J. Chosun Natural Sci.
Vol. 10, No. 4 (2017) pp. 249 – 257
https://doi.org/10.13160/ricns.2017.10.4.249

Improved Exponential Software Reliability Model Based on NHPP with the Uncertainty of Operating Environments

Kwang Yoon Song and In Hong Chang†

Abstract

The main focus when developing software is to improve the reliability and stability of a software system. We are enjoying a very comfortable life thanks to modern civilization, however, comfort is not guaranteed to us. Once software systems are introduced, the software systems used in the field environments are the same as or close to those used in the development-testing environment; however, the systems may be used in many different locations. Development of software system is a difficult and complex process. Generally, existing software reliability models are applied to software testing data and then used to make predictions on the software failures and reliability in the field. In this paper, we present an improved exponential NHPP software reliability model in different development environments, and examine the goodness-of-fit of improved exponential model and other model based on two datasets. The results show that the proposed model fits significantly better than other NHPP software reliability model.

Keywords: Exponential NHPP Model, Non-homogeneous Poisson Process, Software Reliability, Fault Detection Rate

1. Introduction

Many existing NHPP software reliability models have been developed through the fault intensity rate function and the mean value functions m(t) within a controlled testing environment to estimate reliability metrics such as the number of residual faults, failure rate, and reliability of the software[1-4]. The pioneering attempt in NHPP based on software reliability model was made by Goel and Okumoto[1]. Goel and Okumoto[1] presented a stochastic model for the software failure phenomenon based on a nonhomogeneous Poisson process, and this model describes the failure observation phenomenon by an exponential curve. Once software systems are introduced, the software systems used in the field environments are the same as or close to those used in the development-testing environment. Generally, many existing models are applied to software testing data and then used to make predictions on the software failures and reliability in the field[1-4]. Here, the important point is that the test environment and operational environment are different from each other. Once software systems are introduced, the software systems used in the field environments are the same as or close to those used in the development-testing environment; however, the systems may be used in many different locations. Pham[5-6] and Chang *et al.*[7] developed a software reliability model incorporating the uncertainty of the system fault detection rate per unit of time subject to the operating environment. Pham[8] recently presented a new generalized software reliability model subject to the uncertainty of operating environments. And also, Song *et al.*[9-10] presented a new model with consideration of a three-parameter fault detection rate and a Weibull fault detection rate in the software development process, and relate it to the error detection rate function with consideration of the uncertainty of operating environments.

In this paper, we discuss an improved exponential NHPP software reliability model in different development environments. We examine the goodness-of-fit of improved exponential model and other model based on two datasets. The explicit solution of the mean value function for the new NHPP software reliability models is derived in Section 2. Criteria for model comparisons and selection of the best model are discussed. The model analysis and results are discussed in Section 3.

Department of Computer Science and Statistics, Chosun University, Gwangju, Korea

†Corresponding author : ihchang@chosun.ac.kr
(Received : November 28, 2017, Revised : December 17, 2017, Accepted : December 25, 2017)

Section 4 presents the conclusions and remarks.

2. New NHPP Software Reliability Model

2.1. Non-homogeneous Poisson Process

The NHPP models provide an analytical framework for describing the software failure phenomenon during testing. The main point in the NHPP models are to estimate the mean value function (MVF) of the cumulative number of failures experienced up to a certain point in time. The software fault detection process has been widely formulated by using a counting process. A counting process $\{N(t), t \geq 0\}$, is said to be a NHPP with intensity function $\lambda(t)$, if $N(t)$ follows a Poisson distribution with the mean value function $m(t)$, i.e.,

$$Pr\{N(t)=n\} = \frac{\{m(t)\}^n}{n!}\exp\{-m(t)\}, n=0,1,2,3,\ldots$$

The mean value function m(t), which is the expected number of faults detected at time t with m(0)=0 can be expressed as

$$m(t) = \int_0^t \lambda(s)ds \,.$$

The software reliability R(x|t) is defined as the probability that a failure does not occur in the time interval $[t, t+x]$ $(t \geq 0, x \geq 0)$

