

NHPP 소프트웨어 신뢰도 모형에 대한 모수 추정 비교

김 희 철* · 이 상 식** · 송 영 재***

요 약

본 논문에서는 기존의 소프트웨어 신뢰성 모형인 Goel-Okumoto 모형과 Yamada-Ohba-Osaki 모형을 재조명하고 또, 란리 분포를 이용한 란리 모형을 적용하여 모수 추정방법을 연구하였다. 본 연구에서는 기존의 최우추정법과 잠재변수를 도입하여 깁스 샘플링(Gibbs sampling)을 이용한 베이저안 모수추정 방법을 비교하고 그 특징을 분석하고자 한다. 또, 효율적 모형을 위한 모형선택으로서 잔차제곱합(Sum of the squared errors ; SSE)과 Braun 통계량을 적용하여 모형들에 대한 효율성 입증방법을 설명하였다. 그리고 수치적인 예로서 실제 자료를 이용한 수치 결과를 나열하였다. 이 접근방법을 기초로 하여 수명분포가 중첩(Superposition) 및 혼합(Mixture)인 경우에 대한 접근방법이 연구되었으면 한다.

The Comparison of Parameter Estimation for Nonhomogeneous Poisson Process Software Reliability Model

Hee-Cheul Kim[†] · Sang-Sik Lee^{**} · Young-jae Song^{***}

ABSTRACT

The Parameter Estimation for software existing reliability models, Goel-Okumoto, Yamada-Ohba-Osaki model was reviewed and Rayleigh model based on Rayleigh distribution was studied. In this paper, we discuss comparison of parameter estimation using maximum likelihood estimator and Bayesian estimation based on Gibbs sampling to analysis of the estimator's pattern. Model selection based on sum of the squared errors and Braun statistic, for the sake of efficient model, was employed. A numerical example was illustrated using real data. The current areas and models of Superposition, mixture for future development are also employed.

키워드 : 소프트웨어 신뢰도 모형(Software Reliability Model), 깁스샘플링(Gibbs Sampling), 비동질적인 포아송과정(Nonhomogeneous Poisson Process), 고장 발생률(ROCOF), Goel-Okumoto 모형, Yamada-Ohba-Osaki 모형, 메트 로폴리스알고리즘(Metropolish Algorithm), 소프트웨어 신뢰도(Software Reliability), 잠재변수(Latent Variable), 잔차 제곱합(Sum of the Squared Errors), Braun 통계량

1. 서 론

신뢰도 이론은 소프트웨어 시스템 전체가 규정된 환경조건하에서 의도하는 기간동안에 요구된 기능을 만족스럽게 수행할 수 있는 확률을 예측하고 증대시키기 위한 실제적인 도구가 된다. 제품이 의도된 기간 동안 고장 없이 제 기능을 발휘할 수 있는 확률을 보다 정확한 방법으로 추정하는 연구는 이 분야의 관심사항이 된다.

신뢰도의 정량적인 값을 얻는데 있어 기존의 방법은 최우 추정법(maximum likelihood estimator ; MLE)을 많이 사용하였다. 그러나 새로운 자료가 얻어지면 그 자료를 이제까지 얻었던 자료와 결합시켜 새로운 결론에 도달하려는 이론이 베이저 추정법이다. 즉, 베이저 이론은 알려져 있는 사실에 대한 주관적 의견을 경험이나 지식을 바탕으로 하여 사전정

보를 만든 다음 실험을 통하여 얻어진 자료와 결합시켜 사후정보를 추출하는 과정이다. 그러나 베이저 추정법에서 사전확률분포인 수명분포가 복잡하면 적분이 불가능해지기 때문에 사후정보의 추출이 불가능해진다.

따라서 본 연구에서는 최우추정법과 적분이 난해한 경우에 깁스 샘플링(Gibbs sampling)[1, 3, 5]을 이용하여 근사적 깁스추정량을 유도하여 비교하고 그 특징을 분석하고자 한다.

우리가 가능한 시스템이나 소프트웨어 신뢰도에서 관측시간 $(0, t]$ 사이에 발견된 고장수 $N(t)$ 를 모형화 하는데 비동질적 포아송 과정(Non-homogeneous Poisson process)이 널리 사용되어 왔다. 강도함수(Intensity function) 혹은 고장 발생률(Rate of occurrence of failure) $\lambda(t) = dE[N(t)]/dt$ 은 t 에 대한 단조(Monotonic)함수로 흔히 가정한다. 예를 들어 동질적 포아송 과정(HPP ; Homogeneous poisson process), Musa-Okumoto과정, Weibull 과정 그리고 Cox-Lewis 과정은 각각 ROCOF를 상수, 시간에 관한 부분(Fraction) 함수, 멱(Power) 함수, 대수선형(Log-linear) 함수를 가정하

[†] 정 회 원 : 한라대학교 정보통신학부 겸임교수

^{**} 정 회 원 : 송호대학 정보산업계열 교수

^{***} 중신회원 : 경희대학교 컴퓨터공학과 교수

논문접수 : 2004년 2월 6일, 심사완료 : 2004년 7월 7일

였다[10].

본 논문에서는 기존의 소프트웨어 신뢰성 모형인 Goel-Okumoto 모형[7]과 Yamada-Ohba-Osaki 모형[15]을 재조명하고 또, 랄리 분포(Rayleigh distribution)[10]를 이용한 랄리 모형을 적용하여 모수 추정방법을 연구하였다. 또, 효율적 모형을 위한 모형선택으로서 잔차제곱합(SSE)과 Braun 통계량을 적용하여 모형들에 대한 효율성 입증방법을 설명하고 비교하였다. 모수의 추정을 위해서 최우추정법과 관련된 관계식을 풀기 위하여 일반적 수치해석적 방법인 이분법을 이용하고 사전분포가 필요한 베이지안적 접근 방법에서는 잠재변수를 도입한 깁스 샘플링(Gibbs sampling)을 이용하였다.

