
Kappa(2) 커버리지 함수를 이용한 ENHPP 소프트웨어 신뢰성장모형에 관한 연구

김 희 철*

The Study for ENHPP Software Reliability Growth Model Based on Kappa(2) Coverage Function

Hee-Cheul Kim*

요 약

유한고장수를 가진 비동질적인 포아송 과정에 기초한 모형들에서 잔존 오류 1개당 고장 발생률은 일반적으로 상수, 혹은 단조증가 및 단조 감소 추세를 가지고 있다. 소프트웨어 제품의 정확한 인도시기를 예측하거나 효율성 및 신뢰성을 예측하기 위해서는 소프트웨어 테스트 과정에서 중요한 요소인 테스트 커버리지를 이용하면 보다 효율적인 테스트 작업을 할 수 있다. 이런 모형을 ENHPP 모형이라고 한다. 본 논문에서는 기존의 소프트웨어 신뢰성 모형인 지수 커버리지 모형과 S-커버리지 모형을 재조명하고 이 분야에 적용될 수 있는 Kappa 분포를 이용한 모형인 Kappa 커버리지 모형을 제안하였다. 고장 간격시간으로 구성된 자료를 이용한 모수추정 방법은 최우추정법과 일반적인 수치해석 방법인 이분법을 사용하여 모수 추정을 실시하고 효율적인 모형 선택은 편차자승합(SSE) 및 콜모고로프 거리를 적용하여 모형들에 대한 효율적인 모형 선택도 시도 하였다. 수치적인 예에서는 실제 고장자료를 통하여 분석하였다. 이 자료들에서 기존의 모형과 Kappa 커버리지 모형의 비교를 위하여 산술적 및 라플라스 검정, 편의 검정 등을 이용하였다.

ABSTRACT

Finite failure NHPP models presented in the literature exhibit either constant, monotonic increasing or monotonic decreasing failure occurrence rates per fault. Accurate predictions of software release times, and estimation of the reliability and availability of a software product require Release times of a critical element of the software testing process : test coverage. This model called Enhanced non-homogeneous Poission process(ENHPP). In this paper, exponential coverage and S-shaped model was reviewed, proposes the Kappa coverage model, which make out efficiency application for software reliability. Algorithm to estimate the parameters used to maximum likelihood estimator and bisection method, model selection based on SSE statistics and Kolmogorov distance, for the sake of efficient model, was employed. Numerical examples using real data set for the sake of proposing Kappa coverage model was employed. This analysis of failure data compared with the Kappa coverage model and the existing model(using arithmetic and Laplace trend tests, bias tests) is presented.

키워드

Test Coverage, ENHPP, Kappa Distribution, Kolmogorov Distance

I. 서론

소프트웨어 테스트 단계에서 소프트웨어 고장수 (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 추정되어 미래의 고장시간을 예측할 수도 있다.

이 범주에서 지금까지 알려진 모형들은 Goel-Okumoto 모형, Weibull 모형 그리고 Cox-Lewis 모형 등이 있는데 이러한 모형들에 대한 강도함수는 각각 시간에 의존한 함수, 멱(Power) 함수, 대수 선형(Log-linear) 함수를 가정하였다[1,2].

소프트웨어 제품의 정확한 인도시기(Release times)를 예측하거나 효율성 및 신뢰성을 예측하기 위해서는 소프트웨어 테스트 과정에서 중요한 요소인 테스트 커버리지(Coverage)를 이용하면 보다 효율적인 테스트 작업을 할 수 있다.

이러한 모형은 기존에 존재하는 NHPP모형에서 테스트 커버리지를 포함하는 모형이 된다. 이런 모형을 ENHPP(Enhanced non-homogeneous Poisson process) 이라고 한다[3, 4].

본 논문에서는 분포함수가 효율적으로 사용할 수 있는 2모수 Kappa 분포[5, 6]을 이용한 Kappa 커버리지 함수를 제안하여 모형의 효율성과 그 특성을 알아보고자 한다.

II. 이론적 배경에 대한 고찰

2.1. 테스트 커버리지

프로그램 검사 능력과 테스트 커버리지(Coverage)는 관련성 있는 개념이다.