$$R(x|t) = e^{-[m(t+x)-m(t)]} \,.$$

Many NHPP-based SRGM have been modeled m(t) using the differential equation

$$\frac{dm(t)}{dt} = b(t)[N-m(t)] \tag{1}$$

Solving Eq. (1) makes it possible to obtain different values of m(t) using different values for b(t), which reflects various assumptions of the software testing process. The solution for the mean value function m(t), where the initial condition m(0) = 0, is given

$$m(t) = N\left(1-e^{-\int_0^t b(s)ds}\right) \tag{2}$$

Here, if $b(t) = b$, the following an exponential NHPP software reliability model of Goel-Okumoto (GO Model).

$$m(t) = N(1-e^{-bt}) \,.$$

2.2. New NHPP Software Reliability Model

A generalized NHPP model incorporating the uncertainty of operating environments can be formulated as follows[6]:

$$\frac{dm(t)}{dt} = \eta b(t)[N-m(t)] \tag{3}$$

The solution for the mean value function m(t), where the initial condition m(0) = 0, is given by[6]:

$$m(t) = N\left(1-e^{-\eta\int_0^t b(s)ds}\right) \tag{4}$$

where η is a random variable that represents the uncertainty of the system fault detection rate in the operating environments with a probability density function g, N is the expected number of faults that exists in the software before testing, b(t) is the fault detection rate function, which also represents the average failure rate of a fault, and m(t) is the expected number of errors detected by time t or the mean value function.

If we assume that η has a gamma distribution with parameters α and β, i.e., η~Gamma (α, β) where the probability density function of η is given by

$$g(x) = \frac{\beta^\alpha x^{\alpha-1} e^{-\beta x}}{\Gamma(\alpha)} \tag{5}$$

Then from Eq. (4), we obtain

$$m(t) = N\left(1-\left(\frac{\beta}{\beta+\int_0^t b(s)ds}\right)^\alpha\right)$$

If we assume that η has a gamma distribution with parameters α and β, i.e., η~Exponential (α, β) where the probability density function of η is given by

$$g(x) = \beta e^{-\beta x} \tag{6}$$

Then from Eq. (4), we obtain

$$m(t) = N\left(1-\frac{\beta}{\beta+\int_0^t b(s)ds}\right)$$

In this paper, we consider a fault rate function b(t) to be as follows:

$$b(t) = b, \; b > 0 \tag{5}$$

We obtain a new NHPP software reliability model, m(t), that can be used to determine the expected number of software failures detected by time t by substituting the function b(t) above into Eq. (4):

Table 1. NHPP Software reliability models

No.	Model	m(t)
1	Exponential Model	$m(t)=N(1-e^{-bt})$
2	Proposed New Model 1	$m(t)=N\left(1-\left(\frac{\beta}{\beta+bt}\right)^{\alpha}\right)$
3	Proposed New Model 2	$m(t)=N\left(1-\frac{\beta}{\beta+bt}\right)$

$$m(t)=N\left(1-\left(\frac{\beta}{\beta+bt}\right)^{\alpha}\right) \tag{7}$$

$$m(t)=N\left(1-\frac{\beta}{\beta+bt}\right) \tag{8}$$

Table 1 summarizes the proposed new NHPP software reliability models and the Goel-Okumoto NHPP software reliability model with different mean value functions.

3. Numerical Examples

The model parameters to be estimated in the mean value function m(t) can then be obtained with the help of a developed MATLAB 2016 program based on the least-squares estimate (LSE) method.

3.1. Criteria for Model Comparisons

Criteria for model comparisons will be used as criteria for the model estimation of the goodness-of-fit and to compare the proposed models and other model as listed in Table 1. For all criteria, the smaller the value, the closer the model fits relative to other models run on the same data set.

