A Comparative Study on the Performance Evaluation of NHPP Software Reliability Model with Non-Exponential Family Distribution Property

Seung Kyu Park

Namseoul University, Department of Electronic Engineering, Chungnam, 31020, Korea.

Abstract

In this study, after applying the non-exponential family distributions (Pareto, Log-Logistic) which are widely used in the field of reliability to the finite failure NHPP software reliability model, we analyzed the reliability performance. For this, software failure time data was used, parametric estimation was applied to the maximum likelihood estimation method, and nonlinear equations were calculated using the bisection method. As a result, in the analysis of the intensity function, the log-logistic model was the most efficient in terms of reliability suitability because the strength function increased and decreased as the failure time passed and the mean square error (MSE) was also small. In the analysis of the mean value function, the Pareto model showed the biggest error estimation compared to the true value, but the Log-Logistic model had a smaller margin of error than other models. As a result of evaluating the software reliability after putting the mission time in the future, the Pareto model was high and stable, but the Log-Logistic and Goel-Okumoto basic model showed a decreasing tendency. In conclusion, the Log-Logistic model was the most efficient among the proposed models. In this study, we have newly analyzed and evaluated the reliability performance of non-exponential family distribution which have no previous research cases, and expect it to be used as a basic guideline for software developers to search for the optimal software reliability model.

Keywords: Log-Logistic Distribution, NHPP Model, Non-Exponential Family, Pareto Distribution, Software Reliability Performance, Software Failure Time.

1. INTRODUCTION

As software technology is widely applied in high-tech industries, there is a growing need for high-quality software that can reliably process a variety of data without failure. To solve this problem, software developers are doing a lot of research to explore ways to improve reliability. In particular, software reliability models using the Non-Homogeneous Poisson Process (NHPP) have been extensively studied to improve software quality. Many NHPP software reliability models using the intensity function and the mean value function have been proposed to estimate the reliability attributes such as the number of residual failures and the software failure rate in a controlled test environment [1]. In this process, Pham and Zhang [2] proposed a new model based on software reliability, while Gokhale and Trivedi [3] proposed an improved NHPP model. In particular, Kim [4] analyzed the properties of the finite failure NHPP software model based on the modified Landley type lifetime distribution, while Kim and Shin [5] analyzed the optimal software release time problem based on the exponential and non-exponential distribution models. Also, Yang [6] proposed a new performance analysis results of finite failure NHPP software reliability model with Logistic distribution property, Yang and Park [7] compared the attributes of the NHPP software reliability model based on the Weibull extension and the flexible Weibull extension distribution.

Therefore, in this study, after applying the non-exponential family distributions which are widely accepted in the field of reliability to the finite failure NHPP software reliability model, we were newly evaluated the reliability performance of the proposed models and will present the optimal software reliability model through this research results.

2. RELATED RESEARCH

2.1 NHPP Software Reliability Model

The NHPP model contains property about mean value m(t) and intensity pattern $\lambda(t)$. N(t) is the cumulative number of failures of the software detected up to time t, m(t) is a mean value function when λ(t) is expressed by an intensity function, the cumulative failure number N(t) follows a Poisson probability density function having a parameter m(t). The NHPP software reliability model is a model that measures the reliability using the mean failure rate function by the number of failures generated per unit time. That is

$$P\{N(t) = n\} = \frac{[m(t)]^n \cdot e^{-m(t)}}{n!} \tag{1}$$

Note. $n = 0,1,2,\cdots \infty$.

The mean value function m(t) and the intensity function λ(t) of the NHPP model are as follows.

$$m(t) = \int_0^t \lambda(s)ds \tag{2}$$

$$\frac{dm(t)}{d(t)} = \lambda(t) \tag{3}$$

In terms of software reliability, the mean value function represents a software failure occurrence expected value, the intensity function is the failure rate function and means the failure occurrence rate per defect. Also, the time domain NHPP models are classified into a finite failure that the failure does not occur at the time of repairing the failure, and an infinite failure that the failure occurs at the time of repairing failure. In this study, we will analyze the software reliability performance based on finite failure cases. That is, in the finite-failure NHPP model, if the expected value of the failure that can be found up to time [0, t] is θ, then the mean value function and the intensity function are as follows.

$$m(t|\theta, b) = \theta F(t) \tag{4}$$

$$\lambda(t|\theta, b) = \theta F(t)' = \theta f(t) \tag{5}$$

Considering Eq. 4 and Eq. 5, the likelihood function of the finite-failure NHPP model is derived as follows.

$$L_{NHPP}(\Theta|\underline{x}) = \left(\prod_{i=1}^n \lambda(x_i)\right) \exp[-m(x_n)] \tag{6}$$

Note. $\underline{x} = (x_1, x_2, x_3 \cdots x_n)$.

