

변형된 강도함수를 적용한 소프트웨어 신뢰모형의 신뢰성능 비교 평가에 관한 연구

김희철* · 문송철**

A Study on the Reliability Performance Evaluation of Software Reliability Model Using Modified Intensity Function

Hee Cheul Kim* · Song Chul Moon**

Abstract

In this study, we compared the reliability performance of the software reliability model, which applied the Goel-Okumoto model developed using the exponential distribution, to the logarithmic function modifying the intensity function and the Rayleigh form. As a result, the log-type model is relatively smaller in the mean squared error compared to the Rayleigh model and the Goel-Okumoto model. The logarithmic model is more efficient because of the determination coefficient is relatively higher than the Goel-Okumoto model. The estimated determination coefficient of the proposed model was estimated to be more than 80% which is a useful model in the field of software reliability. Reliability has been shown to be relatively higher in the log-type model than the Rayleigh model and the Goel-Okumoto model as the mission time has elapsed. Through this study, software designer and users can identify the software failure characteristics using mean square error, decision coefficient. The confidence interval can be used as a basic guideline when applying the intensity function that reflects the characteristics of the lifetime distribution.

Keywords : Determination Coefficient, Mean Squared Error, Confidence Interval, Intensity Function, NHPP, Mission Time

Received : 2018. 05. 28. Revised : 2018. 06. 18. Final Acceptance : 2018. 06. 22.

* Professor, Namseoul University, Department of Industrial & Management Engineering, e-mail : kim1458@nsu.ac.kr

** Corresponding Author, Professor, Department of Computer Science, Namseoul University, 91 Daehak-ro, Seonghwan-eup, Seobuk-gu, Cheonan-si, Chungcheongnam-do, 31020, Korea, Tel : +82-41-580-2200, e-mail : moon@nsu.ac.kr

1. 서 론

소프트웨어 시스템은 우리 산업의 매우 중요한 지식 산업의 산물로 인식되고 있다. 이러한 시스템은 정확성(Correctness)과 신뢰성(Reliability)을 담보 할 수 있다면 소프트웨어 운용자에게 더 나은 품질의 서비스를 제공 할 수 있는 환경이 되므로 중요한 문제 중에 하나이다. 그러나 소프트웨어 유지 및 보수는 어렵고 복잡한 시행으로 판단된다. 따라서 소프트웨어 개발자 및 운용자들의 주된 시행과정은 소프트웨어 시스템의 정확성과 신뢰성을 확보하는데 있다. 이로 인해 소프트웨어 유지 및 보수 관련 엔지니어링 연구가 시작되었으며 지난 수십 년 동안 소프트웨어 신뢰 성장모형(Software reliability growth model)이 제안되었다. 즉, 남아 있는 고장의 수, 고장비율 등의 신뢰성 특성 요인을 예측하기 위해 관찰된 테스트 환경 조건하에서 고장강도 함수 및 평균값 함수를 이용하여 비동질 포아송 과정(Non-homogeneous Poisson process; NHPP)에 의존한 소프트웨어 신뢰성 모형이 개발되었다[Song and Chang, 2017].

소프트웨어 신뢰성 모형은 소프트웨어의 신뢰성, 남은 고장 수, 고장강도, 소프트웨어 개발 총비용 등을 예측 및 검정하는 데 사용된다. 현재까지 다양한 소프트웨어 안정성 모형연구가 시행되었다[Yamada and Osaki, 1985]. 이러한 환경에서 최우추정법을 이용하여 평균값 함수의 결과를 예측 할 수 있음을 강조 하였고 평균값 함수의 신뢰구간을 나타내는 그래프를 제시하여 신뢰성 특성을 설명하였다[Huang, 2005]. 또한, 디버깅을 이용한 물류 지수 테스트 커버리지 함수를 통합한 NHPP에 의존한 소프트웨어 신뢰성 모형을 제안했다[Chatterjee and Singh, 2014]. 이런 소프트웨어 환경에서 불확실성과 소프트웨어 고장률에 미치는 영향을 포착하는 일

반화 된 모형에 대해 논의했다[Teng and Pham, 2006].

