한국정보전자통신기술학회논문지(jkiiect)'16-06, Vol.9 No.3
논문 16-09-03-278 http://dx.doi.org/10.17661/jkiiect.2016.9.3.278

선형위험률분포의 절편모수에 근거한 무한고장 NHPP 소프트웨어 신뢰모형에 관한 연구

김희철*, 신현철**

The Study of Infinite NHPP Software Reliability Model from the Intercept Parameter using Linear Hazard Rate Distribution

Hee-Cheul Kim*, Hyun-Cheul Shin**

요 약 소프트웨어 개발과정에서 소프트웨어 신뢰성은 매우 중요한 이슈이다. 소프트웨어 고장분석을 위한 무한고장 비동질적인 포아송과정에서 고장발생률이 상수이거나, 단조 증가 또는 단조 감소하는 패턴을 가질 수 있다. 본 논문에서는 수리시점에서도 고장이 발생할 상황을 반영하는 무한고장 NHPP모형들을 비교 제시하였다. 소프트웨어 경제, 경영, 보험수리분야에서 많이 사용되는 선형 위험률분포의 절편모수에 근거한 무한고장 소프트웨어 신뢰성모형에 대한 비교문제를 제시하였다. 그 결과 절편모수가 비교적 큰 경우가 효율적으로 나타났다. 그리고 모수 추정법은 최우추정법을 이용하였고 모형선택은 평균제곱오차와 결정계수를 이용하였다. 본 연구에서 제안된 방법은 선형 위험률분포의 절편모수를 고려한 모형도 신뢰성 측면에서 효율적이기 때문에 (결정계수가 90% 이상) 이 분야에서 기존 모형의 하나의 대안으로 사용할 수 있음을 확인 할 수 있었다. 이 연구를 통하여 소프트웨어 개발자들은 다양한 수명분포의 절편모수를 고려함으로서 소프트웨어 고장형태에 대한 사전지식을 파악하는데 도움을 줄 수 있으리라 사료 된다.

Abstract Software reliability in the software development process is an important issue. In infinite failure NHPP software reliability models, the fault occurrence rates may have constant, monotonic increasing or monotonic decreasing pattern. In this paper, infinite failures NHPP models that the situation was reflected for the fault occurs in the repair time, were presented about comparing property. Commonly, the software model of the infinite failures using the linear hazard rate distribution software reliability based on intercept parameter was used in business economics and actuarial modeling, was presented for comparison problem. The result is that a relatively large intercept parameter was appeared effectively form. The parameters estimation using maximum likelihood estimation was conducted and model selection was performed using the mean square error and the coefficient of determination. The linear hazard rate distribution model is also efficient in terms of reliability because it (the coefficient of determination is 90% or more) in the field of the conventional model can be used as an alternative model could be confirmed. From this paper, the software developers have to consider intercept parameter of life distribution by prior knowledge of the software to identify failure modes which can be able to help.

Key Words : Software Reliability, NHPP, Laplace Trend Test, Linear Hazard Rate Distribution.

* Corresponding Author : Division of Industrial & Management Engineering, Namseoul University, Seoul, Korea
**Division of Computer Engineering, BaekSeok Culture University, Seoul, Korea
Received June 04, 2016 Revised June 20, 2016 Accepted June 24, 2016