The mean squared error (MSE) measures the distance of a model estimate from the actual data with the consideration of the number of observations, n, and the number of unknown parameters in the model, m. The mean squared error is given by

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

The sum absolute error is similar to the sum squared error, but the way of measuring the deviation is by the use of absolute values, and sums the absolute value of the deviation between the actual data and the estimated curve. The sum absolute error is given by

$$SAE=\sum_{i=0}^{n}\left|\hat{m}(t_i)-y_i\right|.$$

The correlation index of the regression curve equation (R^2) is given by

$$R^2=1-\frac{\sum_{i=0}^{n}(\hat{m}(t_i)-y_i)^2}{\sum_{i=0}^{n}(y_i-\bar{y})^2}.$$

where $m(t_i)$ is the estimated cumulative number of failures at t_i for i = 1, 2, ..., n; and y_i is the total number of failures observed at time t_i.

3.2. Estimation of the Confidence Intervals

we use Eq. (6) to obtain the confidence intervals[11] of the software reliability models in Table 1. The confidence interval is given by

$$\hat{m}(t_i)\pm Z_{\alpha/2}\sqrt{\hat{m}(t_i)} \tag{6}$$

where $Z_{\alpha/2}$ is 100(1-α) percentile of the standard normal distribution.

3.3. Data Information

Dataset1 listed in Table 2, was reported by Pham[11]. The data was collected over a period of 12 weeks during which time the testing started and stopped many times. Errors detection is broken down into subcategories to help the development and testing team to sort and solve the most critical Modification Requests first. Dataset2 listed in Table 3, was reported by Lee *et al.*[12]. The field failure data is the failure data detected in the system test. The size of the exchange software is a large

Table 2. Dataset1

Time Index (Month)	Failures	Cumulative Failures
1	21	21
2	8	29
3	4	33
4	11	44
5	11	55
6	33	88
7	14	102
8	9	111
9	3	114
10	16	130
11	1	131
12	5	136

Table 3. Dataset2

Time Index (Month)	Failures	Cumulative Failures
1	83	83
2	287	370
3	177	547
4	193	740
5	120	760
6	67	927
7	75	1002
8	46	1048
9	24	1072
10	69	1141
11	129	1270
12	117	1387
13	31	1418
14	40	1458
15	34	1492
16	35	1527
17	20	1547
18	5	1552

program with 134 million source code lines and consists of 140 major functional blocks. All faults detected for each system test are registered in the fault management system and are tracked until all faults have been corrected and solved.

3.4. Results

Table 4 summarize the results of the estimated parameters of all models in Table 1 using the least-squares estimation (LSE) technique. We obtained the three common criteria when t = 1, 2, ..., 12 from Dataset1, as can be seen from Table 5, SAE value for the proposed new model1 is the lowest values compared to all models. And R^2 value for the proposed new model2 is better, because, close to 1 than R^2 value for other all models. And, We obtained the three common criteria when t = 1, 2, ..., 18 from Dataset2, as can be seen from Table 6, MSE and SAE values for the proposed new model2 are the lowest values compared to all models. And R^2 value for the proposed new model1 is better, because, close to 1 than R^2 value for other all models. Table 7 and 8 summarize the results of the mean value function and confidence interval each of all models for Dataset1 and 2, respectively.

Fig. 1 and 2 show the graph of the mean value functions for all models for Dataset1 and Dataset2, respectively. Fig. 3 and 4 show the graph of the absolute of relative error value of all models for Dataset1 and 2, respectively, and shows the graph of the absolute value of relative error for all models, and better when close to 0 at each point. Fig. 5-10 show the graph of the mean value function and confidence interval each of all models for Dataset1 and 2, respectively.

4. Conclusions

When new software is introduced, this software will be used into a similar environment or another environment. However, most of the time, it will be used in

Table 4. Model parameter estimation from Dataset1 and 2

Model	LSE's	
	Dataset1	Dataset2
Exponential Model	$\hat{N}$ = 400.86, $\hat{b}$ = 0.0375	$\hat{N}$ = 1821.85, $\hat{b}$ = 0.11
Proposed New Model 1	$\hat{N}$ = 407.01, $\hat{b}$ = 0.002, $\hat{\alpha}$ = 109.00, $\hat{\beta}$ = 5.99	$\hat{N}$ = 2186, $\hat{b}$ = 0.43, $\hat{\alpha}$ = 1.90, $\hat{\beta}$ = 8.00
Proposed New Model 2	$\hat{N}$ = 759.7, $\hat{b}$ = 0.095, $\hat{\beta}$ = 4.87	$\hat{N}$=2573.01, $\hat{b}$ = 0.78, $\hat{\beta}$ = 8.70