본 논문의 구성의 2장에서는 NHPP 모형과 소프트웨어 신뢰도 모형에 대한 최우추정법과 깁스 표본 추출법을 이용한 모수 추정에 대하여 서술 하였고 3, 4장에서는 각각 모형 선택의 방법과 실측자료인 Military software system[12] CPU 고장 자료를 이용하여 각 모형에 대한 모수 추정 및 모형 비교에 대해서 설명하였다. 마지막으로 5장에서는 결론 및 향후 연구방향을 제시하였다.

2. 최우추정법과 깁스 표본추출법을 이용한 모수 추정

시간 $(0, t]$ 까지 조사하기 위한 시간 절단(Time truncated) 모형은 n 번째 까지 고장시점 자료를 $x_k(k=1, 2, \dots, n; 0 \leq x_1 \leq x_2 \leq \dots \leq x_n)$ 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[6, 10, 12].

$$L_{NHPP}(\theta, \beta | D_t) = \left[\prod_{i=1}^n \lambda(x_i) \right] \exp(-m(t)) \quad (2.1)$$

단, $m(t) = \int_0^t \lambda(t) dt$.

우도함수 식 (2.1)에서 t 을 x_n 으로 대치하면 유사한 형태의 고장 절단 모형의 우도함수가 된다[8].

본 연구의 NHPP모형에서는 강도함수 $\lambda(t) = m'(t)$ 을 만족하고 각 모형에서 모수인 θ 는 테스트 개시 전에 소프트웨어 내에 잠재하는 총 기대 고장 수를 나타내고, β 은 각 모형에 대한 잔존고장 1개당 고장 발견율을 의미한다[6].

따라서 식 (2.1)에서 최우추정법을 사용하여 NHPP모형들에 대한 모수 추정을 할 수 있고, 베이지안 모수추정을 위하여 깁스 표본 추출법을 이용할 수 있다.

그러나, 식 (2.1)의 우도함수에서 $\exp[-m(t)] = \exp[-F(t|\beta)]$ 의 형태 때문에 깁스 표본추출을 적용하는데 필요

한 조건부분포의 형태를 구체화하기 어렵다. 그러므로 모수 $\theta[1-F(t|\beta)]$ 을 가지는 포아송 분포를 따르는 잠재변수(Latent variable) $N' = N - n$ 을 이용한다. 이 경우 사후분포 $p(\theta, \beta | D_t)$ 는 결합밀도 $p(\theta, \beta, N' | D_t)$ 을 주변밀도화 함으로서 얻을 수 있다. 이 밀도는 다음과 같은 구체화가 가능한 조건부밀도의 반복을 이용하여 깁스 표본추출을 통하여 근사밀도를 구할 수 있다[8, 9].

$$\text{즉, } p(N' | \theta, \beta, D_t), p(\theta | N', \beta, D_t), p(\beta | N', \theta, D_t).$$

본 논문에서는 $X \perp Y$ 는 X 와 Y 가 독립, $N' = N - n$ 을 의미하고, $\Gamma(a, b)$ 는 평균이 a/b 인 감마 분포를, $P(\theta)$ 는 평균이 θ 인 포아송 분포를 나타내고, β 은 미지의 모수를 의미한다.

본 절에서는 다음과 같은 몇 개의 분포를 적용하여 깁스 추정량과 기존의 최우추정량을 구하고자 한다.

2.1 Goel-Okumoto 모형

이 모형에 대한 평균값 함수와 강도함수는 각각 다음과 같이 알려져 있다[7, 8].

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

위 평균값 함수와 강도함수를 식 (2.1)의 우도함수에 대입한 고장 절단 모형의 우도함수는 다음과 같이 나타낼 수 있고 본 논문에서 수식의 구별을 위하여 데이터 x_n 을 s_n 으로 대치 되었다.

$$L_{NHPP}(\theta, \beta | D_{s_n}) = \left(\prod_{i=1}^n \theta \beta e^{-\beta x_i} \right) \exp[-\theta(1 - e^{-\beta s_n})]. \quad (2.2)$$

2.1.1 최우추정법에 의한 모수 계산

식 (2.2)의 우도함수에 로그를 취하여 각 모수에 대한 편미분을 시행하면 다음과 같이 최우추정치를 계산할 수 있다[7].

$$\frac{\partial \ln L_{NHPP}(\theta, \beta | D_{s_n})}{\partial \theta} = \frac{n}{\theta} - 1 + e^{-\beta s_n} = 0 \quad (2.3)$$

$$\frac{\partial \ln L_{NHPP}(\theta, \beta | D_{s_n})}{\partial \beta} = \frac{n}{\beta} - \sum_{i=1}^n x_i - \theta s_n e^{-\beta s_n} = 0 \quad (2.4)$$

따라서 식 (2.3)와 식 (2.4)의 관계식을 이용한 계산을 수치해석적인 반복법의 하나인 이분법(Bisection method)을 이용하여 모수의 최우추정치 $\hat{\theta}_{MLE}$ 와 $\hat{\beta}_{MLE}$ 를 계산할 수 있다.

