International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 12, Number 12 (2019), pp. 2687-2691

A Specific Feature of SPC for Software Reliability Model using on Type-2 Gumbel Life Distribution

Kyung-Mo Yeon[1], Hee-Cheul Kim[2*]

[1]*Master Student, Department of Big-data AI, Namseoul University, South Korea.*

[2]*Associate Professor, Department of Industrial & Management Engineering, Namseoul University, South Korea.*

*ORCID of Author[2]: 0000-0002-4786-5672, *Corresponding Author*

Abstract

Reliability of software quality in software development process is an important issue. Statistical process control (SPC) in this area is a management method of software development process through application of statistical analysis including definition, measurement, control and improvement of software development process. In this study, the control mechanism using the mean value difference chart were proposed by evaluating the mean value function. The finite failure NHPP model with the shape parameter of Type-2 Gumbel distribution was used for the life-time distribution. The parameter estimation method is used the traditional maximum likelihood method. The software failure data used in this study were obtained by using the data except for abnormal values when an abnormal value was found through the Box plot. The results of this study using the mean value difference chart show that the lower the shape parameter of the Type-2 Gumbel distribution is the more efficient the model because estimation points is the higher than control lower limit. In conclusion, if the system approach using the successive difference of mean value chart can be analysed beforehand, it is possible to improve the quality of the software failure more.

Keywords: Software Reliability Model, Type-2 Gumbel Distribution, NHPP, Control- Limit,

I. INTRODUCTION

Software stability can be a fundamental and necessary factor affecting the reliability of computer systems. Software has a different aspect to hardware stability in terms of design attributes. Thus, the failure of computer system due to a software defect may result in tremendous loss of property to software users. Software reliability analysis techniques to reduce software defects during software development process are basic and essential. In this environment, the reliability requirements of the software operator and the minimum test cost must be met. If you can analyse the reliability trend of software in advance to manage the minimum cost of software testing execution, it may be an economic development practice. Therefore, in order to maintain the minimum cost during software testing, it may be possible to implement cost minimization by predicting and applying the reliability pattern of software. Therefore, the development of software development that predicts reliability, minimum cost, and estimated release time of software release timing to operator is an efficient development process considering time and economic advantages. Many software reliability prediction analysis models have been proposed in this field. Among the many models, the software reliability model based on the Non-Homogeneous Poisson Process (NHPP) [1] is a reliable software model in terms of defect detection analysis. It removes immediately when another defect occurs and has an assumption that a new defect is no occurrence. Goel and Okumoto [2] studied an exponential software reliability growth model using a mean value function with S-shaped or exponential shaped software cumulative defect counts. Using this model, Huang [3] studied the technique of analysing the software reliability using the generalized logistic testing effort function and the change-point parameter. Statistical process control (SPC) can help to improve the quality of software reliability by proactively monitoring software failures. Control charts are widely used tools for software development process management in the software industry [4]. In terms of the statistical process control, Rao, Prasad and Kantham were studied the mechanism using the process control chart based on the Half Logistics lifetime distribution [4]. In addition, Kim [5] was showed that a more efficient model should reduce testing costs and allow software to increase total cost benefits. The cost curves of Burr-Hatke-exponential distribution model used in this study were compared and analysed lifetime distribution of the NHPP software reliability model. Based on the preliminary study, this study was compared the approach using the statistical process control method for the finite-failure NHPP software model using lifetime distribution follows the shape parameter of the Type-2 Gumbel distribution.

II. BACKGROUND WORK

II.I Finite failure NHPP software model using the Type-2 Gumbel distribution

The Type-2 Gumbel distribution is one of the continuous distributions that can represent various reliability attributes. The probability density function $f(t)$ and the cumulative distribution $F(t)$ consist of the shape parameter (a) and the shape parameter (b) [6].

$$f(t) = a\,b\,t^{-a-1}\,e^{-bt^{-a}}\ ,\ F(t) = e^{-bt^{-a}} \quad (a>0,\ b>0,\quad t>0) \qquad (1)$$

In this paper, we try to compare the reliability attributes by using the shape parameters ($a=1$, $a=2$ and $a=3$).
In the finite-failure NHPP model, the intensity function and the mean value function of the NHPP in the Type-2 Gumbel distribution model using the equations (1) are expressed as next forms [7].