Table 5. Comparison criteria from Dataset1 and 2

Model	Dataset1			Dataset2		
	MSE	SAE	R^2	MSE	SAE	R^2
Exponential Model	77.5850	84.3216	0.9631	3038.5616	750.7537	0.9848
Proposed New Model 1	95.3377	84.1716	0.9637	3146.1575	678.4878	0.9862
Proposed New Model 2	85.5004	84.8672	0.9634	2957.7008	682.4817	0.9861

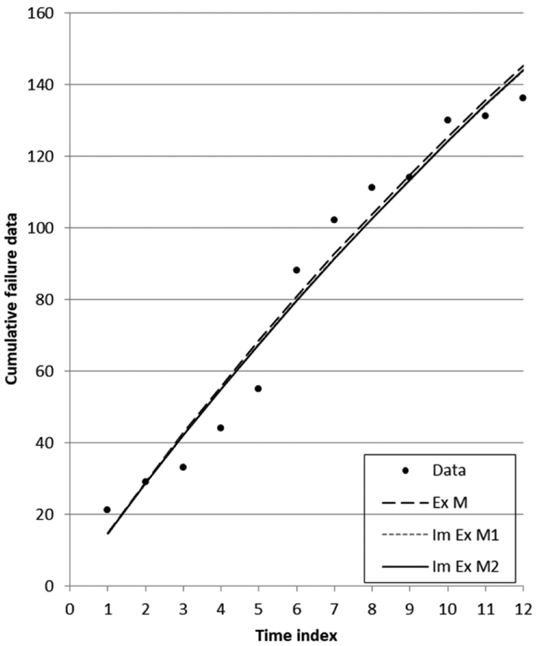

Fig. 1. Mean value function of all models for Dataset1.

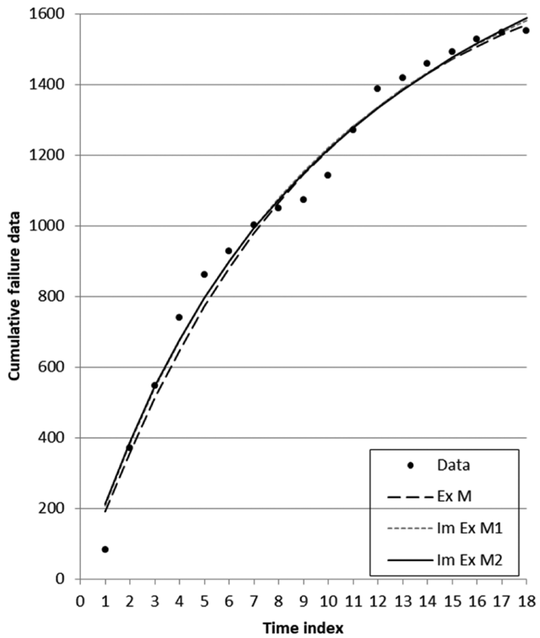

Fig. 2. Mean value function of all models for Dataset2.

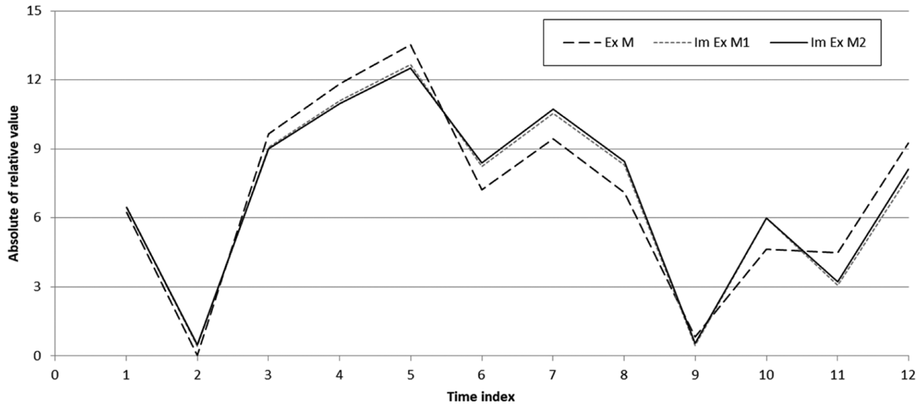

Fig. 3. Absolute of relative error value of models for Dataset1.

