백분위수 변화점을 고려한 NHPP 소프트웨어 신뢰성장모형에 관한 연구

김희철* · 신현철**

요 약

소프트웨어 제품의 정확한 인도시기를 예측하거나 효용성 및 신뢰성을 예측하기 위해서는 소프트웨어 테스팅 과정에서 중요한 요소인 테스트 변화점를 이용하면 보다 효율적인 테스팅 작업을 할 수 있다.

본 논문에서는 기존의 소프트웨어 신뢰성 모형인 지수 모형(Goel-Okumoto 모형)을 적용하여 변화점이 백분위수를 가질 경우를 고려하였다. 고장 간격시간으로 구성된 자료를 이용한 모수추정 방법은 최우추정법과 일반적인 수치해석 방법인 이분법을 사용하여 모수 추정을 실시하고 효율적인 모형 선택은 편차자승합(SSE) 을 적용하여 모형들에 대한 효율적인 모형선택을 시도하였다. 수치적인 예에서는 NTDS 자료를 사용하여 백분위수 변화점을 고려한 결과를 나열하였다.

The Study for NHPP Software Reliability Growth Model of Percentile Change-point

Hee-Cheul Kim* · Hyun-Cheul Shin**

ABSTRACT

Accurate predictions of software release times, and estimation of the reliability and availability of a software product require quantification of a critical element of the software testing process : Change-point problem. In this paper, exponential (Goel-Okumoto) model was reviewed, proposes the percentile change-point problem, which maked 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, for the sake of efficient model, was employed. Using NTDS data, The numerical example of percentilechange-point problemi s presented.

Key words : Percentile Change-point, Non-Homogeneous Poission Process, NHPP

* 남서울대학교 산업경영공학과
** 백석문화대학 컴퓨터정보학부

1. 서 론

신뢰도에서 관측시간에 발견된 고장수를 모형화하는데 비동질적 포아송 과정(Non-homogeneous poission process ; NHPP)이 널리 사용하여 왔다. 이러한 NHPP 모형은 강도함수(Intensity function)와 평균값함수(Mean value function)에 의존한다[1, 2].

이 범주에서 지금까지 알려진 모형들은 Goel-Okumoto 모형, Weibull 모형 그리고 Cox-Lewis 모형등이 있는데 이 모형들에 대한 강도함수는 각각 시간에 의존한 함수, 멱(Power) 함수, 대수선형(Log-linear) 함수를 가정하였다[3, 4].

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

이러한 모형은 기존에 존재하는 NHPP모형에서 테스트 커버리지를 포함하는 모형이 된다. 이런 모형을 ENHPP(Enhanced non-homogeneous poission process)이라고도 한다[5].

또 다른 측면에서 NHPP 모형에서 결함은 서로 독립적이고 확률적으로 발생하는 것으로서 결함탐지과정동안 동일한 분포에 따라 발생한다[6]. 그러나 더 현실적인 상황에서 결함분포는 여러 요인 즉, 작업환경, 테스팅 전략 및 작업 할당과 같은 요인들에 의해서 영향을 받을 수 있다. 이러한 요인들은 소프트웨어 테스팅 단계 과정에서 변화되어 강도함수에 영향을 주어 단조적으로 증가하거나 감소되는 결과를 나타낸다. 이를 변화점 문제(Change-point problem ; CP)라고 한다.

이 분야에서는 Zhao(1993), Chang(2001), Shyur(2003) 등에 의해 연구 되었다[7-9].

본 논문에서는 변화점을 사분위수로 적용하는 문제를 고려하고 이러한 적용의 효율성과 그 특성을 알아보고자한다.

본 논문의 제 2장에서는 관련연구로서 사분위수 변화점을 가지는 ENHPP 모형에 대하여 서술하였고 제 3장에서는 모수 추정방법에 대하여 나열하였으며 제 4장에서는 수치적인 예로 실제 고장자료를 이용하여 각 모형에 대한 모수추정 및 모형비교를 실시하였으며 마지막으로 제 5장에서는 결론을 나열 하였다.

