

# 결함 중요도 단계를 고려한 소프트웨어 신뢰도 성장 모델에 관한 연구

## (A Study on Software Reliability Growth Modeling with Fault Significance Levels)

신 경 애\*  
(Kyung-Ae Shin)

### 요 약

소프트웨어 개발 과정에서 시스템 내에 잔존하는 결함을 발견하거나 수정하기 위해 테스트를 실시한다. 테스트 단계에서 결함을 발견하고 소프트웨어 신뢰성을 평가할 수 있다. 수리적으로 신뢰성을 평가할 수 있는 모델이 소프트웨어 신뢰도 성장 모델이다. 이 모델의 대부분은 결함의 형태가 하나이고 결함율은 일정하다는 가정에서 진행되고 있다. 본 연구에서는 테스트 단계에서 발견되는 결함이 일정하지 않다라는 관점에서 새로운 모델을 제안하고 결함 데이터를 적용해보았다. 또한 기존의 모델과 비교 및 분석하여 타당성을 증명하였다.

### ABSTRACT

In general, software test is carried out to detect or repair errors in system during software development process. Namely, we can evaluate software reliability through collecting and removing the faults detected in testing phase. Software reliability growth model evaluates reliability of software mathematically. Many kinds of software reliability growth modeling which modeling the processes of detecting, revising and removing the faults detected in testing phase have been proposed in many ways. and, it is assumed that almost of these modeling have one typed detect and show the uniformed detection rate. In this study, significance levels of the faults detected in test phase are classified according to how they can affect on the whole system and then the fault detection capability of them is applied. From this point of view, We here by propose a software reliability growth model with faults detection capability according considering fault significance levels and apply some fault data to this proposed model and finally verify its validity by comparing and estimating with the existing modeling.

### 1. 서론

컴퓨터 시스템의 고장은 하드웨어적인 것 보다 소프트웨어적인 원인이 더 많이 있다[1]. 소프트웨어적인 원인을 찾고 개발하기 위하여 소프트웨어 공학 기술을 이용하여 신뢰성이 높은 소프트웨어를 개발하는 것이 중요한 부분이다.

이런 부분을 해결하기 위하여 대부분 테스트 단계를 거친다. 테스트는 소프트웨어 개발 과정에 다 적용할 수 있지만 가장 효율적으로 신뢰성을 평가할 수 있는 단계가 주로 마지막 단계에서 이루어진다. 일반적으로 소프트웨어 개발 과정에서 시스템 내에 잔존하는 결함을 발견하거나 수정하기 위해 테스트를 실시한다. 테스트는 개발 과정의 진행 사항이나

\* 중신회원 : 동주대학 컴퓨터정보기술공학부 교수

논문접수 : 2002. 5. 16.

심사완료 : 2002. 6. 3.

※ 본 논문은 2001학년도 동주대학 학술연구조성비에 의해 연구되었음.