검사능력은 하나의 프로그램 상에서 검사할 수 있는 용이성이라면 커버리지는 프로그램 상에서 숨겨진 잠재적인 모든 오류 위치들을(All potential fault-sites) 충분히 찾아낼 수 있는 척도(Measure)를 제공한다. 여기서 잠재적인 오류 사이트들은 구조적으로나 기능적으로 명백히 설명할 수 있는 프로그램의 요소라고 정의 할 수 있고 이런 요소들은 합당하게 설계된 테스트를 통해 입

증 성이나 타당성이 요구되는 보전성(Integrity)을 가지고 있어야 한다[3, 4].

따라서 프로그램 검사 능력과 테스트 커버리지는 다음과 같은 요인에 의해 영향을 받는다.

(1) 프로그램의 복잡성, (2) 소프트웨어 개발 이론, (3) 사용된 소프트웨어 도구(Tools), (4) 테스트 질(Quality), (5) 테스트의 효율성.

테스트 커버리지에 대한 연구는 이미 여러 학자들에 의하여 연구되었다[7, 8]. 그러나 소프트웨어 개발자나 사용자들은 테스트 커버리지를 측정 할 수 있는 효율적인 평가 도구들을 원하고 있다. 이러한 도구들은 소프트웨어 제품의 인도시기 뿐만 아니라 효율적이고 질적인 테스트가 이루어 져야 한다.

테스트 커버리지는 테스트 작업 하에서 소프트웨어 제품에 숨겨진 잠재적인 모든 오류 사이트들을 얼마나 찾아낼 수 있는지를 평가하는 척도이다. 그러므로 잠재 오류 사이트와 오류 사이트는 서로 민감성을 가지고 있다. 따라서 테스트 커버리지는 소프트웨어 제품 과 테스트 지침이 설정되었을 때 고려대상에서 존재하는 잠재적인 총 오류 사이트수에 대한 테스트 하는 동안에 숨겨진 잠재적인 오류 사이트의 수의 비(Ratio)라고 정의되어 진다[3, 4]

2.2. 소프트웨어 신뢰성에 대한 테스트 커버리지

테스트 커버리지를 유한고장 비동질적 포아송 과정 NHPP에 적용 시킬 수 있다. 이러한 적용 모형을 확장된 비동질적 포아송 과정(ENHPP)이라고 부른다. 이러한 ENHPP 모형의 오류탐색비용은 숨겨진 잠재 오류 사이트의비와 남아있는 오류함의 기대수와의 곱에 비례한다. 이러한 ENHPP 모형은 다음과 같은 가정을 한다[3].

(가정 1) 오류들은 모든 잠재 오류 사이트에 대하여 균일하게(uniformly) 분포되어 있다.

(가정 2) 잠재 오류 사이트가 숨겨졌을 때 사이트에서 오류가 탐색되어 질수 있는 확률은 $K(t)$.

(가정 3) 수리는 효과적으로 즉시 이루어지고 새로운 오류는 발생되지 않는다. (이 가정은 유한 고장 NHPP 모형과 유사하다)

(가정 4) 커버리지는 테스트 시간에 대하여 연속적인 단조 비 감소함수(Monotonic non-decreasing function)를 따른다.

이러한 모형을 해석학적으로 표시하면 다음과 같이 표현된다.

$$\frac{dm(t)}{dt} = \tilde{a}K(t) \frac{dc(t)}{dt} \quad (2.1)$$

$$\text{혹은} \quad m(t) = \tilde{a} \int_0^t K(t)c'(\tau)d\tau.$$

단, \tilde{a} 는 무한 테스트 시간이 주어졌을 때 탐색될 수 있는 오류의 기대수이고 완전한 오류 탐색 커버리지는 $K(t) = 1$ 이고 완전 테스트 커버리지는 다음을 만족한다.

$$\lim_{t \rightarrow \infty} c(t) = 1 \quad (2.2)$$

만약 $K(t)$ 가 상수 K 로 가정하면 t 시점에서 탐색 될 오류의 기대수 $m(t)$ 는 다음과 같이 표현 가능하다.

$$m(t) = \tilde{a}Kc(t) \quad (2.3)$$

(2.3)식은 t 시점에서 탐색 될 오류의 기대 수는 오류의 총 오류의 기대수와 t 시점에서의 오류 탐색 커버리지의 곱과 같다.

