

다항 위험함수에 근거한 NHPP 소프트웨어 신뢰성장모형에 관한 연구*

김희철*

A Study for NHPP Software Reliability Growth Model based on polynomial hazard function

Kim, Hee Cheul

〈Abstract〉

Infinite failure NHPP models presented in the literature exhibit either constant, monotonic increasing or monotonic decreasing failure occurrence rate per fault (hazard function). This infinite non-homogeneous Poisson process is model which reflects the possibility of introducing new faults when correcting or modifying the software. In this paper, polynomial hazard function have been proposed, which can efficiency application for software reliability. Algorithm for estimating the parameters used to maximum likelihood estimator and bisection method. Model selection based on mean square error and the coefficient of determination for the sake of efficient model were employed. In numerical example, log power time model of the existing model in this area and the polynomial hazard function model were compared using failure interval time. Because polynomial hazard function model is more efficient in terms of reliability, polynomial hazard function model as an alternative to the existing model also were able to confirm that can use in this area.

Key Words : NHPP, Polynomial Hazard Function, Coefficient of Determination

I. 서론

소프트웨어 테스트 단계에서 소프트웨어 고장 수 (Number of failure)와 고장간격시간에 의해 소프트웨어 고장현상을 수리적으로 모형 화하면 소프트웨어에 대한 평가를 보다 쉽게 할 수 있으며 신뢰도 모형에 의해 소프트웨어 고장 수, 소프트웨어 고장발생간격시간, 소프트웨어 신뢰도 및 고장률 등의 신뢰성 평가측도들이 추정되

어 미래의 고장시간을 예측할 수도 있다. 소프트웨어 고장시간은 수명자료가 된다. 따라서 비음(Non-negative)의 값을 가지기 때문에 이 분야에서는 주로 지수분포, 와이 블분포, 감마분포 등 일반화 감마 분포(혹은 어랑분포)가 많이 사용되어 왔다[1].

이러한 소프트웨어 모형에 연구들은 대부분 유한 고장 NHPP(Non-Homogeneous Poisson Process)모형을 사용하였다[1-2]. 이러한 유한(Finite)고장 NHPP모형은 소프트웨어가 유한개의 고장이 있고 고장 제거 단계에서는 새로운 고장이 발생하지 않는다는 가정을 한 모형이다. 그러나 실제 고장 제거 단계에서도 새로운 고장이 발

* 이 논문은 2011년도 남서울대학교 학술연구비 지원에 의하여 연구되었음.

** 남서울대학교 산업경영공학과 교수(교신저자)

생할 수도 있다. 따라서 본 연구에서는 무한(Infinite) 고장 NHPP 모형을 이용하여 소프트웨어 신뢰성 모형에 대한 문제를 제안하고자 한다.

이 분야에서는 Musa-Okumoto의 대수 포아송 실행시간(Logarithmic Poisson execution time)[3] 모형과 로그 파워어(Log power)[4-5] 모형 등이 이미 연구되었고 최근까지도 이와 관련된 연구는 계속 수행되고 있다

본 연구에서는 소프트웨어의 결함을 제거하거나 수정 작업 중에도 새로운 결함이 발생될 가능성이 있는 무한 고장수를 가진 비동질적인 포아송 과정에 기초하고 시스템이 복잡해지면 다양한 원인에 의해 발생할 수 있는 다항모형을 이용한 신뢰성장모형에 관한 문제를 다루었다.

본 연구의 2장에서는 관련 연구로서 무한고장 NHPP와 기존의 로그 파워어 실행시간 모형과 평균제곱오차와 결정계수를 나열하였고 3장에서는 제안된 다항 위험함수를 이용한 NHPP모형을 제시하였으며 4장에서는 수치적인 예와 그 결과를 그리고 5장에서는 결론을 나열하였다.

II. 관련연구

2.1 무한고장 NHPP

신뢰도에서 관측시간 $(0, t]$ 사이에 발견된 고장 수 $N(t)$ 을 모형화 하는데 비동질적 포아송 과정(NHPP)이 널리 사용하여 왔다.