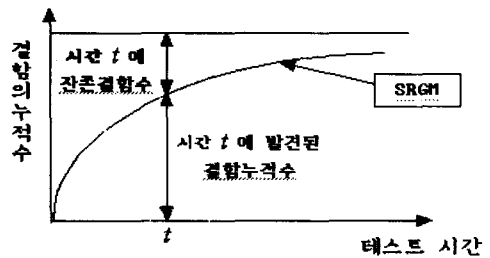
개발의 결과를 정량적인 척도에 의하여 파악할 수 있어야한다. 지금까지 널리 알려진 방법은 소프트웨어 신뢰도 성장 모델(SRGM : Software Reliability Growth model)을 가지고 정량적인 척도를 파악하는 연구가 많이 진행되고 있다[2][3][4][5][6][7]. 이것은 테스트 단계에서 소프트웨어 고장이나 잔존 결함수를 추정하여 신뢰성 목표나 릴리즈 이후에 운용단계에서 신뢰성을 예측하는 것이다. 소프트웨어 신뢰성 평가를 SRGM에 의해 정량적으로 평가하기 위하여 소프트웨어 내에 잔존하는 결함수와 소프트웨어 고장 발생시간을 가지고 한다[4][6]. 즉, 소프트웨어 신뢰성이 높은 소프트웨어는 소프트웨어 내에 잔존하는 결함수가 적고, 또한 소프트웨어 고장 발생시간 간격이 긴 것을 말한다[1][2]. 지금까지는 단일의 소프트웨어 결함수 또는 고장발생 시간을 가지고 헤자드 레이드 모델, NHPP 모델, 통계적 해석 모델, 초기하 신뢰도 모델등 여러 가지 SRGM이 제안 및 연구되었다[4]. 본 연구에서는 기존의 SRGM에서 적용한 결함과는 달리 테스트 단계에서 발견되는 결함이 시스템에 미치는 영향에 따라 4가지로 분류(중요도1, 중요도2, 중요도3, 중요도4)하여 적용하고자한다. 즉, 결함의 분류에 따라서 소프트웨어 고장을 일으키는 정도가 다르고 결함이 발견되는 중요도도 다르다라는 관점에서 살펴보고자 한다. 본 연구의 구성은 2장에서 SRGM모델과 NHPP모델 그리고 결함 난이도에 대하여 고찰하고, 3장에서는 발견되는 결함의 중요도에 따라 소프트웨어 고장을 발생시키는 영향이 다르다는 관점에서 새로운 SRGM을 제안하고, 또한 제안된 모델에 결함데이터를 적용하여 타당성을 평가하고자한다. 4장은 결론 및 향후과제 이다.

## 2. 관련연구

### 2.1 소프트웨어 신뢰도 성장 모델

소프트웨어 신뢰성 평가는 테스트 단계에서 관측된 데이터를 가지고 신뢰도 성장이라는 관점에서 논의 될 수 있다. 테스트 시간이 경과함에 따라 소프트웨어 내에 잔존하는 결함은 발견되고 수정되어 고장의 발생 확률이 감소하여 소프트웨어 신뢰도는 증

가하며 소프트웨어 고장 발생 시간이 점점 길어진다. 이것을 수리적으로 표현한 것이 SRGM이다. 일반적인 SGRM은 [그림 1]과 같다[2][3][4][5][6]. SRGM은 1972년에 Jelinski와 Moranad가 처음 제안하였다. 그 후에 Wolverton, Goel, Okumoto, Yamada등에 의해 연구 되었다. SRGM은 크게 시간계측모델, 개수계측모델, 가용시간모델, 경향곡선모델로 분류한다 [4][5].



[그림 1] 소프트웨어 신뢰도 성장 모델

[Fig. 1] Software Reliability Growth Model

### 2.2 NHPP 모델

NHPP(NonHomogeneous poisson process)모델은 개수계측 모델로서 소프트웨어 개발의 테스트 단계에서 발견되는 결함수나 발생하는 고장수를 측정하여, 이것을 확률변수  $N(t)$ 에 적용하여 포아송 과정을 가정하는 SRGM이다. NHPP 모델을 사용하기 위해서는 다음과 같이 가정한다[4][5][6].

- $N(0) = 0$  즉, 테스트 시작 시각 0 에서는 결함은 발견되지 않는다.
- $\{N(t), t \geq 0\}$ 는 독립 증분을 갖는다. 즉, 서로 다른 테스트 시간 구간에서 발견된 결함수는 통계적으로 독립이다.
- $\Pr\{N(t+\Delta t) - N(t) \geq 2\} = 0(\Delta t)$ 이다. 즉, 임의의 테스트 시간 구간  $\Delta t$ 에서는 2개 이상의 결함이 발견될 확률은 거의 없다.
- $\Pr\{N(t+\Delta t) - N(t) = 1\} = h(t)\Delta t + 0(\Delta t)$ 이다. 즉, 임의의 테스트 시간 구간  $\Delta t$ 에서 1개의 결함이 발견될 확률은 발견율 및 소프트웨어 고장율에 비례한다.