2. 백분위수 변화점을 가지는 NHPP 모형

어떠한 소프트웨어 시스템이 결함이 원인이 되어 고장이 발생하고 고장이 발생하면 즉각 수정이나 제거되고 새로운 결함이 시스템에 도입되지 않는다는 가정을 하자. $\{N(t)\,,\,t \ge 0\}$를 시점 t까지의 소프트웨어 누적 고장수라고 표시하는 계수과정(counting process)이라고 한다면 $N(t)$ 평균값 함수 $m(t)$와 $m(0)=0$을 만족하는 NHPP가 된다. 이 NHPP에서 결함탐색 비율은 단조증가, 감소, 상수가 될 수도 있지만 테스팅 환경, 전략 ,작업할당 등의 원인으로 인하여 변화가 일어날 수 있다. 따라서 시점 t까지의 결함탐색비율을 변화시키는 모형(NHPP-CP)을 다음과 같이 나타낼 수 있다[8].

$$b(t)=\left(\frac{d\,m(t)}{dt}\right)/\left(\theta-m(t)\right)=\begin{cases} b_1\,, & 0 \le t \le \tau \\ b_2\,, & t > \tau \end{cases} \qquad (2.1)$$

단, τ은 변화점, θ은 테스팅 초기시점에서의 소프트웨어 결함의 기대수를 나타내고, $b_1(>0)$과 $b_2(>0)$는 각각 변화점 이전의 결함탐색비율, 변화점이 시작되는 결함탐색비율을 의미한다.

물론 식 (2.1)에서 $b_1=b_2=b$이면 NHPP 모형과 NHPP-CP 모형은 같은 모형이 된다.

Goel-Okumoto 모형[6]을 식 (2.1)에 적용하면 $m(t)$는 다음과 같은 식으로 변환 시킬 수 있다.

$$m(t)=\begin{cases} \theta(1-e^{b_1 t})\,, & 0 \le t \le \tau \\ \theta(1-e^{-b_1\tau-b_2(t-\tau)})\,, & t > \tau \end{cases} \qquad (2.2)$$

또, 강도함수는 다음과 같이 표현된다.

$$\lambda(t)=\frac{dm(t)}{dt}=\begin{cases}\theta b_1\, e^{-b_1 t},\ 0\le t\le\tau\\ \theta\, b_2 e^{-b_1\tau-b_2(t-\tau)},\ t>\tau\end{cases} \tag{2.3}$$

식 (2.2)과 식 (2.3)에서 $m(t)$는 연속함수이지만 $\lambda(t)$는 $t=\tau$의 시점에서는 연속함수가 아니다. $b_1\neq b_2$ 경우에 $\lambda(t)$는 단계적으로 감소함수의 패턴을 가진다. 물론 NHPP-CP에서 정수형(상수) 변화점을 가정 할 수도 있다. 따라서 여러 가지 변화점을 가진 모형을 제시 할 수 있다. 본 논문에서는 고장자료에서 결함 발견율이 변화하는 시점인 변화점 τ을 백분위수(percentile)를 사용하고자 한다.

즉, 85분위수(Q_{85})와 90분위수(Q_{90}) 그리고 95분위수(Q_{95})를 고려하고자 한다.

3. 최우추정법을 이용한 모수 추정

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

$$t_k=\sum_{i=1}^{k}x_k\,(k=1,2,\cdots,n\,;0\le t_1\le t_2\le\cdots\le t_n) \tag{3.1}$$

이라고 하면 데이터 집합 D_t는 $\{n,t_1,t_2\cdots t_n;t\}$와 같이 구성된다. n번째까지 고장시점이 관찰된 고장절단 모형일 경우에 데이터 집합 D_{t_n}은 $\{t_1,t_2\cdots t_n\}$으로 구성된다. 이 시간 절단 모형에서의 우도함수는 다음과 같이 알려져 있다[4, 11].

$$L(\Theta\mid D_t)=\left(\prod_{i=1}^{n}\lambda(t_i)\right)\exp(-m(t)) \tag{3.2}$$

단, Θ은 미지의 모수 집합을 의미하고 우도함수 식 (3.2)에서 t을 t_n으로 대치하면 유사한 형태의 고장 절단 모형의 우도함수가 된다.

따라서 식 (3.2)에 식 (2.2)와 식 (2.3)을 대입하면 하나의 우도함수를 유도 할 수 있다. 즉, 로그 우도함수는 다음과 같이 표현된다[8].

$$\begin{aligned}&\ln L(\tau,\theta,b_1,b_2)\\&=-\theta+\theta\, e^{-b_1\tau-b_2(t-\tau)}+\sum_{i=1}^{N(\tau)}[\ln(\theta b_1)-b_1t_i]\\&+\sum_{i=N(\tau)+1}^{n}[\ln(\theta b_2)-b_1\tau-b_2(t_i-\tau)]\end{aligned} \tag{3.3}$$

식 (3.3)을 이용하여 최우추정법(MLE)방법으로 계산할 경우에 $\ln b_2\rightarrow\infty$로 수렴하지 않기 때문에 사용 할 수 없다고 하였다(Chang, Y. P(2001)). 따라서 본 논문에서는 b_2를 0.5로 고정하여 최우 추정법을 이용하고자 한다.