2.1.2 베이지안적 모수 추정

베이즈 추정법을 시행하기 위하여 사전분포는 $\theta \sim \Gamma(a, b)$, $a, b > 0$, $\beta \sim \Gamma(c, d)$, $c, d > 0$, $N' \sim P(\theta e^{-\beta s_n})$, $\theta \perp \beta \perp N'$ (독립)이라고 하면 사후결합밀도는 사전분포와 우도함수 식 (2.1)를 이용하면 다음과 같은 식이 된다(베이즈 정리 이용)[8].

$$\begin{aligned}
 p(\theta, \beta, N' | D_{s_n}) &\propto L_{NHPP}(\theta, \beta | D_{s_n}) \cdot p(\theta) \cdot p(\beta) \cdot p(N') \\
 &= \frac{(\theta e^{-\beta s_n})^{N'}}{N'!} \cdot e^{-\theta e^{-\beta s_n}} \cdot \theta^{n+a-1} \cdot \beta^{c+n-1} \cdot e^{-b\theta} \cdot e^{-(d+\sum_{i=1}^n x_i)\beta} \\
 &\quad \exp[-\theta(1 - e^{-\beta s_n})] \\
 &= \frac{1}{N'!} \theta^{N'+n+a-1} \cdot \beta^{c+n-1} \cdot e^{-(b+1)\theta} e^{-(d+s_n N' + \sum_{i=1}^n x_i)\beta}
 \end{aligned}$$

이식을 장애(Nuisance)모수 개념을 이용하여 깁스 샘플링 알고리즘에 사용되는 각 모수에 대한 완전 사후조건부밀도(Full posterior conditional densities)는 다음과 같다.

$$N' | \theta, \beta, D_{s_n} \sim P(\theta e^{-\beta s_n}); \tag{2.5}$$

$$\theta | N', \beta, D_{s_n} \sim \Gamma(a+n+N', b+1); \tag{2.6}$$

$$\beta | N', \theta, D_{s_n} \sim \Gamma(c+n, d+s_n N' + \sum_{i=1}^n x_i); \tag{2.7}$$

위 조건부 밀도를 이용한 깁스 알고리즘[1, 3, 5]은 다음과 같은 단계를 이용하여 시행한다.

(0단계)

$\Gamma(a, b)$ 와 $\Gamma(c, d)$ 의 분포에서 데이터를 랜덤추출하여 초기 값을 각각 $\theta^{(0)}, \beta^{(0)}$ 을 정한다. 단, a, b, c, d 는 0보다 큰 임의의 알려진 상수를 의미한다.

(1-1단계)

$\theta = \theta^{(0)}, \beta = \beta^{(0)}$ 로 고정시켰을 경우 식 (2.5)에 대입하여 생성된 랜덤포본 하나를 $N^{(1)}$ 이라 한다. 즉,

$$N^{(1)} \sim P(\theta^{(0)} e^{-\beta^{(0)} s_n}).$$

위와 동일한 방법으로(1-1단계)를 이용하여 다음 (1-2단계), (1-3단계)를 얻는다.

(1-2단계)

$$\theta^{(1)} \sim \Gamma(a+n+N^{(1)}, b+1).$$

(1-3단계)

$$\beta^{(1)} \sim \Gamma(c+n, d+s_n N^{(1)} + \sum_{i=1}^n x_i).$$

(2단계)

(1-1단계), (1-2단계), (1-3단계)로부터 고정시킨 θ, β, N' 의 값을 가장 최근에 생성된 랜덤포본의 값으로 대체 하면서 (1-1단계), (1-2단계), (1-3단계)를 충분히 큰 수(본 논문에서는 70번) 만큼 반복 수행한다. 이렇게 하여 얻은 최종표본을 $(\theta_1^{(70)}, \beta_1^{(70)}, N_1^{(70)})$ 이라 한다.

(3단계)

(1-1단계), (1-2단계), (1-3단계)를 다시 (2000-1)번 반복 수

행하면 총 2000개의 랜덤 표본

$$\begin{aligned}
 &(\theta_1^{(70)}, \beta_1^{(70)}, N_1^{(70)}), (\theta_2^{(70)}, \beta_2^{(70)}, N_2^{(70)}), \\
 &\dots, (\theta_{2000}^{(70)}, \beta_{2000}^{(70)}, N_{2000}^{(70)})
 \end{aligned}$$

이 얻어진다.

(4단계)

최종적인 결과에 의해 θ, β, N' 의 추정은 다음과 같다.

$$\begin{aligned}
 \hat{\theta}_{Gibbs} &= \frac{1}{2000} \sum_{k=1}^{2000} \theta_k^{(70)}, \quad \hat{\beta}_{Gibbs} = \frac{1}{2000} \sum_{k=1}^{2000} \beta_k^{(70)}, \\
 \hat{N}'_{Gibbs} &= \frac{1}{2000} \sum_{k=1}^{2000} N_k^{(70)}
 \end{aligned}$$

그리고 테스트 시점 t 에서 소프트웨어 고장이 일어난다고 하는 가정하에서 신뢰구간 $(t, t+x]$ (단, x 는 임무시간(Mission time) 사이에 소프트웨어의 에러가 일어나지 않을 소프트웨어 신뢰도(Software reliability)는 다음과 같이 알려져 있다[7, 8].

$$\begin{aligned}
 \hat{R}(x|t) &= \exp[-\{m(t+x) - m(t)\}] \\
 &= \exp[-\hat{\theta}\{\exp(-\hat{\beta}t) - \exp(-\hat{\beta}(t+x))\}] \\
 &\quad (t \geq 0, x \geq 0)
 \end{aligned} \tag{2.8}$$

2.2 Yamada-Ohba-Osaki 모형

이 모형에 대한 평균값 함수는 $m(t) = \theta[1 - (1 + \beta t)e^{-\beta t}]$ ($\theta > 0, \beta > 0$)이라고 알려져 있고 강도함수 $\lambda(t) = m'(t)$ 의 관계식에 의해 $\lambda(t) = \theta \beta^2 t e^{-\beta t}$ 가 된다[15]. 따라서 고장 절단 모형의 우도함수는 식 (2.1)과 관련하면 다음과 같다.

$$\begin{aligned}
 L_{NHPP}(\theta, \beta | D_{s_n}) &= \left(\prod_{i=1}^n \theta \beta^2 x_i \exp^{-\beta x_i} \right) \\
 &\quad \exp[-\theta\{1 - e^{-\beta s_n}(1 + \beta s_n)\}]
 \end{aligned} \tag{2.9}$$

2.2.1 최우추정법에 의한 모수 계산

앞의 방법과 유사하게 이 우도함수에 로그를 취하여 각 모수에 대한 편미분을 시행하면 다음과 같이 최우추정치를 계산할 수 있다.

