ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

PERFORMANCE ATTRIBUTES ANALYSIS OF SOFTWARE DEVELOPMENT COST MODEL WITH GAMMA FAMILY DISTRIBUTION CHARACTERISTICS

HYO JEONG BAE

Professor, Department of Drone and GIS Engineering, Namseoul University, 91 Daehak-ro, Seonghwan-eup, 31020, Seobuk-gu, Cheonan-si, Chungnam, Korea
E-mail: baehj@nsu.ac.kr

ABSTRACT

In this study, the performance attributes of the NHPP-type software development cost model with Gamma family distribution characteristics widely known to be suitable for reliability studies were newly analyzed and evaluated. Also, after verifying the cost characteristics by comparing the proposed model with the Goel-Okumoto basic model, the optimal model was also presented. For efficient research, randomly collected software failure time data was used, and the estimation solution for the parameters of the proposed model was computed by maximum likelihood estimation. Conclusively, first, as a result of analyzing the m(t) function that affects the performance properties of the development cost model, the Rayleigh model and the Goel-Okumoto basic model were found to be efficient among the proposed models because the error value in predicting the true value was small. Second, when analyzing the development cost properties, the Rayleigh model was found to be an efficient model with excellent performance. Third, as a result of evaluating performance attributes, it was concluded that the Rayleigh model showed the best performance in this work. Therefore, if a software developer utilizes this research data, it can be used as a fundamental design guideline for the attributes analysis of development cost together with research on improving reliability quality.

Keywords: *Erlang, Log-Logistic, Rayleigh, Software Development Cost Model, Performance Attributes.*

1. INTRODUCTION

In the present intelligent software era, since software application technologies are rapidly converging into various related industries, demand for reliable software development that can handle big data in various fields without defects is also increasing. However, the development cost of high-reliability software will be much higher than that of general application software. Therefore, developers are still conducting research and investment to find a research method capable of developing highly reliable software at the most economical cost. For this reason and purpose, many reliability models using NHPP (Non-homogeneous Poisson Process), which is known to be suitable for reliability prediction and analysis, are still being proposed [1]. Regarding research on the NHPP model, Goel and Okumoto [2] predicted error behavior that could occur when software was running, Huang [3] analyzed reliability performance utilizing a reliability attribute factor, and in particular, XIAO and DOHI [4] demonstrated the superiority of the Weibull-type model through property prediction with the Weibull distribution, which utilized the advantage of the Weibull distribution that can represent various patterns of software failure rate functions. Also, Kim [5] compared and analyzed the predictive power of software failure time using the finite failure NHPP reliability model, Pham [6] proposed a new reliability distribution function applying a failure rate function in the form of Vtub, and this function was used for software reliability modeling. Tokuno, Fukuda, and Yamada [7] investigated and explained the correlation between the characteristics of software reliability and system performance along with the probabilistic performance evaluation of software systems considering real-time properties using the NHPP model. Yang [8] newly explored the attribute factors influencing the development cost and optimal release timing using software development models with basic-Lindley distribution and modified-

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Lindley distribution characteristics. Also, Yang [9], paying attention to the exponential-type distribution characteristics, proposed a new algorithm that can analyze the cost property of a software development model applying the m(t) function of the NHPP reliability model. Also, after applying the proposed cost model to future reliability, the performance of the development cost model simulated by the proposed algorithm was newly evaluated.

Therefore, in this work, the Gamma family distribution, which is well known to be suitable for software reliability quality testing, is applied to the NHPP-type model. Based on this applied model, the cost performance is newly analyzed according to the proposed sequence, and the optimal model is also presented.

2. RELATED RESEARCH

2.1.1 NHPP model

The software reliability model in which software failures depend on the NHPP is classified as a model having a time domain. In this stochastic process, the parameter λ(t) represents the intensity function related to the software execution time.

Therefore, N(t) becomes a Poisson probability function having the mean value function m(t) as a parameter, as shown in Equation (1).

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

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

As such, time-related models can be explained as stochastic failure processes by NHPP. Thus, $m(t)$ and $\lambda(t)$ satisfy the relationship as shown in Equations (2) and (3).

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

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

These NHPP models are classified into finite failures in which failures do not occur during repairs and infinite failures in which failures continue to occur even during repairs. In this paper, we will develop this work based on the finite failure NHPP model by applying the actual software development situation.

2.1.2 NHPP software reliability model

The NHPP models assume that the expected value of a defect has a finite value given sufficient test time. When given sufficient testing time in the NHPP model, if the detectable residual failure rate is θ, the cumulative distribution function is F(t), and the probability density function is f(t), then m(t) and λ(t) can be expressed as the following functional expressions, respectively.

$$m(t|\theta,b)=\theta F(t) \quad (4)$$

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

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

Applying Equations (4) and (5), the likelihood function of the NHPP model is as follows.

