

소프트웨어 NHPP 신뢰성모형에 대한 고장시간 예측능력 비교분석 연구

김희철*, 김경수**

남서울대학교 산업경영공학과*, 백석문화대학교 인터넷정보학부**

Failure Time Prediction Capability Comparative Analysis of Software NHPP Reliability Model

Hee-Cheul Kim*, Kyung-Soo Kim**

Dept. of Industrial & Management Engineering, Namseoul University*

Dept. of Internet information, BaekSeok Culture University**

요약 본 연구는 소프트웨어 NHPP 신뢰성 모형 (Goel-Okumo 모형, 지연된 S-형태 신뢰성모형 및 레일리분포 모형)의 예측능력을 분석하는 것을 목적으로 한다. 예측 능력분석은 두 가지 요인으로 분석이 될 것이다. 하나는 사용 가능한 고장자료에 대한 적용성의 정도이고 다른 하나는 예측능력 정도이다. 각 모형의 모수 추정은 고장시간자료의 첫 번째 고장시점부터 80%가 되는 고장시간 자료를 사용하고 기법은 최우추정법을 이용 하였다. 모형의 예측 능력의 비교에 있어서는 가능한 고장 데이터의 마지막 20%가 되는 선택된 자료를 이용하였다. 이 연구를 통하여 소프트웨어 관리자들에 소프트웨어 고장분석을 하는데 사전정보로 활용 할 수 있다.

주제어 : 비동질적인 포아송과정, 지연된 S-형태 신뢰성 모형, 레일리분포, 고장시간 예측, 최우추정법

Abstract This study aims to analyze the predict capability of some of the popular software NHPP reliability models(Goel-Okumo model, delayed S-shaped reliability model and Rayleigh distribution model). The predict capability analysis will be on two key factors, one pertaining to the degree of fitment on available failure data and the other for its prediction capability. Estimation of parameters for each model was used maximum likelihood estimation using first 80% of the failure data. Comparison of predict capability of models selected by validating against the last 20% of the available failure data. Through this study, findings can be used as priori information for the administrator to analyze the failure of software.

Key Words : NHPP, Rayleigh Distribution, Delayed S-shaped Reliability Model, Prediction of Failure Time, Maximum Likelihood Estimation

1. 서론

이 분야에서 특정한 환경조건에서 특정기간동안 고장

이 나지 않고 운영 할 수 있는 확률이 소프트웨어 신뢰성 이라고 정의 할 수 있다.

따라서 소프트웨어 신뢰성은 소프트웨어 개발과정에

Received 30 August 2015, Revised 9 November 2015
Accepted 20 December 2015
Corresponding Author: Hee-Cheul Kim(Namseoul University)
Email: kim1458@nsu.ac.kr

© The Society of Digital Policy & Management. All rights reserved. This is an open-access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0>), which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

ISSN: 1738-1916

서 핵심과제가 된다. 이러한 사항은 사용자의 요구조건을 만족시키고 테스트 비용을 줄여야 한다. 소프트웨어의 신뢰성의 변동과 테스트 비용을 사전에 알고 있으면 소프트웨어 테스트(디버깅)면에서 비용을 줄일 수 있다. 따라서 소프트웨어 개발과정 고려사항(신뢰도, 비용 및 방출시간)을 만족하는 소프트웨어 개발 과정이 요구되어 진다.

이 분야에서 비동질적 포아송 과정(Non-homogeneous Poisson process; NHPP)에 의존한 모형[1]은 지금까지 많은 소프트웨어 신뢰성모형이 제안 되었다. 이러한 모형은 에리탐색 과정측면에서는 효율적인 모형이고 결함이 발생하면 즉시 제거되며 또한, 디버깅과정에서 새로운 결함이 발생되지 않는다는 가정을 하고 있다.

이 분야에서 고양된 비동질적인 포아송과정모형(Enhanced NHPP model)은 Gokhale과Trivedi [2]에 의해 제시되었고 또한, 결함의 누적수가 S-형태나 지수적 형태(S-shaped or exponential-shaped)를 가진 평균값함수(Mean value function)를 이용한 지수적 소프트웨어 신뢰성모형(Exponential software reliability growth model)은 Goel 과 Okumoto [3]에 의해 제안되었다.