이상과 같은 가정으로 소프트웨어 결함의 누적수를 산출할 수 있는 NHPP는 식(1)과 같다.

$$\Pr\{N(t) = n\} = \frac{\{H(t)\}^n}{n!} \exp[-H(t)] \quad (N=0, 1, 2, 3, \dots) \quad \text{식(1)}$$

여기서,  $H(t)$ 는 평균치 함수이다. 즉, 테스트 시간 구간  $(0, t]$ 에 발견될 총 기대 결함수 또는 발생할 총 기대 소프트웨어 고장 수이다. 그리고 임의의 테스트 시간  $t$ 에서 소프트웨어 내의 잔존 결함수를  $NA(t)$ 라고 할 때 식(2)와 같다.

$$NA(t) = N(\infty) - N(t) \quad \text{식(2)}$$

또한,  $NA(t)$ 의 기대치 및 분산은 식(3)과 같다.

$$n(t) = E[NA(t)] = \alpha - H(t) = \text{Var}[NA(t)] \quad \text{식(3)}$$

여기서,  $\alpha$ 는 테스트를 시작하기 전에 잔존하는 결함수이며, 또한 테스트 시간  $t$ 까지 진행될 때 시간 구간  $(t, t+x)$ 에서 소프트웨어 고장이 발생하지 않을 확률을 소프트웨어 신뢰도이다. 소프트웨어 신뢰도는 식(4)와 같다.

$$R(x | t) = \exp[-H(t+x) - H(t)], \quad X \geq 0, t \geq 0 \quad \text{식(4)}$$

### 2.3 결함 발견 난이도

NHPP모델에서 소프트웨어 결함 수는 소프트웨어 내에 잔존하는 결함 수에 비례한다[2][3][4]. 이것은 미분 방정식으로 표현하면 식(5)와 같다.

$$\frac{dH(t)}{dt} = b(t)[\alpha - H(t)], \quad b(t) > 0, t \geq 0 \quad \text{식(5)}$$

여기서,  $\alpha$ 는 테스트 하기전 잔존 결함수이고,  $b$ 는 결함 발견 수 비례상수이다. 또한, 식(5)에 의하여 1개의 결함 발견율은 식(6)과 같다.

$$b(t) = \frac{dH(t)}{dt} / [\alpha - H(t)] \quad \text{식(6)}$$

## 3. 모델 제안 및 평가

본 연구는 결함 발견 수 데이터가 시스템에 미치는 영향이 일정하지 않고, 또한 결함의 중요도 단계에 따라서 소프트웨어 신뢰도 성장곡선이 달라진다는 관점에서 살펴보고자 한다. 결함중요도 단계에 따른 SRGM을 적용하기 위하여 다음과 같이 가정한다.

- 결함 발견시 발견된 소프트웨어 결함은 모두 수정 및 제거된다. 즉, 수정된 결함은 새로 생기지 않는다.
- 각 소프트웨어 고장은 4가지 결함에 의해 발생한다.
- 각 소프트웨어 고장은 서로 독립적이다.
- 각 소프트웨어 결함의 수정 시간은 무시한다.

### 3.1 결함 중요도

소프트웨어에서 고장의 원인은 소프트웨어 결함이 다[1][2]. 즉, 소프트웨어 고장을 일어나게 하는 소프트웨어 내의 결함을 예러라고 한다. 본 연구에서는 결함이 시스템에 미치는 영향의 정도를 중요도라 하고 <표 1>과 같이 4가지 형태로 분류한다[2].

<표 1> 결함 중요도 종류

<Table 1> Fault significance levels

종류	영향	의미
중요도1	치명적	시스템 다운, 업무의 마비
중요도2	중대	해당 처리 프로그램 중지
중요도3	경미	해당 기능 서비스 정지
중요도4	경고	해당 에러 메시지 알림

<표 2>는 제 3세대언어를 사용하는 50,000행 이상의 소프트웨어에서 나온 결과이다. 이때, 소프트웨어 라이프사이클 단계에서 발생하는 결함과 결함 중요도 단계의 관계를 <표 2>로 나타낼 수 있다[9].