따라서 $a = \tilde{a}K$ 로 하면 탐색 될 오류의 기대수 $m(t)$ 는 다음과 같이 또 다른 형태로 표현 가능하다.

$$m(t) = a c(t) \quad (2.4)$$

이런 결과로 인하여 고장 강도함수(Intensity function) $\lambda(t)$ 을 적용시키면 다음과 같다.

$$\frac{dm(t)}{dt} = \lambda(t) = a c'(t) \quad (2.5)$$

고장 강도 함수 $\lambda(t) = [a - m(t)] h(t)$ 임이 됨이 알려져 있으므로[1, 3] (2.5)식을 변형하면 다음과 같이도 표현도 가능하다.

$$\lambda(t) = [a - m(t)] \frac{c'(t)}{1 - c(t)} = [a - m(t)] h(t) \quad (2.6)$$

그러므로 한 개의 오류 당 오류발생비율인 위험함수(Hazard function) $h(t)$ 는 다음과 같이 유도 된다.

$$h(t) = \frac{c'(t)}{1 - c(t)} \quad (2.7)$$

따라서 이러한 ENHPP모형은 테스트가 진행됨에 따라 각각의 오류가 발생할 수 있는 비율은 시간에 따라 변화하기 때문에 시간에 의존하는 고장 발생 비율이라고 할 수 있다. 또, 이러한 위험함수에 해당하는 분포는 t 시점에서 커버리지 함수를 평가 할 수 있다.

이러한 ENHPP모형은 (불완전한) 커버리지 함수에 대하여 실제적인 상황($c(\infty) < 1$)을 적용시킬 수 있다.

결과적으로 조건부 신뢰도(Conditional reliability) $R(t | s)$ 는 다음과 같이 알려져 있다[3].

$$R(t | s) = e^{-\int_s^{s+t} \lambda(\tau)d\tau} = e^{-a[c(s+t) - c(s)]} \quad (2.8)$$

단, t 는 임무시간(Mission time)이고 s 는 마지막 고장 시간이다.

2.3. 커버리지 함수를 이용한 NHPP 모형(ENHPP)

고장 시간 영역 모형에서는 고장 강도 함수 $\lambda(t)$ 에 따라 다른 NHPP 모형이 된다. 따라서 커버리지 함수 $c(t)$ 도 다른 형태로 존재 한다. 결국 $c(t)$ 도 여러 가지 형태를 제시 할 수 있다. 본절에서는 잘 알려진 지수 커버리지 함수와 S-형 커버리지 함수를 재조명 한다.

2.3.1 지수 커버리지 함수

잘 알려진 Goel-Okumoto 모형[9, 10]은 이 분야에서 기본적인 모형이다. 이러한 모형은 지수 커버리지 함수를 이용한 모형 즉, 오류 당 고장 발생 비율이 상수를 가지는 모형이 된다. 이러한 지수 모형을 NHPP로 접근하면 평균값 함수와 강도 함수는 다음과 같이 알려져 있다.

$$m(t|a, g) = a(1 - e^{-gt}) \quad (a > 0, g > 0) \quad (2.9)$$

$$\lambda(t|a, g) = a g e^{-gt} \quad (2.10)$$

단, a 는 무한하게 테스트 시간이 주어졌을 때 관찰된 고장의 기대수이고 g 는 오류 당 고장 발생 비율을 의미한다.

따라서 지수모형에 대한 커버리지 함수 $c(t)$ 수와 위험함수 $h(t)$ 는 다음과 같이 유도 된다.

$$c(t) = 1 - e^{-g t} \quad (2.11)$$

$$h(t) = \frac{c'(t)}{1-c(t)} = g \quad (2.12)$$

2.3.2 S-형 커버리지 함수

S-형 커버리지 함수 모형[9, 11]은 Yamada-Ohba-Osaki 모형으로 알려져 있고 S-형 모형을 NHPP로 접근하면 평균값 함수와 강도 함수는 다음과 같이 알려져 있다.

$$m(t|a, g) = a [1 - (1 + g t) e^{-g t}] \quad (a > 0, g > 0) \quad (2.13)$$

$$\lambda(t|a, g) = a g^2 t e^{-g t} \quad (2.14)$$

따라서 S-형 모형에 대한 커버리지 함수 $c(t)$ 와 위험 함수 $h(t)$ 는 다음과 같이 유도 된다.