그리고 이 모형을 더욱더 구체화시킨 지연된 S-형태 신뢰성장모형(Delayed S-shaped reliability growth model)과 변곡된 S-형태 신뢰성장모형(Inflection S-shaped reliability growth model)은 Yamada 와 Ohba [4]에 의해 연구되었다.

이 분야에서 소프트웨어 신뢰도측면에서 변환점 문제는 Zhao [5]에 의해 설명되었고 이러한 변환점을 구체화한 일반화한 신뢰도 성장모형은 Shyur [6]에 의해서 제시되었다.

소프트웨어 안정도를 평가 할 수 있는 테스트 커버리지(Coverage)를 이용한 소프트웨어 안정도모형은 Pham 와 Zhang[7]에 의해 연구되었다. 이 모형과 관련하여 일반화 로지스틱 테스트 노력 함수(Generalized logistic testing-effort function)와 변환점 모수(Change-point parameter)를 통합한 효율적인 소프트웨어 신뢰성 예측 기술은 Huang [8]에 의해 제시되었다. 또한, 소프트웨어 관리자들에 소프트웨어 및 검사 도구에 익숙해지는 학습 과정을 설명할 수 있는 모형은 S-형태모형이라는 설명하기도 하였다[9]. 이러한 모형과 관련하여 kim[10]은 수명분포가 반로지스틱분포를 이용한 학습과정에 대

한 연구도 이루어졌다.

본 논문에서는 소프트웨어 NHPP 신뢰성 모형(Goel--Okumo 모형, 지연된 S-형태 신뢰성모형 및 레일리분포 모형)의 예측능력을 분석하는 것을 목적으로 한다. 모형의 예측 능력의 비교에 있어서는 가능한 고장 데이터의 마지막 20%가 되는 선택된 자료를 이용하였다. 예측분석은 고장자료에 대한 적용성의 정도와 관찰시점 이후의 예측능력 정도를 비교하고자 한다.

2. 관련연구

2.1 Goel-Okumoto 모형

소프트웨어 신뢰성이 분야에서 Goel-Okumoto 모형은 가장 기본적인 모형으로 알려져 있다. 이모형은 결함 당 고장발생 시간의 분포를 따르는 수명 분포는 지수분포를 가정하고 또한, 소프트웨어 결함(Fault)당 고장 발생률은 일정한 형태 즉 상수를 가지며 평균값함수 및 강도함수 형태는 다음과 같다[3].

$$m(t|\theta, \beta_1) = \theta F(t) = \theta(1 - e^{-\beta_1 t}) \quad (\theta > 0, \beta_1 > 0) \quad (1)$$

$$\lambda(t|\theta, \beta_1) = \theta f(t) = \theta \beta_1 e^{-\beta_1 t} \quad (2)$$

단, θ 는 결함들(Faults)의 기대 값이고 β_1 는 고장 발생률을 의미한다. (1)식과 (2)식에서 t 을 최종고장시점 x_n 으로 변환하면 우도함수는 다음과 같이 표현 할 수 있다.

$$L_{NHPP}(\theta | \underline{x}) = \left(\prod_{i=1}^n \theta \beta_1 e^{-\beta_1 x_i} \right) \exp(-\theta(1 - e^{-\beta_1 x_n})) \quad (3)$$

단, $\underline{x} = (x_1, x_2, x_3, \dots, x_n)$

최우추정법(MLE)이용하여 모수를 추정하기 위한 로그우도함수는 (3)식을 이용하면 다음과 같이 표현된다.

$$\ln L_{NHPP}(\theta | \underline{x}) = n \ln \theta + n \ln \beta_1 - \beta_1 \sum_{i=1}^n x_i - \theta(1 - e^{-\beta_1 x_n}) \quad (4)$$

따라서 고장절단모형을 가정하면 모수 θ 와 β_1 에 관한 편미분식은 다음과 같은 식을 만족한다[12].