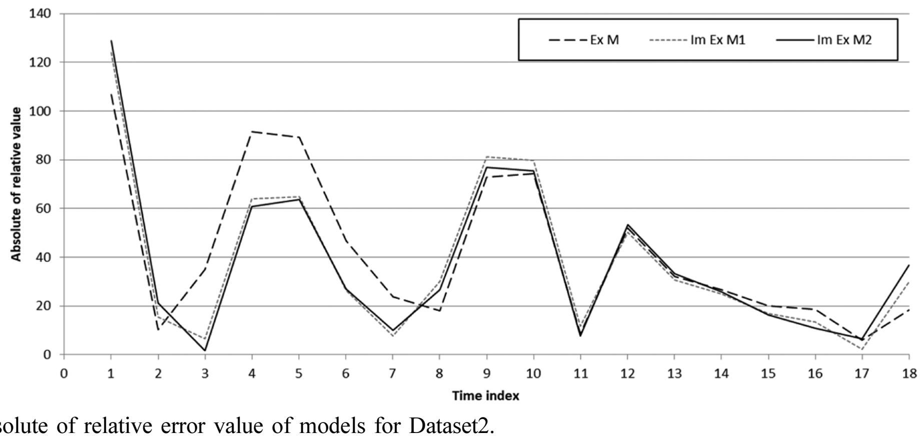

Fig. 4. Absolute of relative error value of models for Dataset2.

Table 6. 95% Confidence limits of all models from Dataset1

Time index \ Model	Data	Exponential Model			Proposed New Model 1			Proposed New Model 2		
		LCL	m(t)	UCL	LCL	m(t)	UCL	LCL	m(t)	UCL
1	21	7.226	14.754	22.282	7.069	14.544	22.019	7.063	14.536	22.009
2	29	18.416	28.965	39.513	18.089	28.564	39.039	18.058	28.526	38.994
3	33	29.852	42.653	55.453	29.364	42.078	54.792	29.299	42.001	54.703
4	44	41.191	55.837	70.482	40.556	55.106	69.655	40.454	54.988	69.522
5	55	52.310	68.535	84.761	51.542	67.664	83.787	51.409	67.513	83.617
6	88	63.153	80.767	98.381	62.265	79.771	97.276	62.114	79.601	97.088
7	102	73.693	92.548	111.403	72.699	91.441	110.183	72.549	91.274	109.999
8	111	83.918	103.896	123.873	82.830	102.692	122.553	82.705	102.553	122.401
9	114	93.823	114.826	135.828	92.654	113.538	134.422	92.581	113.457	134.334
10	130	103.409	125.353	147.297	102.169	123.994	145.819	102.180	124.006	145.832
11	131	112.679	135.493	158.308	111.380	134.074	156.769	111.509	134.216	156.922
12	136	121.638	145.260	168.883	120.289	143.792	167.295	120.575	144.103	167.631