$$c(t) = 1 - (1 + g t) e^{-g t} \quad (2.15)$$

$$h(t) = \frac{c'(t)}{1-c(t)} = \frac{g^2 t}{1 + g t} \quad (2.16)$$

III. 제안한 Kappa(2) 커버리지 함수

소프트웨어 신뢰성이나 의료정보분야 등 여러 분야에서 널리 사용될 수 있는 분포 중에 하나가 3모수 (α, β, γ) Kappa 분포이다[5, 6].

$$f_{kappa(3)}(t) = \frac{\alpha \gamma}{\beta} \left(\frac{t}{\beta}\right)^{\gamma-1} [\alpha + (t/\beta)^\alpha \gamma]^{-(\alpha+1)/\alpha} \quad (3.1)$$

단, $t > 0$ 을 만족하고 $\beta (> 0)$ 는 척도모수이고 $\alpha (> 0)$ 와 $\gamma (> 0)$ 는 형상모수이다.

로그노오말(Lognormal)분포나 감마(Gamma)분포의 누적분포함수가 폐쇄형(Closed form)을 이루지 못하지만 이 Kappa분포는 폐쇄형을 이루고 있으므로 순서 통계량(Order statistics)의 정보를 설명하는데 편리하다는 장점을 가지고 있다[5].

따라서 본 논문에서의 소프트웨어 신뢰성 모형도 일반 순서 통계량에 의존되는 모형이므로 이 분포를 적용하는 것이 합리적이고 3모수 Kappa분포에서 형상모수

$\gamma = 1$ 인 2모수 Kappa분포도 3모수 Kappa 분포의 성질을 그대로 유지하고 있기 때문에 본 논문에서 2모수 Kappa 분포를 적용하고자 한다[4]. 2모수 Kappa 분포의 확률밀도 함수와 누적분포함수는 다음과 같다[6].

$$f_{kappa(2)}(t) = \alpha/\beta [\alpha + (t/\beta)^\alpha]^{-(\alpha+1)/\alpha} \quad (3.2)$$

$$F_{kappa(2)}(t) = [(\alpha/\beta)^\alpha / (\alpha + (t/\beta)^\alpha)]^{1/\alpha} \quad (3.3)$$

이러한 Kappa(2) 모형을 NHPP로 접근하면 평균값 함수와 강도 함수는 다음과 같이 표현 할 수 있다.

$$m(t|\alpha, \beta) = a [(\alpha/\beta)^\alpha / (\alpha + (t/\beta)^\alpha)]^{1/\alpha} \quad (3.4)$$

$$\lambda(t|\alpha, \beta) = a \alpha/\beta [\alpha + (t/\beta)^\alpha]^{-(\alpha+1)/\alpha} \quad (3.5)$$

단, $t > 0, \beta > 0, \alpha > 0$.

따라서 Kappa(2) 모형에 대한 커버리지 함수 $c(t)$ 와 위험함수 $h(t)$ 는 다음과 같이 유도 된다.

$$c(t) = [(\alpha/\beta)^\alpha / (\alpha + (t/\beta)^\alpha)]^{1/\alpha} \quad (3.6)$$

$$h(t) = \frac{\alpha/\beta}{(\alpha + (t/\beta)^\alpha) (\alpha + (t/\beta)^\alpha)^{1/\alpha} - \alpha/\beta} \quad (3.7)$$

IV. ENHPP 모형에 대한 모수 추정

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

$$x_k = \sum_{i=1}^k t_k \quad (k = 1, 2, \dots, n; 0 \leq x_1 \leq x_2 \leq \dots \leq x_n) \quad (4.1)$$

이라고 하면 데이터 집합 D_t 는 $\{n, x_1, x_2, \dots, x_n; t\}$ 와 같이 구성된다. n 번째까지 고장시점이 관찰된 고장 절단 모형일 경우에 데이터 집합 D_{x_n} 은 $\{x_1, x_2, \dots, x_n\}$ 으로 구성된다. 이 시간 절단 모형에서의 우도함수는 다음과 같이 알려져 있다[1, 3].

$$L(\theta | D_t) = \left(\prod_{i=1}^n \lambda(x_i) \right) \exp(-m(t)) \quad (4.2)$$

단, θ 은 미지의 모수 집합을 의미하고 우도함수 (4.2)

식에서 t 을 x_n 으로 대치하면 유사한 형태의 고장 절단 모형의 우도함수가 된다. 본 논문에서 이러한 최우 추정법(Maximum Likelihood Estimation; MLE)으로 모수 추정을 실시하고자 한다.