<표 2> 소프트웨어 결함 발생원인 항목 및 중요도단계

<Table 2> Software fault sources and fault significance levels

결함발생원인항목	중요도 단계				합계
	1	2	3	4	
시스템 요구정의	5	5	3	2	15
분석 및 설계	3	22	10	5	40
코딩	2	10	10	8	30
문서화작성	0	1	2	2	5
수정작업	0	2	5	3	10
총 결함 수	10	40	30	20	100

<표 2>에서 소프트웨어 결함 중요도 단계를 발견 난이도 확률로 표현하면 <표 3>과 같다.

<표 3> 발견 난이도 확률

<Table 3> Probability of the detection capability

0<40<30<20<10<1
-----------------

### 3.2 기존 모델 설정 및 새로운 모델 제안

본 연구에서는 SRGM에서 지수형 모델을 이용한 다. 이것은 단위 시간에 발견된 결함 수가 그 시각의 잔존 결함 수에 비례하는 것으로 식(7)과 같다.

$$h(t) = \frac{dH(t)}{dt} = b[a - H(t)] \quad \text{식(7)}$$

여기서,  $a$ 는 테스트 하기 전 잔존 결함수 이고,  $b$ 는 결함 발견 수 비례상수이다. 또한, 식(7)를  $H(0)=0$ 로 하여 전개하면 식(8)과 같다.

$$H(t) = a(1 - \exp[-\int_0^t b \, dt]) \quad \text{식(8)}$$

따라서 지수형 모델의 평균치 함수는 식(9)와 같다.

$$H_{\alpha(t)} \equiv m(t) = a(1 - e^{-bt}), a > 0, b > 0 \quad \text{식(9)}$$

지수형 모델은 임의의 시각  $t$ 에서 잔존 결함 1개의 발견율은 일정하다고 가정했다[8]. 그러나 본 연구에서 제안하는 모델은 테스트에 의하여 발견되는 결함의 중요도 단계에 따라 발견 난이도가 다르다고 하였다. <표 3>에서 테스트 시간이 경과됨에 따라 중요도2, 중요도3, 중요도4, 중요도1과 같은 순서로 결함이 발견된다. 이때, 테스트 시각  $t$ 까지 발견되는 결함을 지수형 모델의 평균치 함수를 이용하면 식(10)과 같이 나타낼 수 있다.

$$m_i(t) = k_i a (1 - \exp^{-b_i t}), \quad (i=1, 2, 3, 4) \quad \text{식(10)}$$

그리고 본 연구에서 제안하는 모델의 평균치 함수는 식(11)과 같다.

$$H_s(t) = m(t) = a \sum_{i=1}^4 k_i (1 - \exp^{-b_i t}) \quad \text{식(11)}$$

여기서, 파라미터  $b_i$ 는 결함 1개의 발견율이고,  $k_i$ 는 소프트 내에 결함 함유율이다.  $k_i a$ 는 테스트 하기 전에 소프트웨어 내에 잔존하는 중요도  $i$  결함의 총 평균치를 의미한다.

### 3.3 제안 모델 파라미터 추정 식

본 연구에서는 제안된 모델의 파라미터를 추정하기 위하여 최우추정법을 이용한다[5][10]. 최우추정법을 이용하기 위하여 먼저, NHPP모델의 파라미터를 일정한 테스트 시각까지 발견된 총결함수에 관한 측정데이터를 가지고 관측하는 것으로 한다. 이 관측된 데이터에 대한 평균치 함수  $H(t)$ 를 가지는 NHPP모델의 우도함수는 식(12)와 같다

$$L = \Pr N(t_1) = y_1, N(t_2) = y_2, \dots, N(t_n) = y_n \\ = \exp[-H(t_n)] \prod_{k=1}^n \frac{H(t_k) - H(t_{k-1})^{(y_k - y_{k-1})}}{(y_k - y_{k-1})!} \quad \text{식(12)}$$

여기서,  $t_0=0, y_0=0$ 으로 하고, 식(12)에 양변에 자연대수를 취하면 대수 우도함수는 식(13)과 같다.

