

## 소프트웨어 신뢰도 성장 모델의 일반형

### (A Generalized Form of Software Reliability Growth Models)

劉 宰 年 \*

(Jae-Nyun Yoo)

#### 요 약

이 논문에서 우리는 일정 기간 동안 소프트웨어 신뢰도 성장을 표현하는 non-homogeneous 모델에 대하여 미분 방정식 관점에서의 일반형

$$\frac{dN(t)}{dt} = b(t)f(N(t))$$

을 제시하고, 세 가지 잘 알려진 소프트웨어 성장 모델, Goel-Okumoto, S-자형, 그리고 Musa-Okumoto 모델들이 일반형의 각기 특수한 경우임을 보인다. 여기에서  $N(t)$ 는 발견되지 않은 결함 수를 나타내고,  $b(t)$ 는 고장 발생율을 의미한다. 일반형을 확장하여 확장 일반형을

$$\frac{dN(t)}{dt} = b(t, \gamma)f(N(t))$$

을 정의하고 고장 발생에 대한 분포가 주어지면 확장 일반형은 항상 구할 수 있음을 보였다. 고장 발생이 Weibull 밀도를 갖는 소프트웨어 신뢰도 모델이 확장 일반형임을 보인다. 또한 일반형을 참조로 하여  $b(t)$ 와  $\gamma$ 에 따라 세 가지 모델의 특징을 분류한다. 일반형을 이용한 사례연구를 보인다.

#### Abstract

We analyze the software reliability growth models for the specified period from the viewpoint of theory of differential equations. We define a generalized form of reliability growth models as follows:

$$\frac{dN(t)}{dt} = b(t)f(N(t)),$$

where  $N(t)$  is the number of remaining faults and  $b(t)$  is the failure rate per software fault at time  $t$ . We show that the well-known three software reliability growth models - Goel-Okumoto, S-shaped, and Musa-Okumoto model - are special cases of the generalized form. We, also, extend the generalized form into an extended form being

$$\frac{dN(t)}{dt} = b(t, \gamma)f(N(t)).$$

The generalized form can be obtained if the distribution of failures is given. The extended form can be used to describe a software reliability growth model having Weibull density function as a fault exposure rate. As an application of the generalized form, we classify three mentioned models according to the forms of  $b(t)$  and  $f(N(t))$ . Also, we present a case study applying the generalized form.

#### I. 서 론

\* 正會員, 韓國電子通信研究員, 交換技術研究團, 소프트웨어工學部

(ETRI, switching Technology Division, software Engineering Section)

接受日字:1998年2月9日, 수정완료일:1998年4月14日

소프트웨어 신뢰도 연구는 소프트웨어 개발의 단계 별로 내재하는 결함을 수정하는 과정의 추이를 체계적 확률론으로 모델화 하여 시험을 거쳐 얻은 고장 자료를 이용하여, 주어진 시점에서의 소프트웨어 신뢰도를

추정하고 앞으로의 신뢰도를 예측하며, 이를 바탕으로 시험의 방법과 시험의 효율성에 대한 조사와 가용 자원의 재배치, 최적의 배포시기 결정 등의 소프트웨어 개발 관리에 이용된다.

소프트웨어 개발과 관련된 결함의 발생과 제거 과정의 특징을 결정하는 간단한 몇 가지 가정 하에서 소프트웨어 신뢰도 성장에 대한 다양한 모델이 연구되어지고 사용되어 오고 있다. 이 논문에서는 일반적인 용어와 기본 사실은 [1], [2]의 내용을 참조로 한다. 이러한 모델들에서 취급하는 측정 가능한 양은 주로 시간  $t$  까지 발견되는 누적 결함 수  $\mu(t)$ 로서, 시험을 통하여 얻어진 결함 자료를 이용하여 Maximum Likelihood Method를 사용하여 모델 함수의 파라미터를 결정하는 방법을 이용한다.  $\mu(t)$ 에 대한 표현은 모델마다 다르게 주어진다. 세 가지 대표적이며 이 논문에서 다루고자 하는 모델의  $\mu(t)$ 는 아래와 같이 주어진다.

● Goel-Okumoto 모델 :  $\mu_{G(t)} = a(1 - e^{-bt})$ .

● S-자형 (Yamada) 모델 :  $\mu_{S(t)} = a(1 - (1 + bt)e^{-bt})$ .