유사한 방법으로 (3.4)식과 (3.5)식을 연관하여 2모수 kappa(2) 커버리지 모형에 대한 로그우도함수는 다음과 같이 표현된다[5].

$$\begin{aligned} \ln L_{\text{kappa}(2)}(a, \alpha, \beta | D_{x_n}) & \quad (4.3) \\ &= n \ln a + n \ln \frac{\alpha}{\beta} - \left(\frac{\alpha+1}{\alpha} \right) \sum_{i=1}^n \ln (\alpha + (x_i/\beta)^\alpha) \\ & \quad - a \left(\frac{(\alpha/\beta)^\alpha}{(\alpha + (x_n/\beta)^\alpha)} \right)^{1/\alpha} \end{aligned}$$

본 연구에서는 $\alpha=2$ 인 경우를 고려하면 즉, 형상 모수 $\alpha=2$ 값은 상수(고정)라고 가정 했을 때 최우추정법을 이용하기 위하여 (4.3)식을 a 와 β 에 대하여 편미분을 하면 다음과 같은 식을 유도 할 수 있다.

$$\frac{n}{a} = \left(\frac{(2/\beta)^2}{(2 + (x_n/\beta)^2)} \right)^{1/2} \quad (4.4)$$

$$\frac{2n}{\beta} = 3/2 \sum_{i=1}^n \left(\frac{4\beta}{2\beta^2 + x_i^2} \right) + a \left(\frac{4\beta}{\sqrt{2\beta^2 + x_i^2}} \right) \quad (4.5)$$

(4.4) 식과 (4.5) 식을 비선형 연립방정식(수치해석적 방법)을 이용하여 풀면 최우추정치 $\hat{\beta}_{MLE}$ 와 \hat{a}_{MLE} 의 값을 구 할 수 있다.

V. 수치적인 예

이 장에서 실제적인 자료를 가지고 Kappa(2) 커버리지 모형을 분석하고자 한다. 고장자료는 Allen P.Nikora와 Michael R.Lyu가 인용한 SYS2 자료[12]을 이용하고자 한다. 제시하는 신뢰모형들을 분석하기 위하여 우선 자료에 대한 추세 검정이 선행 되어야 한다[1, 3]. 추세 분석에는 산술평균 검정(Arithmetic mean test)과 라플라스 추세 검정(Laplace trend test)등이 있다.

이 검정을 실시한 결과 <그림 1>에서는 산술평균 검정결과 고장수가 증가함에 따라 산술 평균이 거의 증가 추세를 보이고 있으므로 신뢰성장(Reliability growth)이 되고 있음을 나타내고 있고 라플라스 추세 검정의 결과 도 라플라스 요인(Factor)이 음수로서 감소하기 때문에 역시 신뢰성장(Reliability growth)이 되고 있음을 나타내고 있다. 따라서 이 자료를 가지고 신뢰성장 모형을 제시하는 것이 효율적임을 시사하고 있다[3]. 실제 분석에서는 원래의 자료를 변량변환(Variate transformation)시킨 고장 간격 데이터(102594×10^{-5} 시간(Second) 단위에서 고장이 86번 일어남)을 이용하였고 <표 1>에 자료가 나열 되어 있다.

소프트웨어 신뢰성 모형의 모수 추정은 최우추정법을 이용하였고 비선형 방정식의 계산방법은 수치해석적 기본 방법인 이분법(Bisection method)을 사용하였다. 이러한 계산은 초기 값을 10^{-1} 와 10을, 허용 한계(Tolerance for width of interval)는 10^{-10} 을 주고 수렴성을 확인 하면서 충분한 반복 횟수인 100번을 C-언어를 이용하여 모수 추정을 수행하였다 Kappa 모형을 포함한 모수의 추정 값들의 결과는 <표 2>에 요약되었다.