이 과정(Process)에서 강도함수(Intensity function) $\lambda(t) = dE[N(t)]/dt$ 은 t 에 대한 단조(Monotonic)함수로 흔히 가정한다[6].

NHPP 모형에서 평균값 함수 $m(t)$ (Mean value function)와 강도 함수 $\lambda(t)$ 는 다음과 같은 관계로 표현할 수 있다[6-7].

$$m(t) = \int_0^t \lambda(s)ds, \quad \frac{dm(t)}{dt} = \lambda(t) \quad (1)$$

따라서 $N(t)$ 는 모수 $m(t)$ 을 가진 포아송 확률 밀도 함수(Probability density function)는 다음과 같이 알려져 있다. 즉,

$$P\{N(t) = n\} = \frac{[m(t)]^n \cdot e^{-m(t)}}{n!}, \quad n = 0, 1, 2, \dots, \infty \quad (2)$$

이처럼 시간 관련 모형(Time domain models) 들은 NHPP에 의해서 확률 고장 과정으로 설명이 가능하다. 이러한 모형들은 고장 강도 함수 $\lambda(t)$ 가 다르게 표현됨으로서 평균값 함수 $m(t)$ 도 역시 다르게 나타나고 이러한 NHPP 모형들은 유한 고장 모형과 무한 고장 범주로 분류한다.

수리 시점에서도 고장이 발생할 수도 있는 상황을 반영하기 위하여 RVS(Record Value Statistics)모형을 사용하는 NHPP 모형의 평균값함수는 다음과 같이 알려져 있다[7].

$$m(t) = -\ln(1 - F(t)) \quad (3)$$

따라서 (1)식 과 (3)식을 연관시키고 $f(t)$ 을 확률밀도 함수, $F(t)$ 을 분포함수라고 하면 다음과 같은 관계식에 의해 NHPP의 강도함수는 $F(t)$ 의 위험함수($h(t)$)가 된다.

$$\lambda(t) = m'(t) = f(t)/(1 - F(t)) = h(t) \quad (4)$$

시간 $(0, t]$ 까지 조사하기 위한 시간 절단(Time truncated) 모형은 n 번째 까지 고장 시점 자료를

$$x_k = \sum_{i=1}^k t_k \quad (k = 1, 2, \dots, n; 0 \leq x_1 \leq x_2 \leq \dots \leq x_n) \quad (5)$$

이라고 하고 θ 을 모수공간이라고 하면 우도 함수는 다음과 같이 알려져 있다[1, 7].

$$L_{NHPP_{NIN}}(\theta | \underline{x}) = \prod_{i=1}^n \left(\frac{f(x_i)}{1-F(x_i)} \right) (1-F(x_n)) \quad (6)$$

$$= \left(\prod_{i=1}^n h(x_i) \right) (1-F(x_n))$$

단, $1-F(t) = e^{-m(t)}$, $\underline{x} = (x_1, x_2, x_3, \dots, x_n)$.
 NHPP 모형에서 테스트 시점 x_n 에서 소프트웨어 고장이 일어난다고 하는 가정 하에서 신뢰구간 $(x_n, x_n + t]$ (단, t 는 임무시간(Mission time)사이에서 소프트웨어의 고장이 일어나지 않을 확률인 신뢰도(Reliability) $\hat{R}(t | x_n)$ 는 다음과 같이 됨이 알려져 있다[1, 7].

$$\hat{R}(t | x_n) = e^{-\int_{x_n}^{x_n+t} \lambda(\tau) d\tau} \quad (8)$$

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

2.2 기존의 로그 파워어 NHPP 모형

로그 파워어 (Log power)모형[4-5]은 1999년에 Xie와 Homg에 의해서 발전된 무한 고장 소프트웨어 모형으로 평균값함수와 강도함수는 다음과 같이 알려져 있다.

$$m(t) = a \ln^b(1+t) \quad (9)$$

$$\lambda(t) = \frac{a b \ln^{b-1}(1+t)}{1+t} \quad (10)$$

단, $a(> 0)$ 는 척도모수이고 $b(> 0)$ 는 형상모수이며 $t \geq 0$.