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

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

2.2 Goel-Okumoto Basic NHPP Model

In the field of reliability attributes analysis and reliability performance evaluation, the Goel-Okumoto model is known as the basic model. In particular, in the Goel-Okumoto basic model, the lifetime distribution following the distribution of failure occurrence time per software defect assumes an exponential distribution.

Therefore, the property functions of the reliability performance are as follows [10].

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

$$\lambda(t|\theta,b)=\theta f(t)=\theta be^{-bt} \quad (8)$$

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

That is, if Equations (7) and (8) are substituted into Equation (6) and rearranged, the log likelihood function can be written as follows.

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

Therefore, using Equation (9), the estimators $\hat{\theta}_{MLE}$ and $\hat{b}_{MLE}$ for the parameters must satisfy the following conditional expression.

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

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

2.3 NHPP Erlang Distribution Model

Among the software reliability distributions, the Gamma distribution is most widely used in reliability data analysis because it can express various distributions according to the values of shape and scale parameters. Therefore, the attributes functions of the reliability performance are as follows [11].

$$m(t|\theta,b) = \theta\left[1 - e^{-bt}\sum_{i=0}^{a-1}\frac{(bt)^i}{i!}\right] \quad (12)$$

$$\lambda(t|\theta,b) = \theta\left[\frac{b^a}{\Gamma(a)}\, t^{a-1}e^{-bt}\right] \quad (13)$$

Note that $a, b > 0$, $a = 1,2,3\ldots.$, $t \in [0,\infty]$

The Erlang distribution belonging to the Gamma family distribution to be studied in this work considers the case where the shape parameter (a) is 2.

$$\ln L_{NHPP}(\Theta|\underline{x}) = nln\theta - nln\Gamma(a) + naln$$
$$-b\sum_{i=1}^{n} x_i + (a-1)\sum_{i=1}^{n} lnx_i - \theta$$
$$+\theta e^{-bx_n}\left(\sum_{i=0}^{a-1}\frac{(bx_n)^i}{i!}\right) \quad (14)$$

Therefore, in Equation (14), the estimators $\hat{\theta}_{MLE}$ and $\hat{b}_{MLE}$ for the parameters must satisfy the following Equations (15) and (16).

$$\frac{\partial \ln L_{NHPP}(\Theta|\underline{x})}{\partial\theta} = \frac{n}{\theta} - 1 + e^{-bx_n}(1 + bx_n) = 0 \quad (15)$$

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

2.4 NHPP Log-Logistic Distribution Model

In general, the Log-Logistic distribution is a measurable continuous distribution defined using scale and shape parameters and has been used to model binary responses in fields such as biostatistics and economics of growth models. But, compared to a general distribution model in which the failure rate per defect is monotonically increases and decreases, the log-logistic distribution model, which has characteristics similar to actual failure rates phenomenon, is known as an excellent model for reliability analysis and testing [12].

$$\mathrm{m}(\mathrm{t}|\theta,\tau,\mathrm{k}) = \theta\,\frac{(\tau t)^k}{[1+(\tau t)^k]} \quad (17)$$

$$\lambda(\mathrm{t}|\theta,\tau,\mathrm{k}) = \theta\,\frac{\tau k(\tau t)^{k-1}}{[1+(\tau t)^k]^2} \quad (18)$$

As shown in Equations (17) and (18), the Log-Logistic distribution belonging to the Gamma-family lifetime distribution to be studied in this paper considers the case where the shape parameter (k) is 2.

$$\ln L_{NHPP}(\Theta|\underline{x}) = nln2 + nln\theta + 2nln\tau + \sum_{i=1}^{n} x_i$$
$$-2\sum_{i=1}^{n} ln[1+(\tau x_i)^2] - \theta\frac{(\tau x_n)^2}{[1+(\tau x_n)^2]} = 0 \quad (19)$$

That is, if using Equation (19), the estimators $\hat{\theta}_{MLE}$ and $\hat{\tau}_{MLE}$ for the parameters must satisfy the following conditional expression.