$$\ln L = \sum_{k=1}^n (y_k - y_{k-1}) \ln [H(t_k) - H(t_{k-1})] - H(t_n) - \sum_{k=1}^n \ln [(y_k - y_{k-1})!] \quad \text{식(13)}$$

식(13)의  $H(t)$ 에 제안하는 모델의 평균치 함수를 대입하면 식(14)와 같다.

$$\ln L = y_n \cdot \ln a + \sum_{k=1}^n (y_k - y_{k-1}) \ln [\exp^{-b_i(k-1)} - \exp^{-b_i t_k}] - k_i a (1 - \exp^{-b_i t_n}) - \sum_{k=1}^n \ln [(y_k - y_{k-1})!] \quad \text{식(14)}$$

식(14)를 가지고 모델의 파라미터에 대해 편미분하면 식(15)와 같이 나타낼 수 있다.

$$a_i = \frac{y_{i_n}}{(1 - \exp(-b_i t_n))}, \quad (i=1, 2, 3, 4) - b_i t_k$$

$$\frac{A}{B} = \sum_{k=1}^n \frac{C}{D}, \quad (i=1, 2, 3, 4) \quad \text{식(15)}$$

여기서,

$$A = y_i \exp(-b_i t_n),$$

$$B = (1 - \exp(-b_i t_n)),$$

$$C = (y_i - y_{i-1})(t_k \exp(-b_i t_k) - t_{k-1} \exp(-b_i t_{k-1})),$$

$$D = (\exp(-b_i t_k) - t_{k-1}) - \exp(-b_i t_n)$$

이다.

### 3.4 적용 데이터 분석

본 연구에서 적용할 데이터는 Brooks와 Motley가 실험결과에서 얻은 35개월의 결함 데이터를 사용하고 있다[6]. <표 4>은 결함 발견 수 데이터가 한 가지 종류이고, <표 5>는 결함 발견 수 데이터가 결함 중요도 단계별인 경우이다[6]. 여기서 결함 발견 수 데이터가 한가지인 경우는 기존의 지수형 모델, 또는 결함 중요도 단계별인 경우는 결함 중요도형 모델을 사용한다.

<표 4> 한가지 종류 결함 발견 수

<Table 4> Fault detection count of one type

시간	결함수	시간	결함수	시간	결함수
1	14	13	74	25	8
2	19	14	109	26	25
3	25	15	30	27	33
4	49	16	56	28	21
5	30	17	80	29	7
6	23	18	85	30	30
7	21	19	61	31	2
8	44	20	77	32	4
9	50	21	53	33	3
10	37	22	60	34	7
11	49	23	26	35	7
12	44	24	37		

<표 5> 중요도 단계 결함 발견 수 데이터

<Table 5> fault detection count of the significance levels

시간	중요도 단계 결함수				시간	중요도 단계 결함수				시간	중요도 단계 결함수			
	1	2	3	4		1	2	3	4		1	2	3	4
1	1.4	5.6	4.2	2.8	13	7.4	29.6	22.2	14.8	25	0.8	3.2	2.4	1.6
2	4.9	7.6	5.7	3.8	14	10.9	43.6	32.7	21.8	26	2.5	10.0	7.5	5.0
3	2.5	10.0	7.5	5.0	15	3.0	12.0	9.0	6.0	27	3.3	13.2	9.9	6.6
4	4.9	19.6	14.7	9.8	16	5.6	22.4	16.8	11.2	28	2.1	8.4	6.3	4.2
5	3.0	12.0	9.0	6.0	17	8.0	32.0	24.0	16.0	29	0.7	2.8	2.1	1.4
6	2.3	9.2	6.9	4.6	18	8.5	34.0	25.5	17.0	30	3.0	12.0	9.0	6.0
7	2.1	8.4	6.3	4.2	19	6.1	24.4	18.3	12.2	31	0.2	0.8	0.6	0.4
8	4.4	17.6	13.2	8.8	20	7.7	30.8	23.1	15.4	32	0.4	1.6	1.2	0.8
9	5.0	20.0	15.0	10.0	21	5.8	21.2	15.9	10.6	33	0.3	1.2	0.9	0.6
10	3.7	14.8	11.1	7.4	22	6.0	24.0	18.0	12.0	34	0.7	2.8	2.1	1.4
11	4.9	19.6	14.7	9.8	23	6.6	10.4	7.8	5.2	35	0.7	2.8	2.1	1.4
12	4.4	17.6	13.2	8.8	24	3.7	14.8	11.1	7.4					