● Musa-Okumoto 모델 :  $\mu_{M(t)} = a \ln(1 + bt)$ .

여기에서  $a, b$ 는 소프트웨어 성장 곡선의 표현에 사용되는 두 파라미터로서  $a$ 는 시험 초기의 소프트웨어 시스템에 내재하는 결함의 초기 값을 나타내고, 파라미터  $b$ 는 시험기간 동안의 결함 발견율을, 즉, 고장 발생율을 의미한다.

결함은 주로 소프트웨어의 디자인과 코딩 단계의 소프트웨어의 기능 정의의 불확실성과 설계, 개발자의 실수에 의해 주로 발생하므로  $a$ 는 소프트웨어 개발 인력의 개발 경험과 소프트웨어의 규모, 복잡성에 많이 영향을 받는다.  $a$ 의 값은 소프트웨어 개발 과제에서 얻은 경험적 결과로 추정된다.  $b$ 는 시험에서 얼마나 많은 결함을 검출하며, 발견하기 어려운 결함을 찾아낼 수 있는 가의 정도를 표시하는 것으로 주로 소프트웨어의 시험에 투입되는 인적 자원, 그리고 시험에 소요되는 시간 등의 시험자원에 영향을 받는다.  $a, b$ 의 값을 추정하는 데 이용되는 방법은 선형 수행 시간 近似 방법이 있다 [1].  $a, b$ 의 의미에 관한 자세한 해석은 [3], [4]에서 찾아 볼 수 있다.

다양한 소프트웨어 신뢰도 성장 모델들 사이의 특징을 비교하기 위하여 이 연구에서는 소프트웨어 신뢰도 성장 모델을 표현하는 미분 방정식의 일반형을 제시하

고 이 형의 특징을 이용하여 소프트웨어 신뢰도 성장 모델을 분류한다.

2절에서는 일반형을 정의하고, 3절에서 잘 알려진 세 가지의 소프트웨어 신뢰도 성장 모델을 2절의 일반형으로 표현한다.

4절에서 결함의 발견율이 시간의 함수일 때,  $b(t)$ 가 다른 매개변수에 따라 변하는 경우에 대해 일반형을 확장하고, 5절에서  $b(t)$ 와 결함 검출에 대한 확률 밀도 함수와의 관계를 설명하고, 6절에서 사례 연구를 설명하고, 7절에서 결론을 내린다.

## II. 소프트웨어 신뢰도 성장 모델의 일반형

시간  $t$ 에서 소프트웨어에 남은 평균 결함 수를  $N(t)$ 라 하면 시간  $t$ 까지 발견되는 평균 결함수는

$$\mu(t) = N(0) - N(t)$$

가 된다.  $N(0)$ 는  $t=0$ 일 때, 즉, 시험의 초기에 소프트웨어에 내재하는 결함 수를 의미한다.

소프트웨어 신뢰도 성장 모델은  $N(t)$ 의 시간변화율과  $N(t)$ 와 결함 검출율  $b(t)$ 의 관계를 나타내는 미분방정식으로 표현될 수 있다. 시간  $t$ 까지 잔존 결함수의 시간 변화율은  $N(t)$ 와 결함 검출율  $b(t)$ 의 곱에 비례한다는 일반적인 가정에 따라

$$\frac{dN(t)}{dt} = -b(t)N(t) \quad (1)$$

와 같이 일반적으로 표현할 수 있다. 우리는 (1)식을 **소프트웨어 신뢰도 성장 모델의 일반형**이라 부르기로 한다. 여기에서  $b(t)$ 는 시간에 대한 결함의 검출율을 의미하며  $f(N(t))$ 는  $N(t)$ 을 변수로 하는 임의의 함수이다.

소프트웨어의 결함은 시험을 거치는 동안 발견되는 고장에 의해 발견되고 수정되어 진다. 따라서 (1)식은

$$\frac{d\mu(t)}{dt} = -\frac{dN(t)}{dt} = b(t)f(N(t)) = b(t)f(N(0) - \mu(t))$$

로 시간  $t$ 에서  $\mu(t)$ 의 시간 변화율은 결함 검출율  $b(t)$ 와  $N(0)$ ,  $\mu(t)$ 에 종속적임을 의미한다.  $N(0) = a$ 라 하면 소프트웨어의 신뢰도 성장 곡선  $\mu(t)$ 는 결함의 발견과 수정 과정을 정량화 하는 매개변수  $a$ ,  $b(t)$ 에 의해 결정된다.

