VOL. 17, NO. 6, MARCH 2022 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

COMPARATIVE ANALYSIS ON THE COST OF SOFTWARE DEVELOPMENT MODEL BASED ON WEIBULL FAMILY LIFETIME DISTRIBUTION

Hyo-Jeong Bae
Department of Drone and GIS Engineering, Namseoul University, Daehak-ro, Seonghwan-eup, Seobuk-gu, Cheonan-city, Chungnam, Korea
E-Mail: baehj@nsu.ac.kr

ABSTRACT

In this study, properties of software development cost were analyzed by applying the Weibull family lifetime distributions (Lindley, Rayleigh, Type-2 Gumbel) which are utilized in the reliability evaluation field to the software development model. Also, the Weibull family distribution models were compared with the Goel-Okumoto basic model to verify cost property, and the optimal development cost model was presented. For this study, a total solution was performed using software failure time data generated during desktop application operation, parameter calculations were solved using the maximum likelihood estimation (MLE) method, and nonlinear equations were calculated using the binary method. As a result, first, when the testing cost per unit time and the cost of eliminating a single fault detected during the development testing process increase, the development cost increases, but the release time does not change. However, if the fault correction cost detected by the operator during normal system operation increases, the development cost increased along with the delay of the release time. Second, it can be confirmed that the Lindley distribution model is efficient among the proposed models as it has the best performance in terms of development cost and releasing time. Third, if software developers and operators can use this analytical information efficiently, they can explore the cost of economic development by predicting relevant attributes.

Keywords: lindley, cost attributes, rayleigh, software development model, type-2 gumbel, weibull family distribution.

1. INTRODUCTION

With the rapid growth of software convergence technology, the most important issue is to develop reliable software that can accurately process various and complex large amounts of big data without failure. The most important problem in the process of developing such software is the development cost. Therefore, the problem of developing reliable software at an economical cost becomes the most important research topic for software developers. For this reason, studies on software reliability and software development cost are still being actively conducted. Recently, software developers and researchers are actively researching to find the most economical software development cost together with software reliability that determines software quality [1]. Recently, to analyze and predict the reliability performance of software, a new type of software reliability model using the Non-homogeneous Poisson process (NHPP) has been presented. In particular, to estimate the reliability performance, the software reliability models based on finite failure NHPP model using the mean value function were developed [2]. Chatterjee and Singh [3] analyzed the NHPP software reliability and optimal release policy with logistic-exponential testing coverage, while Bajta and Idri, Fernandez-Aleman [4] presented and explained the techniques based on the software cost estimation methods for effective software development. Rashid and Nisar, Mahmood [5] presented a comparative study on the software development cost estimation technique with a software life-cycle model. Kim [6] compared the cost properties of the software development model using Gompertz distribution. Also, Yang [7] compared and presented the research results on the characteristics of the NHPP software reliability model using the Weibull family lifetime distribution, which has not been studied so far.

Therefore, in this study, the Weibull family lifetime distribution model, which is frequently utilized in the reliability evaluation test field, will be applied to the software development model to analyze the properties of development costs and present a new optimal cost model.

2. RELATED RESEARCH

2.1 NHPP Software Reliability Model

The Poisson distribution is a probabilistic model for the number of occurrences of an event in a given time or event domain. It is a distribution model of a random variable with a very low probability of a specific event occurring among many events. Accordingly, in the NHPP model, if N(t) is the cumulative number of software failure times found up to time t, then m(t) is expressed as an average value function representing the expected value of fault occurrence. That is, the equation of the NHPP software reliability model is as follows. That is

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

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

Also, the intensity function $\lambda(t)$ means the fault occurrence rate per defect.

ARPN Journal of Engineering and Applied Sciences

$$m(t) = \int_0^t \lambda(s)ds \quad (2)$$

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

This study is based on the finite failure NHPP model that does not occur new defects during the removal or correction process of software defects.