$$\frac{\partial \ln L_{NHPP}(\theta | \underline{x})}{\partial \theta} = \frac{n}{\theta} - 1 + e^{-\beta_1 x_n} = 0 \quad (5)$$

$$\frac{\partial \ln L_{NHPP}(\theta | \underline{x})}{\partial \beta_1} = \frac{n}{\beta_1} - \sum_{i=1}^n x_i - \theta x_n e^{-\beta_1 x_n} = 0 \quad (6)$$

단, $\underline{x} = (x_1, x_2, x_3, \dots, x_n)$

따라서 (5)식과 (6)식을 요약정리하면 각 모수에 대한 최우추정량 $\hat{\theta}_{MLE}$ 와 $\hat{\beta}_{1MLE}$ 은 다음과 같이 표현 할 수 있다[3].

$$\frac{n}{\hat{\theta}} = 1 - e^{-\hat{\beta}_1 x_n} \quad (7)$$

$$\frac{n}{\hat{\beta}_1} = \sum_{i=1}^n x_i + \hat{\theta} x_n e^{-\hat{\beta}_1 x_n} \quad (8)$$

2.2 지연된 S-형태 신뢰성장모형

지연된 S-형태 신뢰성장모형(Delayed S-shaped reliability growth model)[4]의 유한고장 NHPP 모형의 강도함수와 평균값 함수는 다음과 같이 알려져 있다.

$$\lambda(t|\theta, \beta_2) = \theta F'(t) = \theta \beta_2^2 t e^{-\beta_2 t} \quad (\theta, \beta_2 > 0) \quad (9)$$

$$m(t|\theta, \beta_2) = \theta F(t) = \theta [1 - (1 + \beta_2 t) e^{-\beta_2 t}] \quad (10)$$

이경우의 우도함수는 (9)식과 (10)식을 이용하면 다음과 같다.

$$L_{NHPP}(\theta | \underline{x}) = \left[\prod_{i=1}^n (\theta \beta_2^2 x_i e^{-\beta_2 x_i}) \right] \cdot \exp\{-\theta [(1 + \beta_2 x_n) e^{-\beta_2 x_n}]\} \quad (11)$$

단, $\underline{x} = (x_1, x_2, x_3, \dots, x_n)$

따라서 Goel-Okumoto 모형과 유사하게 로그 우도함수는 (11)식과 관련하면 다음과 같이 유도된다.

$$\ln L_{NHPP}(\theta | \underline{x}) = n \ln \theta + 2n \ln \beta_2 + \sum_{i=1}^n \ln x_i - \beta_2 \sum_{i=1}^n x_i - \theta [(1 + \beta_2 x_n) e^{-\beta_2 x_n}] \quad (12)$$

따라서 (12)식에서 θ 와 β_2 에 대하여 편미분 하면 다음과 같은 식을 만족하는 $\hat{\theta}_{MLE}$ 와 $\hat{\beta}_{2MLE}$ 을 이분법(수치 해석적 방법)을 이용하면 추정 할 수 있다.

$$\frac{\partial \ln L_{NHPP}(\theta | \underline{x})}{\partial \theta} = \frac{n}{\theta} - 1 + e^{-\beta_2 x_n} + \beta_2 x_n e^{-\beta_2 x_n} = 0 \quad (13)$$

$$\hat{\theta} = \frac{n}{1 - e^{-\beta_2 x_n} (1 + \beta_2 x_n)} \quad (14)$$

$$\frac{\partial \ln L_{NHPP}(\theta | \underline{x})}{\partial \beta_2} = \frac{2n}{\beta_2} - \theta \beta_2 x_n^2 e^{-\beta_2 x_n} - \sum_{i=1}^n x_i = 0 \quad (15)$$

2.3 레일리분포 모형

와이블분포(Weibull distribution)의 특수한 형태(형상모수가 2)인 레일리분포(Rayleigh distribution)는 소프트웨어 신뢰성분야에서 수명시험 및 신뢰성이론에 적합한 모형으로 알려진 분포 중에 하나이다.

이러한 레일리분포의 확률밀도함수와 분포함수는 다음과 같이 정리 할 수 있다[12].

$$f(t) = 2bt \exp(-bt^2) \quad (16)$$

$$F(t) = 1 - \exp(-bt^2) \quad (17)$$

단, $b > 0, t \in [0, \infty]$.