즉, 식 (3.3)에서 $b_2=0.5$인 경우 θ와 b_1로 편미분을 이용한 MLE는 다음식이 성립한다.

$$\hat{\theta}_{mle}=\frac{2N(\tau)-n+1}{1-\exp(-b_1\tau-0.5(t-\tau)} \tag{3.4}$$

$$\frac{N(\tau)}{\hat{b}_{1\,mle}}=\sum_{i=1}^{N(\tau)}t_i+ \tag{3.5}$$

$$\tau(n-N(\tau)-1)+\tau\theta\exp(-\hat{b}_{1_{mle}}\tau-0.5(t-\tau))$$

4. 수치적인 예

이 장에서 실제 고장자료를 이용하여 형상모수에 따른 Burr 모형을 분석하고자 한다. 이 고장자료는 NTDS(Naval Tactical Data System)에 의해 발생된 소프트웨어 고장자료로서 Goel과 Okumoto [6], Mazzuchi와 Kuo와 Yang[11] 등이 이 고장자료를 이용하여 소프트웨어 모형을 제안 한 바 있다. 본 연구도 이 자료를 이용하고자 한다. 이 자료는 <표 1>에 나열 되어 있다. <표 1>의 자료에서 $t_n=t_{26}=250$이고 $n=26$이다. 본 논문에서는 위 고장자료에서 결함 발견율이 변화하는 시점인 변화점 τ을 백분위수(percentile)를 사용하고자 한다. 즉, 85분위수(Q_{85}) $=\left(\frac{85}{100}\right)\times 26=22.1$되므로 23번째 자료가 85분위수가 된다[12]. 즉, $Q_{85}=156$이 된다. 유사한 방법으로 90분위수(Q_{90}) = 247 95분위수(Q_{95})

=249로 계산되어 각 분위수를 이용한 변화점은 다음과 같이 표현된다.

$$N(\tau) = N(Q_{85}) = N(156) = 23$$
$$N(\tau) = N(Q_{90}) = N(247) = 24$$
$$N(\tau) = N(Q_{95}) = N(249) = 25$$

90분위수(Q_{90})에서 모수 추정은 식 (3.4)와 식 (3.5)를 이용하면 다음과 같이 표현된다.

$$\hat{\theta}_{mle} = \frac{21}{1-\exp(-156\times b_1 - 94)} \tag{3.6}$$

$$\frac{23}{\hat{b}_{1\,mle}} = 2058+156\cdot\theta\cdot\exp(-156\cdot b_{1_{mle}} - 94) \tag{3.7}$$

식 (3.6)과 식 (3.7)를 이용하면 모수 추정을 할 수 있다.

〈표 1〉 고장 간격 자료

Failure number	Failure Interval Time	Failure Time
1	9	9
2	12	21
3	11	32
4	4	36
5	7	43
6	2	45
7	5	50
8	8	58
9	5	63
10	7	70
11	1	71
12	6	77
13	1	78
14	9	87
15	4	91
16	1	92
17	3	95
18	3	98
19	6	104
20	1	105
21	11	116
22	33	149
23	7	156
24	91	247
25	2	249
26	1	250

유사한 방법으로 90분위수(Q_{90}) = 247 95분위수(Q_{95}) = 249로 추정 할 수 있다. 그 결과는 <표 2>에 요약되었다. 이러한 비선형 방정식의 계산방법은 수치해석적 기본 방법인 이분법(Bisection method)을 사용하였다. 이러한 계산은 초기값을 0과 1을, 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.

〈표 2〉 형상 모수 $b_2 = 0.5$인 경우의 모수 추정값

변화점 $N(\tau)$	MLE		
미고려	$\hat{b}$ = 0.00579		$\hat{\theta}$ = 33.99
고려	$\hat{b_2}$	$\hat{b_1}$	$\hat{\theta}$
$N(Q_{85})$	0.5	0.004311	23.52
$N(Q_{90})$		0.002815	25.88
$N(Q_{95})$		0.002876	32.69

모형 선택의 하나의 방법으로 편차자승합(SSE [2, 6])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 변화점으로 85분위수($N(Q_{85})$)와 90분위수($N(Q_{90})$) 대체 모형이 변화점을 고려하지 않은 모형보다 상대적으로 효율적 모형으로 나타나고 있다.