$$\begin{aligned}
 \frac{\partial \ln L_{NHPP}(\theta, \beta | D_{s_n})}{\partial \theta} &= \frac{n}{\theta} - 1 + e^{-\beta s_n}(1 + \beta s_n) = 0 \tag{2.10} \\
 \frac{\partial \ln L_{NHPP}(\theta, \beta | D_{s_n})}{\partial \beta} &= \frac{2n}{\beta} - \sum_{i=1}^n x_i - \theta e^{-\beta s_n}(\beta s_n^2) = 0
 \end{aligned} \tag{2.11}$$

따라서 식 (2.10)와 식 (2.11)의 관계식을 이용한 계산을 수치해석적인 반복법의 하나인 이분법(Bisection method)을 이용하여 모수의 최우추정치 $\hat{\theta}_{MLE}$ 와 $\hat{\beta}_{MLE}$ 를 계산할 수 있다.

2.2.2 베이저안적 모수 추정

김스 표본 추출 알고리즘은 Goel-Okumoto 모형과 유사한 방법으로 다음과 같은 조건부 밀도를 이용하여 시행한다.

사전분포는 $\theta \sim \Gamma(a, b), \beta \sim \pi_1(\beta), N' \sim P[\theta e^{-\beta s_n}(1 + \beta s_n)]$. 단, π_1 은 $\beta > 0$ 을 만족하는 임의의 사전분포이고 $\theta \perp \beta \perp N'$ 을 만족한다.

$$N' | \theta, \beta, D_{s_n} \sim P[\theta e^{-\beta s_n}(1 + \beta s_n)]; \tag{2.12}$$

$$\theta | N', \beta, D_{s_n} \sim \Gamma(a + n + N', b + 1); \tag{2.13}$$

$$\beta | N', \theta, D_{s_n} \propto \beta^{2n} e^{-\beta \sum x_i} [e^{-\beta s_n}(1 + \beta s_n)]^{N'} \cdot \pi_1(\beta); \tag{2.14}$$

앞의 Goel-Okumoto 모형의 알고리즘과 유사하게 적용되지만 식 (2.14)는 따르는 분포가 구체화 되지 않기 때문에 앞 절의 (1-3단계)에서 메트로폴리스 기법[2, 8, 9]을 적용해야 한다.

즉,

$$\beta | N', \theta, D_{s_n} \propto \beta^{2n} e^{-\beta \sum x_i} [e^{-\beta s_n}(1 + \beta s_n)]^{N'} \cdot \pi_1(\beta).$$

위 식에서 오른쪽 식 즉, 목적분포를 간편하게 $f(\beta)$ 라고 표기하고, N' 는 앞 단계에서 김스추출법으로 추정된 값이고 $\pi_1(\beta) \propto \frac{1}{\beta} (> 0)$ 라고 가정하였고 추이커널(Transitional kernel)은 거의 대칭을 이루는 $\Gamma(a, b)$ 에서 β' 을 발생하고 균등분포 (0, 1)에서 확률변량을 w 라고 하면

$$\log w \leq \log f(\beta') - \log f(\beta)$$

을 만족하면 β' 를 β 로 간주되고 만족되지 않으면 β 를 β' 로 대체되면서 충분한 반복을 시행한다. 따라서 β 는 Metropolis 알고리즘에 의하여 발생되어진다.

앞의 모형의 개념과 유사하게 소프트웨어 신뢰도는 다음과 같이 알려져 있다[4, 11].

$$\begin{aligned} \widehat{R}(x|t) &= \exp[-\{m(t+x) - m(t)\}] \\ &= \exp[-\{\theta\{(1 + \widehat{\beta}t)\exp(-\widehat{\beta}t) - (1 + \widehat{\beta}(t+x))\exp(-\widehat{\beta}(t+x))\}\}] \quad (t \geq 0, x \geq 0) \end{aligned} \tag{2.15}$$

2.3 랄리(Rayleigh) 모형

수명분포로 사용하는 랄리분포의 확률밀도함수와 분포함수를 이용하면 다음과 같은 모형을 제시할 수 있다[12].

$$\begin{aligned} m(t) &= \theta F(t) = \theta(1 - e^{-\beta t^2}), \\ \lambda(t) &= \theta f(t) = 2\theta\beta t \exp(-\beta t^2) \end{aligned}$$

위 평균값 함수와 강도함수를 식 (2.1)과 관련하면 고장 절단 모형의 우도함수는 다음과 같이 나타낼 수 있다.

$$\begin{aligned} L_{NHPP}(\theta, \beta | D_{s_n}) &= \left\{ \prod_{i=1}^n 2\theta\beta x_i \exp(-\beta x_i^2) \right\} \\ &\cdot \exp[-\theta(1 - e^{-\beta s_n^2})]. \end{aligned}$$

2.3.1 최우추정법에 의한 모수 계산

앞의 방법과 유사하게 이 우도함수에 로그를 취하여 각 모수에 대한 편미분을 시행하면 다음과 같이 최우추정치를 계산할 수 있다.

$$\frac{\partial \ln L_{NHPP}(\theta, \beta | D_{s_n})}{\partial \theta} = \frac{n}{\theta} - 1 + e^{-\beta s_n^2} = 0 \tag{2.16}$$

$$\frac{\partial \ln L_{NHPP}(\theta, \beta | D_{s_n})}{\partial \beta} = \frac{n}{\beta} - \sum_{i=1}^n x_i^2 - \theta s_n^2 e^{-\beta s_n^2} = 0 \tag{2.17}$$

그러나 식 (2.16)와 식 (2.17)의 관계식을 이용한 계산을 수치해석적인 반복법의 하나인 이분법(Bisection method)을 이용하여 모수의 최우추정치 $\widehat{\theta}_{MLE}$ 와 $\widehat{\beta}_{MLE}$ 를 계산할 수 있다.