따라서 유한고장 NHPP를 적용하면 강도함수와 평균값함수는 (16)식과 (17)식을 이용하면 다음과 같이 구성 할 수 있다.

$$\lambda(t) = \theta F'(t) = 2\theta bt \exp(-bt^2) \quad (\theta, b > 0) \quad (18)$$

$$m(t) = \theta F(t) = \theta [1 - \exp(-bt^2)] \quad (19)$$

이경우의 우도함수는 다음과 같이 표현 할 수 있다.

$$L_{NHPP}(\theta | \underline{x}) = \left[\prod_{i=1}^n 2\theta b x_i e^{-bx_i^2} \right] \exp[-\theta (1 - e^{-bx_n^2})] \quad (20)$$

따라서 최우추정치(Maximum Likelihood Estimator) $\hat{\theta}_{MLE}$ 와 \hat{b}_{MLE} 는 Goel-Okumoto 모형과 레일리분포 모형과 유사하게 (20)식을 이용한 로그우도함수를 이용하면 다음과 같은 식을 만족하는 방법으로 모수를 추정 할 수 있다.

$$\hat{\theta} = \frac{n}{1 - e^{-bx_n^2}} \quad (21)$$

$$\frac{n}{b} = \sum_{i=1}^n x_i^2 + \theta x_n^2 e^{-bx_n^2} \quad (22)$$

2.4 효율적 모형을 위한 모형의 비교기준

평균제곱오차(Mean square error; MSE)와 결정계수(Coefficient of determination; R^2)를 사용하면 모형에 대한 효율성을 조사하기 위한 기준이 될 수 있다고 알려져 있다[9, 11].

즉, 실제 관찰 값과 예측 값에 대한 차이를 비교하는 도구로서 평균제곱오차(MSE)는 다음과 같이 알려져 있다.

$$MSE = \frac{\sum_{i=1}^n (m(x_i) - \hat{m}(x_i))^2}{n - k} \quad (23)$$

단, $m(x_i)$ 은 시간(0, x_i]까지 발생한 고장들의 실제누적개수(참값)를 의미하고 $\hat{m}(x_i)$ 는 x_i 시점까지 평균값함수로부터 추정된 고장의 누적개수(추정값)가 된다. 또한, n 은 관찰 혹은 측정값의 수이고 k 는 추정하고자 하는 모수의 수를 의미한다. 즉, 효율적인 모형의 비교는 작은 평균제곱오차 값을 가진 모형이 된다.

그리고 다른 측면으로 예측 값의 차이에 대한 설명력(R^2)을 나타내는 도구가 결정계수가 있다. 이 경우는 큰 결정계수를 가진 모형은 효율적인 모형으로 알려져 있다.

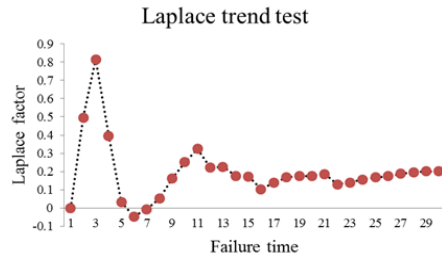
$$R^2 = 1 - \frac{\sum_{i=1}^n (m(x_i) - \hat{m}(x_i))^2}{\sum_{i=1}^n (m(x_i) - \sum_{j=1}^n m(x_j)/n)^2} \quad (24)$$

3. 소프트웨어 고장시간 자료 분석 및 예측

본 논문에서 제시하는 소프트웨어 신뢰모형들을 이용한 소프트웨어 고장시간을 분석을 위하여 소프트웨어 고장시간자료[13] (Failure interval time data)를 이용하였다. 이 자료는 30번의 고장이 발생된 자료이며 <Table 1>에 요약 되었다.

