

## 일반화 감마분포에 근거한 소프트웨어 최적방출시기에 관한 비교 연구\*

김재욱\*\* · 김희철\*\*\*

### *A Study on Optimal Release Time for Software Systems based on Generalized Gamma Distribution*

Kim, Jae Wook · Kim, Hee Cheul

#### 〈Abstract〉

Decision problem called an optimal release policies, after testing a software system in development phase and transfer it to the user, is studied. The applied model of release time exploited infinite non-homogeneous Poisson process. This infinite non-homogeneous Poisson process is a model which reflects the possibility of introducing new faults when correcting or modifying the software. The failure life-cycle distribution used generalized gamma type distribution which has the efficient various property because of various shape and scale parameter. Thus, software release policies which minimize a total average software cost of development and maintenance under the constraint of satisfying a software reliability requirement becomes an optimal release policies. In a numerical example, after trend test applied and estimated the parameters using maximum likelihood estimation of inter-failure time data, estimated software optimal release time.

Key Words : Software Release Policies, Record Value Statistics, Generalized Gamma Distribution, Total Average Software Cost

## I. 서론

소프트웨어 신뢰성은 컴퓨터 시스템에 대한 적용과 이에 대한 연구 분야에서 중요한 역할을 담당해 오고 있다. 소프트웨어 고장으로 인한 컴퓨터 시스템의 고장은 우리 사회에 엄청난 손실을 유발 할 수 도 있다. 따라서

소프트웨어 신뢰도는 현대의 소프트웨어 생산품 개발에서 중요한 분야 가운데 하나이다. 이러한 소프트웨어 신뢰성 엔지니어링에서의 연구 활동은 지난 30년 전부터 행해져 오고 있고 많은 신뢰도 성장 모형들이 소프트웨어에 남아 있는 고장들의 수와 소프트웨어 신뢰도의 추정에 관한 문제들을 제안해 왔다. 일반적으로 소프트웨어 개발과정은 설계단계, 디자인, 코딩 그리고 테스트 단계를 거친다. 이러한 과정을 거친 후 소프트웨어 제품을 방출하게 되는데 방출이후에 발견되지 않은 고장들이 나

\* 이 논문은 2009년도 남서울대학교 학술연구비 지원에 의하여 연구되었음.

\*\* 남서울대학교 전자공학과 교수

\*\*\* 남서울대학교 산업경영공학과 교수(교신저자)

타난다면 이것들에 대한 보전 비용(Maintenance cost)은 크게 증가 할 것이다. 결국, 소프트웨어 시스템 시험을 끝내고 그것을 사용자에게 넘기는 시기 결정은 매우 중요한 사항이 된다.

소프트웨어 방출시간에 대한 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 발생 할 수 있다. 따라서 본 연구에서는 새로운 고장을 까지도 반영하는 무한(Infinite) 고장 NHPP 모형을 이용하여 최적 방출시기에 대한 문제를 제안하고자 한다. 이 분야에서는 Musa-Okumoto의 대수 포아송 실행시간 모형 [3]과 로그-파우어 모형[4]을 이용한 방출 문제에 대한 문제들이 이미 연구되었고 이와 관련된 문제에 대한 연구는 Yang 과 Xie(2000) 와 Huang(2005)에 의해 보증기간 할인을 및 보증기간 이후의 사후 대책 등에 대하여 연구되고 있다[5-6].

본 연구에서는 소프트웨어의 결함을 제거하거나 수정 작업 중에도 새로운 결함이 발생될 가능성이 있는 무한 고장수를 가진 비동질적인 포아송 과정에 기초하고 수명 분포는 다양한 형상모수와 척도모수에 이용 할 수 있기 때문에 신뢰성 분야에서 많이 사용되는 일반화 감마 분포[7]를 이용한 최적 방출시기에 관한 문제를 다루었다.

본 논문의 2절에서는 관련 연구로서 무한고장과 기록 값 통계량, 대수 포아송 실행시간모형 과 로그 파우어 모형에 대한 모수추정에 대하여 약술하였고 3장에서는 요구 신뢰도와 비용 최소화를 고려한 방출시간에 대하여 서술하고 4장에는 제안된 감마 분포 모형에 대하여 서술하고 5장과 6장에서는 각각 수치적인 예와 그 결론을 나열 하였다.

## II. 관련연구

### 2.1 무한고장 NHPP와 기록값 통계량

NHPP 모형에서 평균값 함수  $m(t)$  (Mean value function)와 강도 함수(Intensity function)  $\lambda(t)$ 는 다음과 같은 관계로 표현할 수 있다[1].

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

따라서  $N(t)$ 는 모수  $m(t)$ 을 가진 포아송 확률 밀도 함수(Probability density function; Pdf)로 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[8].

유한 고장 NHPP 모형들은 충분한 테스트 시간이 주어지면 결함들(Faults)의 기대값이 유한 값 ( $\lim_{t \rightarrow \infty} m(t) = \theta < \infty$ )을 가지고 반면에 무한 고장 NHPP 모형들은 무한 값을 가진다고 가정된다.

일반적으로 일반화된 순서 통계량(Generalized Order Statistics; GOS) 모형을 이용한 모형은 소프트웨어 테스트 시점에서 미지의  $N$ 개의 결함들을 가지고 있고 이  $N$ 개의 결함들로부터 임의의 확률 밀도 함수(Pdf)에 따라 발생한  $n$ 개의 순서 통계량이 고장 시점이 된다. 이모형은 각 수리 시점에서 새로운 결함이 발생하지 않는다는 가정을 한다. 그러나 실제 상황에서는 수리 시점에서도

고장이 발생할 수도 있다. 이러한 상황을 추가하기 위하여 기록 멈춤 통계량(Record breaking statistics)을 사용하는 RVS(Record Value Statistics)모형을 사용할 수 있다고 하였고 이 RVS 모형과 NHPP 모형에 관해서 평균값함수는 다음과 같이 된다고 하였다[1, 3].