이 분야에서 기본모형인 지수분포를 이용하여 개발된 결함 탐색비율은 상수(위험함수)인 강도함수로 구성되었다[Goel and Okumoto, 1978]. 본 연구에서는 강도함수를 변형한 로그형 및 Rayleigh 형태를 적용하여 향후 소프트웨어 개발단계에서 신뢰도를 높이기 위해 개발자들이 활용가능한 수준의 소프트웨어 신뢰성모형의 신뢰성능을 비교 연구할 필요가 있다.

2. 다양한 결함발생률을 고려한 NHPP 소프트웨어 신뢰성 모형

2.1 기존의 Goel-Okumoto-NHPP 모형

소프트웨어 신뢰성 분야에서 기본모형인 유한 고장 상황에서 고장의 원인이 되는 결함의 기대값을 θ 라고 표현하고 결함 탐색비율을 β 라고 하면 NHPP 모형에서 결함 탐색비율 β 는 고정상수로 간주하여 정의하였다. 따라서 이 상황에서 $f(t)$ 을 고장밀도함수이고 $F(t)$ 을 누적분포함수라고 하면 유한고장 NHPP 모형의 강도함수와 평균값함수를 적용시키면 다음과 같다[Goel and Okumoto, 1978].

$$\lambda(t|\theta, \beta) = \theta f(t) = \theta \beta e^{-\beta t} \quad (1)$$

$$m(t|\theta, \beta) = \theta F(t) = \theta(1 - e^{-\beta t}) \quad (2)$$

2.2 로그 및 Rayleigh형 강도함수를 적용한 유한고장 NHPP 모형

강도함수를 변형한 로그형 및 Rayleigh 형태를 적용한 강도함수와 평균값 함수는 <Table 1>에 요약되었다.

〈Table 1〉 Intensity Function and Mean Value Function

Model	Intensity function $\lambda(t \theta, \beta)$	Mean value function $m(t \theta, \beta)$
Goel-Okumoto	$\theta \beta e^{-\beta t}$	$\theta(1 - e^{-\beta t})$
Log-type	$\theta \ln(\beta) e^{-\ln(\beta)t}$	$\theta(1 - e^{-\ln(\beta)t})$
Rayleigh type	$2\theta \beta t \exp(-\beta t^2)$	$\theta[1 - \exp(-\beta t^2)]$

Note) θ : Expected value of faults that would be discovered observation time $(0, t]$

β : Fault search probability

한편, 관측시간 $(0, t]$ 까지 관찰하기 위한 정시 중단(Time truncated) 모형은 n 번째까지 고장시점 자료(t_i 은 고장간격시간)는 다음과 같은 관계로 표현 할 수 있다[Yamada and Osaki, 1985].

$$x_n = \sum_{i=1}^n t_i \quad (i = 1, 2, \dots, n; 0 \leq x_1 \leq x_2 \leq \dots \leq x_n) \quad (3)$$

한편, 식 (3)을 이용한 NHPP 모형에 대한 우도함수는 다음과 같다[Goel and Okumoto, 1978].

$$L_{NHPP}(\Theta | \underline{x}) = \left(\prod_{i=1}^n \lambda(x_i) \right) \exp[-m(x_n)] \quad (4)$$

단, $\underline{x} = (0 \leq x_1 \leq x_2 \leq \dots \leq x_n)$

Goel-Okumoto인 경우 식 (4)를 이용한 최우추정량 $\hat{\theta}_{MLE}$ 와 $\hat{\beta}_{MLE}$ 에 대한 모수 추정은 다음조건을 만족해야 한다[Yamada and Osaki, 1985; Goel and Okumoto, 1978].

$$\frac{n}{\hat{\theta}} = 1 - e^{-\hat{\beta} x_n} \quad (5)$$

$$\frac{n}{\hat{\beta}} = \sum_{i=1}^n x_i + \hat{\theta} x_n e^{-\hat{\beta} x_n} \quad (6)$$

Goel-Okumoto 모형과 유사하게 로그형 모형에서도 식 (4)을 이용하여 최우추정 값 $\hat{\theta}_{MLE}$ 와 $\hat{\beta}_{MLE}$ 을 다음과 같은 방법으로 추정 할 수 있다 [Kim and Shin, 2016].