### Ⅲ. 3가지 소프트웨어 신뢰도 성장 모델의 일반형

이제 소프트웨어 신뢰도 성장 모델 이론에서 가장 잘 알려진 세 가지 모델에 대하여 (1)식과 같은 미분 방정식 관점에서의 해석을 한다.

#### 1. Goel-Okumoto 모델

먼저 소프트웨어 신뢰도 성장 모델의 가장 기본적인 Goel-Okumoto(지수형) 성장 모델에 대하여 살펴본다 ([1]에서는 기본형으로 불린다). Goel-Okumoto 모델에서는  $b(t) = -b$ 로 ( $b$ 는 상수), 그리고 (1)의  $J$ 는 항등 함수이다. 이 경우, (1) 식은 간단히 아래와 같이 표현된다.

$$\frac{dN(t)}{dt} = -bN(t). \quad (2)$$

(2) 식을 완전 미분 형식으로 표시하면

$$\frac{dN(t)}{N(t)} = -b dt$$

와 같다. 이것을 적분하면

$$nN(t) = -\int_0^t b ds + c,$$

$c$ 는 적분상수이다. 위 식을  $N(t)$ 로 표시하면

$$N(t) = Ce^{-bt}.$$

이제 초기 조건  $N(0) = a$ 라 하면  $C = a$  이므로

$$N(t) = ae^{-bt} \quad (3)$$

을 얻는다.

$$N(0) - N(t) = a - ae^{-bt} = a(1 - e^{-bt}) = \mu_G(t)$$

임을 쉽게 확인할 수 있다.

#### 2. S-자형 모델

S-자형 모델은 Yamada 등에 의해 제안된 모델로서 결함 검출율은 상수가 아니고 시간의 함수로

$$b(t) = -\frac{b^2 t}{1 + bt}$$

로 주어진다 [5]. 식 (1)의  $J$ 는 Goel-Okumoto 모델과 마찬가지로 항등 함수로 가정된다. 이 모델은 Goel-Okumoto 모델을 결함 검출율의 시간에 대한

의존성을 고려하여 확장한 것이다. 위의  $b(t)$ 를 적용하여 (1) 식의 형으로 나타내면 아래와 같다.

$$\frac{dN(t)}{dt} = -\frac{b^2 t}{1 + bt} N(t) \quad (4)$$

(4) 식을 완전 미분 형식으로 표시하면,

$$\frac{dN(t)}{N(t)} = -\frac{b^2 t}{1 + bt} dt.$$

위 식을 적분하면

$$\begin{aligned} \ln N(t) &= -\int_0^t b \frac{bs}{1 + bs} ds + c \\ &= -b \int_0^t \left(1 - \frac{1}{1 + bs}\right) ds + c \\ &= -b t + \ln(1 + bt) + c. \end{aligned}$$

$c$ 는 적분 상수이다. 따라서

$$\begin{aligned} N(t) &= Ce^{-bt} e^{\ln(1 + bt)} \\ &= C(1 + bt)e^{-bt}. \end{aligned}$$

$N(0) = C = a$ 라 하면

$$N(t) = a(1 + bt)e^{-bt}. \quad (5)$$

이를 이용하여,

$$N(0) - N(t) = a(1 - (1 + bt)e^{-bt}) = \mu_S(t)$$

임을 확인할 수 있다.

#### 3. Musa-Okumoto 모델

이 모델은 무한 시간 동안 발생하는 고장 수를 무한한 것으로 가정하는 모델이지만, 일정 시간 동안(예를 들면, 시스템시험 기간) 발생하는 전체 고장을 유한 것으로 가정할 수 있다. 일정 기간 동안 이 모델의 누적 고장 수를 식 (1)의 일반형으로 표시하기 위하여

$$n(t) = \frac{N(t)}{N(0)}$$

을 도입한다.  $n(t)$ 는 시간  $t$  까지 발견되지 않은 소프트웨어의 결함 수  $N(t)$ 와 초기 결함 수  $N(0)$ 의 비율 나타낸다.  $n(t)$ 를 이용하면 Musa-Okumoto 모델의 일반형은 아래와 같이 표현된다.

$$\frac{dn(t)}{dt} = -be^{n(t)-1}. \quad (6)$$

이 모델은 결함 검출율은 상수  $b$ 이고,  $f(x) = e^{x-1}$ 로 가정된다. (6)의 의미는 소프트웨어의 나머지 결함의 시간 변화율은  $t$  시점에서의 남은 결함 수와 초기 결함 수와의 비에 지수적으로 의존한다는 가정에 따른 것임을 알 수 있다. 즉, 결함 검출율은 상수이지만 남은 결함수의 시간 변화율이 남은 결함 수의 지수에 따라 변하는 것을 모델 화 한 것으로 볼 수 있다. 이러한 면에서, Musa-Okumoto 모델은 Goel-Okumoto 모델과 Yamada 모델과 구별된다.