1. 서론

소프트웨어 고장으로 인한 컴퓨터 시스템의 고장은 우리 사회에 엄청난 손실을 유발 할 수 도 있다. 따라서 소프트웨어 개발 과정에서 소프트웨어 신뢰성은 중요한 문제이다. 이 문제는 사용자의 요구조건과 테스팅 비용을 만족시켜야 한다. 소프트웨어 테스팅(디버깅)면에서 비용을 줄이기 위해서는 소프트웨어의 신뢰성의 변동과 테스팅 비용을 사전에 알고 있어야 효율적이다. 따라서 신뢰도, 비용 및 방출 시간의 고려사항을 가진 소프트웨어 개발 과정은 필수 불가결 하다. 결국 소프트웨어 제품의 결함내용을 예측하기 위한 모형 개발이 필요하다. 지금까지 많은 소프트웨어 신뢰성 모형이 제안 되었다. 이 중에서 비동질적 포아송 과정(Non-homogeneous Poisson process; NHPP)에 의존한 모형[1]은 에러 탐색 과정측면에서는 우수한 모형이고 이 모형은 결함이 발생하면 즉시 제거되고 디버깅 과정에서 새로운 결함이 발생되지 않는다는 가정을 하고 있다. 이 분야에서 Gokhale 과Trivedi [1]은 고양된 비동질적인 포아송 과정 모형(Enhanced NHPP) 모형을 제시하였고 Goel 과 Okumoto [2]은 지수적 소프트웨어 신뢰성 모형(Exponential software reliability growth model)을 제안 하였다. 이 모형은 결함의 누적수가 S 형태나 지수적 형태(S-shaped or exponential-shaped)를 가진 평균값함수(Mean value function)를 이용하였다. 이러한 모형에 의존한 일반화 모형은 Yamada 와 Ohba [3]에 의해 지연된 S-형태 신뢰성장모형(Delayed S-shaped reliability growth model)과 변곡된 S-형태 신뢰성장모형(Inflection S-shaped reliability growth model)이 제안되었다. Zhao [4]는 소프트웨어 신뢰도에서 변환점 문제를 제시하였고 Shyur [5]는 변환점을 이용한 일반화한 신뢰도 성장 모형을 제안하였다. Pham와 Zhang[6]는 테스팅 커버리지(Coverage)를 측정하여 소프트웨어 안정도를 평가 할 수 있는 소프트웨어 안정도 모형을 제시했다. 비교적 최근에, Huang [7]은 일반화 로지스틱 테스팅 노력 함수(Generalized logistic testing-effort function)와 변환점모수(Change-point parameter)를 통합하여 효율적인 소프트웨어 신뢰성 예측 기술을 제시하기도 하였다. 그리고 최근에는 S-형태 모형은 소프트웨어 관리자들이 소프트웨어 및 검사 도구에 익숙해지는 학습 과정을 설명 할 수 있다고 하였고 [8] 또한, 대수 선형 위험함수를 이용한 학습과정 특성을 연구하기도 하였다[9].

따라서 본 논문에서는 선형 위험률 분포의 절편모수에 근거한 무한고장 소프트웨어 신뢰성모형에 대한 효율성과 그 특징을 알아보고자 한다.

2. NHPP 소프트웨어 신뢰성

$N(t)$ 을 시간 t 까지 검출된 소프트웨어의 누적 고장수라고 하고, $m(t)$를 이에 대한 기대값를 나타내는 평균값 함수(Mean Value Function)로 가정하고 $\lambda(t)$ 을 강도함수(Intensity function) (즉, t 에서의 순간 결함 검출비율)이면 비동질 포아송 과정(NHPP)은 누적 고장수인 $N(t)$ 는 모수 $m(t)$ 을 가진 포아송 확률밀도함수 (Probability density function)로 알려져 있다. 즉,

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

따라서, NHPP모형에서 평균값함수 $m(t)$ 와 강도함수 $\lambda(t)$ 는 다음과 같은 관계로 표현할 수 있다[1, 9].

$$m(t)=\int_0^t \lambda(s)ds\ ,\ \frac{d\,m(t)}{dt}=\lambda(t) \quad (2)$$

이처럼 시간관련모형(Time domain models)들은 NHPP에 의해서 확률고장과정으로 설명이 가능하다. 이러한 NHPP모형들은 유한고장모형과 무한고장 범주로 분류한다[1, 9].

무한고장 NHPP모형들은 수리시점에서도 고장이 발생할 상황을 반영하기 위하여 기록 멈춤 통계량(Record breaking statistics)을 사용하는 RVS(Record Value Statistics)모형을 사용할 수 있다고 하였고 이

RVS모형과 NHPP모형에 관해서 평균값함수는 다음과 같이 된다고 하였다[1, 9].

$$m(t) = -\ln(1-F(t)) \tag{3}$$

따라서 (3)식을 이용하고 $f(t)$을 확률밀도함수, $F(t)$을 분포함수라고 하면 NHPP의 강도함수는 다음과 같이 위험함수($h(t)$)가 된다.