Table 7. 95% Confidence limits of all models from Dataset2

Time index \ Model	Data	Exponential Model			Proposed New Model 1			Proposed New Model 2		
		LCL	m(t)	UCL	LCL	m(t)	UCL	LCL	m(t)	UCL
1	83	162.774	189.775	216.775	178.789	206.987	235.185	183.186	211.703	240.221
2	370	322.605	359.781	396.958	347.002	385.484	423.965	352.451	391.218	429.984
3	547	467.726	512.079	556.431	494.957	540.525	586.093	499.595	545.366	591.137
4	740	598.600	648.512	698.425	625.114	676.076	727.037	628.092	679.170	730.249
5	860	716.321	770.734	825.147	740.020	795.293	850.566	741.096	796.408	851.719
6	927	822.075	880.224	938.374	841.896	900.718	959.541	841.178	899.977	958.775
7	1002	917.006	978.310	1039.613	932.608	994.414	1056.221	930.400	992.135	1053.871
8	1048	1002.180	1066.178	1130.175	1013.718	1078.072	1142.425	1010.419	1074.671	1138.923
9	1072	1078.575	1144.893	1211.211	1086.530	1153.085	1219.639	1082.579	1149.016	1215.453
10	1141	1147.079	1215.409	1283.738	1152.137	1220.613	1289.089	1147.976	1216.332	1284.688
11	1270	1208.496	1278.579	1348.662	1211.462	1281.628	1351.794	1207.516	1277.571	1347.626
12	1387	1263.553	1335.170	1406.787	1265.282	1336.947	1408.611	1261.947	1333.520	1405.093
13	1418	1312.901	1385.865	1458.829	1314.262	1387.262	1460.263	1311.900	1384.837	1457.774
14	1458	1357.130	1431.280	1505.430	1358.966	1433.165	1507.363	1357.902	1432.073	1506.243
15	1492	1396.768	1471.964	1547.160	1399.881	1475.159	1550.437	1400.405	1475.697	1550.989
16	1527	1432.288	1508.410	1584.532	1437.426	1513.681	1589.935	1439.792	1516.108	1592.423
17	1547	1464.119	1541.060	1618.001	1471.963	1549.104	1626.246	1476.394	1553.648	1630.903
18	1552	1492.641	1570.309	1647.976	1503.806	1581.756	1659.707	1510.494	1588.613	1666.732

other environments. In this paper, we discussed an improved exponential NHPP software reliability model in different development environments. Table 4 summarized the results of the estimated parameters of all models and the three common criteria value for two Datasets. As a result, the proposed new models are lowest values compared to the exponential model. In other words, the results show the difference between the actual and predicted values of the new models are smaller than the other model. Future work will approach the optimal release policies using the proposed new models.

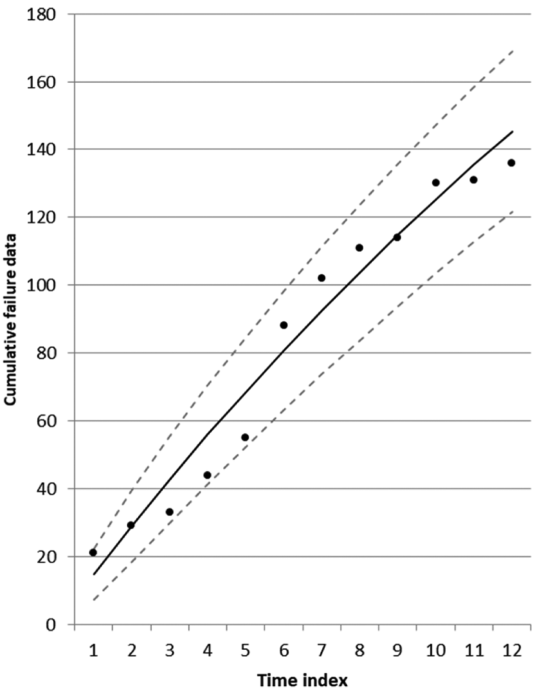

Fig. 5. Confidence intervals of the exponential model for Dataset1.

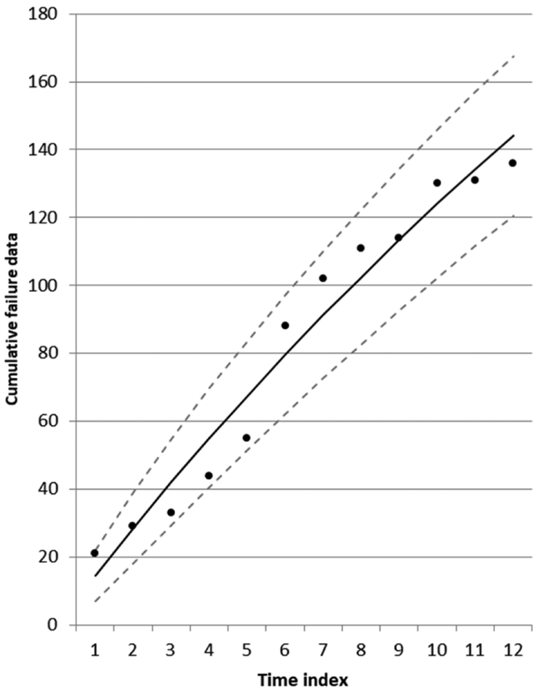

Fig. 7. Confidence intervals of the proposed new model2 for Dataset1.