(6)의 해를 구하기 위하여 (6) 식을 완전 미분 형식으로 표현하면

$$d(e^{-n(t)}) = \frac{b}{e} dt$$

이를 적분하면

$$e^{-n(t)} = \frac{b}{e} t + c,$$

여기에서  $c$ 는 적분 상수이다.

위 식에 초기조건  $n(0) = N(0)/N(0) = 1$ 을 적용하면  $c = 1/e$ 이 된다. 따라서

$$n(t) = 1 - \ln(1 + bt).$$

이제  $N(0) = a$  하고 위 식을  $N(t)$ 로 변환하면,

$$N(t) = a n(t) = a (1 - \ln(1 + bt)) \tag{7}$$

을 얻는다.

$\mu_M(t) = a \ln(1 + bt)$ 임을 상기하면 (7)은  $N(t) = a - \mu_M(t)$ 임을 확인할 수 있다.

#### IV. 소프트웨어 신뢰도 모델의 확장

결함 검출율은 시험에서 발견하는 고장 수에 따라 변하므로, 시험에서 고장의 발견을 위해 시험에 투입된 시간과 노력을 표시하는 양을 고려하여 기술할 수 있다. 이제  $b(t)$ 가 시험에 소요되는 시간과 시험에 투입되는 자원의 특징을 표시하는 유한개의 파라미터에 따라 변하는 경우를 살펴보기로 한다. 이런 경우 소프트웨어 신뢰도의 일반형 미분 방정식 (1)은 아래와 같이 확장된 표현을 갖는다.

$$\frac{dN(t)}{dt} = b(t)\gamma f(N(t)). \tag{8}$$

위의 (8)식은 결함 검출율에 대한 다른 요소의 영향을 (예를 들면 시험의 효과)  $\gamma$  파라미터로서 표현한 것이다. 우리는 식 (8)을 **소프트웨어 신뢰도 성장 모델의 일반형의 확장**이라 부르기로 한다. 만약 결함 검출율이 하나의 파라미터  $\gamma$ 에 대하여 Weibull 분포를 따른다면,

$$b(t, \gamma) = -b\gamma t^{\gamma-1}, \quad \gamma > 0 \tag{9}$$

와 같이 주어지고, 2. 절의 미분 방정식의 해법을 적용하면,

$$N(t, \gamma) = ae^{-bt^\gamma} \tag{10}$$

을 얻는다. 여기에서,  $\gamma = 1$  이면  $b(t, 1) = -b$  이므로  $N(t, 1)$  식은 Goel-Okumoto 모델의 잔존 결함을 나타내는 (3)식과 같다.  $\gamma > 1$ 이면 모든  $t$ 에 대하여  $N(t, \gamma) < N(t, 1)$  이므로  $\gamma > 1$  일 때의 잔존 결함 수는 Goel-Okumoto 모델 보다 적다. 이는 시험의 효과가 보통의 경우 ( $\gamma = 1$ )보다 높은 것을 나타낸다.  $0 < \gamma < 1$  경우,  $N(t, \gamma) > N(t, 1)$ 이 되므로 시험의 효과가 참조 경우보다 낮음을 보이는 것이다. 이 모델은 시험의 투입되는 자원의 효과를 분석하는 데 이용된다.

#### V. 신뢰도 성장 매개변수 $b(t)$ 의 해석

결함 발견의 확률 분포 함수가 시간  $t$ 에 대하여  $F(t)$ 이면  $b(t)dt$ 는 시간  $t$ 까지 소프트웨어 결함이 발생하지 않은 상황에서 구간  $(t, t + dt)$  동안 결함이 검출될 조건부 확률을 나타낸다. 따라서

$$b(t)dt = \frac{P\{t < x < t + dt\}}{P\{x > t\}} = \frac{F(t)dt}{1 - F(t)}$$

이다. 따라서 시간  $t$ 에서 결함 검출 hazard율은

$$b(t) = \frac{F(t)}{1 - F(t)} \tag{11}$$

로 주어진다. 따라서 일정 기간 동안 소프트웨어 고장 발생에 대한 분포가 알려지면 (10)식으로부터 항상 소프트웨어 신뢰도 성장의 확장 일반형을 구할 수 있다.