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

즉,  $e^{-m(t)} = 1-F(t)$ . 따라서 (1)식 과 (3)식과 연관시키고  $f(t)$ 을 확률밀도함수,  $F(t)$ 을 분포함수라고 하면 다음과 같은 관계식에 의해 NHPP의 강도함수는  $F(t)$ 의 위험함수( $h(t)$ )가 된다.

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

결국 무한 고장 NHPP 모형의 평균값 함수와 고장 강도 함수는 각각 다음과 같이 유도된다[8].

$$m(t) = -\ln(1-F(t)) \quad (5)$$

$$\lambda(t) = m'(t) = f(t)/(1-F(t)) = h(t) \quad (6)$$

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

$$x_k = \sum_{i=1}^k t_i \quad (k=1, 2, \dots, n; 0 \leq x_1 \leq x_2 \leq \dots \leq x_n) \quad (7)$$

이라고 하고  $\theta$ 을 모수공간이라고 하면 우도 함수는 다음과 같이 알려져 있다[8].

$$L_{NHPP_{NFB}}(\theta | D_t) = \prod_{i=1}^n \left( \frac{f(x_i)}{1-F(x_i)} \right) (1-F(x_n)) \quad (8)$$

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

## 2.2 기존모형

### 2.2.1 대수 포아송 실행시간모형

대수 포아송 실행시간(Log Poission execution time [3-4])은 1984년에 Musa 와 Okumoto에 의해서 소개된 무한 고장 소프트웨어 모형으로 평균값함수와 강도함수는 다음과 같이 알려져 있다.

$$m(t) = \frac{1}{\theta} \ln(\lambda_0 \theta t + 1) \quad (9)$$

$$\lambda(t) = \frac{\lambda_0}{\lambda_0 \theta t + 1} \quad (10)$$

한편, (9)식과(10)식을 (8) 식에 대입하면 우도함수는 다음과 같이 유도 할 수 있다.

$$L(\theta, \lambda_0 | \underline{x}) = \left( \prod_{i=1}^n \frac{\lambda_0}{\lambda_0 \theta x_i + 1} \right) \cdot e^{-\frac{1}{\theta} \ln(\lambda_0 \theta x_n + 1)} \quad (11)$$

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

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

$$\ln L(\theta, \lambda_0 | \underline{x}) \quad (12)$$

$$= n \ln \lambda_0 - \ln \sum_{i=1}^n (\lambda_0 \theta x_i + 1) - \frac{1}{\theta} \ln(\lambda_0 \theta x_n + 1)$$

(12)식을 이용하여 최우추정치  $\hat{\theta}_{MLE}$  와  $\hat{\lambda}_{MLE}$ 은 다음과 같이 구할 수 있다고 하였다[5-6, 9].

$$\hat{\theta}_{MLE} = \frac{1}{n} \ln(\hat{\phi} x_n + 1) \quad (13)$$

$$\hat{\lambda}_{MLE} = \hat{\phi} / \hat{\theta}_{MLE} \quad (14)$$

단,  $\phi(=\hat{\lambda}_{0,MLE} \cdot \hat{\theta}_{MLE})$ 는 (15) 식의 근이 된다[3, 9].

즉, 이  $\phi$  근을 구하기 위해서는 수치 해석적 방법으로 다음과 같은 식을 이용하여 계산할 수 있다.

$$\frac{\partial \ln L(\phi | \underline{x})}{\partial \phi} = \frac{n}{\phi} - \sum_{i=1}^n \frac{x_i}{\phi x_i + 1} - \frac{n x_n}{(\phi x_n + 1) \ln(\phi x_n + 1)} = 0 \quad (15)$$

### 2.2.2 로그 파워어 모형

로그 파워어 (Log Power)모형[4, 10]은 1999년에 Xie와 Hong에 의해서 발전된 무한 고장 소프트웨어 모형으로 평균값함수와 강도함수는 다음과 같이 알려져 있다.

$$m(t) = a \ln^b(1+t) \quad (16)$$

$$\lambda(t) = \frac{a b \ln^{b-1}(1+t)}{1+t} \quad (17)$$

한편, (3)식과 (8)식을 이용하면 우도함수는 다음과 같이 유도 할 수 있다.

$$L(a, b | \underline{x}) = \left( \prod_{i=1}^n \frac{a b \ln^{b-1}(1+x_i)}{1+x_i} \right) \cdot e^{-a \ln^b(1+x_n)} \quad (18)$$

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

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

$$\ln L(a, b | \underline{x}) = n \ln a + n \ln b - (b-1) \ln \left( \sum_{i=1}^n \ln(1+x_i) \right) - \sum_{i=1}^n \ln(1+x_i) - a \ln^b(1+x_n) \quad (19)$$

(19)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.

$$\frac{\partial \ln L(a, b | \underline{x})}{\partial a} = \frac{n}{a} - \ln^b(1+x_n) = 0 \quad (20)$$

$$\frac{\partial \ln L(a, b | \underline{x})}{\partial b} = \frac{n}{b} - \ln \left( \sum_{i=1}^n \ln(1+x_i) \right) - a \ln^b(1+x_n) \ln(\ln(1+x_n)) = 0 \quad (21)$$

### 2.3 일반화 감마 분포에 기초한 NHPP모형

본 절에서는 일반화 감마분포[7]를 이용한 신뢰성 모형을 설명 하고자 한다. 우선 일반화 감마분포의 확률밀도 함수는 다음과 같다.

$$f_{GG}(t | k, \alpha, \beta) = \frac{\alpha \beta}{\Gamma(k)} (\alpha t)^{k\beta-1} e^{-(\alpha t)^\beta}, \quad t > 0, \alpha, \beta, k > 0. \quad (22)$$

따라서 분포함수는 다음과 같이 나타낼 수 있다.

$$F_{GG}(t | k, \alpha, \beta) = I(k, (\alpha t)^\beta) \quad (23)$$

$$\text{단, } I(k, x) = \frac{1}{\Gamma(k)} \int_0^x \mu^{k-1} e^{-\mu} d\mu.$$

따라서 위험함수는 다음과 같이 정의된다.

$$h_{GG}(t | k, \alpha, \beta) = \frac{f_{GG}(t | k, \alpha, \beta)}{1 - F_{GG}(t | k, \alpha, \beta)} \quad (24)$$

위 일반화 감마분포는 3가지 특성의 분포로 구분해 볼 수 있다. 형상모수(Shape parameter)가  $\beta=1$  인 경우는 감마분포  $Gamma(k, \alpha)$  가 되고  $k=1$  인 경우에는 와이블분포  $Weibull(\alpha, \beta)$  가 되고  $k \rightarrow \infty$  일 경우는 로그 정규분포(극한분포)가 된다[7].