한편, (6)식과 (9)식을 이용하면 우도함수는 다음과 같이 유도 할 수 있다.

$$L(a, b | \underline{x}) = \left(\prod_{i=1}^n \frac{a b \ln^{b-1}(1+x_i)}{1+x_i} \right) \cdot e^{-a \ln^b(1+x_n)} \quad (11)$$

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

최우추정법(MLE)을 이용하기 위한 로그 우도 함수는 다음과 같이 유도된다.

$$\ln L(a, b | \underline{x}) \quad (12)$$

$$= n \ln a + n \ln b - (b-1) \ln \left(\prod_{i=1}^n \ln(1+x_i) \right)$$

$$- \sum_{i=1}^n \ln(1+x_i) - a \ln^b(1+x_n)$$

(12)식에서 a 와 b 에 대하여 편미분 하여 다음과 같은 식을 만족하는 \hat{a}_{MLE} 와 \hat{b}_{MLE} 을 수치 해석적 방법으로 계산할 수 있다.

$$\frac{\partial \ln L(a, b | \underline{x})}{\partial a} = \frac{n}{a} - \ln^b(1+x_n) = 0 \quad (13)$$

$$\frac{\partial \ln L(a, b | \underline{x})}{\partial b} = \frac{n}{b} - \ln \left(\prod_{i=1}^n \ln(1+x_i) \right) - a \ln^b(1+x_n) \ln(\ln(1+x_n)) = 0 \quad (14)$$

(8) 식을 이용한 신뢰도(Reliability) $\hat{R}(t | x_n)$ 는 다음과 같이 표현 할 수 있다.

$$\hat{R}(t | x_n) = \exp[-a(\ln^b(1+(x+t)) + \ln^b(1+t))] \quad (15)$$

2.3 관측 자료에 대한 모형 비교

최근에 모형에 대한 효율성을 조사하기 위한 기준으로 MSE(평균제곱오차)와 R^2 (결정계수)를 사용한다[8].

2.3.1 평균제곱오차(Mean square error)

평균제곱오차는 실제 관찰 값과 예측 값에 대한 차이를 측정하는 도구로서 다음과 같이 정의 된다[9].

$$MSE = \frac{\sum_{i=1}^n (m(x_i) - \hat{m}(x_i))^2}{n-k} \quad (16)$$

단, $m(x_i)$ 은 시간 $(0, x_i]$ 까지 나타난 고장들의 누적 함수를 의미하고 $\hat{m}(x_i)$ 는 x_i 시점까지 평균값 함수로부터 추정된 고장의 누적개수를 의미한다. 그리고 n 은 관찰 값의 수이고 k 는 모수의 수를 의미한다.

2.3.2 R^2

R^2 (결정계수)는 관찰 값의 차이에 대한 설명력을 나타내는 도구로서 다음과 같이 정의 된다.

$$R^2 = 1 - \frac{\sum_{i=1}^n (m(x_i) - \hat{m}(x_i))^2}{\sum_{i=1}^n \left(m(x_i) - \sum_{j=1}^n m(x_j)/n \right)^2} \quad (17)$$

III. 제안된 다항 위험함수를 이용한 NHPP모형

다항 위험함수(Polynomial hazard function) 형태는 다음과 같이 알려져 있다[6].

$$h(t) = \beta_0 + \beta_1 t + \dots + \beta_{m-1} t^{m-1}, \quad t \geq 0 \quad (18)$$

단, β_i ($i = 0, 1, 2, \dots, m-1$)는 다항위험함수의 계수이고 무한고장 NHPP의 강도함수는 위험함수($h(t)$)가 된다고 하였기 때문에 다항형태의 무한고장 NHPP 강도함수 $\lambda(t)$ 는 (4)식과 동일한 위험함수 $h(t)$ 가 된다.