(1) 혹은 (8) 식은 남은 고장수의 변화율을(발견 고장수의 변화율과 같은 변화율을 갖는다) 이용하므로 소프트웨어 시험 시기의 초기에 소프트웨어 고장 데이터가 충분히 주어지지 않더라도 고장 발생을  $b$ 를 추

정하는 데 이용될 수 있다.

먼저 Weibull 분포에 대하여 (8)식의  $b(t)$ 을 구하기로 하자.

1. Weibull 분포의  $b(t)$   
Weibull 분포의 밀도 함수는

$$f(t) = b\gamma t^{\gamma-1} e^{-bt^\gamma}$$

로 주어지고, 분포 함수는

$$F(t) = 1 - e^{-bt^\gamma}$$

이다. 따라서

$$b(t) = \frac{f(t)}{1-F(t)} = b\gamma t^{\gamma-1}$$

이는 식 (9)와 일치한다.

2. Goel-Okumoto 모델과 Musa-Okumoto 모델의  $b(t)$

이제, Goel-Okumoto 모델과 Musa-Okumoto 모델에 사용되는 상수  $b$ 의 해석을 생각한다. 결합 검출의 밀도를

$$f(x) = be^{-bx}$$

로 가정하면(exponential density), (11)에 따라

$$b(t) = \frac{be^{-bt}}{1-(1-e^{-bt})} = b$$

와 같다.

Goel-Okumoto 모델과 Musa-Okumoto 모델은 고장의 발생 밀도 함수를  $f(x) = be^{-bx}$ 로 갖는다는 같은 가정을 사용하고 있음을 알 수 있다. 이런 면에서 두 모델은 결합의 검출에 관하여 같은 가정을 사용하는 것으로 분류할 수 있다.

3. S-자형 모델의  $b(t)$

이제, S-자형 모델에 사용되는  $b(t)$ 의 해석을 생각한다.

확률 밀도 함수가

$$f(x) = b^2 x e^{-bx}$$

이면 (즉, Gamma density)

$$F(x) = (1 - bxe^{-bx} - e^{-bx}).$$

따라서, (11)에 의해

$$b(t) = \frac{b^2 t e^{-bt}}{bte^{-bt} + e^{-bt}} = \frac{b^2 t}{1 + bt}.$$

S-자형 모델은 결합의 hazard율로서 gamma density를 가정하고 있음을 알 수 있다. 이런 면에서, S-모델은 Goel-Okumoto, Musa-Okumoto 모델과 차이점을 보인다.

표 1. 매 월 발생 고장 수

Table 1. Number of failures per month.

Month	1	2	3	4	5	6	7	8	9
고장수	15	12	42	91	69	79	34	37	34

### VI. 사례 연구

이 논문에서 제안한 일반형을 Hanbit ACE 64 ATM 교환 시스템 소프트웨어에 대한 고장 데이터에 대하여 적용하여  $b(t)$ 를 구한다. 이 소프트웨어는 SV3.4로 불리며, 시험 초기 단계에 있다. 그 규모는 약 850 K 소스라인 정도이다. 약 9 개월 동안 발생한 고장 수는 표 1과 같다.

이 고장 자료는 일정 기간 동안 발생한 고장 수이므로, 고장 수를 이용하는 소프트웨어 신뢰도 성장 모델, Goel-Okumoto 모델과, Yamada 모델 만을 생각하기로 한다(Musa-Okumoto 모델은 고장 사이의 시간을 측정 자료로 이용한다). 이를 forward difference의 Log 값을 이용하여 G-O 모델의  $b(t) = b$ 와 Yamada 모델의  $b(t) = \frac{b^2 t}{1 + bt}$ 로 least square approximation을 이용하여 nonlinear curve fitting 시키면  $b(t) = b = 0.136385$ 와  $b(t) = \frac{0.0266025t}{1 + 0.163103t}$ 를 얻는다.