<Table 1> Failure time data

Failure number	Failure time (hours)	Failure interval (hours)
1	0.479	4.79
2	0.745	2.66
3	1.022	2.77
4	1.576	5.54
5	2.61	10.34
6	3.559	9.49
7	4.252	6.93
8	4.849	5.97
9	4.966	1.17
10	5.136	1.7
11	5.253	1.17
12	6.527	12.74
13	6.996	4.69
14	8.17	11.74
15	8.863	6.93
16	10.771	19.08
17	10.906	1.35
18	11.183	2.77
19	11.779	5.96
20	12.536	7.57
21	12.973	4.37
22	15.203	22.3
23	15.64	4.37
24	15.98	3.4
25	16.385	4.05
26	16.96	5.75
27	17.237	2.77
28	17.6	3.63
29	18.122	5.22
30	18.735	6.13



[Fig. 1] Laplace trend test

또 한 추세검정을 통하여 극단값(Extreme value)이 존재하는지를 알아보기 위하여 추세검정을 실시하였다.[14]. 추세분석에는 플라스 추세검정(Laplace trend test)을 이용하였고 이 검정을 실시한 결과 [Fig. 1]에서 라플라스요인(Laplace factor)이 -2와 2사이에 존재함으로써 라플라스 추세검정의 결과는 극단값이 나타나지 않기 때문에 이 자료를 이용하여 신뢰도 모형을 비교하는

작업이 가능함이 알려져 있다[11, 14].

3.1 예측 능력 접근 방법

예측타당성과정은 다음과 같은 과정으로 구성되었다[15].

· [단계 1]

첫 시점부터 고장자료의 80%을 이용하여 각 모형에 대한 모수를 추정(본 연구에서는 사용된 고장수는 $30 \times 0.8 = 24$, 모형은 Goel-Okumo 모형, 지연된 S-형태 신뢰성모형 및 레일리분포 모형)

· [단계 2]

데이터의 마지막 20%가 되는 선택된 자료(본 연구에서는 사용된 고장 수는 $30 \times 0.2 = 6$)를 이용하고 [단계 1]에서 추정된 모수를 이용하여 각 모형에 대하여 고장시간을 예측한다.

· [단계 3]

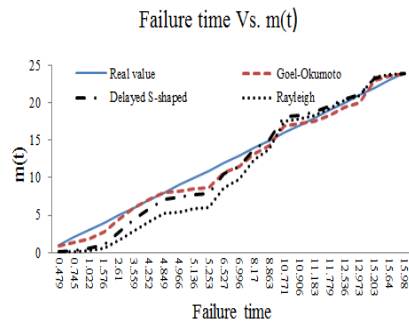
[단계 2]에서 예측된 고장시간에 대하여 평균제곱오차(MSE)와 결정계수(R^2)을 예측성능을 비교 분석한다.

전체 30개의 시간고장자료에서 24개의 자료에 대한 모수추정방법은 최우추정법과 비선형방정식의 계산방법은 이분법(Bisection method)을 사용하였다. 그리고 이러한 계산은 초기 값 0.0001과 10을 이용하고 허용한계(Tolerance for width of interval)는 10^{-5} 을 기준으로 하였고 수렴성을 확인하면서 100번인 충분한 반복횟수를 C-언어를 이용하여 모수 추정의 결과를 시행한 결과는 <Table 2>에 요약되었다. 또한 이 표에서 MSE(평균제곱오차)와 R^2 (결정계수)가 요약되었다. 이러한 수치는 모형에 대한 효율성을 조사하기 위한 기준으로서 활용할 수 있다. 결국 24개의 자료에 대해서 MSE 측면에서는 Rayleigh모형이 Delayed S-shaped모형과 Goel-Okumoto 모형보다 작게 나타나고 있다. 즉 실제 관찰(측정값) 값과 예측 값(추정값)에 대한 차이 측면에서는 비교적 Rayleigh모형이 효율적으로 판단되고 R^2 측면에서는 Delayed S-shaped모형이 크게 나타나고 있다. 즉 설명력이(예측 값의 차이에 대한) 높게 나타나기 때문에 설명력부분에서는 비교에 있어서 다른 모형보다 효율적인 모형으로 설명할 수 있다.

<Table 2> Parameter estimation of each model

Model	MLE	Model Comparison	
		MSE	R^2
Goel-Okumoto	$\hat{\theta}_{MLE} = 90.8840,$ $\hat{b}_{MLE} = 0.0192$	13.0604	0.9867
Delayed S-shaped	$\hat{\theta}_{MLE} = 29.4832,$ $\hat{b}_{MLE} = 0.1934$	12.9010	0.9896
Rayleigh	$\hat{\theta}_{MLE} = 26.3633,$ $\hat{b}_{MLE} = 0.0095$	11.9359	0.9881

Note. MLE : Maximum likelihood estimation;
 MSE : Mean square error;
 R^2 : Coefficient of determination.