따라서 평균값함수 $m(t) = \int_0^t \lambda(s) ds$ 은 다음과 같이 유도된다.

$$m(t) = \beta_0 t + \frac{1}{2} \beta_1 t^2 + \frac{1}{3} \beta_2 t^3 + \dots + \frac{1}{m} \beta_{m-1} t^m \quad (19)$$

(19) 식에서 $m = 1$ 이면 지수분포가 되고 $m = 2$ 일 경우에는 선형 위험률 분포(Linear hazard rate distribution)라고 알려져 있다[10]. 그리고 $m > 2$ 경우에는 비 단조 위험 함수(non-monotone hazard function) 형태를 가진다고 하였다[6]. 따라서 본 연구에서는 $m = 2$ 인 경우를 이용한 다항모형을 적용하고자 한다.

이 경우의 다항 모형의 우도함수는 다음과 같이 표현 된다.

$$L_{NHPP_{poly}}(\Theta | \underline{x}) = \prod_{i=1}^n (\beta_0 + \beta_1 x_i) e^{-\beta_0 x_n - \frac{1}{2} \beta_1 x_n^2} \quad (20)$$

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

모수추정은 최우추정법(MLE)을 이용하였다. 이러한 추정 법을 사용하기 위한 로그 우도 함수는 (20)식을 연 관 하면 다음과 같이 유도된다.

$$\ln L_{NHPP_{m=1}}(\Theta | \underline{x}) = \beta_0 n \cdot e^{-\beta_0 x_n} \quad (21)$$

(21)식에서

$$\frac{\partial \ln L_{NHPP_{poly}}(\Theta | \underline{x})}{\partial \beta_0} = 0, \quad \frac{\partial \ln L_{NHPP_{poly}}(\Theta | \underline{x})}{\partial \beta_1} = 0$$

을 만족하는 $\hat{\beta}_{0,MLE}$ 와 $\hat{\beta}_{1,MLE}$ 는 다음과 같이 구할 수 있다[11].

$$\frac{\partial \ln L_{NHPP_{poly}}(\Theta | \underline{x})}{\partial \beta_0} = \sum_{i=1}^n \frac{1}{\beta_0 + \beta_1 x_i} - x_n = 0 \quad (23)$$

$$\frac{\partial \ln L_{NHPP_{poly}}(\Theta | \underline{x})}{\partial \beta_1} = \sum_{i=1}^n \frac{x_i}{\beta_0 + \beta_1 x_i} - \frac{1}{2} x_n^2 = 0 \quad (24)$$

그러나 위 식들은 근을 구하는데 근이 수렴되지 않음

므로 본 연구에서는 β_0 의 추정치를 고정된 상수로 간주하여 전개하고자 한다. 또한, (8)식을 이용한 신뢰도는 다음과 같다.

$$\widehat{R}_{Poly}(x | x_n) = \exp \left[-\beta_0(x_n + x) + \beta_0 x_n - \frac{1}{2} \beta_1(x_n + x)^2 + \frac{1}{2} \beta_1 x_n^2 \right] \quad (25)$$

21	0.437	12.973
22	2.230	15.203
23	0.437	15.640
24	0.340	15.980
25	0.405	16.385
26	0.575	16.96
27	0.277	17.237
28	0.363	17.600
29	0.522	18.122
30	0.613	18.735

IV. 수치적인 예

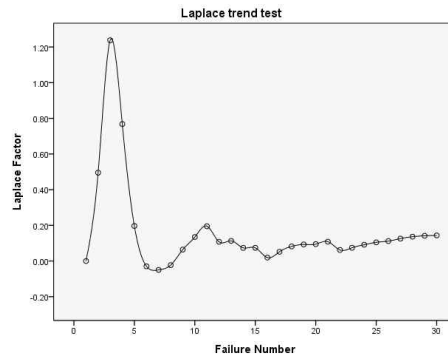
이 장에서 소프트웨어 고장 간격 시간 자료[11] (Failure interval time data)를 가지고 제시하는 신뢰모형들을 분석하고자 한다. 이 자료의 고장 시간은 18.735 시간단위에 30번의 고장이 발생된 자료이며 <표 1>에 나열되어 있다 이 고장간격자료의 기초통계량은 <표 2>에 요약되었다. 이 표에서 왜도와 첨도는 각각 양수로 나타나 정규분포와 비교하여 좀 더 뾰족하고 오른 쪽 꼬리를 가지는 분포 속성을 가지고 있다.