$$\lambda(t|\theta,a,b)=\theta f(t)=\theta\, a\, b\, t^{-a-1}\, e^{-bt^{-a}} \quad (2)$$

$$m(t\,|\,\theta,a,b)=\theta F(t)=\theta\, e^{-bt^{-a}} \quad (3)$$

If the time truncated model is used to the observation time $(0,t]$, the likelihood function can be derived by the following equation using the equations (2) and (3) [7, 8].

$$\begin{aligned}L_{NHPP}(\Theta\,|\,\underline{x}) &= \prod_{i=1}^{n}\lambda(x_i)\,\exp[-m(x_n)]\\ &= \prod_{i=1}^{n}\theta\left[a\,b\,x_i^{-a-1}\,e^{-b x_i^{-a}}\right]\times\exp\left[-\theta\, e^{-b x_n^{-a}}\right]\end{aligned} \quad (4)$$

Note. $\underline{x}=(x_1\le x_2\le x_3\le\ldots\le x_n)$, $i=1,2,\ldots,n$, $\Theta=\{\theta,a,b\}$ indicates parameter space. Therefore, the log-likelihood function for using the maximum likelihood estimation is derived as follows [8].

$$\begin{aligned}&\ln L_{NHPP}(\Theta\,|\,\underline{x})=\\ &n\ln\theta+n\ln a+n\ln b\\ &-(a+1)\sum_{i=1}^{n}\ln x_i-b\sum_{i=1}^{n}x_i^{-a}-\theta\, e^{-b x_n^{-a}}\end{aligned} \quad (5)$$

When the shape parameter (a) is fixed, the estimator $\hat{\theta}_{MLE}$ and $\hat{b}_{MLE}$ must be meted the following condition for the maximum likelihood estimation about each parameter using equation (5).

$$\frac{\partial\ln L_{NHPP}(\Theta\,|\,\underline{x})}{\partial\theta}=\frac{n}{\theta}-e^{-b x_n^{-a}}=0 \quad (6)$$

In equation (6), solving for θ , $\hat{\theta}_{MLE}=\dfrac{n}{e^{-b x_n^{-a}}}$

$$\frac{\partial\ln L_{NHPP}(\Theta\,|\,\underline{x})}{\partial b}=\frac{n}{b}-\sum_{i=1}^{n}x_n^{-a}+\hat{\theta}_{MLE}\,x_n^{-a}\,e^{-b x_n^{-a}}=0 \quad (7)$$

II. II A statistical process control method based on the shape parameters of the Type-2 Gumbel distribution

Statistical process control is used to check whether the process is in a stable state or to keep the process stable. The control chart reasonably distinguishes between chance and assignable causes and identifies the cause of the abnormal cause. According to the result, it is the purpose of the control chart to perform the necessary steps to maintain the process in a stable state [4]. The control limits of the control chart consist of upper control limit (UCL), center line (CL) and low control limit (LCL). The case of the control chart goes beyond the control limit, can be made-up for the aim of improving the quality as well as maintaining the quality by detecting the process abnormality and implementing the measures to prevent recurrence [4]. We can be estimated UCL (t_U), low control limit LCL (t_L), and center line CL (t_C) by applying the standard probability of 6-sigma (0.99865, 0.00135 and 0.5) for the distribution function of the lifetime distribution ($F(t)$) for the software reliability model [4]. Therefore, in this paper, using the distribution function of the Type-2 Gumbel distribution the upper limit is as follows [4].

$$F(t)=e^{-bt^{-a}}=0.99865 \quad (8)$$

In terms of $t>0$, the upper control limit is derived as follows using expressing (8) [4, 9].

$$t=\exp\left(\frac{\ln\left[\ln(0.99865/\text{-b})\right]}{-a}\right)=\mathrm{t_U} \quad (9)$$

Similar to the upper control limit, the center line and the lower control limit are derived as follows.

$$t=\exp\left(\frac{\ln\left[\ln(0.5/\text{-b})\right]}{-a}\right)=\mathrm{t_C} \quad (10)$$

$$t=\exp\left(\frac{\ln\left[\ln(0.00135/\text{-b})\right]}{-a}\right)=\mathrm{t_L} \quad (11)$$

Therefore, the upper control limit ($m(t_U)$), ($m(t_C)$) is the centred line and lower control limit ($m(t_L)$) are derived as follows [4, 9] using mean value function.