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

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

$$x_n = \sum_{i=1}^{n} t_i \quad (i = 1,2,\cdots,n\,;\, 0 \le x_1 \le x_2 \le \cdots \le x_n) \tag{5}$$

이라고 하면 n번째까지 고장시점이 관찰된 고장절단 모형일 경우에 데이터 집합 D_{x_n}은 $\{x_1, x_2 \cdots x_n\}$으로 구성되며, 이 고장절단모형에서 Θ을 모수공간이라고 표시하면 NHPP모형의 우도함수는 다음과 같이 알려져 있다[1, 9].

$$L_{NHPP}(\Theta | \underline{x}) = \left(\prod_{i=1}^{n} \lambda(x_i)\right) \exp[-m(x_n)] \tag{6}$$

단, $\underline{x} = (x_1, x_2, x_3, \cdots, x_n]$

NHPP 모형에서 테스트 시점 x_n에서 소프트웨어 고장이 일어난다고 하는 가정 하에서 신뢰구간 $(x_n, x_n + t]$ (단, t는 임무시간(Mission time))사이에서 소프트웨어의 고장이 일어나지 않을 확률인 신뢰도(Reliability) $\hat{R}(t \mid x_n)$는 다음과 같이 됨이 알려져 있다[9].

$$\hat{R}(t \mid x_n) = e^{-\int_{x_n}^{x_n+t} \lambda(\tau) d\tau} = \exp[-\{m(t+x_n) - m(x_n)\}] \tag{7}$$

2.1 관측 자료에 대한 모형 비교

최근에 모형에 대한 효율성을 조사하기 위한 기준으로서 MSE(평균자승오차)와 Rsq(결정계수)를 사용한다[8, 9].

2.1.1 평균제곱오차(Mean square error)

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

$$MSE = \frac{\sum_{i=1}^{n} \left(m(x_i) - \hat{m}(x_i)\right)^2}{n-k} \tag{8}$$

단, $m(x_i)$은 시간$(0, x_i]$까지 나타난 에러들의 누적함수를 의미하고 $\hat{m}(x_i)$는 x_i시점까지 평균값 함수로부터 추정된 에러의 누적개수를 의미한다. 그리고 n은 관찰 값의 수이고 k는 모수의 수를 의미한다. 즉 비교에 있어서 평균제곱오차 값이 작으면 상대적으로 효율적인 모형이 된다.

2.1.2 R^2

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

$$R^2 = 1 - \frac{\sum_{i=1}^{n} \left(m(x_i) - \hat{m}(x_i)\right)^2}{\sum_{i=1}^{n} \left(m(x_i) - \sum_{j=1}^{n} m(x_j)/n\right)^2} \tag{9}$$

즉 비교에 있어서 결정계수 값이 크면 상대적으로 효율적인 모형이 된다.

3. 제안된 선형위험률분포의 형상모수를 고려한 NHPP 모형

선형 위험률분포는 다항위험함수(Polynomial hazard function) 형태의 특수한 경우로 다음과 같이 알려져 있다[10].

$$h(t\,|) = \beta_0 + \beta_1 t + \cdots + \beta_{m-1} t^{m-1},\ t \geq 0 \qquad (10)$$

단, $\beta_i\,(i = 0,1,2,\cdots,m-1)$ 는 다항위험함수의 계수이고 (10)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. 따라서 (4)식에서 보여 주듯이 무한고장 NHPP의 강도함수는 위험함수($h(t)$)가 된다고 하였기 때문에 다항형태의 무한고장 NHPP 강도함수 $\lambda(t)$는 (10)식과 동일한 위험함수 $h(t)$가 된다.

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

$$m(t) = \beta_0\, t + \frac{1}{2}\beta_1 t^2 + \frac{1}{3}\beta_2 t^3 + \cdots + \frac{1}{m}\beta_{m-1}\, t^m \qquad (11)$$

따라서 본 연구에서는 $m = 2$인 경우를 이용한 선형 위험률 분포를 적용하고자 한다.

