-removal phenomenon, Software Engineering Journal, 291-294, July 1992.

[Kapur, 11] Kapur, P. K., Pham, H., Anand, S., Yadav, K.: A unified approach for developing software reliability growth models in the presence of imperfect debugging and error generation, IEEE Trans. on Reliability, Vol. 60, Number 1, 331-340, 2011.

[Li, 17,] Li, Q., Pham, H.: NHPP software reliability model considering the uncertainty of operating environments with imperfect debugging and testing coverage, Applied Mathematical Modelling, Vol. 51, Number 11, 68-85, 2017.

[Ohba, 84] Ohba M., Yamada, S.: S-shaped software reliability growth models, Proc. 4th Int. Conf. Reliability and Maintainability, 430-436, 1984.

[Ohba, 84] Ohba, M.: Inflexion S-shaped software reliability growth models, Stochastic Models in Reliability Theory, ed. S. Osaki and Y. Hatoyama, Springer-Verlag, p. 144-162 (1984)

[Pham, 06] Pham, H.: System Software Reliability, Springer, 2006.

[Pham, 93] Pham, H.: Software reliability assessment: imperfect debugging and multiple failure types in software development, EG&G-RAAM-10737; Idaho ational Engineering Laboratory, 1993.

[Pham, 96] Pham, H.: A software cost model with imperfect debugging, random life cycle and penalty cost, International Journal of Systems Science, Vol. 27, Number 5, 455-463, 1996.

[Pham, 03] Pham, H., Zhang, X.: NHPP software reliability and cost models with testing coverage, European Journal of Operational Research, Vol. 145, 443-454, 2003.

[Pham, 99] Pham, H., Nordmann, L., Zhang, X.: A general imperfect software debugging model with s-shaped fault detection rate, IEEE Transactions on Reliability, Vol. 48, Number 2, 169-175, 1999.

[Pham, 97] Pham, H., Zhang, X.: An NHPP software reliability model and its comparison, International Journal of Reliability, Quality and Safety Engineering, Vol. 4, Number 3, 269-282, 1997.

[Pham, 03] Pham, H., Deng, C.: Predictive-ratio risk criterion for selecting software reliability models, Proc. inth International Conf. on Reliability and Quality in Design, 2003.

[Pham, 07] Pham, H.: An imperfect-debugging fault-detection dependent-parameter software, International Journal of Automation and Computing, Vol. 4, Number 4, 325-328, 2007.

[Pham, 13] Pham, H.: A software reliability model with vtub-shaped fault-detection rate subject to operating environments, Proc. 19th ISSAT Int'l Conf. on Reliability and Quality in Design, Hawaii, 2013.

[Pham, 14] Pham, H.: A new software reliability model with Vtub-shaped fault-detection rate and the uncertainty of operating environments, Optimization, Vol. 63, 1481-1490, 2014.

[Pham, 13] Pham, H.: A software reliability model with vtub-shaped fault-detection rate subject to operating environments, Proc. 19th ISSAT International Conf. on Reliability and Quality in Design, Hawaii, 2013.

[Pham, 17] Pham, H.: A New Error Dependent Fault Detection Software Reliability Model based on Normalized-rank Euclidean Distance Criteria, Amity Journal of Interdisciplinary Research, vol.1, no. 1. (2017)

[Pham, 14] Pham, H., Pham, D. H., Pham, H.: A new mathematical logistic model and its applications, International Journal of Information and Management Sciences, Vol. 25, Number 2, 79-99, 2014.

[Pham, 14] Pham, H.: Log fault-detection rate and testing coverage software reliability models subject to random environments, Vietnam Journal of Computer Science, Vol.1, Number 1, 2014.

[Pham, 00] Pham, L., Pham, H.: Software reliability models with time-dependent hazard function based on Bayesian approach, IEEE Trans. on Systems, Man, and Cybernetics - Part A, Vol. 30, Number 1, 25-35, 2000.

[Pham, 16] Pham, H.: A generalized fault-detection software reliability model subject to random operating environments, Vietnam Journal of Computer Science, Vol.3, Number 3, 145-150, 2016.

[Sgarbossa, 10] Sgarbossa, F., Pham, H.: A cost analysis of systems subject to random field environments and reliability, IEEE Trans. on Systems, Man, and Cybernetics – Part C, Vol. 40, Number 4, 429-437, 2010.

[Tamura, 14] Tamura, Y., Yamada, S.: Reliability analysis based on AHP and software reliability models for big data on cloud computing, International Journal of Statistics - Theory and Applications, Vol. 1, 43-49, 2014.

[Yamada, 83] Yamada, S., Ohba, M., Osaki, S.: S-shaped reliability growth modeling for software fault detection, IEEE Trans. on Reliability, Vol.12, 475-484, 1983.

[Yamada, 85] Yamada, S., Osaki, S.: Software reliability growth modeling: models and applications, IEEE Trans. on Software Engineering, Vol.11, 1431-1437, 1985.

[Yamada, 92] Yamada, S., Tokuno, K., Osaki, S.: Imperfect debugging models with fault introduction rate for software reliability assessment, International Journal of Systems Science, Vol. 23, Number 12, 1992.

[Zhang, 03] Zhang, X., Teng, X., Pham, H.: Considering fault removal efficiency in software reliability assessment, IEEE Trans. on Systems, Man, and Cybernetics – Part A, Vol. 33, Number 1, 114-120, 2003.

[Zhang, 06] Zhang, X., Pham, H.: Software field failure rate prediction before software deployment, Journal of Systems and Software, Vol. 79, 291-300, 2006.

[Zhu, 18] Zhu, M., Pham, H.: A two-phase software reliability modeling involving with software fault dependency and imperfect fault removal, Computer Languages, Systems & Structures (in press). 2018.