본 연구에서는 감마분포  $Gamma(k, \lambda)$  일 경우와 와이블분포  $Weibull(\lambda, \beta)$  일 경우에 대해서만 고려하고자 한다.

### III. 신뢰도 및 비용최소화를 고려한 방출시간

NHPP 모형에서 테스트 시점  $x_n$ (마지막 고장시점)에서 소프트웨어 고장이 일어난다고 하는 가정 하에서 신뢰구간  $(x_n, x_n + x)$  (단,  $x$ 는 임무시간(Mission time))동안 소프트웨어의 고장이 일어나지 않을 확률인 신뢰도  $\hat{R}(x | x_n)$ 는 다음과 같이 됨이 알려져 있다[1, 8].

$$\hat{R}(x | x_n) = \exp\left(-\int_{x_n}^{x_n+x} \lambda(\tau) d\tau\right) = \exp[-\{m(x+x_n) - m(x_n)\}] \quad (25)$$

따라서 대수 포아송 실행시간모형에 대한 신뢰도는 평균값함수 (25)식과  $t = x_n$ 을 이용하면 다음과 같이 표현된다.

$$R(x | T_R) = \exp\left(-\frac{1}{\theta} [\ln(\lambda_0\theta(x+t)+1) - \ln(\lambda_0\theta t+1)]\right) \quad (26)$$

따라서 소프트웨어 방출시간  $T_R$ 이 신뢰도  $R_0 = \hat{R}(x | T_R)$ 을 확보해야 한다면 다음 방정식을 만족해야 한다.

$$\ln R_0 = -\frac{1}{\theta} [\ln(\lambda_0\theta(x+T_R)+1) - \ln(\lambda_0\theta T_R+1)] \quad (27)$$

단, 신뢰도  $R_0 = \hat{R}(x | T_R)$ . 비용 최소화와 관련된 최적 방출시간은 신뢰도와 함께 비용모형에 의해서 결정된다. 소프트웨어 방출시간을  $T$ 로 표현하고  $m(T)$ 와  $m(\infty)$ 을 각각  $(0, T)$ 와  $(0, \infty)$ 의 기간에 발견된 기대 고장수라고 표현하고  $C(T)$ 을 소프트웨어 라이프사이클(life cycle) 동안에 기대되는 소프트웨어 비용이라고 하면  $C(T)$ 는 다음과 같이 표현된다[10].

$$C(T) = c_1 m(T) + c_2 [m(\infty) - m(T)] + c_3 T \quad (28)$$

위 식에서  $c_1$ 는 테스트 동안에 하나의 고장을 수리하는 비용이고  $c_2$ 가동 중에 하나의 고장을 수리하는 비용 ( $c_2 > c_1$ ), 그리고  $c_3$ 는 단위 시간당 테스트 비용을 나타낸다. 이와 관련하여 총비용의 최소화는 무한고장 평균값 함수를 가진 NHPP 모형에 대하여 발생 할 수 있다. 무한 수명에 대한 비용함수  $C(T)$ 식인 (28)식에서  $m(\infty)$ 은 직접 추정 할 수 없기 때문에 이 식을 사용하기 위해서는 소프트웨어 수명시간 인  $T_{LC}$ 을 지정하여 분석한다[6]. 이러한  $T_{LC}$ 는 소프트웨어마다 서로 다른 임의의 값이기 때문에 유한 고장 NHPP 모형이라고 할 수는 없다. 따라서 비용함수를 고려하여 소프트웨어의 모든 수명에서 총비용을 최소화함으로서 최적 테스트 시간을 결정 할 수 있고 다음과 같은 식을 만족하면 비용함수  $C(T)$ 는 유일한 최소값을 가진다[6, 10].

$$\frac{dC(T)}{dT} = 0, \quad \frac{d^2 C(T)}{d^2 T} > 0 \quad (28)$$

결국 소프트웨어 지정된 수명  $T_{LC}$ 을 이용한 로그 포아송 실행시간모형 비용함수  $C(T)$ 는 다음과 같이 유도된다.

$$C(T) = c_1 m(T) + c_2 [m(\infty) - m(T)] + c_3 T = (c_1 - c_2) \frac{1}{\theta} \ln(\lambda_0\theta T + 1) + \frac{c_2}{\theta} \ln(\lambda_0\theta T_{LC} + 1) + c_3 T \quad (29)$$

$T$ 에 관해서 비용함수  $C(T)$ 을 미분하면 다음과 같은 방정식을 만족하는 최적방출시간  $T_C$ 를 계산 할 수 있다[10].

$$\frac{(c_1 - c_2)\lambda_0}{\lambda_0\theta T_C + 1} + c_3 = 0 \quad (30)$$

위 식에서 최적방출시간은 소프트웨어 지정 수명시간인  $T_{LC}$  와 의존하지 않는다는 것을 알 수 있다. 이러한 사실은 무한 고장 평균값 함수를 가진 NHPP모형들이 새로운 결점들이 발생함으로써 몇 개의 고장이 야기 될 수 있는 점을 고려한 모형으로 적합 시킬 수 있다. 실제로 만족할 만한 신뢰도가 부여되고 동시에 시스템 고장과 연계된 기대 총비용을 최소화시키기 위하여 필요하다면 충분한 테스트를 계속해야 한다. 따라서 신뢰성 요구를 만족하고 총 비용을 최소화하는 상황이 최적 방출 시간이다. 따라서 대수 포아송 실행시간모형을 사용한 최적 방출시간  $T_{OP}$  는  $T_R$  과  $T_C$  에 대하여 다음을 만족한다[10].