모형 선택의 하나의 방법으로 편차자승합(SSE [1,9])을 이용할 수 있는데 이 편차자승합이 작으면 상대적으로 효율적인 모형이 된다. 주어진 자료를 이용하여 제시된 모형들에 대한 편차자승합의 값은 <표 3>에 요약되었다. 이 표에서 Kappa모형이 기존에 알려진 모형인 S-커버리지 모형이나 지수 커버리지 모형에 비해 상대적으로 효율적 모형으로 나타나고 있다.

예측오류(Prediction error)의 비정상성(Non-stationarity)에 대한 측도는 Kolmogorov 거리(distance)[1, 3]로 측정되는데 이 거리가 클수록 상대적으로 비정상성을 내포하고 있다. <그림 2>은 S-Plus 소프트웨어[13]를 이용하여 Kolmogorov 검정에 대한 그림을 보여주고 있고 이 그림에서도 전체적으로 Kappa모형이 상대적으로 비정상성 속성이 덜 내포하고 있음을 알 수 있다. <표 4>는 Kolmogorov 거리를 요약한 표로서 모형에 대한 치우침(Bias)을 나타내고 있다. 이 표에서도 Kappa 커버리지 모형의 형태가 상대적으로 비정상성 속성이 덜 내포하고 있음을 알 수 있다.

표 1. 고 장 간격 자료
Table 1. Failure Interval Time Data

Failure number	Failure interval(second)	Failure interval(second)	Failure transformation interval(second)	Failure number	Failure interval(second)	Failure interval(second)	Failure transformation interval(second)
1	475	479	0.00429	44	2441	2441	0.31336
2	266	745	0.00745	45	4860	32846	0.32346
3	277	1022	0.01022	46	565	32911	0.32911
4	354	1576	0.01576	47	1119	34030	0.34030
5	1034	2610	0.02610	48	437	34467	0.34467
6	249	2859	0.02859	49	927	35394	0.35394
7	693	3552	0.03552	50	4462	39856	0.39856
8	597	4149	0.04149	51	714	40570	0.40570
9	117	4266	0.04266	52	181	40751	0.40751
10	170	4436	0.04436	53	1485	42236	0.42236
11	117	4553	0.04553	54	757	42993	0.42993
12	1274	5827	0.05827	55	3154	46147	0.46147
13	469	6296	0.06296	56	2115	48262	0.48262
14	1174	7470	0.07470	57	884	49146	0.49146
15	693	8163	0.08163	58	2037	51183	0.51183
16	1908	10071	0.10071	59	1481	52664	0.52664
17	135	10206	0.10206	60	559	53223	0.53223
18	277	10483	0.10483	61	490	53713	0.53713
19	596	11079	0.11079	62	593	54506	0.54506
20	757	11836	0.11836	63	1789	56075	0.56075
21	437	12273	0.12273	64	85	56160	0.56160
22	2230	14903	0.14903	65	2836	58996	0.58996
23	437	14949	0.14949	66	213	59299	0.59299
24	340	15280	0.15280	67	1866	61075	0.61075
25	405	15683	0.15683	68	490	61565	0.61565
26	335	16220	0.16220	69	1487	63052	0.63052
27	277	16497	0.16497	70	4322	67374	0.67374
28	363	16860	0.16860	71	1418	68792	0.68792
29	522	17382	0.17382	72	1023	69815	0.69815
30	613	17993	0.17993	73	5490	75305	0.75305
31	277	18272	0.18272	74	1520	76125	0.76125
32	1300	19572	0.19572	75	3281	80106	0.80106
33	821	20393	0.20393	76	2716	82822	0.82822
34	213	20606	0.20606	77	2175	84997	0.84997
35	1620	22226	0.22226	78	3505	88502	0.88502
36	1801	23827	0.23827	79	725	89227	0.89227
37	298	24125	0.24125	80	1963	91190	0.91190
38	374	24999	0.24999	81	3979	95169	0.95169
39	618	25617	0.25617	82	1090	96259	0.96259
40	2840	28127	0.28127	83	245	96504	0.96504
41	5	28262	0.28262	84	1194	97698	0.97698
42	149	28411	0.28411	85	994	98692	0.98692
43	1034	29443	0.29443	86	3902	102594	1.02594