### 3.5 두 모델 파라미터 추정 및 비교

모델을 평가하기 위하여 <표 4>~<표 5>의 데이터를 가지고 <표 6>과 같이 각각의 파라미터의 값을 식(15)에 의하여 추정할 수 있다.

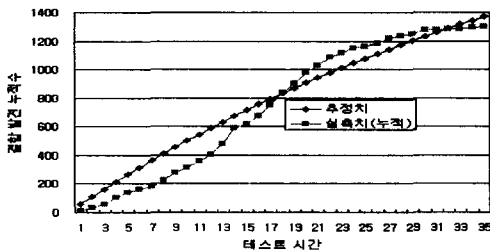
<표 6> 두 모델 파라미터 추정 값

<Table 6> The parameter estimation values of the two models

지수형 SRGM(기준)		결함 중요도 단계 SRGM	
테스트 하기 전에 존재하는 결함	결함 발견율	테스트 하기 전에 존재하는 결함	중요도 i 결함 발견율
$a=2592$	$b=0.02152$	$a1=245.680$ $a2=980.699$ $a3=736.340$ $a4=491.355$	$b1=0.021520$ $b2=0.021580$ $b3=0.021550$ $b4=0.021520$

<표 5>에서 지수형 모델의 파라미터 값을 식(9)에 대입하면 식(16)과 같다. 식(16)을 이용하면 [그림 2]와 같이 지수형 모델을 추정할 수 있다.

$$He(t) = 2592(1 - e^{-0.02152t}) \quad \text{식(16)}$$



[그림 2] 지수형 모델

[Fig. 2] The exponential type model

그리고 <표 5>에서 결함 중요도 단계를 고려한 모델의 파라미터 값을 식(11)에 대입하면 식(17), 식(18), 식(19), 식(20)과 같다.

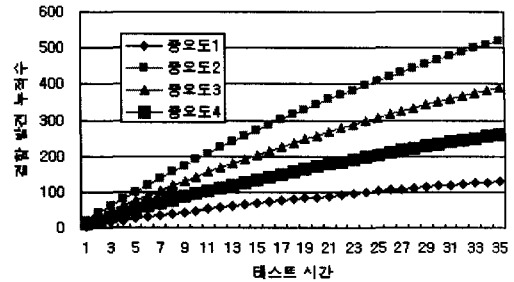
$$Hs1(t) = 245.680(1 - \exp^{-0.021520t}) \quad \text{식(17)}$$

$$Hs2(t) = 980.699(1 - \exp^{-0.021580t}) \quad \text{식(18)}$$

$$Hs3(t) = 736.340(1 - \exp^{-0.021550t}) \quad \text{식(19)}$$

$$Hs4(t) = 491.355(1 - \exp^{-0.021520t}) \quad \text{식(20)}$$

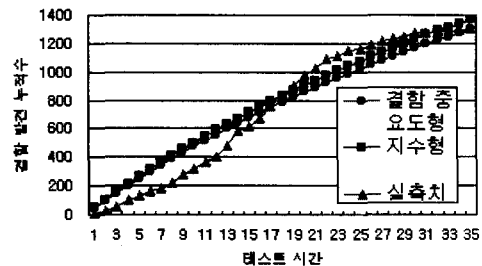
식(17), 식(18), 식(19), 식(20)을 이용하면 [그림 3]과 같이 결함 중요도 단계를 고려한 모델을 추정할 수 있다.