$$T_{OP} = \text{Max}(T_C, T_R) \quad (31)$$

(31) 식에서  $T_R$  과  $T_C$  는 다음 두 방정식에 의해서 계산 될 수 있다.

$$\ln R_0 = -\frac{1}{\theta} [\ln(\lambda_0 \theta (x + T_R) + 1) - \ln(\lambda_0 \theta T_R + 1)] \quad (32)$$

$$\frac{(c_1 - c_2) \lambda_0}{\lambda_0 \theta T_C + 1} + c_3 = 0 \quad (33)$$

유사한 방법으로 로그 파워어 모형에서도  $T_R$  과  $T_C$  는 다음 두 방정식에 의해서 계산 될 수 있다.

$$\ln R_0 = -a(\ln^b(1 + (x + T_R)) - \ln^b(1 + T_R)) \quad (34)$$

$$(c_1 - c_2) \frac{a b \ln^{b-1}(1 + T_C)}{1 + T_C} + c_3 = 0 \quad (35)$$

#### IV. 제안된 일반화 감마분포 모형

소프트웨어 신뢰성 분야에서 널리 사용되는 분포 중에서 일반화 감마분포는 형상(Shape) 및 척도(Scale)모수

의 값에 따라 다양한 분포를 표현할 수 있어 신뢰성 데이터분석에 가장 널리 사용된다. (22) 식에 대한 일반화 감마분포  $\text{Gamma}(k, \alpha, \beta)$  는 형상모수  $\beta=1$  인 경우는 감마분포  $\text{Gamma}(k, \alpha)$  가 되고  $k=1$  인 경우에는 와이블분포  $\text{Weibull}(\alpha, \beta)$  가 된다[7].

#### 4.1 감마 분포 모형

감마 분포의 확률밀도함수는 다음과 같이 정의된다 [7].

$$f(t) = \frac{\alpha (\alpha t)^{k-1}}{\Gamma(k)} \exp(-\alpha t), \alpha, k, t > 0 \quad (36)$$

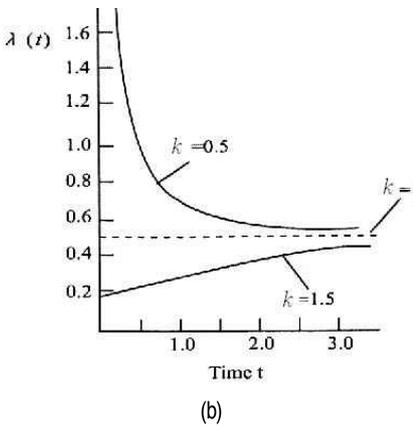
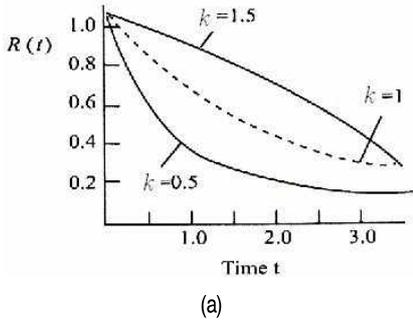
위 식에서  $k$  는 형상모수이고  $\alpha$  는 척도 모수이다. 형상모수  $k$  는 이 분포의 비대칭성(왜도, Skewness), 첨도(Kurtosis) 등 분포의 형상을 결정하는 모수로 중요한 위치를 차지하고 있다.  $k$  에 따른 감마분포의 신뢰도와 고장률은 <그림 1>에 나타내었다.

이러한 감마분포는  $k=1$  일 때 확률 밀도 함수는 모수  $\alpha$  을 갖는 지수 분포와 동일하고 고장률은 상수가 된다. 그리고 대기행렬 이론에서 서비스 시간과 도착간격 시간을 표현 하는데 자주 사용되어 지며  $k=n$  일 때( $n$  은 정수) 어랑 분포로 알려져 있으며 신뢰도 함수는 다음과 같다[7].

$$R(t) = \sum_{k=0}^{n-1} \frac{(\alpha t)^k}{k!} \exp(-\alpha t) \quad (37)$$

(37)식에서 본 논문에서 사용하고자 하는 어랑분포(Erlang distribution)모형은  $k=2$  인 특수한 경우로 확률 밀도함수는 다음과 같다.

$$f(t) = \alpha^2 t \exp(-\alpha t) \quad (38)$$



<그림 1> 감마분포의 신뢰도(a)와 고장률(b)

따라서 식(3)과 식(4)을 연관 시킨 무한 고장 NHPP 모형에 적용시킨 평균값 함수와 강도함수는 다음과 같다.

$$\lambda(t) = \alpha^2 t / (\alpha t + 1) \quad , \quad m(t) = \alpha t - \ln(1 + \alpha t) \quad (39)$$

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

$$\ln L(2, \alpha | \underline{x}) = 2n \ln \alpha - \sum_{i=1}^n \ln(\alpha x_i + 1) + \sum_{i=1}^n \ln x_i - \alpha x_n + \ln(1 + \alpha x_n) \quad (40)$$

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

(40)식을 이용하여  $\alpha$ 에 대하여 미분하고 그 값을 0으로 하는 최우추정치  $\hat{\alpha}_{MLE}$ 을 추정 할 수 있다. 따라서 추우추정치를 추정하기 (41)식은 선형이 아니므로 수치해석적 방법을 동원하면 최우추정치  $\hat{\alpha}_{MLE}$ 를 추정 할 수 있다.

$$\frac{2n}{\alpha} = \sum_{i=1}^n \frac{x_i}{\alpha x_i + 1} + x_n - \frac{x_n}{1 + \alpha x_n} \quad (41)$$

감마 분포 모형에서  $k=3$ 인 경우에는  $k=2$ 인 경우와 유사하게 (3)식과 (4)식을 이용하면 무한 고장 NHPP 모형에 적용시킨 평균값 함수와 강도함수는 다음과 같다.

$$\lambda(t) = \left( \frac{1}{2} \alpha^3 t^2 \right) / \left( 1 + \alpha t + \frac{\alpha^2 t^2}{2} \right) \quad (42)$$

$$m(t) = \alpha t - \ln \left( 1 + \alpha t + \frac{\alpha^2 t^2}{2} \right) \quad (43)$$