$$m(t_U)=\theta\exp(-bt_U^{-a}) \quad (12)$$

$$m(t_C)=\theta\exp(-bt_C^{-a}) \quad (13)$$

$$m(t_L)=\theta\exp(-bt_L^{-a}) \quad (14)$$

III. STATISTICAL PROCESS CONTROL ANALYSIS

Table 1. Failure time data

Failure Number	*Failure Time (hours)*	*Failure Number*	*Failure Time (hours)*
1	9	16	92
2	21	17	95
3	32	18	98
4	36	19	104
5	43	20	105
6	45	21	116
7	50	22	149
8	58	23	156
9	63	24	247
10	70	25	249
11	71	26	250
12	77	27	337
13	78	28	384
14	87	29	396
15	91	30	405
		31	**540**

International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 12, Number 12 (2019), pp. 2687-2691

In this section, we analyze the process attributes of the software reliability model proposed in this paper using the software failure time data [10]. In order to confirm the reliability of the data in terms of data, a trend test should be preceded [11]. In this study, a Box plot is used for trend analysis. In Figure 1, the result of the box plot trend test shows that the 31th data item is the extreme value, so in this study, only the 30th data is used for the parameter estimation except the 31th data item [12, 13].

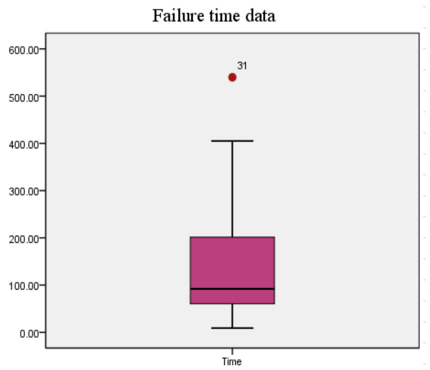

Fig. 1. Result of Box plot.

The approximation value of the parameters for the projected model was used the maximum likelihood method. In this paper, the mathematical change documents (Failure time (hours)$\times 0.01$) for shorten the parameter approximation was used. A consequence of the parameter approximation was attained from the Table 2. In this section, outcome of parameter estimation was listed in Table 2.

Table 2. Parameter estimation of the each model.

Model	Shape parameter	*MLE*
Type-2 Gumbel	$a=1$	$\hat{\theta}_{MLE}=35.876$ $\hat{b}_{MLE}=7.247\times10^{-1}$
	$a=2$	$\hat{\theta}_{MLE}=30.277$ $\hat{b}_{MLE}=1.507\times10^{-1}$
	$a=3$	$\hat{\theta}_{MLE}=30.001$ $\hat{b}_{MLE}=1.885\times10^{-2}$

Note. *MLE* : Maximum likelihood estimation.

Table 3. Control limits of the each model

Model	Shape parameter	Control limits		
		$m(t_U)$	$m(t_C)$	$m(t_L)$
Type-2 Gumbel	$a=1$	32.816	7.662	0.048
	$a=2$	30.236	15.139	0.041
	$a=3$	29.968	15.004	0.042

These calculations to estimate the root, solving mathematically, because the initial values were given 0.001 and 5.000 and tolerance value for the measurement of interval (10^{-5}) were specified, were accomplished repetition of 100 times using C-language checking acceptable convergent. The control limits using the results of the maximum likelihood method are summarized in Table 3.

Table 4 is a table of successive differences for the mean values, and the mean value chart for applying the resulting values to the control limits is summarized in Figures 2, 3 and 4.

Table 4. Successive difference of the mean value

Failure Number	Failure Time (i)	$m(i+1)-m(i), i=1,2,\cdots,29$ Successive difference		
		$a=1$	$a=2$	$a=3$
1	0.09	1.127253621	7.581426488	3.922005899
2	0.21	2.589384626	9.09508822	12.96217117
3	0.32	1.066428476	2.221915117	3.152549605
4	0.36	1.858754662	2.85766067	3.639170168
5	0.43	0.517382601	0.633010844	0.726389588
6	0.45	1.25268657	1.319103499	1.406646638
7	0.5	1.863655122	1.533569931	1.436750955
8	0.58	1.072163877	0.707534507	0.584323441
9	0.63	1.384467947	0.769024977	0.574248278
10	0.7	0.187141753	0.093207894	0.065090003
11	0.71	1.070218982	0.490406481	0.325666015
12	0.77	0.169926116	0.071663673	0.045167754
13	0.78	1.429303843	0.54385353	0.321710897
14	0.87	0.581658315	0.194137371	0.105474737
15	0.91	0.140651660	0.044811747	0.023617561
16	0.92	0.411029933	0.126432153	0.065130013
17	0.95	0.395279334	0.115492398	0.05751788
18	0.98	0.746413279	0.202855946	0.096320695
19	1.04	0.118996905	0.030597141	0.013994482
20	1.05	1.216830787	0.287656079	0.12443406
21	1.16	2.850174776	0.52107899	0.189651267
22	1.49	0.486663429	0.07105613	0.021879032
23	1.56	4.208016623	0.448049617	0.111085825
24	2.47	0.063114649	0.004793389	0.000895925
25	2.49	0.031233555	0.00235395	0.000437279
26	2.5	2.085983883	0.131814395	0.021398681
27	3.37	0.771561595	0.037177169	0.004786546
28	3.84	0.170346239	0.007442347	0.000880418
29	3.96	0.12172904	0.005154093	0.000593539
30	4.05	-------	-------	--------