따라서 최우추정법과 김스추정량은 앞 절의 경우와 유사한 방법으로 시행하여 그 결과를 얻을 수 있다.

2.3.2 베이저안적 모수 추정

사전분포를 $\theta \sim \Gamma(a, b), \beta \sim \Gamma(c, d), N' \sim P(\theta e^{-\beta s_n^2})$, $\theta \perp \beta \perp N'$ 라고 하면 김스추출법에 사용되는 조건부 밀도는 다음과 같다.

$$N' | \theta, \beta, D_{s_n} \sim P(\theta e^{-\beta s_n^2});$$

$$\theta | N', \beta, D_{s_n} \sim \Gamma(a + n + N', b + 1);$$

$$\beta | N', \theta, D_{s_n} \sim \Gamma(n + c, N' s_n^2 + d + \sum_{i=1}^n x_i^2);$$

앞의 모형의 개념과 유사하게 소프트웨어 신뢰도는 다음과 같이 적용할 수 있다.

$$\begin{aligned} \widehat{R}(x|t) &= \exp[-\{m(t+x) - m(t)\}] \\ &= \exp[-\{\theta\{e^{-\beta t^2} - e^{-\beta(t+x)^2}\}\}] \quad (t \geq 0, x \geq 0) \end{aligned} \tag{2.18}$$

3. 모형의 선택

효율적 모형의 비교를 위해 고장번호(i)는 구간 $(0, x_i]$ 에 관찰된 실제고장의 수 $n_i(x_i)$ 이고 $\widehat{m}(x_i)$ 는 구간 $(0, x_i]$ 에서 추정된 고장의 수를 나타내므로 실제 고장수와 추정된 고장수의 잔차제곱합을 계산해서 작은 모형이 효율적인 모형이라고 간주 할 수 있다. 즉, 다음과 같이 나타낼 수 있다[7].

$$C_{SSE} = \sum_{i=1}^n (n_i(x_i) - \widehat{m}(x_i))^2 \tag{3.1}$$

단, $n_i(x_i)$ 는 $(0, x_i]$ 사이에 관찰된 오류의 수이고 $\hat{m}(x_i)$ 는 평균값 함수의 추정값을 의미한다.

그리고 또 다른 비교 평가의 기준으로 Braun 통계량으로 평가할 수 있는 데 이 통계량은 다음과 같이 정의 된다[11].

$$\hat{E}[N(k)]_{Braun} = \frac{\sum_{k=1}^n (n_k - \hat{E}[N(k)])^2 x_k}{\sum_{k=1}^n (n_k - \bar{n})^2 x_k} \quad (3.2)$$

단, $x_k(k=1, 2, \dots, n)$ 은 고장 시간을 나타내고 n_k 는 $(0, x_k]$ 사이에 관찰된 오류의 수, $\hat{E}[N(k)]$ 는 추정된 오류수를 의미한다. Braun 통계량 $\hat{E}[N(k)]_{Braun}$ 의 값이 작으면 정확도(accuracy)가 높게 됨으로서 모수 추정이 잘 된 효율적인 모형이 된다.

4. 수치적인 예

고장 시간 x_i 에 대한 자료는 Military software system[12] CPU 고장 자료(25번 고장까지 사용, 단위 : 초)를 이용하여 모수를 추정하고 신뢰도를 구하고자 한다. 관측 자료는 최종고장시간이 25번 까지를 고려하면 $x_{25} = 488230(x_0 = 0)$ 이고 각 고장 간격시간에 대한 자료는 <표 1>에 요약하였고 본 연구는 이 자료를 이용하여 모수추정과 신뢰도, 모형선택을 시행하는 방법을 설명하고자 한다. 본 연구에서는 각 모형에 대한 모수 추정방법은 최우추정법을 사용하고 비선형근을 구하기 위한 반복기법은 일반적인 수치해석적 방법인 이분법(Bisection)으로 추정치를 구하였다. θ 와 β 의 근을 구하는 과정에서 IMSL[14] 소프트웨어를 사용하였으며 θ 의 초기치는 20과 30을 β 는 0.000001과 0.05를 주고 100번 반복을 수행한 결과를 제시하였다. 그리고 베이지안 모수 추정에 있어서 사전분포는 분산이 큰 $\beta \sim \Gamma(1, 10^{-5})$, $\theta \sim \Gamma(50, 2)$ 을 선택 이용하였고, 메트로폴리스 기법의 적용이 용이하도록

하기 위해 사전 분포 $\pi_1(\beta) = \frac{1}{\beta}$ (상수), $\beta > 0$ 을 가정하였고, 이러한 모수를 추정하기 위하여 2000번(S) 적용을 하고 70번 반복(R)하여 나타난 깃스 추정치(사후 평균)들이 <표 2.1> ~ <표 2.3>에 나타나 있고 또, 이 표에서 수렴성을 확인하기 위해서 500, 2,000번 적용의 결과도 나타내었다.

모형 선택에 있어서는 잔차제곱합(SSE)을 이용한 모형비교를 설명하기 위하여 <표 3>에서는 Goel-Okumoto모형에 대한 최우추정치 및 베이즈 추정치(깃스 추정치)와 잔차제곱값과 잔차제곱합을 계산하는 방법을 나타내었다.