Therefore, when the residual fault value in the finite failure software system is θ, the m(t) and λ(t) functions are as follows:

$$m(\theta, b) = \theta F \quad (4)$$
$$\lambda(\theta, b) = \theta F(t)' = \theta f(t) \quad (5)$$

The likelihood function of the finite failure NHPP model using Equations (4) and (5) is as bellows:

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

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

2.2 Goel-Okumoto: NHPP Basic Model

The Goel-Okumoto model is the best-known basic type among the Weibull family lifetime distribution. In the finite failure NHPP, if f(t) is a probability density function (PDF) and F(t) is a cumulative distribution function (CDF), then the mean value function $m(t)$ and the intensity function $\lambda(t)$ are as follows.

$$m(\theta, b) = \theta F(t) = \theta(1 - e^{-bt}) \quad (7)$$
$$\lambda(\theta, b) = \theta f(t) = \theta b e^{-bt} \quad (8)$$

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

After substituting Equations (7) and (8) into Equation (6) to get the likelihood function, and then taking the logarithm of both sides, the log-likelihood function can be solved as follows:

$$ln(\Theta|\underline{x}) = nln\theta + nlnb - b\sum_{k=1}^{n} x_k - \theta(1 - e^{-bx_n}) \quad (9)$$

Thus, the calculation of the parameters $\hat{\theta}_{MLE}$ and $\hat{b}_{MLE}$ using maximum likelihood estimation (MLE) in Equations (10) and (11) can be solved by the bisection method.

$$\frac{(\Theta|\underline{x})}{\partial\theta} = \frac{n}{\hat{\theta}} - 1 + e^{-\hat{b}x_n} = 0 \quad (10)$$

$$\frac{(\Theta|\underline{x})}{\partial b} = \frac{n}{\hat{b}} - \sum_{i=1}^{n} x_n - \hat{\theta}x_n e^{-\hat{b}x_n} = 0 \quad (11)$$

2.3 Lindley Distribution: NHPP Model

The Lindley distribution is widely known as a suitable model in the reliability evaluation field tests among the Weibull family distribution.

In the finite failure NHPP, if f(t) is a probability density function and F(t) is a cumulative distribution function, then the average value function $m(t)$ and the intensity function $\lambda(t)$ are as follows [7] [8].

$$m(\theta, b) = \theta F(t) = \theta\left[1 - \left(\frac{b+1+bt}{b+1}\right) \times e^{-bt}\right] \quad (12)$$

$$\lambda(\theta, b) = \theta f(t) = \theta\left[\frac{b^2}{b+1}(1+t) \times e^{-bt}\right] \quad (13)$$

After substituting Equations (12) and (13) into Equation (6) to solve the likelihood function, and then taking the logarithm of both sides, the log-likelihood function can be solved as follows:

$$ln(\Theta|\underline{x}) = -\theta\left[1 - \left(\frac{b+1+bt}{b+1}\right) \times e^{-bt}\right] + nln\theta$$
$$+2nlnb - nln(b+1) + \sum_{i=1}^{n}(1 + x_i) - b\sum_{i=1}^{n} x_i \quad (14)$$

Thus, the calculation of the parameters $\hat{\theta}_{MLE}$ and $\hat{b}_{MLE}$ using maximum likelihood estimation (MLE) in Equations (15) and (16) can be solved by the bisection method as bellows.

$$\frac{(\Theta|\underline{x})}{\partial\theta} = \frac{n}{\theta} - \left[1 - \left(\frac{b+1+bt}{b+1}\right) \times e^{-bt}\right] = 0 \quad (15)$$

$$\frac{(\Theta|\underline{x})}{\partial b} = \frac{2n}{b} - \frac{n}{b+1} - \sum_{i=1}^{n} x_i$$
$$-\theta e^{-bx_n}\left(x_n - b^2 x_n^2 + b - b^3 x_n^3 - b^3\right) = 0 \quad (16)$$

2.4 Rayleigh Distribution: NHPP Model

The Rayleigh distribution is widely used in the lifetime reliability test among the Weibull family distribution. In the finite failure NHPP, if f(t) is a probability density function and F(t) is a cumulative distribution function, then the average value function $m(t)$ and the intensity function $\lambda(t)$ are as follows [7] [9].

$$m(\theta, b) = \theta F(t) = \theta\left(1 - e^{-bt^2}\right) \quad (17)$$

$$\lambda(\theta, b) = \theta f(t) = 2\theta bte^{-bt^2} \quad (18)$$

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

After substituting Equations (17) and (18) into Equation (6) to solve the likelihood function, if arranged in the same way as in Equation (9), the log-likelihood function can be solved as follows:

$$ln(\Theta|\underline{x}) = nln2 + nln\theta + nlnb + \sum_{i=1}^{n} lnx_i - b\sum_{i=1}^{n} x_i^2 - \theta\left(1 - e^{-bx_n^2}\right) \quad (19)$$