[Fig. 2] Laplace trend test

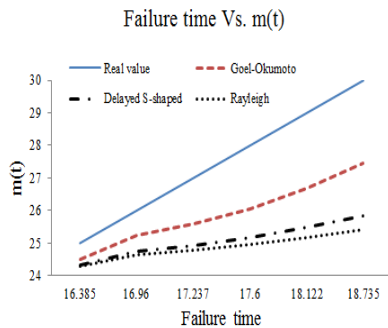
[Fig. 2]에서 24개의 자료에 대한 평균값함수를 나타내었다. 이 그림에서 모든 모형이 유사한 패턴으로 나타났지만 전반부에서는 Delayed S-shaped모형이 상대적으로 참값과 유사하고 후반부에서는 Goel-Okumoto모형이 유사하게 나타나고 있다.

마지막 20%가 되는 선택된 자료($30 \times 0.2 = 6$)를 이용한 각 모형에 대한 예측값에 대한 MSE와 R^2 의 값이 <Table 3>에 요약되었다. 이 표에서 Goel-Okumoto모형이 보다 작게 나타나고 있다. 따라서 실제 관찰값(측정값)과 예측 값에 대한 차이 측면에서는 비교적 Goel-Okumoto모형이 효율적으로 판단되고 R^2 측면에서도 Goel-Okumoto모형이 크게 나타나고 있기 때문에 상대적으로 다른 모형보다 효율적인 모형으로 설명이 가능하다.

〈Table 3〉 MSE and R^2 of each model

Model	Model Comparison	
	MSE	R^2
Goel-Okumoto	4.6118	0.9893
Delayed S-shaped	10.9814	0.9862
Rayleigh	13.0668	0.9733

[Fig. 3]에서는 예측 값에 대한 평균값함수의 패턴을 요약하였다. 이 그림에서 참값과 비교하여 제안한 모형이 모두 과소평가 추정되었다.



[Fig. 3] Mean value function of each model

5. 결론

빅 데이터를 포함하는 소프트웨어는 실행하는 과정에서 수정과 변경하는 과정에서 결점의 발생을 피할 수 없는 경우가 자주 발생하는 환경이 많다.

소프트웨어 신뢰성은 소프트웨어 개발의 최종단계에 있는 테스트 공정 혹은 실제 사용단계에 있어서 소프트웨어 내에 발생하는 고장 수나 고장 발생시간에 의해서 구체적으로 비교할 수 있는 환경으로 그 평가기술이 핵심 관심사가 된다. 결국 소프트웨어개발의 테스트공정이나 실제사용단계에 있는 고장정보를 이용하면 고장 발생환경이나 고장 발생환경을 구체적으로 모형화가 가능하면 그 기능에 대한 효율성을 평가를 할 수 있다. 테스트시간이나 혹은 실행시간, 발생된 고장 수와 고장 발생시간과의 관계를 효율적으로 관리함으로써 소프트웨어 신뢰도를 향상 시킬 수 있다. 이러한 과정을 소프트웨어 성장과정이라고 알려져 있다.

본 연구에서는 소프트웨어 NHPP 신뢰성 모형 (Goel-Okumoto 모형, 지연된 S-형태 신뢰성모형 및 레일리분포 모형)의 예측능력을 분석하는 것을 목적으로 한다. 그 결과 기존의 80%개의 자료에 대해서 본 논문에서 제안된 모형 중에서 실제 관찰 값과 예측 값에 대한 차이에 대해서 비교적 Rayleigh모형이 효율적으로 설명되고 예측 값의 차이에 대한 설명력을 의미하는 설명력 부분에서는 Delayed S-shaped모형이 효율적 모형으로 설명된다. 그러나 모형의 예측 능력의 비교를 위하여 마지막 20%가 되는 고장자료에 대한 예측에서는 실제 관찰 값과 예측 값에 대한 차이와 설명력에 관련해서 Goel-Okumoto 모형이 다른 모형보다 효율적인 모형으로 설명할 수 있다.