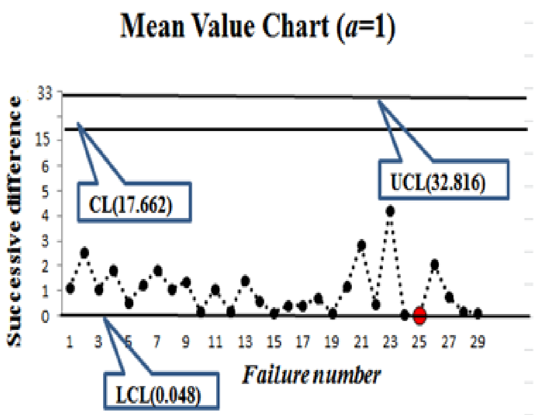

Fig. 2. A chart of the successive difference of mean value case of $a=1$

Table 4 and Figure 2 show the successive difference of mean value control chart when the shape parameter $a=1$ of the Type-2 Gumbel distribution. In this control chart, only the 25th estimation value appears to be lower than the lower control limit. Therefore, the systematic measures should be taken considering improvement of work method, training of workers

International Journal of Engineering Research and Technology. ISSN 0974-3154, Volume 12, Number 12 (2019), pp. 2687-2691

and an equalization of working environment.

In Figure 3, the shape parameter $a=2$ of the Type-2 Gumbel distribution is shown as 6 points in the successive difference of mean value chart with lower than the lower control limit. In addition, the case of the shape parameter $a=3$ of the Type-2 Gumbel distribution is shown as 8 points in the successive difference of mean value chart with lower than the lower control limit. As a result, in terms of efficiency comparison, the smaller the shape parameter of the Type-2 Gumbel distribution is the more efficient. This means that the failure interval time is not relatively long in the case of estimation points lower than the lower control limit. Therefore, the systematic measures using the successive difference of mean value chart can improve the quality of software failure.