한편, 고장의 초기치, 즉 궁극적으로 발견되는 총 고장수를  $850 = 850K \times 1/K$ 로 가정하면(이예측치는 TDX-ISDN 교환 시스템의 고장 자료를 토대로 추정한 것이다 [6]) S-자형 모델이 현재의 시스템 시험에서의 고장 발견을 가장 잘 설명한다는 것을 알 수 있다(아래 그림 1 참조). 이로부터 소프트웨어 신뢰도 성장을 분석하는 모델 선택과 이 모델에서 사용하는 매개변수들, 소프트웨어에 내재하는 초기의 결합수와 시험에서의 결합 검출율을 쉽고 정확하게 추정할 수 있다.

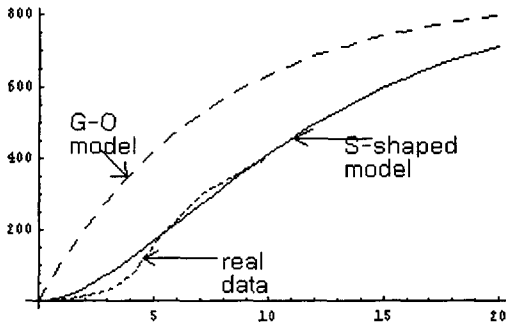


그림 1. 실제 누적 고장 수와 G-O, S-자형 모델의 예측 고장수 그래프의 비교

Fig. 1. Comparison of the real cumulative failures and the predicted graph of G-O and S-shaped models.

## VII. 결 론

우리는 두개의 매개변수를 갖는 일정 기간 동안 소프트웨어 신뢰도 성장 모델의 일반적인 형을 식 (1)과 같은 상 미분 방정식으로 정의하고 이를 이용하여 지금까지 잘 알려져 있고 널리 사용되는 Goel-Okumoto, S-자형, 그리고 Musa-Okumoto 모델들을 일반형으로 표시됨을 보였다. 또한 일반화 모델을 확장하여 결함 검출율이 유한한 파라미터에 의존하는 경우에 대해 살펴보았다. 이 확장 일반형은 고장발생에 대한 분포 함수가 주어진 경우에는 항상 구할 수 있다.

소프트웨어 신뢰도 성장 모델의 일반형, 식 (1)을 참조로 하여 위에서 살펴 본 세 가지 대표적인 소프트웨어 신뢰도 성장 모델을 분류할 수 있다. (1) 식의 오른쪽 편의  $f$ 의 모양에 따라 Goel-Okumoto 모델과 S-자형 모델은 같은 것으로 분류할 수 있다. 그러나  $b(t)$ 가 상수인지의 여부에 따라서는 Goel-Okumoto

모델과 Musa-Okumoto 모델을 같은 범주로 분류할 수 있다.

일반형 식을 이용할 경우, 소프트웨어 시험의 초기에 발생하는 고장 자료에 대해 소프트웨어 신뢰도 성장의 연구에 있어 모델의 선택과 모델 분석에 사용되는 매개 변수를 쉽고 정확하게 선택할 수 있음을 보였다. 일반형은 소프트웨어 신뢰도 성장 모델간의 비교를 쉽게 구별할 수 있게 하여 소프트웨어 신뢰도 성장 연구에 하나의 통일적인 방법을 제공할 수 있을 것이다.

## 참 고 문 헌

- [1] J. Musa, A. Lannino, and K. Okumoto, "Software Reliability : Measurement, Prediction, Application", McGraw-Hill, 1987.
- [2] Michael R. Lyu, "Handbook of Software Reliability Engineering", McGraw-Hill, 1995.
- [3] Y. K. Malaiya, A. von Mayrhauser, and P. K. Srimani, "An Examination of Fault Exposure Ratio," IEEE Transactions on Software Engineering, vol. 19, no. 11, Nov. 1993, pp. 1087-1094.
- [4] L. Naixin and Y. K. Malaiya, "Fault Exposure Ratio Estimation and Applications," ISSRE '96, pp. 372-381.
- [5] S. Yamada and S. Osaki, "Nonhomogeneous Error Detection Rate Models for Software Reliability Growth," Stochastic Models in Reliability Theory, Springer-Verlag, Berlin, 1984, pp. 120-143.
- [6] J. N. Yoo, et al, "기능 블록으로 구성된 대형 교환 소프트웨어의 신뢰도 성장," 전자공학회논문지, 35권, S편, 1호, 1998. 1

## 저 자 소 개

劉宰年(正會員) 第35卷 S編 第1號 參照

현재 전자통신연구원 교환단 선임연구원