$$\frac{\partial \ln L_{NHPP}(\Theta | \underline{x})}{\partial \theta} = \frac{n}{\theta} - 1 + e^{-\ln(\beta) x_n} = 0 \quad (7)$$

$$\frac{\partial \ln L_{NHPP}(\Theta | \underline{x})}{\partial \beta} = \quad (8)$$

$$\frac{n}{\beta \ln(\beta)} - \frac{1}{\beta} \sum_{i=1}^n x_i + \theta \frac{x_n}{\beta} e^{-\ln(\beta) x_n} = 0$$

Rayleigh형 모형에서도 유사하게 최우추정 값 $\hat{\theta}_{MLE}$ 와 $\hat{\beta}_{MLE}$ 을 다음과 같은 방법으로 추정 할 수 있다[Kim, 2016].

$$\frac{\partial \ln L_{NHPP}(\Theta | \underline{x})}{\partial \theta} = \frac{n}{\theta} - 1 + e^{-\beta x_n^2} = 0 \quad (9)$$

$$\frac{\partial \ln L_{NHPP}(\Theta | \underline{x})}{\partial \beta} = \quad (10)$$

$$\frac{n}{\beta} - \sum_{i=1}^n x_i^2 - \theta x_n^2 e^{-\beta x_n^2} = 0$$

2.3 평균제곱오차와 결정계수 및 신뢰구간을 이용한 효율적 모형 판단 기준

평균제곱오차(Mean square error; MSE)는 참 값(실제 관찰 값)과 추정 값(예측 값)에 대한 차이를 비교하는 기준으로서 다음과 같다[Kim, 2016; Yoo, 2018].

$$MSE = \frac{\sum_{i=1}^n (m(x_i) - \hat{m}(x_i))^2}{n - k} \quad (11)$$

단, $m(x_i)$ 은 관찰시점 $(0, x_i]$ 까지 고장들의 누적 개수를 나타내고 $\hat{m}(x_i)$ 는 x_i 시점까지 평균값 합수로부터 추정된 고장의 누적개수를 나타내고 n 은 관찰된 고장의 수이고 k 는 추정된 모수의 수를 나타낸다. 따라서 평균제곱오차 값이 작으면 상대적으로 효율적인 모형으로 판단된다. 그리고 관찰 값의 차이에 대한 설명력을 의미하는 기준인 결정계수(Determination coefficient; R^2)는 다음과 같다[Kuei et al., 2008].

$$R^2 = 1 - \frac{\sum_{i=1}^n (m(x_i) - \hat{m}(x_i))^2}{\sum_{i=1}^n \left(m(x_i) - \frac{\sum_{j=1}^n m(x_j)}{n} \right)^2} \quad (12)$$

즉 비교에 있어서 결정계수 값이 크면 오차가 작아 상대적으로 유용성 있는 모형으로 간주된다.

소프트웨어의 신뢰모형에 대한 신뢰 구간은 신뢰성 특징을 이용하여 추정 될 수 있다. 특히 평균값 함수에 대한 신뢰구간은 다음과 같다[Song and Chang, 2017].

$$\hat{m}(t) \pm Z_{\alpha/2} \sqrt{\hat{m}(t)} \quad (13)$$

단, $Z_{\alpha/2}$ 는 표준화 정규분포의 $100(1-\alpha)\%$ 백분위수를 의미한다.

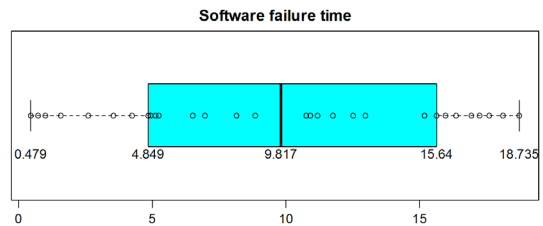
3. 변형된 강도함수를 적용한 소프트웨어 신뢰모형에 관한 고장시간신뢰도 분석

이 장에서 소프트웨어 고장시간 정보자료(Failure time information data)를 적용하여 신뢰모형들의 특징을 비교 분석하고자 한다. 이 자료[Hayakawa and Telfar, 2000]는 <Table 2>에 나열 되었고 극단값이 발생하는지를 알아보기 위하여 자료에 대한 추세 검정이 선행 되어야 하는데 본 연구에서는 추세검정을 박스-플롯(Box-plot)을 이용하였다[Kim, 2017]. <Figure 1>의 결과에서 보여주듯이 상한(= $15.04 + 1.5 \times (15.04 - 4.849) = 30.3265$)과 하한(= $4.849 + 1.5 \times (15.04 - 4.849) = 40.6408$)사이를 벗어난 자료가 없기 때문에 이상 값 혹은 극단 값이 존재하지 않음을 보여주고 있다. 따라서 이 자료를 사용하여 신뢰성 모형의 특성을 추정하는 작업이 효율적이다[Kim, 2016; Yoo, 2018].