즉, 위험률 분포 모형의 강도함수와 평균값함수는 다음과 같이 유도된다.

$$\lambda(t) = \beta_0 + \beta_1 t,\ m(t) = \beta_0\, t + \frac{1}{2}\beta_1 t^2 \qquad (12)$$

모수추정은 최우추정법(MLE)을 이용하였다. 이러한 추정 법을 사용하기 위한 로그 우도 함수는 (6)식과 (12)식을 연관 하면 다음과 같이 유도된다.

$$\ln L_{NHPP_{NHF}}(\Theta\,|\,\underline{x}) = \sum_{i=1}^{n} \ln(\beta_0 + \beta_1 x_i) - \beta_0 x_n - \frac{1}{2}\beta_1 x_n^2 \qquad (13)$$

(13)식에서

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

을 만족하는 $\hat{\beta}_{0\,MLE}$ 와 $\hat{\beta}_{1\,MLE}$ 는 다음과 같이 구할 수 있다[8]

$$\frac{\partial \ln L_{NHPP_{poly}}(\Theta\,|\,\underline{x})}{\partial \beta_0} = \sum_{i=1}^{n} \frac{1}{\beta_0 + \beta_1 x_i} - x_n = 0 \qquad (14)$$

$$\frac{\partial \ln L_{NHPP_{poly}}(\Theta\,|\,\underline{x})}{\partial \beta_1} = \sum_{i=1}^{n} \frac{x_i}{\beta_0 + \beta_1 x_i} - \frac{1}{2}x_n^2 = 0 \qquad (15)$$

단, $\underline{x} = (x_1, x_2, x_3, \cdots, x_n]$

그러나 위 식들은 근을 구하는데 근이 수렴되지 않으므로 본 연구에서는 β_0 의 추정값(철편모수)를 고정된 상수로 간주하여 전개하고자 한다.

4. 소프트웨어 고장시간 분석

이 절에서는 소프트웨어 고장 시간자료 [11] (Failure time data)를 이용하여 본 논문에서 제시하는 소프트웨어 신뢰모형들을 이용한 소프트웨어 고장시간을 분석하고자 한다. 이 자료의 고장시간은 18.735 시간단위에 30번의 고장이 발생된 자료이며 [표 1]에 나열 되어 있다.

또 한 제시하는 신뢰모형들을 분석하기 위하여 우선 자료에 대한 추세검정이 선행 되어야 한다[12]. 추세분석에는 일반적으로 라플라스 추세검정(Laplace trend test)을 사용한다. 이 검정을 실시한 결과 [그림 1]에서 라플라스요인(Factor)이 -2와 2사이에 존재함으로서 즉, 극단값(Extreme value)이 존재하지 않으므로 이 자료를 이용하여 신뢰성장모형을 제시하는 것이 효율적임을 시사하고 있다[9].

표 1. 소프트웨어의 고장시간자료
Table 1. Failure time data of software

Failure number (i)	Failure time (hours) x_i	Failure number (i)	Failure time (hours) x_i
1	0.479	16	10.771
2	0.745	17	10.906
3	1.022	18	11.183
4	1.576	19	11.779
5	2.61	20	12.536
6	3.559	21	12.973
7	4.252	22	15.203
8	4.849	23	15.64
9	4.966	24	15.98
10	5.136	25	16.385
11	5.253	26	16.96
12	6.527	27	17.237
13	6.996	28	17.6
14	8.17	29	18.122
15	8.863	30	18.735

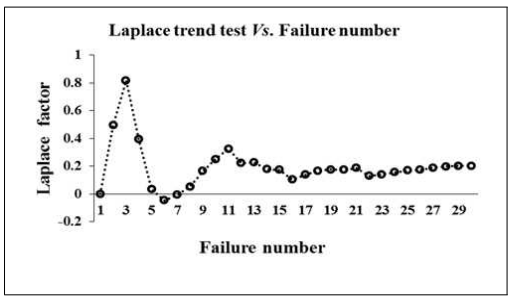

그림 1. 라플라스 추세검정
Fig. 1. Laplace trend test

모수추정은 최우추정법을 이용하고 비선형 방정식의 계산방법은 수치 해석적 기본 방법인 이분법(Bisection method)을 사용하였다. 이러한 계산은 초기 값을 10^{-4}과 5을, 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. 최우추정법의 결과와 모형에 대한 효율성을 조사하기 위한 기준으로서 MSE(평균제곱오차)와 R^2(결정계수)가 [표 2]에 요약되었다.