$$\frac{\partial \ln L_{NHPP}(\Theta|\underline{x})}{\partial\theta} = \frac{n}{\hat{\theta}} - \frac{(\hat{\tau}x_n)^2}{[1+(\hat{\tau}x_n)^2]} = 0 \quad (20)$$

$$\frac{\partial \ln L_{NHPP}(\Theta|\underline{x})}{\partial\tau} = \frac{2n}{\hat{\tau}} - 2\hat{\tau}\sum_{i=1}^{n} {x_i}^2\,\frac{1}{ln[1+(\hat{\tau}x_i)^2]}$$
$$-\hat{\theta}\left(\frac{2\hat{\tau}{x_n}^2(1+\hat{\tau}^2 x_n - \hat{\tau}^2 {x_n}^2}{[1+(\hat{\tau}x_n)^2]^2}\right) = 0 \quad (21)$$

2.5 NHPP Rayleigh Distribution Model

The Rayleigh distribution was originally known as a distribution widely used as a distance distribution in spatial Poisson process and specific functional modeling of electromagnetic waves, but it is also known as a model suitable for reliability analysis. Also, the Rayleigh distribution, which has been found to be an appropriate model for lifetime testing and reliability theory, is a special case with a shape parameter of 2 in the Weibull distribution. Therefore, the property functions (m(t), λ(t)) of the NHPP Rayleigh reliability model considering the shape parameter (α) can be written as follows.

$$m(t|\theta,b) = \theta\left(1-e^{-bt^{\alpha}}\right) = \theta\left(1-e^{-bt^{2}}\right) \quad (22)$$

$$\lambda(t|\theta,b) = \theta\left(2bt^{a-1}e^{-bt^{2}}\right) \quad (23)$$

Note that $\theta > 0,\ b = \frac{1}{2\beta^2} > 0,\ t \in [0,\ \infty]$

That is, the log likelihood function can be written as follows.

$$\ln L_{NHPP}(\Theta|\underline{x}) = nln2 + nln\theta + nlnb + \sum_{i=1}^{n} lnx_i - b\sum_{i=1}^{n} {x_i}^2 - \theta\left(1-e^{-bx_n^2}\right) \quad (24)$$

That is, if using Equation (24), the estimators $\hat{\theta}_{MLE}$ and $\hat{b}_{MLE}$ for the parameters must satisfy the following conditional expression.

$$\frac{\partial lnL_{NHPP}(\Theta|\underline{x})}{\partial\theta} = \frac{n}{\hat{\theta}} - 1 + exp\left(-\hat{b}x_n^2\right) = 0 \quad (25)$$

$$\frac{\partial lnL_{NHPP}(\Theta|\underline{x})}{\partial b} = \frac{n}{\hat{b}} - \sum_{i=1}^{n} x_i^2 - \hat{\theta}x_n^2\, exp\left(-\hat{b}x_n^2\right) = 0 \quad (26)$$

2.6 Software Development Cost Model Applying the NHPP Reliability Model

When the $m(t)$ representing the performance attributes of the NHPP model proposed in this work is applied to the software development cost model, it is said that the total software development cost (E_t) is expressed as the sum of each cost element ($E_1 \sim E_4$) as in Equation (27) [13].

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

Note that E_t represents the total software development cost.

① E_1 is the development cost invested in the initial stage.

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.

③ E_3 is the cost of eliminating one defect.

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

Note that C_3 is the cost of eliminating one error found in the development test phase, and m(t) represents the reliability performance attributes of the NHPP model applied as an error occurrence expectation value.

④ E_4 is the cost of eliminating all remaining flaws.

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

Note that C_4 is the cost of repairing flaws detected by the user during normal operation of the system, and t' is the time that the system can be maintained with the released software after the developed software is released.

Also, software developers will want to release developed software at the point in time when the total software development cost is minimized. Therefore, as shown in Equation (31) below, the optimal release time should be equal to the point at which the total development cost (E_t) is minimized.

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

3. PERFORMANCE ATTRIBUTES ANALYSIS OF SOFTWARE DEVELOPMENT COST MODEL

In this study, software failure time was used to reflect the attributes that the probability of a specific event (software failure) occurring in relation to time is very small.

Accordingly, the software failure time was applied to the NHPP reliability model to analyze the properties of the software development cost model having Gamma family distribution characteristics, and then the performances of the applied model were compared and evaluated.

Table 1: Software Failure Time Data.