감마 분포 모형에서  $k=3$ 인 경우 최우추정법(MLE)을 이용하기 위한 로그 우도 함수는 다음과 같이 유도된다.

$$\begin{aligned} \ln L(3, \alpha | \underline{x}) &= -n \ln 2 + 3n \ln \alpha - \sum_{i=1}^n \ln \left( 1 + \alpha x_i + \frac{\alpha^2 x_i^2}{2} \right) \\ &\quad + \sum_{i=1}^n \ln x_i^2 - \alpha x_n + \ln \left( 1 + \alpha x_n + \frac{\alpha^2 x_n^2}{2} \right) \end{aligned} \quad (44)$$

(44)식을 이용하여  $\alpha$ 에 대하여 미분하고 그 값을 0으로 하는 최우추정치  $\hat{\alpha}_{MLE}$ 을 추정 할 수 있다. 따라서 추우추정치를 추정하기 (45)식은 선형이 아니므로 수치해석적 방법을 동원하면 최우추정치  $\hat{\alpha}_{MLE}$ 를 추정 할 수 있다.

$$\frac{3n}{\alpha} = \sum_{i=1}^n \frac{x_i + x_i^2 \alpha}{1 + \alpha x_i + \frac{\alpha^2 x_i^2}{2}} + x_n - \frac{x_n + x_n^2 \alpha}{1 + \alpha x_n + \frac{\alpha^2 x_n^2}{2}} \quad (45)$$

소프트웨어의 수명분포의 모수는 일반적으로 형상모수와 척도 모수로 이루어진다. 형상모수는 수명분포의 비대칭성, 첨도 등 분포의 형상을 결정하는 모수로 중요한 위치를 차지하고 있다. 본 논문에서는 감마분포에서 형상모수가 고정된 경우를 제안하고자 한다.

감마분포에서 형상모수가 2를 가지는 Erlang 분포 모형에 대한 신뢰도는 (25)식에서  $t = x_n$  라고 하면 다음과 같이 표현된다.

$$R(x | T_R) = \exp [-\alpha x + \ln(1 + \alpha(T_R + x)) - \ln(1 + \alpha T_R)] \quad (46)$$

따라서 소프트웨어 방출시간  $T_R$ 이 신뢰도  $R_0 = \hat{R}(x | T_R)$ 을 확보해야 한다면 다음 방정식을 만족해야 한다.

$$\ln R_0 = -\alpha x + \ln(1 + \alpha(T_R + x)) - \ln(1 + \alpha T_R) \quad (47)$$

단, 신뢰도  $R_0 = \hat{R}(x | T_R)$ .

최소비용 관련 방출시간은 (28)식과 (39)을 연관하면 다음과 같이 유도 된다.

$$(c_1 - c_2) \left( \alpha - \frac{\alpha}{1 + \alpha T_C} \right) + c_3 = 0 \quad (48)$$

따라서 Erlang 분포 모형을 사용한 최적 방출시간  $T_{OP}$ 는  $T_R$ 과  $T_C$ 에 대하여 다음을 만족한다[6, 10].

$$T_{OP} = \text{Max}(T_C, T_R) \quad (49)$$

유사한 방법으로 형상모수가 3인 분포 모형에 대한 신뢰도는 (25)식에서  $t = x_n$  라고 하면 다음과 같이 표현 된다.

$$\begin{aligned} \ln R_0 = & \quad (50) \\ & -\alpha x + \ln(1 + \alpha(T_R + x) + \frac{\alpha^2}{2}(x + T_R) \cdot (x + T_R)) \\ & - \ln(1 + \alpha T_R + \frac{\alpha^2}{2} \cdot T_R^2) \end{aligned}$$

단, 신뢰도  $R_0 = \hat{R}(x | T_R)$ .

최소비용 관련 방출시간도  $k=2$  인 경우와 유사하게 다음과 같이 유도 된다.

$$(c_1 - c_2) \left( \alpha - \frac{\alpha + \alpha^2 T_C}{1 + \alpha T_C + \frac{\alpha^2 T_C^2}{2}} \right) + c_3 = 0 \quad (51)$$

## 4.2 와이블 분포 모형

와이블 분포는 신뢰도 함수에 포함된 모수로서 다른 분포들을 표현하거나 데이터를 적용하기 쉽게 변형시킬 수 있다. 이러한 와이블 분포는  $\alpha(>0)$ 가 척도 모수,  $\beta(>0)$ 는 형상 모수일 때, 확률 밀도 함수와 분포함수는 각각 다음과 같다.

$$f_{Wei}(t | \alpha, \beta) = \alpha \beta (\alpha t)^{\beta-1} e^{-(\alpha t)^\beta} \quad (52)$$

$$F_{Wei}(t | \alpha, \beta) = 1 - e^{-(\alpha t)^\beta} \quad (53)$$

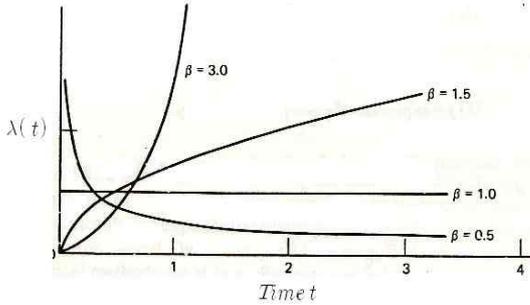
따라서 무한 고장 NHPP 모형의 평균값 함수  $m(t) = -\ln(1 - F(t)) = (\alpha t)^\beta$  가 되고 강도함수는 다음과 같이 표현된다.

$$\lambda(t) = m'(t) = f(t)/(1 - F(t)) = \alpha \beta (\alpha t)^{\beta-1} \quad (54)$$

이러한 와이블 분포는 많은 분포들에 적합시키는 데 사용될 수 있다고 하였고 다음과 같은 특성을 가지고 있다고 알려져 있다[11].