<표 1> 고장 자료

Failure Number	Failure Interval (second)	Failure Time(second)
1	0.479	0.479
2	0.266	0.745
3	0.277	1.022
4	0.554	1.576
5	1.034	2.610
6	0.949	3.559
7	0.693	4.252
8	0.597	4.849
9	0.117	4.966
10	0.170	5.136
11	0.117	5.253
12	1.274	6.527
13	0.469	6.996
14	1.174	8.170
15	0.693	8.863
16	1.908	10.771
17	0.135	10.906
18	0.277	11.183
19	0.596	11.779
20	0.757	12.536

<표 2> 고장 간격 시간자료의 기술통계량

기술 통계량	
평균	0.6245
중앙값	0.5005
첨도	3.8572
왜도	1.8807
범위	2.113
관측수	30
신뢰 구간(95.0%)	(0.4406, 0.8084)



<그림 1> 라플라스 추세 검정

또한 제시하는 신뢰 모형들을 분석하기 위하여 우선 자료에 대한 추세 검정이 선행 되어야 한다[12].

추세 분석에는 일반적으로 라플라스 추세 검정 (Laplace trend test)을 사용한다. 이 검정을 실시한 결과 <그림 1>에서 라플라스 추세 검정의 결과는 라플라스 요인(Factor)이 -2와 2사이에 존재함으로써 신뢰성장 (Reliability growth) 속성을 나타내고 있다. 따라서 이 자

료를 이용하여 신뢰 성장모형을 제시하는 것이 효율적임을 시사하고 있다[13].

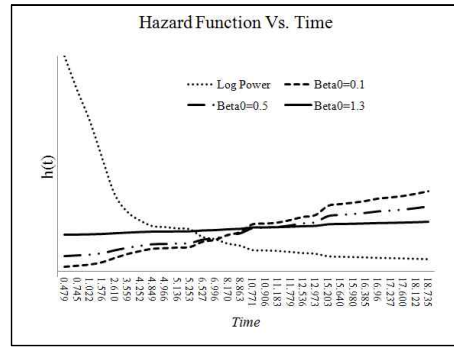
모수 추정에는 최우추정법을 이용하고 비선형 방정식의 계산방법은 수치 해석적 기본 방법인 이분법(Bisection method)을 사용하였다. 이러한 계산은 초기값을 0와 20을, 허용 한계(Tolerance for width of interval)는 10^{-5} 을 주고 수렴성을 확인 하면서 충분한 반복 횟수인 100번을 C-언어를 이용하여 모수 추정을 수행하였다. 이 과정에서 다항모형에서는 근 수렴이 되지 않아 모수추정을 용이하게 하기 위하여 β_0 을 0.1에서부터 1.6까지 고정된 상수(1.7 부터는 수렴하지 않아)로 가정하여 적용하였다. 그 결과는 <표 2>에 요약되었다. 그리고 이 표에 모형에 대한 효율성을 조사하기 위한 기준으로서 MSE(평균제곱오차)와 R^2 (결정계수)도 요약되었다.

<표 2> 각 모형의 모수 추정값 및 MSE와 R^2

Model	MLE		모형 비교		
			MSE	R^2	
Log Power	$\hat{a}_{MLE}=12.3516$,	$\hat{b}_{MLE}=0.8121$	65.1804	0.8813	
Polynomial	β_0	0.1	$\hat{\beta}_{1MLE}=0.1492$	33.0560	0.8822
		0.3	$\hat{\beta}_{1MLE}=0.1412$	20.7766	0.9373
		0.5	$\hat{\beta}_{1MLE}=0.0987$	14.0081	0.9486
		0.7	$\hat{\beta}_{1MLE}=0.0786$	7.9771	0.9580
		0.9	$\hat{\beta}_{1MLE}=0.0601$	7.7391	0.9846
		1.1	$\hat{\beta}_{1MLE}=0.0427$	6.6941	0.9879
		1.3	$\hat{\beta}_{1MLE}=0.0262$	3.2699	0.9894
		1.5	$\hat{\beta}_{1MLE}=0.0104$	1.6726	0.9879
		1.6	$\hat{\beta}_{1MLE}=0.0027$	1.6458	0.9899