2.2 Goel-Okumoto Basic Model

The Goel-Okumoto model is a well-known basic model in the software reliability field. Let f(t) and F(t) for the Goel-Okumoto model be a probability density function and a cumulative density function, respectively. Assuming that the expected value of the number of failures of the observation point [0, t] is θ, the finite failure strength function and the mean value function are as follows [8].

$$\lambda(t|\theta, b) = \theta f(t) = \theta b e^{-bt} \tag{7}$$

$$m(t|\theta, b) = \theta F(t) = \theta(1 - e^{-bt}) \tag{8}$$

Note that $\theta > 0, \ b > 0$.

Using the Eq. 7 and Eq. 8, the likelihood function of the finite-failure NHPP model is derived as follows.

$$L_{NHPP}(\theta, b|\underline{x}) = \tag{9}$$

$$\left(\prod_{i=1}^n \theta b e^{-bx_i}\right) exp[-\theta(1 - e^{-bx_n})]$$

Note that $\underline{x} = (0 \le x_1 \le x_2 \le \cdots \le x_n)$.

The log-likelihood function, using the Eq. 9, is simplified to the following log conditional expression.

$$\ln L_{NHPP}(\Theta|\underline{x}) = \tag{10}$$

$$n\ln\theta + n\ln b - b\sum_{k=1}^n x_k - \theta(1 - e^{-bx_n})$$

When the Eq. 10 is partially differentiated concerning θ and b, the result is as follows.

$$\frac{\partial \ln L_{NHPP}(\Theta|\underline{x})}{\partial \theta} = \frac{n}{\hat{\theta}} - 1 + e^{-\hat{b}x_n} = 0 \tag{11}$$

$$\frac{\partial \ln L_{NHPP}(\Theta|\underline{x})}{\partial b} = \frac{n}{\hat{b}} - \sum_{i=1}^n x_n - \hat{\theta}x_n e^{-\hat{b}x_n} = 0 \tag{12}$$

Therefore, the maximum likelihood estimator $\hat{\theta}_{MLE}$ and $\hat{b}_{MLE}$ can be calculated using the bisection method.

2.3 NHPP Log-Logistic Distribution Model

The Log-Logistic distribution is a non-exponential family distribution which is widely used in the field of reliability.
That is, the Log-Logistic distribution has a property that increases and decreases in the form of failure occurring rate and thus is widely applied in the reliability field. The probability density function and the cumulative distribution function considering the shape parameter(k) are as follows [9].

$$f(t|\tau, k) = \frac{\tau k(\tau t)^{k-1}}{[1 + (\tau t)^k]^2} \tag{13}$$

$$F(t|\tau, k) = \frac{(\tau t)^k}{[1 + (\tau t)^k]} \tag{14}$$

Note that $\tau > 0, \ k > 0$

Therefore, the intensity function and the mean value function of the finite fault NHPP Log-Logistic model are as follows.

$$\lambda(t|\theta, \tau, k) = \theta f(t) = \theta \frac{\tau k(\tau t)^{k-1}}{[1 + (\tau t)^k]^2} \tag{15}$$

$$m(t|\theta, \tau, k) = \theta F(t) = \theta \frac{(\tau t)^k}{[1 + (\tau t)^k]} \tag{16}$$

Therefore, the log-likelihood function to maximum likelihood estimation by using Eq. 15 and Eq. 16 is derived as follows.

$$\ln L_{NHPP}(\Theta|\underline{x}) = n\ln 2 + n\ln \theta + 2n\ln\tau$$

$$+ \sum_{i=1}^n x_i - 2\sum_{i=1}^n \ln[1 + (\tau x_i)^2] - \theta \frac{(\tau x_n)^2}{[1 + (\tau x_n)^2]} = 0 \tag{17}$$

In this study, we apply the case where the shape parameter(k) that determines the shape of failure lifetime distribution is 2. And, if the Eq. 17 is partially differentiated concerning θ and τ, the maximum likelihood estimator $\hat{\theta}_{MLE}$ and $\hat{\tau}_{MLE}$ can be calculated using the bisection method.