표 2. 각 모형의 모수 추정값
Table 2. Estimator of each model

Model	MLE	
exponential coverage	$\hat{g}_{MLE}=0.08215$	$\hat{a}_{MLE}=84.87$
S-coverage	$\hat{g}_{MLE}=0.82564$	$\hat{a}_{MLE}=85.43$
Kappa(2) coverage	$\hat{\beta}_{MLE}=3.79872$	$\hat{a}_{MLE}=89.78$

표 3. 모형들에 대한 SSE의 값
Table 3. SSE of each model

Model	exponential coverage	S-coverage	Kappa(2) coverage
SSE	7926.23	6057.38	5682.12

VI. 결 론

<그림 3> 은 각 모형에 대하여 위험함수를 그린 그림이다. 이 그림에서는 지수 커버리지 모형은 일정하고 S-커버리지 모형은 증가형태, Kappa(2) 커버리지 모형은 감소형태를 나타내고 있으며 <그림 4>에서는 커버리지 함수를 그린 그림이다. ENHPP 모형에서 가정한 대로 연속적인 단조 비감소 함수형태로 나타나고 있고 Kappa(2) 형태가 제일 높은 확률을 보이고 있다.

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[13].

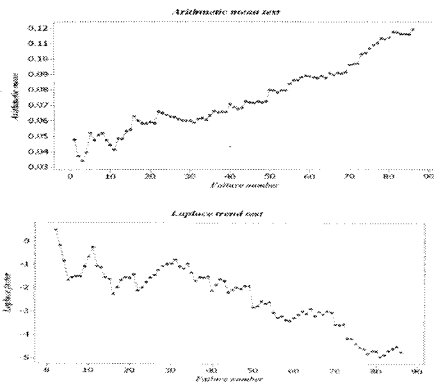
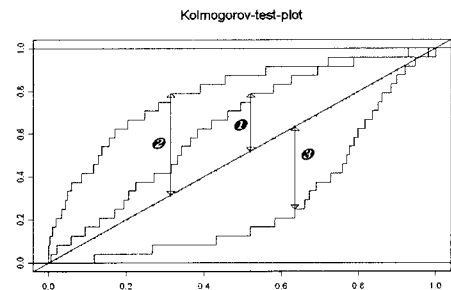


그림 1. 산술평균 검정과 라플라스 추세 검정
Fig. 1. Arithmetic mean and Laplace trend test



단. ① Kappa(2) coverage Model ② S-coverage Model
③ exponential coverage Model

그림 2. 모형에 대한 Kolmogorov 검정
Fig. 2. Kolmogorov test of each model

표 4. 모형들에 대한 Kolmogorov 거리
Table. 4. Kolmogorov distance of each model

Model	Kolmogorov distance
exponential coverage Model	0.159593
S-coverage Model	0.155121
Kappa(2) coverage Model	0.132452