유사한 방법으로 다른 모형에 대한 잔차제곱합을 계산한 결과는 <표 4>에 요약 되었다. 이 결과표에서 Military software system CPU 고장 자료를 적용한 결과 모든 모형이 최우추정법을 이용하여 모수를 추정한 결과 보다는 베이지안 추정법(깃스 추정법)을 이용한 결과가 잔차제곱합을 최

<표 1> Military software system CPU 고장 자료

고장번호 (i)	고장간격시간(CPU sec) ($x_i - x_{i-1}$)	(누적)고장시간 (x_i)
1	320	320
2	14,390	14,710
3	9,000	23,710
4	2,880	26,590
5	5,700	32,290
6	21,800	54,090
7	26,800	80,890
8	113,540	194,430
9	112,137	306,567
10	660	307,227
11	2,700	309,927
12	28,793	338,720
13	2,173	340,893
14	7,263	348,156
15	10,865	359,021
16	4,230	363,251
17	8,460	371,711
18	14,805	386,516
19	11,844	398,360
20	5,361	403,721
21	6,553	410,274
22	6,499	416,773
23	3,124	419,897
24	51,323	471,220
25	17,010	488,230

<표 2.1> Goel-Okumoto모형

S	R	$\hat{\theta}_{Gibbs}$	$\hat{\theta}_{MLE}$	$\hat{\beta}_{Gibbs}$	$\hat{\beta}_{MLE}$
500	50	25.01334	25.00881	0.00008824	0.0000207
500	70	25.01434		0.00008922	
2,000	50	25.01833		0.00008904	
2,000	70	25.01831		0.00008701	

<표 2.2> Yamadaz-Ohba-Osaki 모형

S	R	$\hat{\theta}_{Gibbs}$	$\hat{\theta}_{MLE}$	$\hat{\beta}_{Gibbs}$	$\hat{\beta}_{MLE}$
500	50	25.48748	25.7150	0.0000102	0.0000199
500	70	25.18952		0.0000110	
2,000	50	25.52896		0.0000108	
2,000	70	25.50198		0.0000103	

<표 2.3> 랄리분포모형

S	R	$\hat{\theta}_{Gibbs}$	$\hat{\theta}_{MLE}$	$\hat{\beta}_{Gibbs}$	$\hat{\beta}_{MLE}$
500	50	32.37362	32.4723	0.001491	0.001391
500	70	32.16149		0.001428	
2,000	50	32.84469		0.001437	
2,000	70	32.01269		0.001429	

<표 3> Goel-Okumoto모형에 대한 평균값함수의 최우 추정치, 베이지 추정값($\hat{m}(x_i)$)과 잔차제곱합(C_{SSE})

고장번호(i)	고장간격시간 [A]	(누적) 고장시간 [B]	실제 고장수 [C]	평균값 함수의 최우 추정값[D]	잔차제곱의 값 [E]	평균값 함수의 깃스추정값[F]	잔차 제곱의 값 [G]
1	320	320	1	0.16511084	0.697039909	0.137872507	0.743263815
2	14,390	14,710	2	6.564954033	20.83880532	5.622380104	13.12163762
3	9,000	23,710	3	9.699955046	44.88939762	8.429524039	29.47973129
4	2,880	26,590	4	10.58593546	43.37454582	9.240771251	27.4656833
5	5,700	32,290	5	12.19112904	51.71233681	10.7338081	32.87655537
6	21,800	54,090	6	16.84618353	117.6396973	15.27731746	86.06861921
7	26,800	80,890	7	20.32174355	177.4688512	18.98263807	143.5836151
8	113,540	194,430	8	24.5619236	274.2973134	24.4683109	271.205264
9	112,137	306,567	9	24.9649381	254.8792484	25.2602438	264.3955286
10	660	307,227	10	24.96553326	223.9671858	25.26178515	232.922086
11	2,700	309,927	11	24.96788511	195.1018143	25.26791352	203.5733563
12	28,793	338,720	12	24.98625577	168.642839	25.31840721	177.3799707
13	2,173	340,893	13	24.98724737	143.6940995	25.32130672	151.8145994
14	7,263	348,156	14	24.99025586	120.7857239	25.33025581	128.3746966
15	10,865	359,021	15	24.9939908	99.87985219	25.34173933	106.9515723
16	4,230	363,251	16	24.99523236	80.91420526	25.34566673	87.34148663
17	8,460	371,711	17	24.99741194	63.95859776	25.35272249	69.76797297
18	14,805	386,516	18	25.00041787	49.00585038	25.36287143	54.2118757
19	11,844	398,360	19	25.00224039	36.02688971	25.36933787	40.56846491
20	5,361	403,721	20	25.00292939	25.02930249	25.37186297	28.85691181
21	6,553	410,274	21	25.00367407	16.02940609	25.37465257	19.13758512
22	6,499	416,773	22	25.00431929	9.025934385	25.37712848	11.40499676
23	3,124	419,897	23	25.00459987	4.018420644	25.37822467	5.655952557
24	51,323	471,220	24	25.00734831	1.014750616	25.38992046	1.931878885
25	17,010	488,230	25	25.00777916	6.05154E-05	25.39202827	0.153686164
C_{SSE}					2222.892168		2188.986991

단, [A] = $x_i - x_{i-1}$, [B] = x_i , [C] = $n_i(x_i)$, [D] = $\hat{m}_{MLE}(x_i)$, [E] = $[n_i(x_i) - \hat{m}_{MLE}(x_i)]^2$, [F] = $\hat{m}_{Gibbs}(x_i)$, [G] = $[n_i(x_i) - \hat{m}_{Gibbs}(x_i)]^2$, $C_{SSE} = \sum_{i=1}^{25} (n_i(x_i) - \hat{m}(x_i))^2$.