Note. $\underline{x} = (0 \leq x_1 \leq x_2 \leq \cdots \leq x_n,$

Θ *is parameter space.*

Thus, the calculation of the parameters $\hat{\theta}_{MLE}$ and $\hat{b}_{MLE}$ using maximum likelihood estimation (MLE) in Equations (20) and (21) can be solved by the bisection method.

$$\frac{(\Theta|\underline{x})}{\partial\theta} = \frac{n}{\hat{\theta}} - 1 + exp\left(-\hat{b}x_n^2\right) = 0 \quad (20)$$

$$\frac{(\Theta|\underline{x})}{\partial b} = \frac{n}{\hat{b}} - \sum_{i=1}^{n} x_i^2 - \hat{\theta}x_n^2\, exp\left(-\hat{b}x_n^2\right) = 0 \quad (21)$$

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

2.5 Type-2 Gumbel Distribution: NHPP Model

The Type-2 Gumbel distribution is widely used in the lifetime reliability test among the Weibull family distribution.

In the finite failure NHPP, if f(t) is a probability density function and F(t) is a cumulative distribution function, then the average value function $m(t)$ and the intensity function $\lambda(t)$ are as follows [7] [10].

$$m(\theta, a, b) = \theta F(t) = \theta\left(e^{-bt^{-a}}\right) \quad (22)$$

$$\lambda(\theta, a, b) = \theta f(t) = \theta\left(abt^{-a-1}e^{-bt^{-a}}\right) \quad (23)$$

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

After substituting Equations (22) and (23) into Equation (6) to obtain the likelihood function, if arranged in the same way as in Equation (9), the log-likelihood function can be solved as follows:

$$ln(\Theta|\underline{x}) = nln\theta + nlna + nlnb - (a+1)\sum_{i=1}^{n} lnx_i - b\sum_{i=1}^{n} x_i^{-a} - \theta e^{-bx_n^{-a}} \quad (24)$$

Thus, the calculation of the parameters $\hat{\theta}_{MLE}$ and $\hat{b}_{MLE}$ using maximum likelihood estimation (MLE) when the shape parameter a=2 in Equations (25) and (26) can be solved by the bisection method.

$$\frac{(\Theta|\underline{x})}{\partial\theta} = \frac{n}{\theta} - e^{-bx_n^{-2}} = 0 \quad (25)$$

$$\frac{(\Theta|\underline{x})}{\partial b} = \frac{n}{b} - \sum_{i=1}^{n} x_i^{-2} + \theta x_n^{-2}e^{-bx_n^{-2}} = 0 \quad (26)$$

2.6 Software Development Model using the NHPP Model

When the mean value function m(t) of the proposed NHPP model is applied to the software development cost model, it is composed of the sum of each cost element as follows [11].

$$E_t = E_1 + E_2 + E_3 + E_4 = E_1 + C_2 \times t + C_3 \times m(t) + C_4 \times [m(t+t') - m(t)] \quad (27)$$

Note that E_t is the total cost of software development.

① E_1 is the initial development cost as a constant.

② E_2 is the testing cost per unit time.

$$E_2 = C_2 \times t \quad (28)$$

Note that C_2 is the testing cost per unit time.

③ E_3 is the cost of removing one fault.

$$E_3 = C_3 \times m(t) \quad (29)$$

Note that C_3 refers to the cost of eliminating one fault detected in the development testing stage, and the mean values function m(t) refers to the expected value of failure.

④ E_4 is the cost of removing all remaining faults in the software system.

$$E_4 = C_4 \times [m(t+t') - m(t)] \quad (30)$$

ARPN Journal of Engineering and Applied Sciences

Note that C_4 is the fault correction cost found in the software operation stage. and t' is the normal operating time of the software system.