모수추정은 최우추정법을 이용하고 비선형 방정식의 계산방법은 수치 해석적 방법인 이분법(Bisection method)을 활용하였다. 이러한 계산

<Table 2> Failure Time Data

Failure number	Failure time (hours)	Failure number	Failure time (hours)
1	0.479	16	10.771
2	0.745	17	10.906
3	1.022	18	11.183
4	1.576	19	11.779
5	2.610	20	12.536
6	3.559	21	12.973
7	4.252	22	15.203
8	4.849	23	15.640
9	4.966	24	15.980
10	5.136	25	16.385
11	5.253	26	16.960
12	6.527	27	17.237
13	6.996	28	17.600
14	8.170	29	18.122
15	8.863	30	18.735



<Figure 1> Result of Box-plot

은 초기 값을 0.01과 2.00을, 허용한계(Tolerance for width of interval)는 10^{-5} 을 투여하고 수렴성을 확인하면서 R 소프트웨어²⁾를 이용하여 모수추정을 수행하였다. 그 결과는 <Table 3>에 나타내었다. 또한, <Table 3>에서 실제 값과 예측 값에 대한 차이에 대한 척도를 의미하는 평균 제곱오차(MSE)는 로그형 모형인 경우가 Rayleigh 모형과 Goel-Okumoto 모형보다 상대적으로 가장 작고 예측 값의 차이에 대한 설명력에 대한 척도를 나타내는 결정계수도 로그형 모형인 경우가 Goel-Okumoto 모형과 Rayleigh 모형보다 상대

2) <https://www.r-project.org/>.

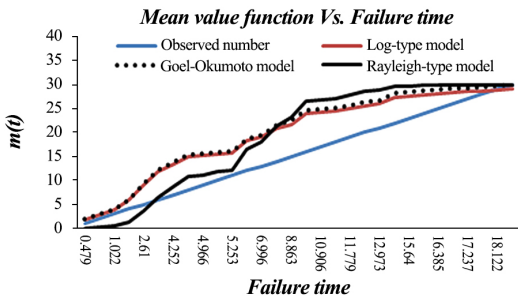
적으로 높은 성향으로 나타나기 때문에 로그형 모형이 다른 모형보다 효율적이라 판단된다. 로그형 모형, Goel-Okumoto 모형 그리고 Rayleigh 모형의 결정계수 값이 80% 이상 추정되어 비교적 소프트웨어 신뢰성 분야에서 유용성이 있는 모형으로 판단 할 수 있다[Kim, 2016; 2017].

<Table 3> Parameter Estimation and MSE, R^2

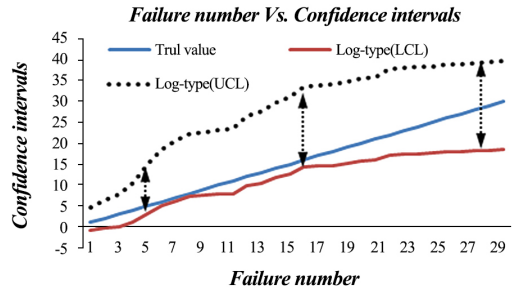
Model	MLE	Model Comparison	
		MSE	R^2
Goel-Okumoto	$\hat{\theta} = 32.9261$ $\hat{\beta} = 0.1297$	32.3491	0.916
Log-type	$\hat{\theta} = 31.9269$ $\hat{\beta} = 1.1379$	26.2298	0.917
Rayleigh type	$\hat{\theta} = 30.0412$ $\hat{\beta} = 0.01884$	32.2867	0.897

Note) MLE : Maximum Likelihood Estimation.
 MSE : Mean square error.
 R^2 : Determination coefficient.