Failure number	Failure time (hours)	Failure time Interval	Failure time (hours)× 10^{-2}
1	30.02	30.02	0.30
2	31.46	1.44	0.31
3	53.93	22.47	0.53
4	55.29	1.36	0.55
5	58.72	3.43	0.58
6	71.92	13.20	0.71
7	77.07	5.15	0.77
8	80.90	3.83	0.80
9	101.90	21.00	1.01
10	114.87	12.97	1.14
11	115.34	0.47	1.15
12	121.57	6.23	1.21
13	124.97	3.40	1.24
14	134.07	9.10	1.34
15	136.25	2.18	1.36
16	151.78	15.53	1.51
17	177.50	25.72	1.77
18	180.29	2.79	1.80
19	182.21	1.92	1.82
20	186.34	4.13	1.86
21	256.81	70.47	2.56
22	273.88	17.07	2.73
23	277.87	3.99	2.77
24	453.93	176.06	4.53
25	535.00	81.07	5.35
26	537.27	2.27	5.37
27	552.90	15.63	5.52
28	673.68	120.78	6.73
29	704.49	30.81	7.04
30	738.68	34.19	7.38

The failure time data applied in this work refers to errors generated randomly due to insufficient testing and basic design errors in the software development stage. Table 1 shows the software failure time data applied in this work [14]. Applied data indicates that the number of failures occurred was 30 in a total of 738.68 hours.

In general, the failure time of software is constant regardless of the testing time, or has the property of monotonically increasing and decreasing.

Therefore, as a data scale method for analyzing this type of data, a trend test technique for data has been developed.

Thus, in the case of analysis using the Laplace trend test technique, if the estimated value is existed between "-2 and 2", the cited data is said to be reliable because it is stable. Figure 1 shows the simulation results of the Laplace trend test, and the results of the analysis indicate that the cited data are distributed between "0 and 2".

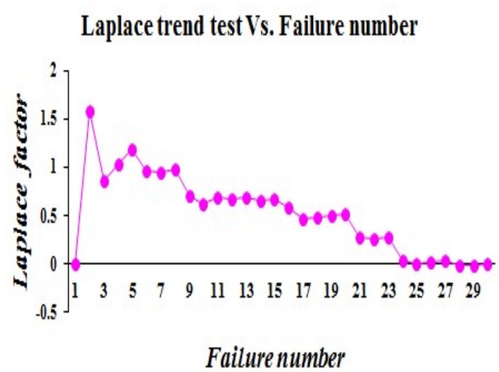

Figure 1: Simulation Results of Laplace Trend Test

Accordingly, it is concluded that the software failure time cited in this work is applicable to this study because it can be judged to be stable data without extreme values.

3.1. Parameter Calculation of the Proposed NHPP Reliability Model.

As presented in Table 1, numerically converted data was used to easily calculate the parameters of the applied model. Also, the solution of the parameter estimator ($\hat{\theta}_{MLE}$, $\hat{b}_{MLE}(\hat{\tau}_{MLE})$) was computed by applying the maximum likelihood estimation (MLE).

Journal of Theoretical and Applied Information Technology
31st May 2023. Vol.101. No 10

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 2: Parameter Estimator Solution Applying the MLE.

Type	NHPP Model	*Parameter Estimates of the Proposed Model*	
		$\hat{\theta}_{MLE}$	$\hat{b}_{MLE}(\hat{\tau}_{MLE})$
Basic	Goel-Okumoto	33.4092	0.3090
Gamma family distribution	Erlang	30.5978	0.7922
	Log-Logistic	32.2412	0.4953
	Rayleigh	24.0116	0.3707

Table 2 shows the results of calculating parameter estimators of the proposed model by applying MLE [15]. Therefore, Table 2 represents the estimated results of the parameters $(\hat{\theta}, \hat{b}(\hat{\tau}))$ using the cited failure time data in this work.

3.2. Mean Value Function (m(t)) of the Proposed NHPP Model.

Table 3 shows the method of calculating the m(t) of the proposed NHPP model and the method of calculating the cost of the software development model by applying the m(t) as an equation [16].

Also, Table 4 shows in detail the estimated value of the m(t) representing the attribute of reliability performance representing the predictive power for the true value in the NHPP model.

Table 3: Applying m(t) of the Proposed NHPP Model to the Cost Calculation of the Development Model.

Type	NHPP Model	$m(t)$ of the Proposed Model	$m(t)$ of Software Development Cost Model
Basic	Goel-Okumoto	$m(t) = \theta(1 - e^{-bt})$	$E_3 = C_3 \times m(t)$ $E_4 = C_4 \times [m(t+t') - m(t)]$
Exponential	Erlang	$m(t) = \theta(1 - e^{-bt}[1 + (bt)])$	
	Log-Logistic	$\mathrm{m(t)} = \theta \dfrac{(\tau t)^2}{[1 + (\tau t)^2]}$	
	Rayleigh	$m(t) = \theta\left(1 - e^{-bt^2}\right)$	

Table 4: Performance Trends for Predicting True Values Using m(t).