$0 < \beta < 1$  일 경우에는 (오류 수정 단계에서 발생하는)

감소하는 강도함수를 표현하고  $\beta=1$ 인 경우에는 (정상 운영 단계에서 발생하는) 일정한 강도함수를 나타내며  $\beta>1$ 인 경우에는 (마모 단계에서 발생하는) 증가하는 강도함수를 표현한다. 이러한 패턴을 <그림 2>과 같이 나타낼 수 있다.



<그림 2>  $\alpha=0.5$ 인 경우  $\beta$ 에 따른 강도함수 패턴

모수 최우추정법(MLE)을 이용하기 위한 로그 우도함수는 (5)식과 (8)식을 연관 하여 다음과 같이 유도된다.

$$\ln L(\alpha, \beta | x) = n \ln \beta + n \beta \ln \alpha + (\beta - 1) \sum_{i=1}^n \ln x_i - (\alpha x_n)^\beta \quad (55)$$

(55)식을 이용하여  $\beta$ 는 사전에 알고 있다고 가정하면 최우추정치  $\hat{\alpha}_{MLE}$ 는 다음과 같이 구할 수 있다[5, 11]

$$\hat{\alpha}_{MLE} = \left( \frac{n}{x_n^\beta} \right)^{\frac{1}{\beta}} \quad (56)$$

따라서 본 논문에서는 형상(shape)모수  $\beta$ 가 1.5, 2, 3인 경우를 선택하여 분석 비교하고자 한다.

신뢰성 요구를 만족하고 총 비용을 최소화하는 상황이 최적 방출 시간이다. 따라서 와이블 특성모형을 사용한 최적 방출시간  $T_{OP}$ 는  $T_R$ 과  $T_C$ 에 대하여 다음을 만족한다.

$$T_{OP} = \text{Max}(T_C, T_R) \quad (57)$$

(57) 식에서  $T_R$ 과  $T_C$ 는 (25)식과 (28)식을 이용하면 다음 두 방정식에 의해서 계산 될 수 있다.

$$\ln R_0 = -\alpha [(x + T_R)^\beta - (T_R)^\beta] \quad (58)$$

$$(c_1 - c_2) \beta \alpha^\beta T_C^{\beta-1} + c_3 = 0 \quad (59)$$

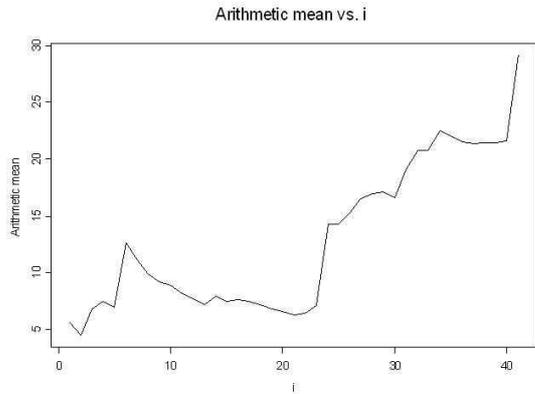
단, 신뢰도  $R_0 = \hat{R}(x | T)$ .

## V. 수치적인 예

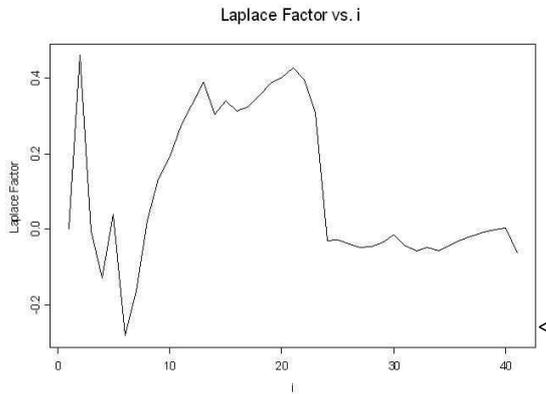
이 장에서 S27[12]가 인용한 고장 간격 시간 자료(Failure interval time data)를 가지고 와이블 특성 모형에 근거한 최적 방출시기를 분석하고자 한다. 이 자료는 1197.945 시간단위에 41번의 고장이 발생된 자료이며 <표 1>에 나열 되어 있고 제시하는 신뢰 모형들을 분석하기 위하여 우선 자료에 대한 추세 검정이 선행 되어야 한다 [12].

추세 분석에는 산술평균 검정(Arithmetic mean test)과 라플라스 추세 검정(Laplace trend test) 등이 있다. 이 검정을 실시한 결과 <그림 3>에서 산술평균 검정결과 고장시간이 증가함에 따라 산술 평균이 거의 증가 추세를 보이고 있으므로 신뢰성장(Reliability growth) 속성을 가지고 있고 <그림 4>의 라플라스 추세 검정의 결과도 라플라스 요인(Factor)이 -2와 2사이에 존재함으로써 신뢰성장(Reliability growth) 속성을 나타내고 있다. 따라서 이 자료를 이용하여 신뢰도와 소프트웨어 방출시기를 추정하는 것이 가능하다.

각 모형의 모수 추정값은 최우추정법을 이용하였고 비선형 방정식의 계산방법은 수치 해석적 기본 방법인 이분법(Bisection method)을 사용하였다. 이러한 계산은 초기값을 0.001 와 10 을, 허용 한계(Tolerance for width of interval)는  $10^{-10}$  을 주고 수렴성을 확인 하면



<그림 3> 산술평균 검정



<그림 4> 라플라스 추세 검정

서 충분한 반복 횟수인 100번을 C-언어를 이용하여 모수 추정을 수행하였다.

감마분포 모형은 형상모수  $k$ 가 정수(Erlang 분포 모형)인  $k$ 가 2, 3인 경우 모수  $\beta$ , 그리고 와이블 특성 모형은  $\beta=1.5$ 와  $\beta=2, \beta=3.0$  일 때를 고정하여 모수  $\alpha$ 을 추정 한 값이 <표 2>에 요약되었다.