이 연구를 통하여 소프트웨어 개발자 혹은 관리자들은 여러경우의 소프트웨어 고장시간 환경을 고려함으로써 소프트웨어 고장형태에 대한 기번 사전지식을 이해하는데 기초적인 사전정보를 파악 할 수 있는 지침으로 활용 할 수 있으리라 사료된다.

REFERENCES

- [1] L. Kuo and T. Y. Yang., Bayesian Computation of Software Reliability, Journal of the American Statistical Association, Vol.91, pp. 763-773, 1996.
- [2] Gokhale, S. S. and Trivedi, K. S. A time/structure based software reliability model, Annals of Software Engineering. 8, pp. 85-121. 1999.
- [3] Goel A L, Okumoto K, Time-dependent fault detection rate model for software and other performance measures, IEEE Trans. Reliab. 28, pp.206-11, 1978.
- [4] Yamada S, Ohba H. and Osaki S., S-shaped software reliability modeling for software error detection", IEEE Trans. Reliab, 32, pp.475-484, 1983.
- [5] Zhao M., Change-point problems in software and hardware reliability", Commun. Stat. Theory Methods, 22(3), pp.757-768, 1993.
- [6] Shyur H-J., A stochastic software reliability model with imperfect debugging and change-point, J.

- Syst. Software 66, pp.135-141, 2003.
- [7] Pham H, Zhang X., "NHPP software reliability and cost models with testing coverage", Eur. J. Oper. Res, 145, pp.445-454, 2003.
- [8] Huang C-Y., "Performance analysis of software reliability growth models with testing-effort and change-point", J. Syst. Software 76, pp. 181-194, 2005.
- [9] 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 93, pp. 1410 - 1421, 2008.
- [10] Hee-Cheul KIM, "The Comparative Study of NHPP Half-Logistic Distribution Software Reliability Model using the Perspective of Learning Effects", Journal of Next Generation Information Technology, Vol. 4, No. 8, pp. 132-139, 2013.
- [11] Hee-Cheul KIM, "The Comparative Study of NHPP Delayed S-Shaped and Extreme Value Distribution Software Reliability Model using the Perspective of Learning Effects", International Journal of Advancements in Computing Technology, Vol. 5, No.9, pp. 1210 -1218, 2013.
- [12] Kim, Hee-cheul, " The Assessing Comparative Study for Statistical Process Control of Software Reliability Model Based on Rayleigh and Burr Type", Journal of the Korea Society of Digital Industry and Information Management, pp. 1-11, 2014.
- [13] Y. HAYAKAWA and G. TELFAR, "Mixed Poisson - Type Processes with Application in Software Reliability", Mathematical and Computer Modelling, 31, pp. 151-156, 2000.
- [14] K. Kanoun and J. C. Laprie, "Handbook of Software Reliability Engineering", M.R.Lyu, Editor, chapter Trend Analysis. McGraw-Hill New York, NY, pp. 401-437, 1996.
- [15] D. R. Prince Williams, "Prediction Capability Analysis of Two and Three Parameters Software Reliability Growth Models", Information Technology Journal, 5(6), pp.1048-1052, 2006.

김희철(Kim, Hee Cheul)



- 1992년 2월 : 동국대학교 통계학과 (이학 석사)
- 1998년 8월 : 동국대학교 통계학과 (이학박사)
- 2005년 3월 ~ 현재 : 남서울대학교 산업경영공학과 교수
- 관심분야 : 소프트웨어신뢰성 공학, 전산 통계

· E-Mail : kim1458@nsu.ac.kr

김경수(Kim, Kyung Soo)



- 2001년 8월 순천향대학교 전산학과(공학 박사)
- 2005년 3월 ~ 2007년 2월: VCU DBLab Visiting Scholar
- 1998년 3월 ~ 현재 : 백석문화대학교 인터넷정보학부 교수
- 관심분야 : 소프트웨어신뢰성 공학, 정보보안, 웹 프로그래밍

· E-Mail : kkskim@bscu.ac.kr