Also, it can be seen that the time point at which the software development cost is the minimum becomes the optimal software releasing time point. The optimal software releasing time point is as follows:

$$\frac{\partial E_t}{\partial t} = E' = (E_1 + E_2 + E_3 + E_4)' = 0 \qquad (31)$$

3. SOLUTIONS USING SOFTWARE FAILURE TIME DATA

As shown in Table-1, the cost properties of the proposed distribution model are compared and analyzed using the software failure time data [12] that occurred 30 times during the testing time.

Table-1. Software failure time data.

Failure Number	Failure Time (hours)	Failure Time (hours) × 10^{-2}	Failure Number	Failure Time (hours)	Failure Time (hours) × 10^{-2}
1	30.02	0.3	16	151.78	1.51
2	31.46	0.31	17	177.50	1.77
3	53.93	0.53	18	180.29	1.8
4	55.29	0.55	19	182.21	1.82
5	58.72	0.58	20	186.34	1.86
6	71.92	0.71	21	256.81	2.56
7	77.07	0.77	22	273.88	2.73
8	80.90	0.8	23	277.87	2.77
9	101.90	1.01	24	453.93	4.53
10	114.87	1.14	25	535	5.35
11	115.34	1.15	26	537.27	5.37
12	121.57	1.21	27	552.90	5.52
13	124.97	1.24	28	673.68	6.73
14	134.07	1.34	29	704.49	7.04
15	136.25	1.36	30	738.68	7.38

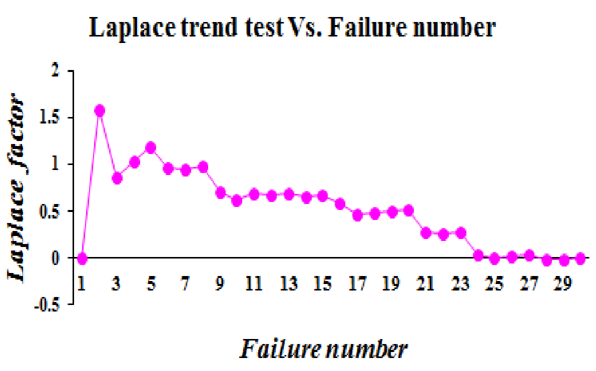

Figure-1. Results of the Laplace trend test.

The software failure time data applied in this paper means random faults caused by software design and analysis errors and insufficient testing during the normal system operation of desktop applications.

In Figure-1, the estimated result of the Laplace trend test existed between 0 and 2. Therefore, this failure data can be used because there are no extreme values [13].

In this study, the parameter estimation was calculated using the maximum likelihood estimation (MLE) method with numerical conversion data to facilitate parameter estimation as in Table-1 [14].

Table-2. below represented the parameter estimated results of the proposed models using MLE.

Type	NHPP Model	MLE (Maximum Likelihood Estimation)	
Basic Model	Goel-Okumoto	$\hat{\theta} = 33.4092$	$\hat{b} = 0.3090$
Weibull family Lifetime Distribution	Lindley	$\hat{\theta} = 30.4691$	$\hat{b} = 1.3460$
	Rayleigh	$\hat{\theta} = 24.0116$	$\hat{b} = 0.3707$
	Type-2 Gumbel	$\hat{\theta} = 30.3852$	$\hat{b} = 0.6960$

ARPN Journal of Engineering and Applied Sciences

The calculating methods of the mean value function that determines the cost attributes of software development cost are shown in Table-3 [15]. For this purpose, the calculation method of the mean value function of the proposed NHPP model is also specified.

Table-3. Calculating methods of the mean value function m(t)

Type	NHPP model	$m(t)$ of Weibull Family Lifetime Distribution	$m(t)$ of Software Development Cost Model
Basic model	Goel-Okumoto	$m(t) = \hat{\theta}\left(1 - e^{-\hat{b}t}\right)$	$E_3 = C_3 \times m(t)$ $E_4 = C_4 \times [m(t + t') - m(t)]$
Weibull family Lifetime distribution	Lindley	$m(t) = \theta\left[1 - \left(\frac{b + 1 + bt}{b + 1}\right) \times e^{-bt}\right]$	
	Rayleigh	$m(t) = \theta(1 - e^{-bt^2})$	
	Type-2 Gumbel	$m(t) = \theta\left(e^{-bt^{-a}}\right)$	

Also, the estimated result values of mean value function $m(t)$ are shown in Table-4. After all, these estimated values of Table-4 can be substituted and used as the mean value function m(t) to calculate the total software development cost of the proposed distribution model.