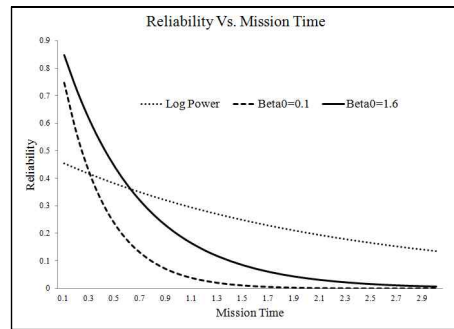
즉, 기존의 로그 파우어 모형 보다 다항모형이 평균제곱오차의 값이 상대적으로 작은 값을 나타내므로 다항모형이 효율적인 모형으로 나타났고 다항 모형에서도 β_0 의 값이 클수록 효율적인 모형으로 나타나고 있다. 그리고 설명력 측면에서도 유사하게 결정 계수값이 0.8보다

크게 나타나 실제 값과 추정된 값에 대한 차이의 대한 설명력이 높고 특히 β_0 의 값이 클수록 설명력이 높게 나타나고 있다.



<그림 2> 위험함수의 형태

<그림 2>에서는 위험함수의 패턴을 보여주고 있는데 기존의 로그 파우어 모형은 비증가함수 형태를 보이는 반면 다항모형은 비감소 형태를 보여주고 있고 β_0 의 값이 클수록 낮은 패턴으로 나타나고 있다.

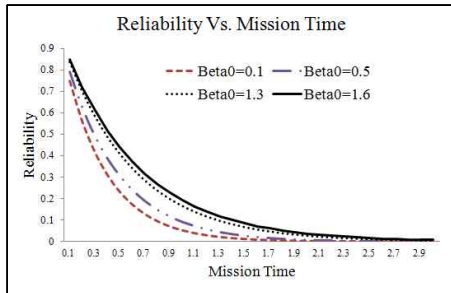


<그림 3> 신뢰도 비교

<그림 3>의 신뢰도 그림에서 보여 주듯이 기존의 로그 파우어 모형은 임무시간이 지남에 따라 완만히 감소하는 형태를 보여 주지만 다항 모형은 급격한 감소 형태를 보여주고 있고 <그림 4>에서는 β_0 의 값이 클수록 높은 신뢰도를 보이고 있다.

따라서 평균제곱오차 와 결정계수, 그리고 신뢰도 측

면에서도 기존의 로그 과우어 모형보다 위험함수가 다항인 경우가 효율적 모형으로 판단할 수 있다.



<그림 4> 다항 위험 함수의 신뢰도

V. 결론

소프트웨어 신뢰성은 개발의 최종단계에 있는 테스트 공정이나 실제 사용단계에 있어서 소프트웨어 내에 존재하는 고장 수나 고장 발생 시간에 의해서 효과적으로 평가할 수 있는 상황으로 그 평가 기술이 중요하게 된다. 따라서 소프트웨어 개발의 테스트 공정이나 실제사용단계에 있어서 고장 발생 환경이나 고장 발생 현상을 수리적으로 모형화가 가능하면 평가를 할 수 있다. 테스트 시간이나 혹은 실행 시간, 발생된 고장 수와 고장 발생시간과의 관계를 효율적으로 관리함으로써 소프트웨어 신뢰도를 성장 시킬 수 있다. 이러한 과정을 소프트웨어 성장 과정이라고 볼 수 있다. 본 연구에서는 소프트웨어의 결함을 제거하거나 수정 작업 중에도 새로운 결함이 발생할 가능성이 있는 무한고장수를 가진 비동질적인 포아송 과정에 기초하고 시스템이 복잡해지면 다양한 원인에 의해 발생 할 수 있는 다항모형을 이용한 신뢰성장모형에 관한 문제를 다루었다. 그 결과 기존의 로그 과우어 모형보다 다항 위험함수 모형이 우수한 모형이 됨을 확인 할 수 있다. 따라서 본 연구에 제안된 다항 위험함수 모형도 신뢰성 측면에서 더 효율적이기 때문에 이 분야