〈표 3〉 모형들에 대한 SSE의 값

변화점 $N(\tau)$	SSE
미고려	129.7239
$N(Q_{85})$	126.4596
$N(Q_{90})$	129.6952
$N(Q_{95})$	144.4886

5. 결 론

소프트웨어 신뢰성은 개발의 최종단계에 있는 테스트 공정이나 실제 사용단계에 있어서 소프트

웨어 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 발생시간 과의 관계를 효율적으로 관리함으로서 소프트웨어 신뢰도를 성장 시 킬 수 있다. 이러한 과정을 소프트웨어 성장 과정이라고 볼 수 있다[6].

본 논문에서는 NHPP 모형에서 결함은 서로 독립적이고 확률적으로 발생하는 것으로서 결함탐지 과정동안 동일한 분포에 따라 발생한다[6]. 그러나 더 현실적인 상황에서 결함분포는 여러 요인 즉, 작업환경, 테스팅 전략 및 작업 할당과 같은 요인들에 의해서 영향을 받을 수 있다. 이러한 요인들은 소프트웨어 테스팅 단계 과정에서 변화되어 강도함수에 영향을 주어 단조적으로 증가하거나 감소되는 결과를 나타낸다.

본 논문에서는 변화점을 백분위수로 적용하는 문제를 고려하고 이러한 적용의 효율성과 그 특성을 알아보고자한다.. 효율적인 모형 비교를 위한 편차자승합의 결과는 변화점이 백분위수로 적용한 모형이 변화점을 무시한 모형보다 상대적으로 효율적인 모형으로 로 간 주 할 수 있다. 이러한 변화점을 고려한 베이지안적 접근 방법이나 수리적인 추정과 검정 부분에 대한 수리적인 접근이 기대 된다.

참고문헌

[1] J. F. Lawless, “Statistical Models and Methods for Lifetime Data”, John Wiley & Sons, New York, 1981.

[2] 김희철, “일반화감마분포를 이용한 NHPP 소프트웨어 신뢰도 모형에 관한 연구”, 한국컴퓨터정보학회 논문지, 제10권, 제6호, pp. 27-35, 2005.

[3] 김희철, “지수화 지수분포에 의존한 NHPP 소프트웨어 신뢰성장 모형에 관한 연구”, 한국컴퓨터정보 학회논문지, 제11권, 제5호, pp. 9-18, 2006.

[4] S. S. Gokhale, and K. S. Trivedi., “A time/structure based software reliability model”, Annals of Software Engineering, Vol. 8, pp. 85-121. 1999.

[5] 김희철, “Burr 커버리지 함수에 기초한 ENHPP 소프트웨어 신뢰성장모형에 관한 연구”, 한국컴퓨터정보 학회논문지, 제12권, 제4호, pp. 1-10, 2007.

[6] 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.

[7] M. Zhao, “Change-point problems in software and hardware reliability”, Commun. Statistical-Treory Math., Vol. 22, pp. 757-768, 1993.

[8] Y. P. Chang, “Estimation pf parameters for honhomogeneous Poission process : software reliability with Change-point model”, Commun. Statist.-Simula., Vol. 30, No. 3, pp. 623-635, 2001.

[9] H. J. Shyur, “A stocastic software reliability model with imperfect-debugging and change-point”, The Journal of System and Software, Vol, 66, pp. 135-141, 2003.

[10] 김희철, 박종구, “Burr분포를 이용한 NHPP 소프트웨어 신뢰성장모형에 관한 연구”, 한국해양정보통신 학회논문지, 제11권, 제3호, pp. 514-522, 2007.

[11] L. Kuo and T. Y. Yang, “Bayesian Computation of Software Reliability”, Journal of

the American Statistical Association, Vol. 91, pp. 763-773, 1996.
[12] 류귀열, 박일렬, 최승두역, "앤더슨의 통계학", 한올출판사, pp. 104-109, 2007.

김 희 철

1992년 동국대학교 통계학과 (이학석사)
1998년 동국대학교 통계학과 (이학박사)
2005년~현재 남서울대학교 산업경영공학과 전임강사

신 현 철

2002년 원광대학교 컴퓨터 공학과(공학박사)
1994년~현재 백석문화대학 컴퓨터정보학부 교수
2004년~현재 (주)아이비루션 자문위원
2005년~현재 한국정보처리학회 이사
한국사이버테러정보전학회 부회장