Failure Number	Failure Time (hours)	True Value	Basic Model	Gamma Family Distribution Model		
			Goel-Okumoto	Erlang	Log-Logistic	Rayleigh
1	0.3002	1	2.959701903	0.739580008	0.697384445	0.986145334
2	0.3146	2	3.094889003	0.806206794	0.764269396	1.081255931
3	0.5393	3	5.128273246	2.111234846	2.147222681	3.067700165
4	0.5529	4	5.246871639	2.203764021	2.249234632	3.215670863
5	0.5872	5	5.54377951	2.442763262	2.51452034	3.601354687
6	0.7192	6	6.657486934	3.428317002	3.630485578	5.236986519
7	0.7707	7	7.079830921	3.836564603	4.100544946	5.931823322
8	0.809	8	7.38959451	4.147249614	4.460441062	6.465944252
9	1.019	9	9.024392104	5.930372224	6.545531831	9.589489077
10	1.1487	10	9.982345064	7.073562178	7.884420283	11.6110598
11	1.1534	11	10.01634319	7.115300508	7.933163842	11.68472679
12	1.2157	12	10.46236517	7.66968272	8.579115146	12.6606697
13	1.2497	13	10.70218265	7.972774635	8.930912369	13.19162042
14	1.3407	14	11.33178785	8.783806147	9.866405017	14.59926874
15	1.3625	15	11.48000594	8.977735913	10.08865555	14.93236695
16	1.5178	16	12.50748713	10.34941267	11.64180325	17.23691115
17	1.775	17	14.10435094	12.55398172	14.05581205	20.67975079
18	1.8029	18	14.27006463	12.78629488	14.30358006	21.01886578
19	1.8221	19	14.38327712	12.94527775	14.47237454	21.24799111
20	1.8634	20	14.62453752	13.28472677	14.83067108	21.72894216
21	2.5681	21	18.30023296	18.45775987	19.9256821	27.41098336
22	2.7388	22	19.07652191	19.52069291	20.88933326	28.15361819
23	2.7787	23	19.25214607	19.7583244	21.10116668	28.29945124
24	4.5393	24	25.19234435	26.74003388	26.91640772	30.00004455
25	5.35	25	27.01314553	28.28455939	28.221968	30.01375987
26	5.3727	26	27.0578524	28.31798665	28.25166556	30.01382372
27	5.529	27	27.35731187	28.53604739	28.44787545	30.01414041
28	6.7368	28	29.24235101	29.66501101	29.58406352	30.01449852
29	7.0449	29	29.62074907	29.83888049	29.79414045	30.01449969
30	7.3868	30	30.00057116	29.99512675	30.00004176	30.01449995

Figure 2 shows the performance trend of the mean value function (refer to the values presented in Table 4), which means the attribute value that can estimate the true value with numerically converted data to facilitate calculation.
That is, if we analyze the properties of the proposed model to be used as the m(t) value that affects the performance of the development cost model, it was concluded that the ability to predict the true value of the Rayleigh model and the Goel-Okumoto basic model was the best.

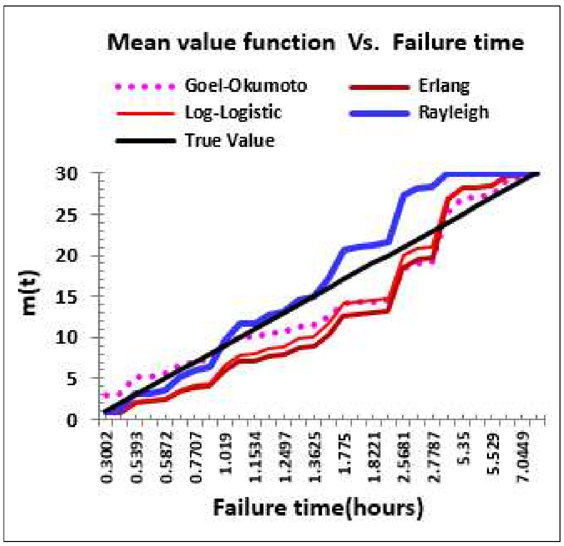

Figure 2: Performance Trend Curve Predicting True Value Using m(t).

3.3. Simulation of Software Development Cost Model using m(t) of the Proposed NHPP Model

In this work, in order to analyze under conditions similar to the actual development environment, the conditions of the software development cost model such as Equation (27) were assigned as [Assumptions 1 through 4] [17].

3.3.1. Assumption 1: basic conditions.

$$E_1 = 40\$,\ \ C_2 = 5\$,\ \ C_3 = 1.5\$,\ \ C_4 = 10\$$$

$$t' = 40(\text{hours}) \qquad (32)$$

Figure 3 shows the results of analyzing the development cost and release time by substituting the calculated value of the m(t) presented in Table 3 into the cost model equation such as Equation (27).