VOL. 17, NO. 6, MARCH 2022 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

Table-4. Estimated result values of mean value function m(t).

Failure Number	Release Time (hours)	Basic Model	Weibull family Lifetime Distribution		
		Goel-Okumoto	Lindley	Rayleigh	Type-2 Gumbel
1	0.3	2.957820066	6.620405287	0.787882862	0.013308163
2	0.6	5.653775151	12.20484333	2.999781883	4.395680325
3	0.9	8.111048972	16.71095179	6.228118036	12.86739238
4	1.2	10.35077272	20.23885719	9.931970353	18.7392886
5	1.5	12.39220679	22.94117459	13.58393837	22.3007947
6	1.8	14.25290636	24.97708068	16.78716488	24.51146171
7	2.1	15.94887243	26.49110959	19.32948446	25.94898561
8	2.4	17.49468937	27.60531347	21.17301682	26.9268065
9	2.7	18.90365034	28.418248	22.4017612	27.61840496
10	3.0	20.18787163	29.00711344	23.15754779	28.12397216
11	3.3	21.35839683	29.43106636	23.58775436	28.50398215
12	3.6	22.4252918	29.73468794	23.81483299	28.79644506
13	3.9	23.39773125	29.95113929	23.92614881	29.02612478
14	4.2	24.28407762	30.10482899	23.97688596	29.20967136
15	4.5	25.091953	30.21356858	23.99840791	29.35859292
16	4.8	25.82830466	30.29026193	24.00691032	29.4810391
17	5.1	26.49946483	30.34420023	24.01004047	29.58290643
18	5.4	27.11120511	30.38203817	24.01111486	29.66854326
19	5.7	27.66878613	30.40852029	24.01145882	29.74121136
20	6.0	28.17700277	30.42701567	24.01156157	29.80339504
21	6.3	28.64022541	30.43990817	24.01159021	29.85701273
22	6.6	29.06243752	30.44887924	24.01159767	29.90356447
23	6.9	29.44726988	30.45511148	24.01159948	29.94423655
24	7.2	29.79803184	30.45943454	24.01159989	29.97997689
25	7.5	30.11773975	30.4624291	24.01159998	30.01155021
26	7.8	30.40914293	30.46450074	24.0116	30.0395788
27	8.1	30.67474729	30.46593216	24.0116	30.06457321
28	8.4	30.91683686	30.46692012	24.0116	30.08695539
29	8.7	31.1374935	30.46760127	24.0116	30.10707655
30	9.0	31.33861472	30.46807043	24.0116	30.12523086

In this study, to test the same cost conditions as the actual development environment, the software development cost is presented as [Assumption 1] ~ [Assumption 5].

3.1 [Assumption 1: Basic Conditions]

$$E_1 = 50\$,\ C_2 = 5\$,\ C_3 = 1.5\$,\ C_4 = 10\$,\ t' = 50 \quad (32)$$

If substituting the cost parameter value given in Equation (32) and the estimated result of the mean value function m(t) in Table-4 into the software development cost model equation as in Equation (27), the test simulation result is as bellows:

VOL. 17, NO. 6, MARCH 2022 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

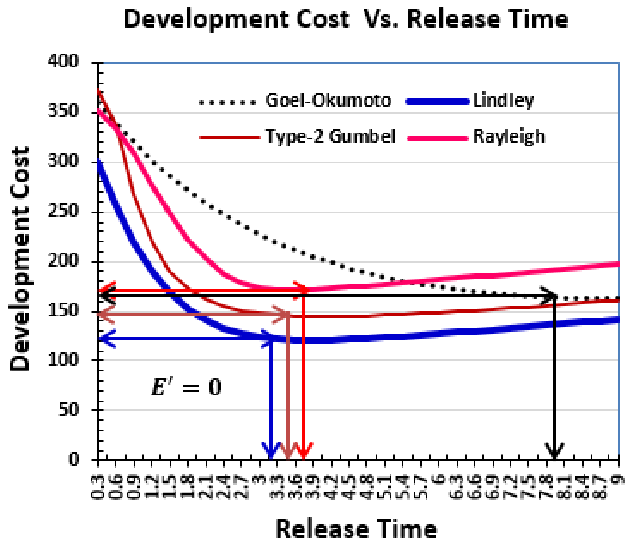

Figure-2. Shape of the cost curve tested by the condition of [Assumption 1].