$$\frac{\partial \ln L_{NHPP}(\Theta|\underline{x})}{\partial \theta} = \frac{n}{\hat{\theta}} - \frac{(\hat{\tau}x_n)^2}{[1 + (\hat{\tau}x_n)^2]} = 0 \qquad (18)$$

$$\frac{\partial \ln L_{NHPP}(\Theta|\underline{x})}{\partial \tau} = \frac{2n}{\hat{\tau}} - 2\hat{\tau}\sum_{i=1}^{n} x_i^2 \frac{1}{ln[1 + (\hat{\tau}x_i)^2]}$$

$$-\hat{\theta}\left(\frac{2\hat{\tau}x_n^2(1 + \hat{\tau}^2 x_n - \hat{\tau}^2 x_n^2)}{[1 + (\hat{\tau}x_n)^2]^2}\right) = 0 \qquad (19)$$

Note that $x = (x_1, x_2, x_3 \cdots x_n)$.

2.4 NHPP Pareto Distribution Model

The Pareto distribution is a non-exponential family distribution applied in the field of reliability that reflects mathematical and observable phenomena such as scientific, and physics. The probability density function and the cumulative density function considering the shape parameter (a) and the scale parameter (b) are as follows [10].

$$f(t) = \frac{a}{b}\left[1 + \frac{t}{b}\right]^{-(a+1)} \qquad (20)$$

$$F(t) = 1 - \left[1 + \left(\frac{t}{b}\right)\right]^{-a} \qquad (21)$$

Note. $a, b > 0, \ t \in [0, \infty]$

Assuming that the expected value of the number of failures of the observation point [0, t] is θ, the finite failure strength function and the mean value function are as follows.

$$\lambda(t|\theta, b) = \theta f(t) = \theta\left(\frac{a}{b}\left[1 + \frac{t}{b}\right]^{-(a+1)}\right) \qquad (22)$$

$$m(t|\theta, b) = \theta F(t) = \theta\left(1 - \left[1 + \left(\frac{t}{b}\right)\right]^{-a}\right) \qquad (23)$$

Therefore, the log-likelihood function to maximum likelihood estimation by using Eq. 22 and Eq. 23 is derived as follows.

$$\ln L_{NHPP}(\Theta|\underline{x}) = n\ln\theta + n\ln a - n\ln b$$

$$+\sum_{i=1}^{n} ln\left[1 + \frac{x_i}{b}\right]^{-(a+1)} - \theta\left(1 - \left[1 + \left(\frac{x_n}{b}\right)\right]^{-a}\right) \qquad (24)$$

In this study, we apply the case where the shape parameter(a) that determines the shape of failure lifetime distribution is 2. When the Eq. 24 is partially differentiated concerning θ and b, the result is as follows.

$$\frac{\partial \ln L_{NHPP}(\Theta|\underline{x})}{\partial \theta} = \frac{n}{\hat{\theta}} - \left(1 - \left[1 + \left(\frac{x_n}{\hat{b}}\right)\right]^{-2}\right) = 0 \qquad (25)$$

$$\frac{\partial \ln L_{NHPP}(\Theta|\underline{x})}{\partial b} = -\frac{n}{\hat{b}} + \frac{2x_n\hat{\theta}}{\hat{b}^2}\left(1 + \frac{x_n}{\hat{b}}\right)^{-3}$$

$$+\frac{3}{4}\sum_{i=1}^{n} ln\left(1 + \frac{x_i}{2}\right) = 0 \qquad (26)$$

Note that $x = (x_1, x_2, x_3 \cdots x_n)$.

Therefore, the maximum likelihood estimator $\hat{\theta}_{MLE}$ and $\hat{b}_{MLE}$ can be calculated using the bisection method.

3. RELIABILITY PERFORMANCE EVALUATION USING SOFTWARE FAILURE TIME

We will compare and evaluate the performance of the proposed reliability models using the software failure time data [11] as shown in Table 1. This software failure time is the data that was occurred 30 times in 738.68 unit time.