이 표에서 절편(Intercept)이 비교적 큰 경우가 실제 값과 예측 값에 대한 차이를 측정하는 평균제곱오차가 낮고 예측 값의 차이에 대한 설명력을 의미하는 결정계수도 높게 나타나기 때문에 절편이 비교적 큰 경우의 모형이 효율적인 모형으로 간주할 수 있다. 그러나 둘 모형이 결정계수 값이 90%이상 나타나 이 분야에서 효율적 모형임을 확인 할 수 있다[9].

표 2. 모수 추정값 및 MSE , R^2
Table 2. Parameter estimation of the each model and MSE , R^2

Intercept	MLE	Model Comparison	
β_0	$\widehat{\beta}_{1_{MLE}}$	MSE	R^2
0.5	0.0987	24.7269	0.9626
0.9	0.0601	11.9780	0.9794
1.5	0.0104	1.7324	0.9898

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

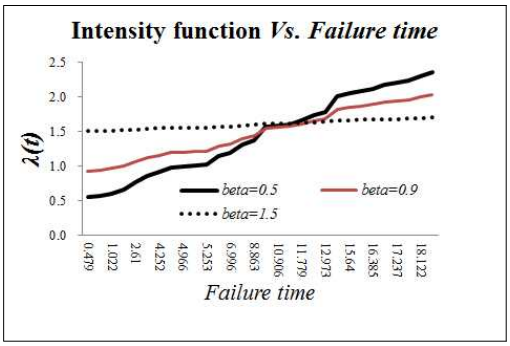

그림 2. 강도함수의 형태
Fig. 2. Pattern of intensity function

[그림 2]는 강도함수에 대한 패턴이 요약 되었다. 이 그림에서 절편모수가 비교적 큰 경우는 거의 고정된 상수처럼 나타났지만 절편모수가 작은 경우는 비감소 패턴을 보이고 있다.

[그림 3]은 평균값 함수에 대한 패턴이 요약 되었다.이 그림에서도 절편모수가 비교적 큰 경우가 작은 경우보다 높게 나타나고 있다.

[그림 4]에서 보여 주듯이 임무시간에 대한 신뢰도 그림에서도절편모수가 비교적 큰 경우가 작

은 경우보다 높게 나타나고 있다.

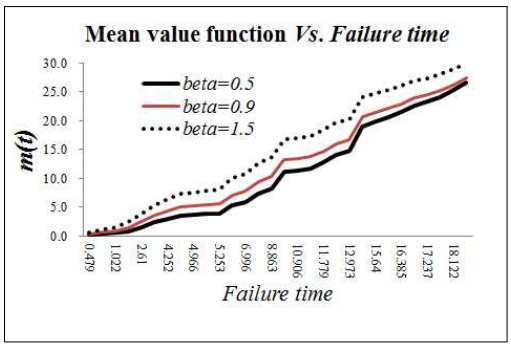

그림 3. 평균값 함수의 형태
Fig. 3. Pattern of mean value function

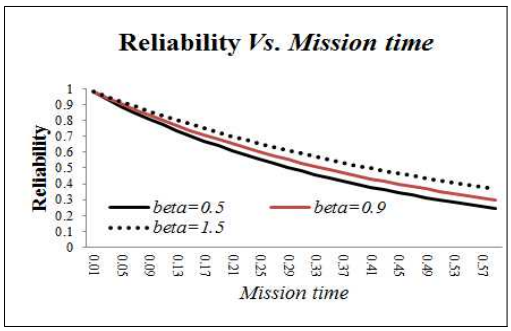

그림 4. 신뢰도 함수의 형태
Fig. 4. Pattern of reliability function

5. 결론

소프트웨어 신뢰성은 개발의 최종단계에 있는 테스트 공정이나 실제 사용단계에 있어서 소프트웨어 내에 존재하는 고장 수나 고장 발생시간에 의해서 효과적으로 평가할 수 있는 상황으로 그 평가 기술이 중요하게 된다. 따라서 소프트웨어 개발의 테스트공정이나 실제사용단계에 있어서 고장 발생 환경이나 고장 발생현상을 수리적으로 모형화가 가능하면 평가를 할 수 있다.