<표 3>에서는  $c_1=5(\$)$ ,  $c_2=20(\$)$  그리고  $c_3=0.5(\$)$  라고 가정하고 시스템 수명시간은 2000시간이고 임무시간을  $x$ 을 1.5이고  $R_0$ 을 0.95(95%)를 투입하여 수치해석적 기본 방법인 이분법(Bisection method)을 사용하였다. 이러한 계산은 초기값을 0.1와 1000을, 허용 한계(Tolerance for width of interval)는  $10^{-10}$ 을 주고 수렴

<표 1> 고장 간격자료[12]

| Failure number | Failure Time(hours) | Failure Interval (hours) |
|----------------|---------------------|--------------------------|
| 1              | 5.649               | 5.649                    |
| 2              | 8.92                | 3.271                    |
| 3              | 20.29               | 11.37                    |
| 4              | 29.955              | 9.665                    |
| 5              | 34.715              | 4.76                     |
| 6              | 75.95               | 41.235                   |
| 7              | 78.171              | 2.221                    |
| 8              | 78.625              | 0.454                    |
| 9              | 83.022              | 4.397                    |
| 10             | 89.114              | 6.092                    |
| 11             | 89.804              | 0.69                     |
| 12             | 92.86               | 3.056                    |
| 13             | 93.66               | 0.8                      |
| 14             | 110.655             | 16.995                   |
| 15             | 111.988             | 1.333                    |
| 16             | 122.545             | 10.557                   |
| 17             | 127.045             | 4.5                      |
| 18             | 128.712             | 1.667                    |
| 19             | 128.99              | 0.278                    |
| 20             | 131.768             | 2.778                    |
| 21             | 131.829             | 0.061                    |
| 22             | 141.712             | 9.883                    |
| 23             | 164.212             | 22.5                     |
| 24             | 342.85              | 178.638                  |
| 25             | 356.144             | 13.294                   |
| 26             | 399.144             | 43                       |
| 27             | 446.494             | 47.35                    |
| 28             | 476.644             | 30.15                    |
| 29             | 497.144             | 20.5                     |
| 30             | 497.661             | 0.517                    |
| 31             | 591.161             | 93.5                     |
| 32             | 665.644             | 74.483                   |
| 33             | 686.444             | 20.8                     |
| 34             | 765.944             | 79.5                     |
| 35             | 772.977             | 7.033                    |
| 36             | 774.944             | 1.967                    |
| 37             | 791.561             | 16.617                   |
| 38             | 815.978             | 24.417                   |
| 39             | 837.145             | 21.167                   |
| 40             | 861.945             | 24.8                     |
| 41             | 1197.945            | 336                      |

성을 확인 하면서 충분한 반복 횟수인 100번을 C-언어를 이용하여 추정을 수행하였다. 각 모형에 대한 추정시간의 결과와 최적방출시간은 <표 3>에 요약되었다.

이 표에서 보여 주듯이 고려한 모형 중에서 어랑분포(감마분포 중에서 형상모수가 상수인 경우)에서  $k=3.0$ 인 경우와 로그파우어 모형이 최적 방출시간이 다른 모형에 비해 짧기 때문에 효율적으로 나타나고 있다. 그리고 어랑분포인 경우( $k=2.0, 3.0$ )와 와이블 분포  $k=1.5$ 인 경우는 기존의 로그 포아송 실행 시간 모형

<표 2> 각 모형의 모수 추정값

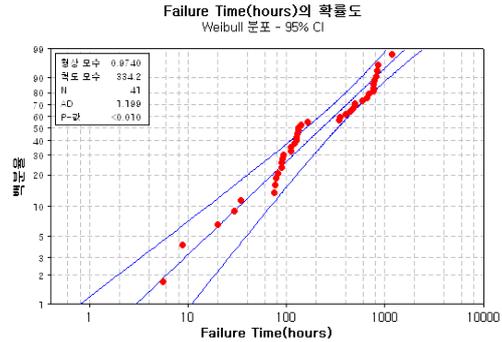
| Model                      | MLE                       |                              |
|----------------------------|---------------------------|------------------------------|
| Log Poisson execution time | $\hat{\theta} = 0.020031$ | $\hat{\lambda}_0 = 0.053067$ |
| Log Power                  | $\hat{a} = 16.0583379$    | $\hat{b} = 0.478584$         |
| Erlang                     | $k = 2.0$ (fixed)         | $\hat{\beta} = 0.068428$     |
|                            | $k = 3.0$ (fixed)         | $\hat{\beta} = 0.047951$     |
| Weibull                    | $\beta = 1.5$ (fixed)     | $\hat{\alpha} = 0.009925$    |
|                            | $\beta = 2.0$ (fixed)     | $\hat{\alpha} = 0.005345$    |
|                            | $\beta = 3.0$ (fixed)     | $\hat{\alpha} = 0.002878$    |

<표 3> 각 최적 방출시간( $R_0 = 0.95$ )

| Model                      | 추정시간  | 최적방출시간( $T_{OP}$ ) |
|----------------------------|---|--------------------|
| Log Poisson execution time | $\hat{T}_R = 518.4197$<br>$\hat{T}_C = 556.9223$                | 556.9223           |
| Log Power                  | $\hat{T}_R = 136.7676$<br>$\hat{T}_C = 102.5704$                | 136.7676           |
| Erlang                     | $k = 2.0$<br>$\hat{T}_R = 291.9701$<br>$\hat{T}_C = 13.8804$    | 291.9701           |
|                            | $k = 3.0$<br>$\hat{T}_R = 8.5369$<br>$\hat{T}_C = 112.7180$     | 112.7180           |
| Weibull                    | $\beta = 1.5$<br>$\hat{T}_R = 2.0426$<br>$\hat{T}_C = 505.1074$ | 505.1074           |
|                            | $\beta = 2.0$<br>$\hat{T}_R = 1.9053$<br>$\hat{T}_C = 583.3824$ | 583.3824           |
|                            | $\beta = 3.0$<br>$\hat{T}_R = 1.2511$<br>$\hat{T}_C = 682.7199$ | 682.7199           |

보다 효율적으로 나타나고 있다. 한편, 지수분포가 되는  $k=1$  과  $\beta=1$  인 경우는 고장률(위험함수)이 상수이기 때문에 방출시간을 추정하는 것이 의미가 없으므로 고려 대상에서 제외 하였다. 그리고 와이블 모형에서 신뢰도 측면에서의 방출시간은 너무 짧고 비용측면에서는 너무 길게 나타나 최적 방출시간은 비용측면 방출시간에 의존 하는 모형이 되었다. 그 이유는 적용한 자료에 대한 와이 블 분포 확률도를 그려본 결과 와이블 분포를 따르지 않

기 때문이다. 이러한 사실을 입증하기 위하여 미니탭 통계 패키지(14버전)[13]을 적용한 와이블 분포에 대한 확률도는 각각 <그림 5> 에 나타내었다.