Table 1. Software Failure Time Data

Failure Number	Failure Time (hours)	Failure Number	Failure Time (hours)
1	30.02	16	151.78
2	31.46	17	177.50
3	53.93	18	180.29
4	55.290	19	182.21
5	58.720	20	186.34
6	71.920	21	256.81
7	77.070	22	273.88
8	80.900	23	277.87
9	101.90	24	453.93
10	114.87	25	535.00
11	115.34	26	537.27
12	121.57	27	552.90
13	124.97	28	673.68
14	134.07	29	704.49
15	136.25	30	738.68

Laplace trend test was used to verify the reliability of the software failure time data as shown in Figure 1 [12].
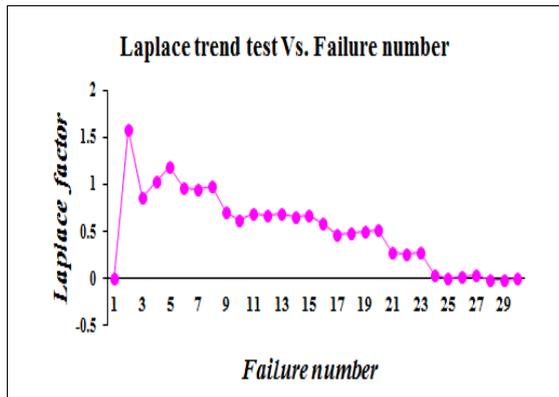
Fig 1. Estimation Results of Laplace Trend Test

In general, if the Laplace factor estimates are distributed between -2 and 2, the data are reliable because the extreme values do not exist and are stable. As a result of this test, the estimated value of the Laplace factor was distributed between 0 and 2, as shown in Figure 1. Therefore, it is possible to apply this data because there is no extreme value.

In this study, the maximum likelihood estimation (MLE) was used to perform parameter estimation. The calculation method of the nonlinear equations is solved by using the bisection method, and the results are shown in Table 2.

Table 2. Parameter Estimation of Each Model

Type	Model	MLE		Model Comparison	
				MSE	R^2
Basic	Goel-Okumoto	$\hat{\theta} = 33.409$	$\hat{b} = 0.309$	5.842	0.981
Non-exponential Family Distribution	Log-Logistic	$\hat{\theta} = 32.241$	$\hat{\tau} = 0.495$	8.973	0.972
	Pareto	$\hat{\theta} = 31.209$	$\hat{b} = 1.810$	45.580	0.855

Explanatory notes. MLE = Maximum Likelihood Estimation
MSE = Mean Square Error, $R^2 = Coefficient\ of\ Determination$.

As the basis for determining the efficient model, the mean square error is defined as follows.

$$MSE = \frac{\sum_{i=1}^{n} \left(m(x_i) - \hat{m}(x_i)\right)^2}{n - k} \qquad (27)$$

Note that $m(x_i)$ is the total accumulated number of errors observed within time is $(0, x_i)$, $\hat{m}(x_i)$ is the estimated cumulative number of errors at time x_i obtained from the fitting mean value function, n is the number of observations and k is the number of parameters to be estimated. When selecting an efficient model, the smaller the mean square error, the more efficient the model [13].

The coefficient of determination (R^2) is a measuring value to the explanatory power of the difference between the target value and the observed value. When selecting an efficient model, the larger the value of the decision coefficient, the more efficient the model because the error is relatively small.

It is defined as

$$R^2 = 1 - \frac{\sum_{i=1}^{n} \left(m(x_i) - \hat{m}(x_i)\right)^2}{\sum_{i=1}^{n} \left(m(x_i) - \sum_{j=1}^{n} m(x_j)/n\right)^2} \qquad (28)$$

As shown in Table 2, we can see that the Log-Logistic model is more efficient than the Pareto model. Also, the Goel-Okumoto basic model having the largest coefficient of determination and the smallest mean square error is more efficient than the other models.

Figure 2 shows the transition of mean square error (MSE) according to each failure number. In this figure, the Log-Logistic model shows an efficient performance in terms of fitness because the mean square error tends to be smaller than the Pareto model.
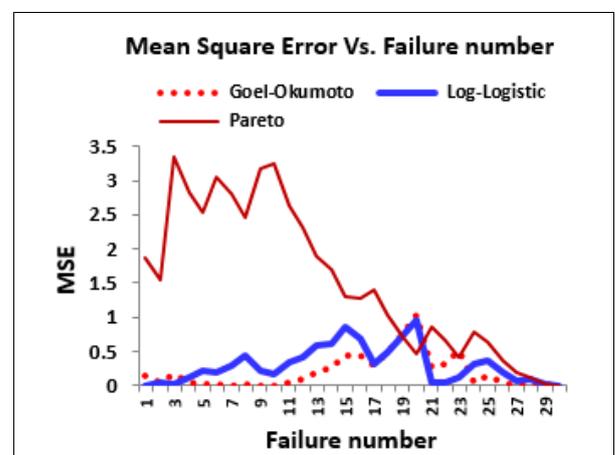
Fig 2. Transition of Mean Square Error