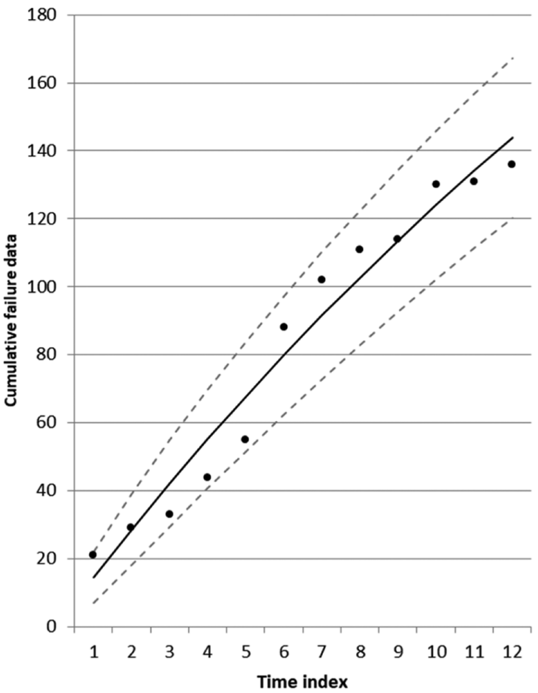

Fig. 6. Confidence intervals of the proposed new model1 for Dataset1.

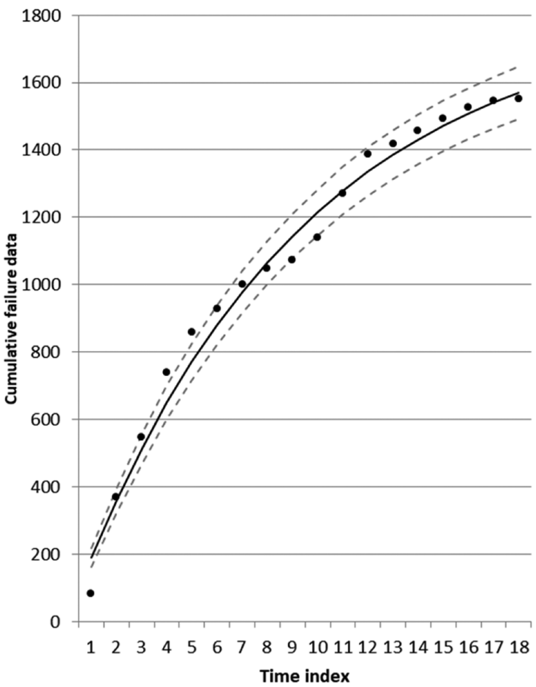

Fig. 8. Confidence intervals of the exponential model for Dataset2.

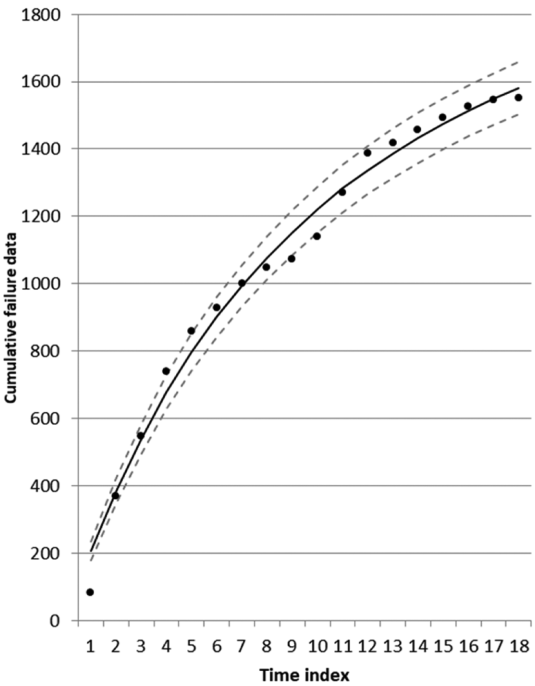

Fig. 9. Confidence intervals of the proposed new model1 for Dataset2.