Analysis of Figure-2 represented that the development cost curve shows a decreasing shape in the initial stage and a constant shape for a short period, but eventually increases with the release time. It is confirmed that the cost increases because the probability of finding a residual failure in the software gradually decreases in the later stage than in the early stage [16].

As shown in Figure-2, the Lindley model showed the best performance among the proposed models (Goel-Okumoto, Rayleigh, Type-2 Gumbel) as it showed the properties of the lowest development cost and the fastest release time.

3.2 [Assumption 2: Assume that only the Cost C_2 has Doubled Under the Conditions of Assumption 1]

$$E_1 = 50\$,\ C_2 = 10\$,\ C_3 = 1.5\$,\ C_4 = 10\$,\ t' = 50 \quad (33)$$

[Assumption 2] is a case where only the cost (C_2) has doubled under the condition of [Assumption 1].

If substituting the cost parameter value given in Equation (33) and the estimated result of the mean value function m(t) in Table-4 into the software development cost model equation as in Equation (27), the test simulation result is as shown in Figure-3.

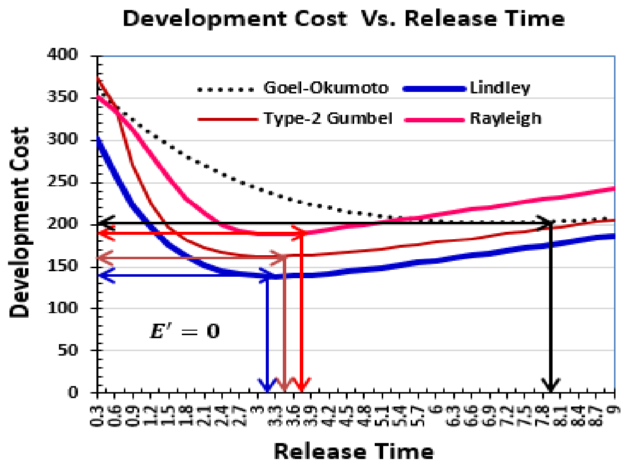

Figure-3. Shape of the cost curve tested by the condition of [Assumption 2].

Analysis of Figure-3 showed that the development cost increased but the release time did not change at all. Thus, it was confirmed that accurate testing is necessary so that the cost does not increase.

Also, as a result of simulation analysis, the Lindley model with the lowest development cost and the fastest release time among the proposed models was the best.

3.3 [Assumption 3: Assume that only the Cost C_3 has Doubled under the Conditions of Assumption 1]

$$E_1 = 50\$,\ C_2 = 5\$,\ C_3 = 3\$,\ C_4 = 10\$,\ t' = 50 \quad (34)$$

[Assumption 3] is a case where only the cost (C_3) has doubled under the condition of [Assumption 1].

If substituting the cost parameter value given in Equation (34) and the estimated result of the mean value function m(t) in Table-4 into the software development cost model equation as in Equation (27), the test simulation result is as shown in Figure-4. Analysis of Figure 4 showed that the development cost increased but the release time did not change at all.

In this case, it was confirmed that as many faults as possible should be eliminated at once so that the cost of removing one fault does not increase during the testing stage of the development process.

Also, as a result of simulation analysis, the Lindley model with the lowest development cost and the fastest release time among the proposed models was the best.

VOL. 17, NO. 6, MARCH 2022 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences

www.arpnjournals.com

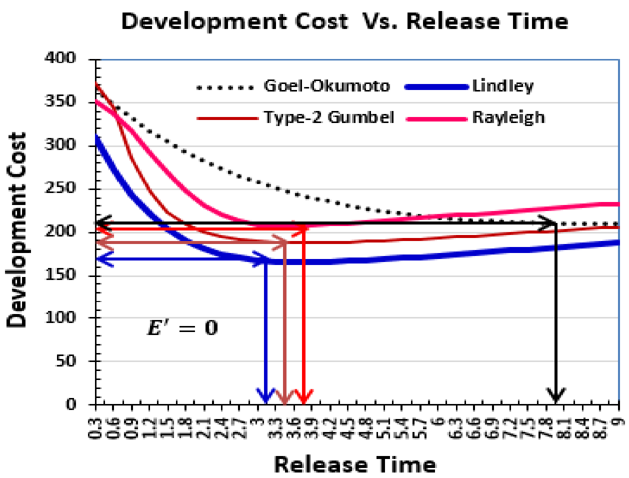

Figure-4. Shape of the cost curve tested by the condition of [Assumption 3].