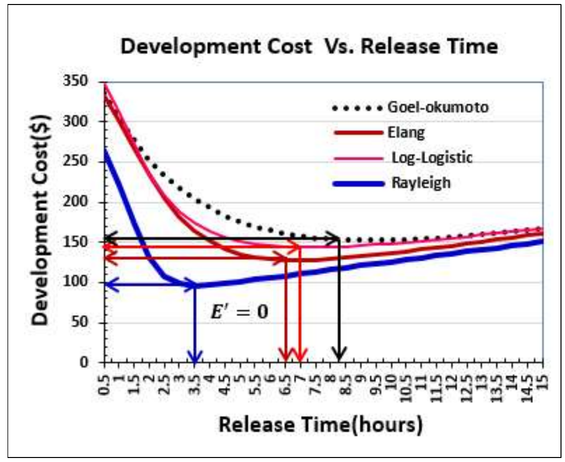

Figure 3: Trend Analysis of Development Cost Model Applying the Condition of [Assumption 1].

Analyzing Figure 3, the trend of the cost curve showed a pattern of rapidly decreasing at the beginning and gradually increasing with time.
The reason is that the number of flaws inherent in the software decreases during the process of eliminating flaws in the early stage, so the cost decreases, but the probability of finding remaining flaws gradually decreases in the later stage. Thus, development costs increase proportionately over time. Eventually, the pattern of development cost curves tends to increase over release time.

3.3.2. Assumption 2: under the condition of Assumption 1, the situation where only the C_2 cost is doubled.

$$E_1 = 40\$,\ \ C_2 = 10\$,\ \ C_3 = 1.5\$,\ \ C_4 = 10\$$$

$$t' = 40(\text{hours}) \qquad (33)$$

The condition of [Assumption 2] is a situation in which all conditions are the same compared to the condition of [Assumption 1], but only the testing cost per unit time (C_2) is doubled (5$→10$). Figure 4 shows the results of analyzing the cost and release time of the software development model under [Assumption 2] conditions after substituting the value of m(t) as described in the previous section.

Figure 4 shows the trend curve for analyzing the development cost attributes under the condition of Assumption 2. The result of comparing the simulations of Assumption 2 and Assumption 1 represents a situation in which only the cost attribute increased, but the time attribute did not change at all.

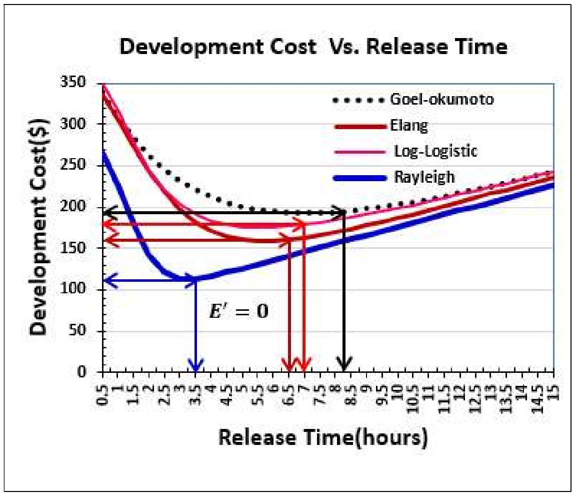

Figure 4: Trend Analysis of Development Cost Model Applying the Condition of [Assumption 2].

Therefore, in this case, it was confirmed that an accurate test is necessary in the development process in order not to increase the development cost. Therefore, under these conditions, it can be seen that the Rayleigh model with the best attributes (cost, release time) is the most efficient in this work.

3.3.3. Assumption 3: under the condition of Assumption 1, the situation where only the C_3 cost is doubled.

$$E_1 = 40\$,\ \ C_2 = 5\$,\ \ C_3 = 3\$,\ \ C_4 = 10\$$$

$$t' = 40(\text{hours}) \tag{34}$$

The condition of [Assumption 3] is a situation in which all conditions are the same compared to the condition of [Assumption 1], but only the cost of removing one error found in the development test stage (C_3) is doubled (1.5\$ → 3\$). Figure 5 shows the trend curve for analyzing the development cost attributes under the condition of Assumption 3.

In the same way, Figure 5 shows the results of analyzing the development cost and release time by substituting the calculated value of the m(t) presented in Table 3 into the cost model equation such as Equation (27). As shown in Figure 5, the analysis results of the simulation showed that the optimal release time was 3.5H when the cost of the Rayleigh model was \$130, and the optimal release time was 6.5H when the cost of the Erlang model was \$170, and the optimal release time was 7.5H when the cost of the Log-Logistic model was \$190.