Figure 3 shows trends in the strength function, which is the failure occurring rate per defect. The Log-Logistic model has a property that increases and decreases in the form of failure occurring rate as the failure time passes, showing that it is the most efficient model in terms of reliability suitability.
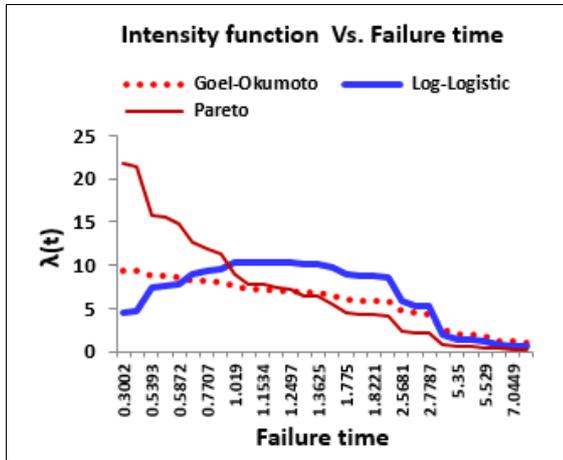

Fig 3. Transition of Intensity Function λ(t)

Figure 4 shows the pattern trend for the mean value function, which is the failure occurring expected value. In this figure, all models show underestimated from the difference between the true values, but the Pareto model has the biggest error estimation. Therefore, the Log-Logistic model is more efficient than the Pareto model.
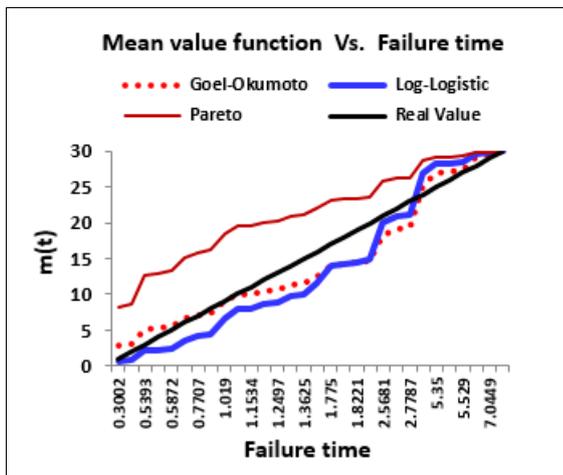

Fig 4. Pattern of Mean Value Function $m(t)$

Let analyze and evaluate the reliability performance of the proposed models for future mission time. Here, reliability is the probability that a software failure will occur when testing at $x_n = 738.68 \times 10^{-2} = 7.3868$, and no software failure will occur between confidence intervals $[x_n, \; x_n + \tau]$ where τ

is the future mission time. Therefore, the reliability of future mission time is as follows [14].

$$\hat{R}(\tau|x_n) = e^{-\int_{x_n}^{x_n+\tau} \lambda(\tau)d\tau}$$
$$= \exp[-\{m(\tau + x_n) - m(x_n)\}]$$
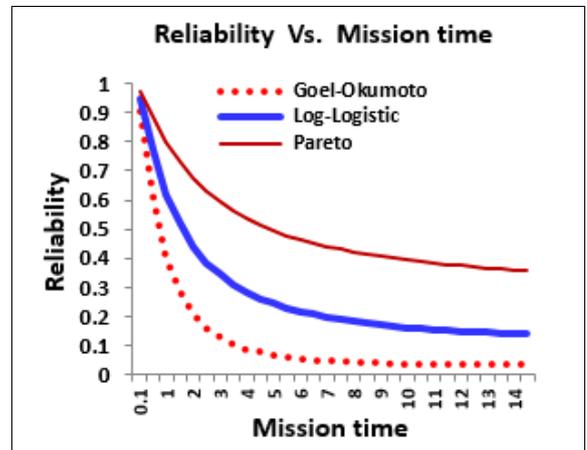$$= \exp[-\{m(\tau + 7.3868) - m(7.3868)\}] \quad (29)$$

Fig 5. Transition of Reliability $\hat{R}(t)$

As shown in Figure 5, the Pareto model shows a higher reliability trend than the other models in which the reliability decreases with the mission time, but the Goel-Okumoto basic model showed the biggest decreasing trend. That is, the Pareto model is more efficient than the other models because of the highest value in terms of future reliability.