3.4 [Assumption 4: Assume that only the Cost C_4 has Doubled under the Conditions of Assumption 1]

$$E_1 = 50\$,\ C_2 = 5\$,\ C_3 = 1.5\$,\ C_4 = 20\$,\ t' = 50 \quad (35)$$

[Assumption 4] is a case where only the cost (C_4) has doubled under the condition of [Assumption 1].

If substituting the cost parameter value given in Equation (35) and the estimated result of the mean value function m(t) in Table-4 into the software development cost model equation as in Equation (27), the test simulation result is as shown in Figure-5.

Analyzing Figure-5, it can be seen that the release time is also delayed as the development cost increases. Thus, in this case, it can be seen that to reduce possible faults before releasing the software, it is necessary to eliminate as many faults as possible in the testing stage rather than the actual operation stage [17].

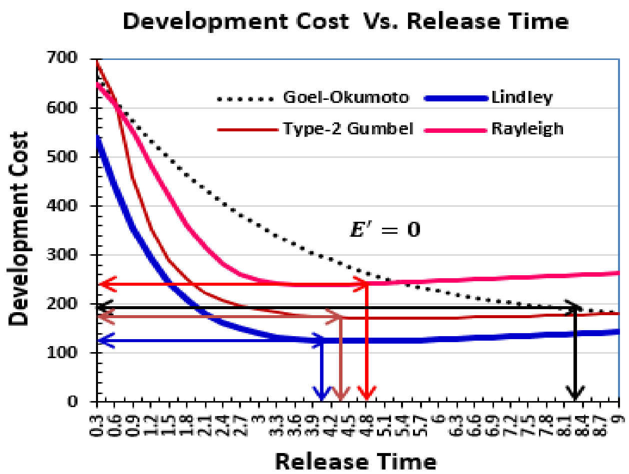

Figure-5. Shape of the cost curve tested by the condition of [Assumption 4].

As can be seen from the simulation results, the Lindley model was relatively efficient compared to the proposed model because the development cost was low and the release time was fast.

3.5 [Assumption 5: Assume that only the Time t' has Doubled under the Conditions of Assumption 1]

$$E_1 = 50\$,\ C_2 = 5\$,\ C_3 = 1.5\$,\ C_4 = 10\$,\ t' = 100 \quad (36)$$

[Assumption 5] is a case where only the time (t') has doubled under the conditions of [Assumption 1].

[Assumption 5] is a case where the time t' that the software system solution can be operated normally after releasing the software is twice increased in [Assumption 1].

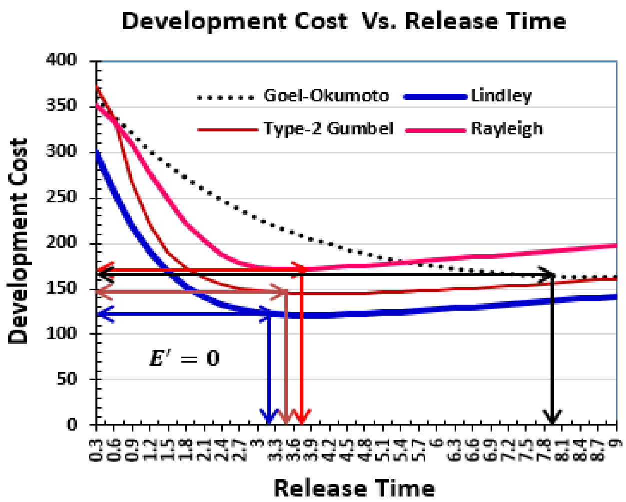

Figure-6. Shape of the cost curve tested by the condition of [Assumption 5].

As shown in Figure-6, it can be seen that the simulation result of assumption 5 is the same as the result of assumption 1 (basic condition). That is, it shows that even if the time t' increases, development cost and release time are not affected.