<Figure 2>에서는 각 모형에 대한 평균값 함수 패턴에 대한 추세를 나타내었다. 적용고장시간에서 대체적으로 비감소 특성을 가진다. 특히 로그형 모형이, Goel-Okumoto 모형과 Rayleigh 모형보다 상대적으로 참값(Observed number)과의 차이의 폭이 작게 예측되는 패턴을 보이고 있다. <Figure 3>에서는 식 (13)을 이용한 로그형 모형에 대한 평균값에 대한 신뢰구간(유의수준(α)이 0.05)이 요약되었다. 이 그림에서 고장수가 증가할수록 상한과 하한의 폭이 증가함을 알 수 있다.



<Figure 2> Mean Value Function Trend for Each Model

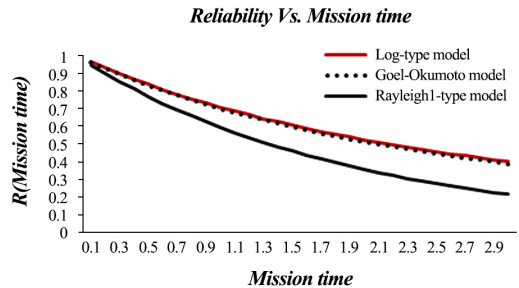


<Figure 3> Confident Interval of Mean Value Function

NHPP 모형에서 테스트 시점 $x_{30} = 18.735$ 에서 소프트웨어 고장이 발생하고 $(18.735, 18.735+t]$ (단, t 는 임무시간(Mission time))사이에서 소프트웨어의 고장이 발생하지 않을 확률인 신뢰도(Reliability) $\hat{R}(t | 18.735)$ 는 다음과 같은 구조로 표현된다[Kim, 2016; Yoo, 2018].

$$\hat{R}(t | 18.735) = e^{-\int_{18.735}^{18.735+t} \lambda(\tau) d\tau} \quad (14)$$

$$= \exp[-\{m(t+18.735) - m(18.735)\}]$$



<Figure 4> Transition of Reliability

<Figure 4>의 임무시간에 대한 신뢰도 그림에서는 임무시간이 지남에 따라 로그형 모형인 경우가 Rayleigh 모형과 Goel-Okumoto 모형보다 상대적으로 높은 신뢰도의 양상으로 나타나고 있다.

4. 결 론

소프트웨어 개발과정에서 테스트 작업이나 실제 소프트웨어 사용과정에서 고장발생 특성 혹은

고장 발생상황을 정량적으로 모형화하면 소프트웨어 유용성을 비교 분석하여 상대적으로 효율성 평가를 할 수 있다. 본 논문에서는 소프트웨어 신뢰성 분야에서 기본모형인 지수분포를 이용하여 개발된 Goel-Okumoto 모형과 강도함수를 변형한 로그형 및 Rayleigh 형태를 적용하여 소프트웨어 신뢰성모형의 신뢰성능을 비교 제시하였다. 그 결과 본 연구에서 평균제곱오차는 로그형 모형인 경우가 Rayleigh 모형과 Goel-Okumoto 모형보다 상대적으로 가장 작고 결정계수도 로그형 모형인 경우가 Goel-Okumoto 모형과 지수형 모형보다 상대적으로 높은 성향으로 나타나기 때문에 로그형 모형이 다른 모형보다 효율적이라 판단된다. 모든 모형의 결정계수 값이 80% 이상 추정되어 비교적 소프트웨어 신뢰성 분야에서 유용성이 있는 모형으로 판단 할 수 있다. 각 모형에 대한 평균값 함수 패턴은 비감소 특성을 가지고 로그형 모형이 Goel-Okumoto 모형 그리고 Rayleigh 모형보다 상대적으로 참값과의 차이의 폭이 작게 예측되는 패턴을 보이고 있다. 로그형 모형에 대한 평균값에 대한 신뢰구간은 고장수가 증가 할수록 상한과 하한의 폭이 증가함을 알 수 있었다. 신뢰도는 임무시간이 경과함에 따라 로그형 모형인 경우가 Rayleigh 모형과 Goel-Okumoto 모형보다 상대적으로 높은 신뢰도의 양상으로 나타나고 있다.