<표 4> 잔차제곱합에 의한 모형비교

	추정법	Goel-Okumoto 모형	Yamada-Ohba-Osaki 모형	Rayleigh 모형
C_{SSE}	깃스추정법	2188.987	2072.653	2173.1290
	최우추정법	2222.892	2165.873	2298.972

<표 5> Braun 통계량에 의한 모수 추정 비교

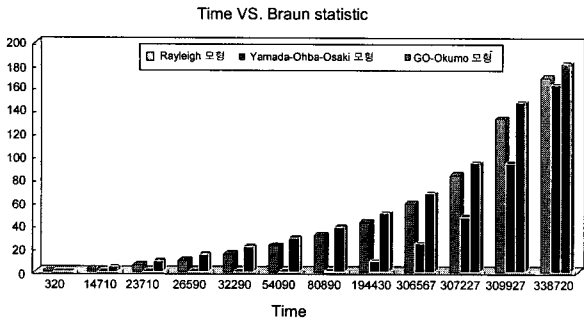
	Goel-Okumoto 모형	Yamada-Ohba-Osaki 모형	Rayleigh 모형
$E[N(k)]_{Braun}$	1563504.097	1388186.874	1444863.9790

소로 하기 때문에 효율적 모형으로 간주할 수 있고 모든 추정법에서도 Yamada-Ohba-Osaki 모형이 효율적 모형으로 나타나고 있다(<표 4> 참조). 각 모형에 대한 베이지안 추정법을 이용한 모수 추정 결과를 이용한 Braun 통계량의 결과는 <표 5>에 나타내었다. 이 표에서도 다른 모형에 비해서

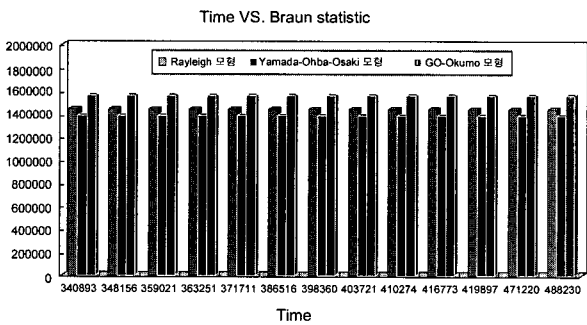
Yamada-Ohba-Osaki 모형이 Braun 통계량값이 다른 모형에 비해 작으므로 모수 추정이 잘된 모형으로 나타나고 있다. 그리고 각 고장시점에서의 Braun 통계량을 전반부(12번 고장까지)와 후반부(13번 고장부터 25번까지)의 비교 결과는 각각 (그림 1), (그림 2)에 나타내었다. 이 그림에서도 전 고장시점에서의 Braun 통계량의 결과값이 Yamada-Ohba-Osaki 모형이 제일 낮게 나타나고 특히 전반부 보다 후반부가 보다 더 큰 차이를 보이고 있다.

(그림 3)에서는 각 모형에 대한 베이지안 추정법(깃스 추정법)을 이용한 결과를 가지고 예측 고장수를 도표로 그린 결과에서도 다른 모형에 비해 Yamada-Ohba-Osaki 모형이 실제 고장수에 가장 근접한 형태로 나타나고 있다.

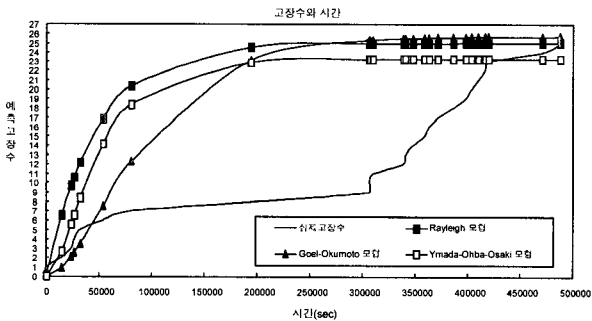
(그림 4)은 신뢰도 함수를 이용하여 미래의 신뢰도 즉, 실제 고장수가 25이후의 신뢰도($\hat{R}(x|_{x_{25}} = 488, 230)$)를 계산하여 비교하고자 한다. 이 그림에서 시간이 지날수록 모형들이 감소추세를 보이고 있고 다른 모형보다 Yamada-Ohba-Osaki 모형이 비교적 높음을 보이고 있다.



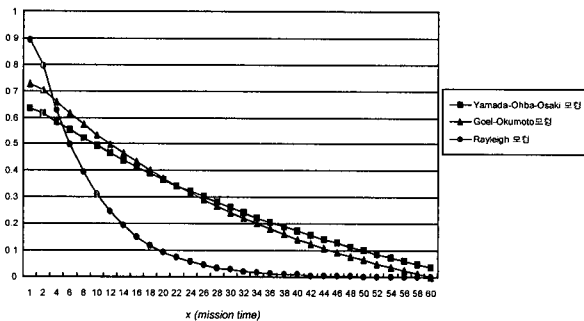
(그림 1) Braun 통계량에 의한 전반부(12번 고장까지) 모수 추정 비교



(그림 2) Braun 통계량에 의한 후반부(13번 고장부터 25번까지) 모수 추정 비교



(그림 3) 각 모형에 대한 예측 고장수 비교



(그림 4) 각 모형에 대한 신뢰도

5. 결론 및 향후 연구방향

소프트웨어의 신뢰성은 개발의 최종단계에 있는 테스트

공정이나 실제 사용 단계에 있어서 소프트웨어 내에 존재하는 고장 수나 소프트웨어의 고장발생시간에 의해 효과적 평가를 할 수 있는 것으로 그 평가 기술이 중요하게 된다. 소프트웨어 개발의 테스트 공정이나 실제 사용단계에 있어서 고장 발생상황이나 소프트웨어 고장 발생현상을 수리적 모형화가 가능하다면 평가를 할 수 있다.

본 연구에서는 비동질 포아송 과정(NHPP)에 기초한 소프트웨어 고장 현상에 대한 확률 모형을 고려하였고 고장 패턴은 NHPP에 대한 강도함수와 평균값 함수로서 나타낼 수 있다. 본 연구에서는 기존의 소프트웨어 신뢰성 모형인 Goel-Okumoto 모형과 Yamada-Ohba-Osaki 모형을 재조명하고 또, 칼리 분포를 이용한 칼리 모형을 적용하여 모수 추정방법을 연구하였다. 본 연구에서는 기존의 최우추정법과 잠재변수를 도입하여 깁스 샘플링(Gibbs sampling)을 이용한 베이저안 모수추정 방법을 비교 설명하였고 추정하고자 하는 모수가 같을 때 효율적 모형을 위한 모형선택으로서 잔차제곱합(Sum of the squared errors ; SSE)을 적용하여 모형들에 대한 효율성 입증방법을 설명하였다.