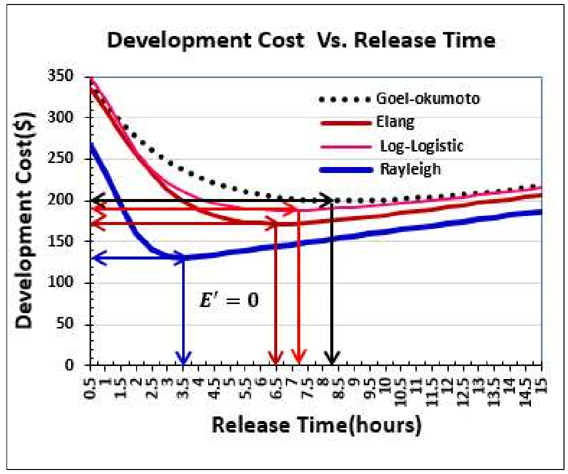

Figure 5: Trend Analysis of Development Cost Model Applying the Condition of [Assumption 3]

That is, in the situation of Assumption 3, the cost increased as in Assumption 2, but the optimal release time did not change at all. Therefore, in order to save development costs in this situation, it was found that as many defects as possible should be removed at one time from the testing process in advance.

3.3.4. Assumption 4: under the condition of Assumption 1, the situation where only the C_4 cost is doubled.

$$E_1 = 40\$,\ \ C_2 = 5\$,\ \ C_3 = 1.5\$,\ \ C_4 = 20\$$$

$$t' = 40(\text{hours}) \tag{35}$$

The conditions of [Assumption 4] are all the same compared to the conditions of [Assumption 1], but only the cost (C_4) of repairing failures found by users in the actual operation stage after the release of the software is doubled (\$10 → 20\$).

Figure 6 shows the results of analyzing the performance attributes of the software development cost model under [Assumption 4] conditions after substituting the value of m(t) as described in the previous section.

Evaluating the simulation results as shown in Figure 6, it was found that the Rayleigh model is the best among the proposed models.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

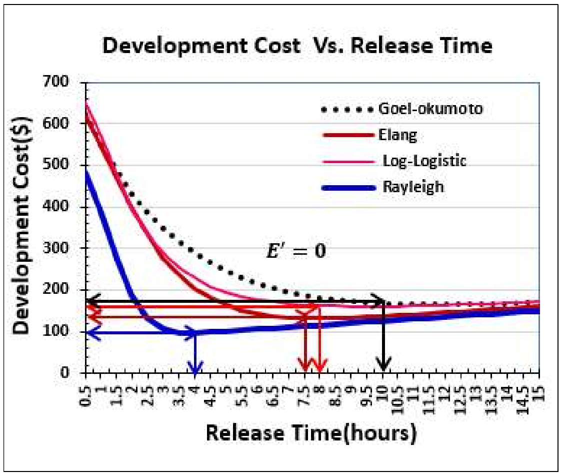

Figure 6: Trend Analysis of Development Cost Model Applying the Condition of [Assumption 4]

Therefore, unlike Assumptions 2 and 3, the situation of Assumption 4 showed a pattern trend in which the release time was delayed along with the increase in development cost. Thus, in this case, all possible defects should be eliminated in the testing process rather than in the operational process, so that defects can be reduced before the software is released.

3.4 Performance Evaluation on the Attributes of the Proposed NHPP Software Development Cost Model.

Table 5 shows the evaluation results of the reliability performance based on the performance attributes. As shown in Table 5, as a result of comprehensively evaluating the performance attributes (m(t), cost, release time) of the software development cost model proposed in this work, the Rayleigh model was found to be the best among the Gamma family distribution [18].

Table 5: Performance Attributes Evaluation.

NHPP model	Performance Attributes		
	m(t)	Cost	Release Time
Goel-Okumoto	Good	Worst	Worst
Erlang	Bad	Good	Good
Log-Logistic	Bad	Good	Good
Rayleigh	Best	Best	Best

4. CONCLUSION

If a software developer has reliable downtime data collected during software system operation, it will be possible to efficiently predict failures that can actually occur by applying it to the development process or testing stage. Accordingly, the developer will be able to proceed with the task of developing reliable software at a more reasonable cost. Therefore, the performance properties of the NHPP software development cost model were explored and newly analyzed using the Gamma distribution.

The results of this study are as follows.
First, as a result of analyzing the m(t) that affects the cost properties, it was found that the Rayleigh model and the Goel-Okumoto basic model are efficient because the error in estimating and predicting the true value is small.