이 연구를 통하여 소프트웨어 운용자들은 강도함수의 특징을 응용할 경우에 소프트웨어 신뢰속성에 대한 고장유형을 파악하기 위하여 평균제곱오차와 결정계수, 신뢰구간을 사용하면 소프트웨어 고장 속성을 파악하는데 기본척도로 사용할 수 있으리라 판단된다.

References

- [1] Kuei-Chen, C., Yeu-Shiang, H. B., and Tzai-Zang, L., "A study of software reliability growth from the perspective of learning effects", *Reliability Engineering and System Safety*, Vol. 93, No. 10, 2008, pp. 1410-1421.
- [2] Huang, C. Y., "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.
- [3] Goel, A. L. and Okumoto, K., "Time-dependent fault detection rate model for software and other performance measures", *IEEE Trans. Reliab.*, Vol. 28, 1978, pp. 206-211.
- [4] Kim, H.-C., "A Performance Analysis of Software Reliability Model using Lomax and Gompertz Distribution Property", *Indian Journal of Science and Technology*, Vol. 9, No. 20, 2016, pp. 1-6.
- [5] Kim, H.-C. and Shin, H. C., "The Study for Performance Analysis of Software Reliability Model using Fault Detection Rate based on Logarithmic and Exponential Type", *The Journal of Korea Institute of Information, Electronics, and Communication Technology*, Vol. 9, No. 3, 2016, pp. 306-311.
- [6] Kim, H.-C., "A Comparative Study for Statistical Process Control of Software Reliability Model Base on Finite and Infinite NHPP Using Rayleigh Distribution", *International Journal of Soft Computing*, Vol. 11, No. 3, 2016, pp. 165-171.
- [7] Kim, H.-C., "A Comparative Study on Software Reliability Models with Shape Parameter of Type-2 Gumble Life Distribution", *International Journal of Soft Computing*, Vol. 12, No. 5-6, 2017, pp. 351-354.

[1] Kuei-Chen, C., Yeu-Shiang, H. B., and Tzai-

- [8] <https://www.r-project.org/>.
- [9] Song, K. Y., Chang, I. H., and Pham, H., "A Software Reliability Model with a Weibull Fault Detection Rate Function Subject to Operating Environments", *Applied Science*, Vol.7, No. 983, 2007, pp. 1-16.
- [10] Chatterjee, S. and Singh, J. B., "A NHPP based software reliability model and optimal release policy with logistic-exponential test coverage under imperfect debugging", *International Journal of System Assurance Engineering and Management*, Vol. 5, No. 3, 2014, pp. 399-406.
- [11] Yamada, S. and Osaki, S., "Software reliability growth modeling : models and applications", *IEEE Transactions on Software Engineering*, Vol. 11, No. 12, 1985, pp. 1431-1437.
- [12] Teng, X. and Pham, H., "A new methodology for predicting software reliability in the random field environments", *IEEE Transactions on Reliability*, Vol.55, No.3, 2006, pp. 458-468.
- [13] Hayakawa, Y. and Telfar, G., "Mixed poisson-type processes with application in software are reliability", *Mathematical and Computer Modelling*, Vol. 31, 2000, pp. 151-156.
- [14] Yoo, T. H., "A Performance Valuation for NHPP Software Reliability Model depend on Weibull-Type Distribution", *International Journal of Pure and Applied Mathematics*, Vol. 118, No. 19, 2018, pp. 1021-1033.

■ 저자소개



Hee Cheul Kim

He received a MS and a Ph.D from the Department of Statistics at Dongguk University in 1992 and 1998, He received a MS and a Ph.D from the

Department of Statistics at Dongguk University in 1992 and 1998, respectively. He has been a professor in the Department of Industrial & Management Engineering at Namseoul University since 2005. His current research interests include software reliability engineering, computer statistics, and information systems.



Song Chul Moon

He received a MS and a Ph.D from the KAIST. Dept. Info. Engineering and Kookmin Univ. Dept. Info. Mgt. He is past-Chairman for JITAM(Journal

if Information Technology Application and Management) in Korea Information Technology Application and Management Society. Main interesting field : S/W Engineering, Ubiquitous computing, Datamining, Mgt. Info. Sys..