에서 기존의 모형의 대안으로 다항 위험함수 모형도 사용할 수 있음을 확인 할 수 있었다. 이 연구를 통하여 소프트웨어 개발자들은 단순 보다는 다항 모형을 고려함으로써 소프트웨어 고장 형태에 대한 사전 지식을 파악하는데 어느 정도 도움을 줄 수 있으리라 사료 된다.

참고문헌

- [1] Gokhale, S. S. and Trivedi, K. S. "A time / structure based software reliability model," *Annals of Software Engineering*, 8, 1999, pp. 85-121.
- [2] Hee-Cheul KIM and Hyung-Keun Park, "Exponentiated Exponential Software Reliability Growth model," *International Journal of Advancements in Computing Technology*, Volume 1, Number 2, 2009, pp. 57-64.
- [3] Musa, J. D, Iannino, A. and Okumoto, K. "Software Reliability: Measurement, Prediction, Application," McGraw Hill, New York, 1987, pp. 289-291.
- [4] Almering, V. and Genuchten, M, V and Cloudt, G. and Sonnemans, P. J. M, "Using Software Reliability Growth Models in Practice," *IEEE SOFTWARE*, 2007, pp. 82-88.
- [5] Hee-Cheul KIM and Hyung-Keun Park, "The Comparative Study of Software Optimal Release Time Based on Burr Distribution," *International Journal of Advancements in Computing Technology*, Volume 2, Number 3, 2010, pp. 119-128.
- [6] J. F. Lawless. *Statistical Models and Methods for Lifetime Data*. John Wiley & Sons, New York, 1981.
- [7] Kuo, L. and Yang, T. Y, "Bayesian Computation

of Software Reliability," Journal of the American Statistical Association, Vol. 91, 1996, pp. 763-773.

- [8] Kuei-Chen Chiu, Yeu-Shiang Huang, Tzai-Zang Lee. "A study of software reliability growth from the perspective of learning effects," Reliability Engineering and system safety, Vol. 93, 2008, pp. 1410-1420.
- [9] 김희철, "NHPP 극값 분포 소프트웨어 신뢰모 형에 대한 학습효과 기법 연구," 디지털산업정 보학회 논문지, 제 7권, 제 2호, 2011, pp. 1-8.
- [10] Bain, L. J. "Analysis for the linear failure rate distribution," Technometrics, 16, pp. 551-559, 1974.
- [11] Y. HAYAKAWA and G. TELFAR "Mixed Poisson-Type Processes with Application in Software Reliability," Mathematical and Computer Modelling, 31, 2000, pp. 151-156.
- [12] K. Kanoun, J. C. Laprie, "Handbook of Software Reliability Engineering," M. R. Lyu, Editor, chapter Trend Analysis. McGraw-Hill New York, NY, 1996, pp. 401-437.
- [13] 이상식, 김희철, "혼합와이블 NHPP 모형에 근 거한 소프트웨어 최적방출시기에 관한 연구," 디지털산업 정보학회 논문지, 제 6권, 제 2호, 2010, pp. 183-191.

논문접수일 : 2011년 10월 31일
수 정 일 : 2011년 11월 29일
게재확정일 : 2011년 12월 02일

■ 저자소개 ■



김 희 철
Kim, Hee Cheul

2005년 3월~현재
남서울대학교 산업경영공학과 교수
1998년 2월 동국대학교 통계학과(이학박사)
1992년 2월 동국대학교 통계학과(이학석사)
관심분야 : 소프트웨어 신뢰성공학, 웹
프로그래밍
E-mail : kim1458@nsu.ac.kr