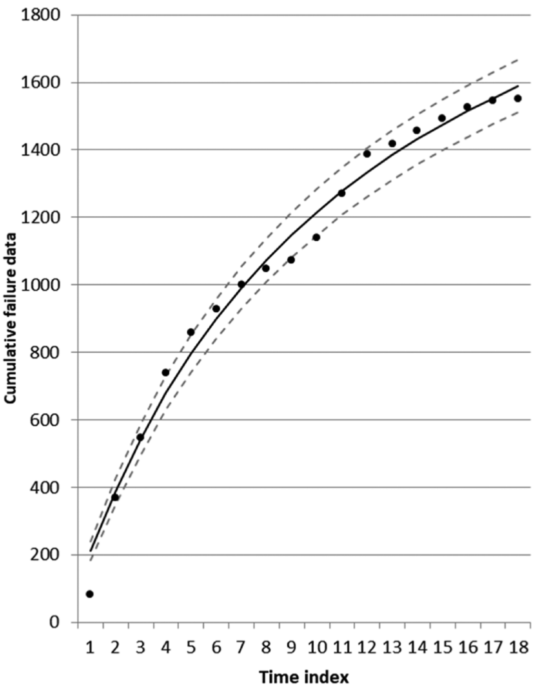

Fig. 10. Confidence intervals of the proposed new model2 for Dataset2.

Acknowledgements

This study was supported by research funds from Chosun University, 2017.

References

[1] A. L. Goel and K. Okumoto, "Time dependent error detection rate model for software reliability and other performance measures", IEEE Transactions on Reliability, Vol. R-28, pp. 206-211, 1979.

[2] M. Ohba, "Inflection S-shaped software reliability growth models", Stochastic Models in Reliability Theory, ed. S. Osaki, Y. Hatoyama, Berlin: Springer-Verlag, pp. 144-162, 1984.

[3] M. Ohba and S. Yamada, "S-shaped software reliability growth models" in Proceedings of the 4th International Conference on Reliability and Maintainability, pp. 430-436, 1984.

[4] S. Yamada, M. Ohba, and S. Osaki, "S-shaped reliability growth modeling for software fault detection", IEEE Transactions on Reliability, Vol. 12, pp. 475-484, 1983.

[5] H. Pham, "A software reliability model with Vtub-Shaped fault-detection rate subject to operating environments", In Proceedings of the 19th ISSAT International Conference on Reliability and Quality in Design, Honolulu, Hawaii, August 5–7, pp. 33–37, 2013.

[6] H. Pham, "A new software reliability model with Vtub-Shaped fault detection rate and the uncertainty of operating environments", Optimization, Vol. 63, pp. 1481–1490, 2014.

[7] I. H. Chang, H. Pham, S. W. Lee, and K. Y. Song. "A testing-coverage software reliability model with the uncertainty of operation environments", International Journal of Systems Science: Operations & Logistics, Vol. 1, pp. 220–227, 2014.

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

[9] K. Y. Song, I. H. Chang, and H. Pham, "A three-parameter fault-detection software reliability model with the uncertainty of operating environments", Journal of Systems Science and Systems Engineering, Vol. 26, pp. 121–132, 2017.

[10] K. Y. Song, I. H. Chang, and H. Pham. "A software reliability model with a weibull fault detection rate function subject to operating environments", Appl.

Sci. (Basel), Vol. 7, p. 983, 2017.
[11] H. Pham, "System software reliability", London: Springer, 2006.
[12] J. K. Lee, S. K. Shin, and Y. M. Lee, "A study on hypothetical switching software through of the analysis of failure data", The Journal of Korean Institute of Communications and Information Sciences, Vol. 23, pp. 1915-1925, 1998.