4. CONCLUSIONS

If software developers and operators can quantitatively analyze the attributes of software development costs during the testing process, they can effectively predict economic development costs. Therefore, in this study, the attributes of software development cost were analyzed and predicted based on the Weibull family distribution model.

The results of this study are as follows:

First, under the condition of Assumption 1, development cost decreased in the initial stage but increased again in the later stage. The reason is that the probability of finding residual faults in the later stages is gradually decreasing.

Second, in the testing process, if the test cost per unit time (C_2) and the cost of removing one fault (C_3)

increased, the development cost increased but the release time did not change at all. However, after the software system was released, if the fault correction cost (C_4) found by the operator increased, the development cost increased and the release time was also delayed.

Third, when the simulation results are analyzed comprehensively, the Lindley model showed the best performance among the proposed Weibull family distribution models because it has the characteristics of the lowest software development cost and the fastest release time.

In conclusion, if software developers and operators can efficiently use this research data, it is possible to predict the optimal release time together with the trend of development costs. Also, after exploring more diverse lifetime model distributions, additional research is needed to find the optimal cost model by applying the software failure time data applied in this paper to the development cost model.

ACKNOWLEDGEMENTS

Funding for this paper was provided by Namseoul University.

REFERENCES

[1] Gokhale S.S. and Trivedi K.S. 1999. A time/structure-based software reliability model. Annals of Software Engineering. 8(1): 85-121.

[2] Song K.Y., Chang I.H. and Pham H. 2017. A software reliability model with a Weibull fault detection rate function subject to operating environments. Applied Sciences. 7(10): 1-16.

[3] Chatterjee S. and Singh J. B. 2014. An NHPP based software reliability model and optimal release time policy with logistic-exponential test coverage under imperfect debugging. International Journal of System Assurance Engineering and Management. 5: 399-406.

[4] Bajta M. E. and Fernandez-Aleman J. L., Ros J. N. 2015. Software Cost Estimation for Global Software Development. 10th international Conference on Evaluation of Novel Software Approaches to Software Engineering. 197-206.

[5] Rashid J. and Nisar M. W, Mahmood T. 2020. A study of software Development Cost Estimation Technique and Models. Mehran University Research Journal of Engineering and Technology. 38(2): 413-431.

[6] Kim H.C. 2015. Software Development Model Based on NHPP Gompertz Distribution. Indian Journal of Science and Technology. 8(12): 1-5.

[7] Yang T.J. 2021. Comparative Study on the Performance Attributes of NHPP Software Reliability Model based on Weibull Family Distribution. International Journal of Performability Engineering. 17(4): 343-353.

[8] Yang T.J. 2021. Comparative study on the Attributes Analysis of Software Development Cost Model Based on Exponential-Type Lifetime Distribution. International Journal of Emerging Technology and Advanced Engineering. 11(10): 166-176.

[9] Yang T.J. 2019. 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. 12(11): 1890-1896.

[10] Yang T. J. 2019. A Comparative Study on Reliability Attributes of Software Reliability Model Based on Type-2 Gumbel and Erlang Life Distribution. Journal of Engineering and Applied Sciences. 14(10): 3366-3370.

[11] Zhang Y. and Wu K. 2012. Software Cost Model Considering Reliability and Time of Software in Use. Journal of Convergence Information Technology. 7(13): 135-142.

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

[13] Yang T. J. 2018. The Analysis and Predict of Software Failure Time Based on Nonlinear Regression. Journal of Engineering and Applied Sciences. 13(12): 4376-4380.

[14] Chiu K.C., Huang Y.S. and Lee T.Z. 2008. A Study of Software Reliability Growth from the Perspective of Learning Effects. Reliability Engineering & System Safety. 93(10): 1410-1421.

[15] Yang T. J. 2020. A Comparative Study on the Cost and Release Time of Software Development Model Based on Lindley-Type Distribution. International Journal of Engineering Research and Technology. 13(9): 2185-2190.

[16] Pham H. 2019. Distribution Function and Its Application in Software Reliability. International

Journal of Performability Engineering. 15(5): 1306-1313.

[17] Maza S. and Megouas O. 2021. Framework for Trustworthiness in Software Development. International Journal of Performability Engineering. 17(2): 241-252.