따라서 본 연구에서는 소프트웨어 관리자들이 소프트웨어 고장원인 및 검사 도구에 활용 할 수 있는 수명분포가 다양한 특성을 가지는 선형위험률분포의 절편모수 모형에 대하여 비교 연구 하였다.

그 결과 절편모수가 비교적 큰 경우가 작은 경우보다 실제 값과 예측 값에 대한 차이를 측정하는 평균제곱오차가 낮고 예측 값의 차이에 대한 설명력을 의미하는 결정계수도 제일 높게 나타나기 때문에 절편모수가 비교적 큰 경우가 작은 경우보다 효율적인 모형으로 간주할 수 있다. 그리고 미래 임무시간에 대한 신뢰도 그림에서도 절편모수가 비교적 큰 경우가 작은 경우보다 임무시간이 증가할수록 신뢰도의 상승으로 나타나고 있다.

따라서 본 연구에 제안된 선형 위험률분포 모형도 신뢰성 측면에서 효율적이기 때문에 (결정계수가 90% 이상) 이 분야에서 기존의 모형의 하나의 대안으로 사용할 수 있음을 확인 할 수 있었다. 이 연구를 통하여 소프트웨어 개발자들은 다양한 수명분포를 고려함으로서 소프트웨어 고장형태에 대한 사전지식을 파악하는데 도움을 줄 수 있으리라 사료 된다.

REFERENCES

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

[2] 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.

[3] Yamada S, Ohba H. S-shaped software reliability modeling for software error detection. IEEE Transaction on Reliability 3, pp. 475 - 84, 1983.

[4] Zhao M. Change-point problems in software and hardware reliability. Communication Stat Theory Methods 22(3), pp. 757 - 68, 1993.

[5] Shyur H-J. A stochastic software reliability

model with imperfect debugging and change-point. J Syst Software 66(2), pp. 135 - 41, 2003.

[6] Pham H, Zhang X., "NHPP software reliability and cost models with testing coverage", Eur. J. Oper. Res, 145, pp.445-454, 2003.

[7] Huang C-Y. Performance analysis of software reliability growth models with testing-effort and change-point. J Syst Software. 2005; 76:181 - 94.

[8] 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.

[9] Hee-Cheul KIM, "The Assessing Comparative Study for Statistical Process Control of Software Reliability Model Based on Rayleigh and Burr Type", Journal of korea society of digital industry and information management, Volume 10, No.2, pp. 1-11,2014.

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

[11] Y. HAYAKAWA and G. TELFAR, "Mixed Poisson-Type Processes with Application in Software Reliability", Mathematical and Computer Modelling, 31, pp.151-156, 2000.

[12] 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.

저자약력

김 희 철 (Hee-Cheul Kim) **[정회원]**

- 1992년 2월 : 동국대학교 통계학과 (이학석사)
- 1998년 8월 : 동국대학교 통계학과 (이학박사)
- 2005년 3월 ~ 현재 : 남서울대학교 산업경영공학과 교수

<관심분야> 소프트웨어신뢰성 공학, 전산 통계, 웹프로그래밍

신 현 철(Hyun-Cheul Shin) **[정회원]**

- 2002년 2월 : 원광대학교 컴퓨터공학과 (공학박사)
- 2005년 3월 : 한국정보처리학회 이사
- 1994년 3월 ~ 현재 : 백석문화대학교 컴퓨터공학부 교수

<관심분야> 통신공학, 컴퓨터과학, 소프트웨어공학, 소프트웨어공학