<그림 5> 와이블 분포를 적용한 확률도

이 그림에서 Anderson-Darling(A-D) 검정을 실시한 결과 유의수준 5%에서 유의확률(P-값)이 0.05보다 작은 값(<0.010)이 나타나 통계적으로 유의하기 때문이다. 즉, 귀무가설( $H_0$ )이 와이블 분포를 따른다는 가정을 기각한다. 결과적으로 적용 자료가 와이블 분포를 따르는 것으로 판명날 경우에는 다른 모형보다 효율적인 모형이 될 수 있는 여지가 있다.

따라서 제안된 감마모형은 형상모수가 클수록 효율적인 모형이 되고 반대로 와이블 분포 모형은 형상모수가 작은 경우수록 효율적인 모형이 될 수 있었다. 이러한 감마모형 및 와이블 모형도 이 분야에 새로운 모형으로 선택 할 수 있음을 보여주고 있다.

## VI. 결론

본 연구는 형상모수를 고정시킨 일반화 감마 분포 모형을 적용한 무한고장 NHPP 모형을 이용하여 최적 방출시기에 관한 문제를 알아보았다. 즉, 대용량 소프트웨어가 수정과 변경하는 과정에서 결점의 발생을 거의 피

할 수 없는 상황이 현실이다.

실제로 만족할 만한 신뢰도가 부여되고 동시에 시스템 고장과 연계된 기대 총비용을 최소화되는 정보를 알기 위하여 필요하다면 충분한 테스트를 계속해야 한다. 따라서 신뢰성 요구를 만족하고 총 비용을 최소화하는 상황이 최적 방출 시간이다.

본 연구에서는 일반화 감마 분포모형을 적용한 방출 시기 추정을 시도하였다. 그 결과 감마모형은 형상모수가 클수록 효율적인 모형이 되고 반대로 와이블 분포 모형은 형상모수가 작은 경우수록 효율적인 모형이 됨을 알 수 있었다. 즉, 형상모수가 모수의 효율성을 판단하는데 중요한 역할을 하고 한다. 따라서 왜도와 첨도 측면에서 효율적인 분포로 알려진 카과분포, 지수화지수분포 등 업데이트된 분포에 대한 방출 시기 문제를 비교 분석하는 연구도 가치 있는 일이라 판단되고 이 연구를 통하여 소프트웨어 개발자들은 방출최적시기를 파악 하는데 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] 김희철, "지수화 지수 분포에 의존한 NHPP 소프트웨어 신뢰성장 모형에 관한 연구," *한국 컴퓨터 정보학회논문지*, 제11권 5호, 2006, pp. 9-18.
- [3] Musa, J. D and Okumoto, K, "A Logarithmic Poisson Execution Time Model for Software Reliability Measurement," *Proceeding the 7th International Conference on Software engineering*, 1984, pp. 230-238.
- [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] Yang, B. and Xie, M, "A study of operational and testing reliability in software reliability analysis," *RELIABILITY ENGINEERING & SYSTEM SAFETY*, Vol, 70, 2000, pp. 323-329.
- [6] Huang, C. Y, "Cost-Reliability-optimal release policy for software reliability models incorporating improvements in testing efficiency," *The journal of Systems and software*, Vol, 77, 2005, pp. 139-1555.
- [7] Lawless, J. F, "Statistical Models and Methods for Lifetime Data," *John Wiley & Sons*, New York, 1981.
- [8] Kuo, L. and Yang, T. Y, "Bayesian Computation of Software Reliability," *Journal of the American Statistical Association*, Vol. 91, 1996, pp. 763-773.
- [9] 김희철, "Log-Logistic 분포모형에 근거한 소프트웨어 최적방출시기에 관한 비교연구," *한국 컴퓨터정보학회 논문지*, 제13권7호, 2008, pp. 1-9.
- [10] Xie, M. and Homg, G. Y, "Software release time determination based on unbound NHPP model," *Proceeding of the 24th International Conference on Computers and Industrial Engineering*, 1999, pp. 165-168.
- [11] Musa, J. D, Iannino, A. and Okumoto, K, "Stware Reliability: Measurement, Prediction, Application," *McGraw Hill*, New York, 1987.
- [12] K. Kanoun and J. C. Laprie, "Handbook of Software Reliability Engineering," M. R. Lyu, Editor, chapter Trend Analysis. *McGraw-Hill New York, NY*, 1996.
- [13] 이레테크, "새 MINITAB 실무완성," *이레테크 미니탭 사업부*, 2005.

■ 저자소개 ■



김 재 우  
Kim, Jae Wook

2006년 3월~현재  
남서울대학교 전자공학과 교수  
2003년 8월 호서대학교 전자공학과 (공학박사)  
1998년 8월 호서대학교 전자공학과 (공학석사)  
1993년 2월 호서대학교 전자공학과 (공학사)

관심분야 : 칩인터더, 자성체 박막소자  
E-mail : jjwkim@nsu.ac.kr



김 회 철  
Kim, Heel Cheul

2005년 3월~현재  
남서울대학교 산업경영공학과 교수  
1998년 2월 동국대학교 통계학과 (이학박사)  
1992년 2월 동국대학교 통계학과 (이학석사)  
1991년 2월 청주대학교 응용통계학과  
(경제학사)

관심분야 : 소프트웨어 신뢰성공학, 웹 프로그  
래밍  
E-mail : kim1458@nsu.ac.kr

|                      |
|----------------------|
| 논문접수일 : 2010년 1월 27일 |
| 수 정 일 : 2010년 2월 16일 |
| 게재확정일 : 2010년 2월 25일 |