Second, as a result of analyzing the attributes of development cost after doubling the cost factors (C2, C2, C4) under the conditions of Assumptions 2 to 4 developed in this study, the Rayleigh model showed the best performance in all conditions.

Third, as a result of comprehensively evaluating the performance attributes (m(t), cost, release time) of the software development cost model proposed in this paper, it was confirmed that the Rayleigh model was the best [18].

In conclusion, if software developers use this study information efficiently, it is expected that it can be used as a basic design data for exploring the attributes of development cost along with reliability problem. Also, after collecting reliable downtime data for each software industry to be applied and applying them to various development cost models, research to find the optimal cost model suitable for the applied industry will be continuously needed.

ACKNOWLEDGEMENTS
Funding for this paper was provided by Namseoul University.

REFERENCES

[1] T. H. Yoo and H. C. Kim, "The Infinite NHPP Software Reliability Model based on Monotonic Intensity Function", *Indian Journal of Science and Technology,* Vol. 8, No. 14, 2015, pp. 1-7.

[2] A. L. Goel and K. Okumoto, “Time-dependent fault detection rate model for software and other performance measures”, *IEEE Trans. Reliability,* Vol. 28, 1978, pp. 206-211.
[3] C. Y. Huang “Performance analysis of software reliability growth models with testing-effort and change-point”, *Journal of Systems and Software,* Vol. 76, No. 2, 2005, pp.181-194.
[4] X. XIAO and T. DOHI, “On the Role of Weibull-type Distributions in NHPP-based Software Reliability Modeling”, *International Journal of Performability Engineering.* Vol. 9, No. 2, 2013, pp. 123-132.
[5] H. C. Kim, “A Capability Comparative of Failure Predicting Time Using Software NHPP Finite Failure Reliability Model”, *International Journal of Soft Computing.* Vol.12, 2017, pp. 308-312.
[6] H. Pham, “Distribution Function and its Application in Software Reliability”, *International Journal of Performability Engineering,* Vol. 15, No.5, 2019, pp. 1306-1313.
[7] K. Tokuno and M. Fukuda, Yamada S. Yamada, “Stochastic Performance Evaluation for Software System Considering NHPP Task Arrival”, *International Journal of Performability Engineering,* Vol. 4, No. 1, 2008, pp. 57-70.
[8] T. J. Yang, “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.* Vol. 13, No. 9, 2020, pp. 2185-2190.
[9] T. J. Yang, “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.* Vol. 11, Issue. 10, 2021, pp. 166-176.
[10] T. J. Yang, “Comparative Analysis on the Reliability Performance of NHPP Software Reliability Model Applying Exponential-Type Lifetime Distribution”, *International Journal of Performability Engineering.* Vol. 18, No. 10, 2022, pp. 679-689.
[11] T. J. Yang, “Comparative Study on the Performance Evaluation of Infinite Failure NHPP Software Reliability Model with Log-Type Distribution Property”, *ARPN Journal of Engineering and Applied Sciences.* Vol. 17, No. 11, 2022, pp. 1209-1218.
[12] Y. Zhang and K. Wu, “Software Cost Model Considering Reliability and Time of Software in Use”, *Journal of Convergence Information Technology,* Vol. 7, No. 13, 2012, pp. 135-142.
[13] T.J. Yang, “A comparative study on the attributes of finite failure NHPP software reliability with logistic distribution property”, *International Journal of Engineering Research and Technology,* Vol. 13, No. 3, 2020, pp. 438-443.
[14] R.S. 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,* Vol. 21, No. 7, 2011, pp. 1-5.
[15] K.C. Chiu, Y.S. Huang and T.Z. Lee, “A Study of Software Reliability Growth from the Perspective of Learning Effects”, *Reliability Engineering & System Safety,* Vol. 93, No. 10, 2008, pp. 1410-1421.
[16] T. J. Yang and J. G. Park, “A Comparative Study of the Software NHPP Based on Weibull Extension Distribution and Flexible Weibull Extension Distribution”, *International Journal of Soft Computing.* Vol. 11, No. 4, 2016. pp. 276-281.
[17] A. Alhaddad, B. Al-ibrahi, A. bualkishik, “A Systematic Mapping Study on Software Effort Estimation”, *Journal of Theoretical and Applied Information Technology,* Vol. 98, No. 17, 2020, pp. 3619-3643.
[18] T. J. Yang, “Performance Analysis on the Reliability Attributes of NHPP Software Reliability Model Applying Exponential and Inverse-Exponential Lifetime Distribution”, *Journal of Theoretical and Applied Information Technology,* Vol. 100, No. 22, 2022, pp. 6645-6656.