4. CONCLUSION

This study is based on the finite failure NHPP that no new defects occur during the removal or correction process of software defects. Also, the software reliability performance was evaluated by analyzing the reliability attributes factor(m(t), λ(t), R(t)) using non-exponential family distribution known as efficient distribution in the field of reliability.

The results of this study can be summarized as follows.
First, in the analysis of the strength function, the Log-Logistic model was the most efficient in terms of reliability suitability. This is because the failure occurring rate of the Log-Logistic distribution shows an increase and decreases as the failure time passes and the mean square error (MSE) is also small.

Second, in the analysis of the mean value function, all the proposed models showed slightly underestimation patterns in

the error estimation for true values, but the Pareto model shows the biggest error estimation. Therefore, the Log-Logistic model is more efficient than the Pareto model.

Third, in the analysis of mission reliability, the Pareto model shows a higher reliability trend than the other models in which the reliability decreases with the mission time. That is, the Pareto model is more effective than other models because of its high reliability.

As a result, we have newly analyzed the software reliability attributes of non-exponential family distributions, which have no previous research cases, and expect it to be used as a basic guideline for software developers to search for the optimal software reliability model. Also, further research will be needed to find the optimal model through the reliability performance evaluation after applying the same type of software failure time data to various reliability models.

Acknowledgements

REFERENCES

[1] L. Kuo and T. Y. Yang., "Bayesian Computation of Software Reliability", Journal of the American Statistical Association, Vol.91, pp.763-773, 1996.

[2] Pham H, Zhang X., "NHPP Software Reliability and Cost Models with testing coverage", Eur.J. Oper. Res. 145, pp.445-454, 2003.

[3] Gokhale, S. S. and Trivedi, K. S. A, "time/structure-based software reliability model", Annals of Software Engineer-ing. 8, pp. 85-121. 1999.

[4] H. C. KIM, "A Comparative Study on the Finite Failure Soft-ware Reliability Model with Modified Lindley Type Lifetime Distribution", International Journal of Engineering Resear-ch and Technology, Vol.12, No.6, pp.760-764, 2019.

[5] H. C. KIM, H. C. Shin, "A Comparative Study of Software Optimal Release Time Based on Gamma Exponential and Non- exponential Family Distribution Model", The Journal of Korea Society of Computer and Information, Vol.15, No.5, pp.125-132, 2010.

[6] T. J. Yang, "A Comparative Study on the Performance Attributes of Finite Failure NHPP Software Reliability Model with Logistic Distribution Property", International Journal of Engineering Research and Technology, Vol.13, No.3, pp.1890-1896, 2020.

[7] T. J. Yang, J.G. Park "A Comparative Study of the Soft-ware NHPP Based on Weibull Extension Distribution and Flexible Weibull Extension Distribution", International Journal of Soft Computing, Vol.11, No.4, pp. 276-281, 2016.

[8] Goel A L, Okumoto K, "Time-dependent fault detection rate model for software and other performance measures", IEEE Transactions on Software Engineering, Vol.28, pp.206-211, 1978

[9] T. J. Yang, "A Study on the Reliability Performance Analysis of Finite Failure NHPP Software Reliability Model Based on Weibull Life Distribution", International Journal of Engineering Research and Technology, Vol.12, No.11, pp.1890-1896, 2019.

[10] H. C. KIM, "A Study on the Utility of Finite-fault NHPP Soft-ware Reliability Model Applying Pareto Distribution Feature",Asia-pacific Journal of Multimedia Service, Vol.7, No.10, pp.915-923, 2017.

[11] R. Satya Prasad, K. R. H. Rao, and R.R.L Kantha, "Software Reliability Measuring using Modified Maximum Likelihood Estimation and SPC", International Journal of Computer Applications (0975-8887) Vol.21, No.7, pp.1-5, 2011.

[12] Kuei-Chen, C., Yu-Shiang, H. and Tzai-Zang, L., "A Study of Software reliability growth from the perspective of learning effects", Reliability Engineering and System Safety 93, pp. 1410-1421, 2008.

[13] Tae-Jin Yang, "The Analysis and Predict of Software Failure Time Based on Nonlinear Regression," Journal of Engineering and Applied Sciences, Vol. 13, Issue 12, pp. 4376-4380, 2018.

[14] Ye Zhang and Kaigui Wu, "Software Cost Model Considering Reliability and Time of Software in Use", Journal of Convergence Information Technology, Vol.7, No.13, pp.135-142, 2012.