모수의 추정을 위해서 최우추정법(MLE)과 일반적인 수치해석적 방법인 이분법을 이용하였다. 수치적인 예에서는 실측자료인 Military software system CPU 고장 자료를 이용하여 모수 및 신뢰도를 추정하는 방법을 설명 하였고 또, 잔차제곱합을 이용한 모형비교의 결과를 나열하였다.

잔차제곱합이 다른 모형에 비해 작으므로 Yamada-Ohba-Osaki 모형보다 효율적인 모형으로 간주 할 수 있다. 각 모형에 대한 신뢰도 추정을 위하여 n 이 25이후의 신뢰도를 계산하여 비교한 결과도 Yamada-Ohba-Osaki 모형이 다른 모형에 비해 신뢰 성장이 됨을 알 수 있었다.

베이저안 모수 추정 결과를 이용한 Braun 통계량의 결과에 있어서도 다른 모형에 비해서 Yamada-Ohba-Osaki 모형이 Braun 통계량값이 다른 모형에 비해 작으므로 모수 추정이 잘된 모형으로 나타나고 있다.

따라서 본 논문은 고장간격시간이 주어진 자료에 대한 베이저안 모수추정 및 모형 선택의 접근방법에 대하여 제시하였고 이 접근방법을 기초로 하여 수명분포가 중첩(Superposition) 및 혼합(Mixture)인 경우에 대한 접근방법이 연구되어야 하고 이러한 접근방법에 대한 분포 이론 및 응용에 관한 연구가 기대된다.

참 고 문 헌

[1] Casella, G. and George, E. I., "Explaining the Gibbs Sampler," The American Statistician, 46, pp.167-174, 1992.
 [2] Chib, S ang Greenberg, E., "Understanding the Metropolis-Hastings Algorithm," The American Statistician, Vol.49, pp.327-335, 1995
 [3] Gelfand, A. E. and Smith, A. F. M., "Sampling-Based

Approaches to Calculating Marginal Densities," Journal of the American Statistical Association, 85, pp.398-409, 1990.

[4] Geman, S. and Geman, D., "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," IEEE Transactions on Pattern Analysis and Machine Intelligence, 6, pp.721-741, 1984.

[5] Gelman, A. E. and Rubin D., "Inference from Iterative Simulation Using Multiple Sequences," Statistical Science, 7, pp.457-472, 1992.

[6] Goel, A. L., Software Reliability Model : Assumption, Limitations, and Applicability, *IEEE Trans. Software Eng.*, Vol.SE-11, No.12, pp.1411-1423, 1985.

[7] Goel, A. L. and Okumoto, K., "Time Dependent Error Detection Rate Model for Software Reliability and Other Performance Measures," IEEE Transactions on Reliability, 28, pp.206-211, 1979.

[8] Kuo, L. and Yang, T. Y., "Bayesian Computation of Software Reliability," Journal of Computational and Graphical Statistics, 1995.

[9] Kuo, L. and Yang, T. Y., "Bayesian Computation for Non-homogeneous Poisson process in Software Reliability," Journal of the American Statistical Association, 91, pp. 763-773, 1996.

[10] Lawless, J. F., *Statistical Models and Methods for Lifetime Data*, New York : John Wiley & Sons, 1981.

[11] Lyu, M. R., *Handbook of Software Reliability Engineering*, New York, NY : McGraw Hill, pp.128-131, 1996.

[12] Musa, J. D., Iannino, A. and Okumoto, K., *Software Reliability : Measurement, Prediction, Application*, New York : McGraw Hill, 1987.

[13] Okumoto, K., "A Statistical Method for Software Quality Control," IEEE Transactions on Software Engineering, Vol.se-11, No.12, pp.1424-1430, 1985.

[14] User Manual *STAT/LIBRARY Fortran Subroutines for statiatical analysis*, IMSL, Vol.3, 1987.

[15] Yamada, S., Ohba, M. and Osaki, S., "S-shaped Reliability Growth Modeling for Software Error Detection," *IEEE Trans. Rel.*, Vol.R-32, No.5, pp.475-47, 1983.

[16] Yamada, S., Ohtera, H. and Narihisa, H., "Software Reliability Growth Models with Testing-Effort," *IEEE Trans.*

Rel., Vol.R-35, No.1, pp.19-23, 1986.



김희철

e-mail : khc@songho.ac.kr
 1998년 동국대학교 대학원 통계학과
 이학박사
 2004년~현재 한라대학교 정보통신공학부
 겸임교수
 관심분야 : 소프트웨어신뢰성공학, 웹프로
 그래밍, 전산통계



이상식

e-mail : leess@songho.ac.kr
 2000년 경희대학교 대학원 전자계산공학과
 공학석사
 2002년 경희대학교 대학원 전자계산공학과
 공학박사
 2001년~현재 송호대학 정보산업계열
 조교수

관심분야 : 소프트웨어공학, 소프트웨어신뢰성, S/W 재사용



송영재

e-mail : yjsong@khu.ac.kr
 1969년 인하대학교 전자공학과 공학사
 1976년 일본 Keio 대학교 전산학과
 공학석사
 1980년 명지대학교 전산학과(공학박사)
 1982년~1983년 미국 Maryland 대학교
 객원교수

1984년~1989년 전국 전산소장 협의회 부회장
 1984년~1989년 경희대학교 전자 계산소장
 1986년~1988년 대한전자공학회 전자계산 연구회 전문위원장
 1990년~1991년 일본 Keio 대학교 객원 교수
 1993년~1995년 경희대학교 교무처장
 1996년~1998년 경희대학교 공과대 학장
 1999년~2000년 경희대학교 기획조정실장
 1976년~현재 경희대학교 컴퓨터공학과 교수
 2001년~현재 경희대학교 산업정보대학 원장
 관심분야 : 소프트웨어공학, OOP/S, CASE 도구, S/W 재사용