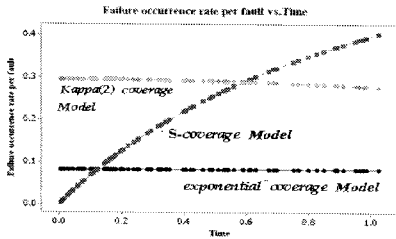


그림 3. 각 모형에 대한 위험함수
Fig. 3. Hazard function of each model

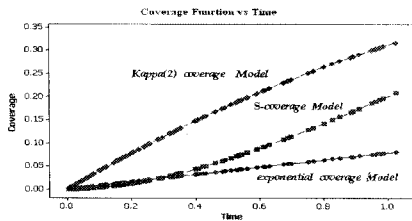


그림 4. 각 모형에 대한 커버리지함수
Fig. 4. Coverage function of each model

본 논문에서는 기존에 존재하는 NHPP 모형에서 테스트 커버리지를 포함하는 모형인 ENHPP 모형에 대하여 연구하였다. 잔존 결함 1개당 고장 발생률이 감소추세를 가진 Kappa(2) 분포를 이용한 카파 커버리지모형을 제안하였다. 고장 간격시간으로 구성된 자료를 이용하여 기존의 모형과 카파 모형에 대하여 최우 추정법을 이용하여 모수 추정을 실시하였다. 소프트웨어 고장분석 자료는 고장수가 비교적 큰 실측 자료(고장수가 86)인 Allen P. Nikora 와 Michael R. Lyu 가 인용한 SYS2 자료를 통하여 분석하였다. 효율적인 모형 비교를 위한 편차자승합의 결과는 Kappa 커버리지모형이 기존에 잘 알려진 지수 커버리지모형이나 S-커버리지 모형보다 우수함을 보이고 있고 콜모고로프 거리의 결과도 Kappa 커버리지 모형이 상대적으로 정상성에 가까운 모형으로 간주할 수 있다. 분석된 자료에 대한 평가에서도 산술평균 검정과 라플라스 추세 검정을 실시한 결과도 신뢰성장이 되

고 있음을 나타내고 있다.

따라서 Kappa 분포를 이용한 ENHPP 모형도 이 분야에서 효율적으로 이용할 수 있는 모형이 됨을 알 수 있었다. 향후 이러한 Kappa 분포를 이용한 베이지안적 접근 방법이나 수리적인 추정과 검정 부분에 대한 수리적인 접근이 기대 된다.

참고문헌

- [1] 김희철, 최유순, 박 종구, "어랑 분포를 이용한 NHPP 소프트웨어 신뢰성장 모형에 관한 연구", 한국 해양정보통신학회 논문지, 10권1호. pp.7-14, 2006
- [2] J. F. Lawless, "Statistical Models and Methods for Lifetime Data", John Wiley & Sons, New York, 1981.
- [3] S. S. Gokhale and K. S. Trivedi. "A time/structure based software reliability model". Annals of Software Engineering, 8, pp. 85-121. 1999
- [4] S. S. Gokhale and T. Philip and P. N. Marinos and k. S. Trivedi, "Unification of Finite Failure Non-Homogeneous Poisson Process Models through Test Coverage ", In Proc. of Intl. Symposium on software Reliability Engineering," White Plains. NY, 1996.
- [5] J. R. M. Hosking, "The Four-parameter Kappa distribution". IBM J. RES, DEVELOP, Vol. 38, No 3, May, 1994.
- [6] M. I. Mielke, "Another Family Distribution for Describing and Analying Precipitation Data", J. Appl. Meteorol 12, pp. 275-280, 1973.
- [7] J.R. Horgan, S. London, and M.R. Lyu, " Achieving Software Quality with Testing Coverage Measure," IEEE Computer, pp. 60-69, Sept. 1994.
- [8] R. Jacoby and K. Masuzawa, "Test Coverage De-pendant Software Reliability Estimation by the HGD Model," 3rd Intl. Symposium on Software Reliability Engineering, 1992.
- [9] H. Pham and L. Nordmann and X. Zhang, "A General Imperfect-Software -Debugging Model with S-Shaped Fault-Detection Rate", IEEE Trans. on reliability, Vol, 48, No 2, pp, 169-175, 1999.

- [10] A. L. Goel and K. Okumoto, "Time-Dependent Error-Detection Rate Models for Software Reliability and Other Performance Measures". IEEE Trans. on Reliability, R-28(3):pp. 206-211, Aug. 1979.
- [11] S. Yamada, M. Ohba and S. Osaki. "S-Shaped Reliability Growth Modeling for Software Error Detection". IEEE Trans. on Reliability. R-32(5): pp. 475-485, Dec. 1983.
- [12] A. P. Nikora and M. R. Lyu., "*Handbook of Software Reliability Engineering*, M.R. Lyu, Editor, chapter *Software Reliability Measurement Experience*, pp. 255-301. MacGraw-Hill, New York, 1996.
- [13] S. Selvin, "Modern Applied Biostatistical Methods Using S-Plus", pages 141-184. Oxford University Press, New York, 1998.

저자소개

김 희 철(Hee-Cheul Kim)



1992년 동국대학교 통계학과 석사
1998년 동국대학교 통계학과 박사
2000년 3월 ~ 2004년 2월

송호대학 정보산업계열 조교수
2005년 3월~현재 남서울대학교 산업경영공학과 전임
강사

※관심분야 : 소프트 웨어 신뢰성공학, 웹프로그래밍,
전산통계, 인터넷비즈니스