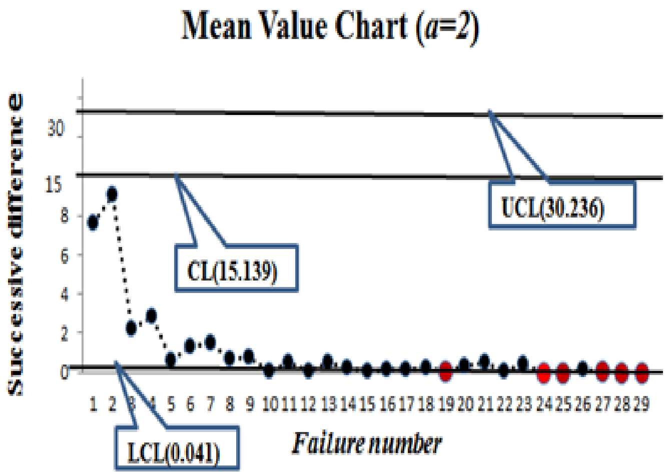

Fig. 3. A chart of the successive difference of mean value case of $a=2$

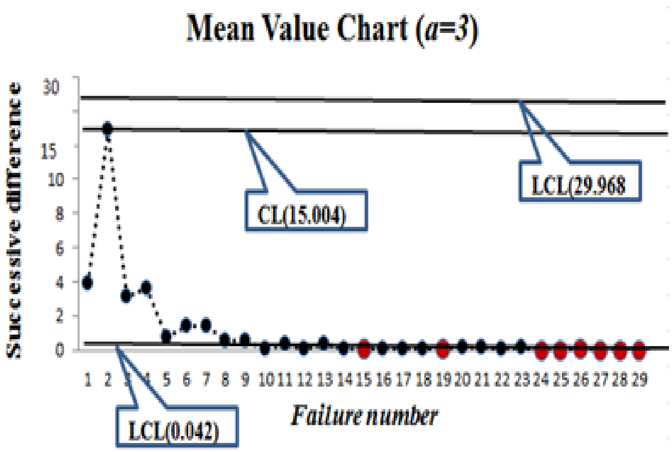

Fig. 4. A chart of the successive difference of mean value case of $a=3$

IV. DISCUSSION

Reliability of software quality in software development process is an important issue. Statistical process control in this area is a management method of software development process through application of statistical analysis including definition, measurement, control and improvement of software development process. Software reliability information can be used to select an efficient model by applying a scale that allows comparative evaluation if the failure occurrence attribute or failure occurrence trend can be quantitatively modelled in the final stage of software development execution. Therefore, the NHPP software model using the statistical process control process that can be applied to the cause of the software failure and the inspection tool was discussed by software operators. We have proposed the statistical process control chart control mechanism using the mean value function of the software NHPP trust model that follows the shape parameters of the Type-2 Gumbel distribution widely used in the reliability field and compared the reliability characteristics. As a result, in terms of efficiency comparison, the smaller the shape parameter of the Type-2 Gumbel distribution is the more efficient. This means that the failure interval time is not relatively long in the case of estimation points lower than the lower control limit.

V. CONCLUSION

The results of this study are relatively efficient for the comparison of the shape parameter efficiency of the Type-2 Gumbel distribution. The reason for this is that the failure interval time is not relatively long if estimation points are lower than the lower control limit. Therefore, if the system approach using the successive difference of mean value chart is recognized beforehand, it is possible to improve the quality of the software failure more. Through this study, it is predicted that software operators can help to recognize statistical process management information about software failure mode by applying process management considering various parameters of the shape of life distribution.

REFERENCES

[1] Gokhale, S.S., and Trivedi, K.S. A time/structure based software reliability model. Annals of Software Engineering 8. 1999; 5-12.

[2] Goel AL, Okumoto K. Time dependent error - detection rate model for software reliability and other performance measure. IEEE Trans Reliability, 1979; R-28(3):206–211.

[3] Huang C-Y. Performance analysis of software reliability growth models with testing-effort and change-point. J. Syst. Software. 2005; 75: 181-194.

[4] K, H Rao, R. S, Prasad and. R. L. Kantham, Software Reliability Measuring using Modified Maximum Likelihood Estimation and SPC. International Journal of Computer Applications (0975 -.8887). 2011; 21(7): 1-5.

[5] Hee-Cheul Kim. A Comparative Study on the Cost of Software Development Model Based on Burr–Hatke-Exponential Distribution. International Journal of

Engineering Research and Technology. 2019; 12(11): 2036-2040.

[6] https://en.wikipedia.org/wiki/Type-Gumbel_distribution

[7] Hong-kyu Kwon, Hee-cheul Kim. A Comparative Software Development Cost Model based on Hazard Function of Lindley Distribution. Information, 2016; 19 (11(A)): 5137-5144.

[8] Kim H-C. A Performance Analysis of Software Reliability Model using Lomax and Gompertz Distribution Property, Indian Journal of Science and Technology, 2016; 9(20):1-6.

[9] K, H Rao, R. S, Prasad and R. L. Kantham. Assessing Software Reliability Using SPC – An Order Statistics Approach, International Journal of Computer Science. Engineering and Applications. 2011; 1(4): 606-619.

[10] Kuo, L. and Yang, T. Y. Bayesian Computation of Software Reliability. Journal of the American Statistical Association. 1996; 91: 763 -773.

[11] K. Kanoun and J. C. Laprie, Handbook of Software Reliability Engineering' McGraw-Hill New York, NY, 1996; 401-437.

[12] Kuei-Chen, C., Yeu-Shiang, H., and Tzai-Zang, L. A study of software reliability growth from the perspective of learning effects. Reliability Engineering and System Safety. 2008; 93:1410–1421.

[13] Hee-Cheul Kim and Kyung-Soo Kim. Software Development Cost Model based on NHPP Gompertz Distribution. Indian Journal of Science and Technology. 2015; 8(12):1-¬5.