[그림 3] 결함 중요도 단계 모델

[Fig. 3] The model of fault significance levels

따라서 [그림 4]는 지수형 모델과 결함 중요도 단계를 고려한 모델을 서로 비교하였다. [그림 4]에서 살펴본 바와 같이 결함 중요도 단계를 고려한 모델이 기존의 지수형 모델과 매우 유사한 곡선이 나타남을 알 수 있고, 또한 결함 중요도 단계를 고려한 모델이 기존의 NHPP 모델과 유사하다는 것을 알 수 있다. 이것은 신뢰성을 평가할 수 있는 SRGM이라는 것이다.



[그림 4] 지수형과 결함 중요도 단계 모델 비교

[Fig. 4] The comparison of exponential type and fault significance level type

### 3.6 모델 적합도 검정

본 연구에서는 제안한 모델에 대하여 적합여부를 검정하기 위해서 K-S 적합도 검정법을 사용하였다. K-S 적합도 검정법은 식(21)과 같다.

$$D_i = \max \left\{ \left| \frac{H(t_i)}{H(t_n)} - \frac{y_i}{y_n} \right|, \left| \frac{H(t_i)}{H(t_n)} - \frac{y_{i-1}}{y_n} \right| \right\} \quad \text{식(21)}$$

여기서,  $H(t_n)$ 은 모델의 평균치 함수이고,  $y_n$ 은 관측된 결함데이터이다. 본 연구에서는 기각한계를 90% 신뢰구간을 정하였다. 두 모델의 K-S검정 통계량은 <표 7>과 같다. <표 7>에서 살펴보면 두 모델 모두 K-S검정통계량이 기각한계의 값 보다 작으므로 실제 데이터에 적합한 모델이라는 것을 알 수 있다.

<표 7> K-S 검정 통계량  
<Table 7> A statistical of K-S verification

모델	K-S통계량	기각한계	적합도 판정
지수형SRGM	0.15968	0.20185	적합
결함중요도단계 SRGM	중요도1:0.015969 중요도2:0.15987 중요도3:0.15980 중요도4:0.15969	0.20185	적합

#### 4. 결론 및 향후 연구 과제

본 연구에서는 개수계측 모델 종류의 NHPP모델 중에서 가장 기본적인 지수형 모델과 결함 중요도 단계를 고려한 모델을 적용 및 비교 분석하였다. 기존의 지수형 모델은 결함의 종류가 한가지이며, 결함의 발견율이 일정하다고 가정하였다. 또한 지금까지 제안하는 대다수의 SRGM은 지수형 모델을 따르고 있다. 그러나 현실적으로 볼 때 실제 테스트 단계에서 발견되는 결함의 유형과 다를 수도 있고 이런 데이터를 가지고 소프트웨어의 신뢰성을 평가할 경우에는 현실적으로 부적절할 수도 있다. 본 연구에서는 기존의 SRGM에서 적용한 결함과는 달리 테스트 단계에서 발견되는 결함이 시스템에 미치는 영향에 따라 4가지로 분류(중요도1,중요도2, 중요도3, 중요도4)하여 살펴보았다. 즉, 결함의 분류에 따라서 소프트웨어 고장을 일으키는 정도가 다르고 결함이 발견되는 중요도도 다르다라는 관점에서 살펴보았다. 우선 제안된 모델의 타당성을 평가하기 위하여 결함 데이터를 적용하여 두 모델의 파라미터를 추정하여

평균치 함수를 가지고 두 모델을 비교하였다. 비교 결과 두 모델이 유사한 곡선임을 알 수 있었다. 이것은 NHPP모델을 기본적으로 가지는 SRGM이라는 것을 알 수 있다. 향후 연구 과제는 실제로 개발할 소프트웨어를 동일한 환경에서 테스트를 동시에 실시하여 정확한 결함 수를 측정하여 중요도단계를 고려한 SRGM에 적용하여 보다 타당한 신뢰성을 평가하는 것이다.

#### ※ 참고문헌

- [1] 신경애, "NVP신뢰도 분석을 위한 유전자 알고리즘 적용", 한국컴퓨터산업교육학회 논문지, "2000, 10, Vol. 1., No. 1, October.
- [2] 권대곤 : "결함 중요도에 따른 소프트웨어 신뢰도 성장모델 제안 및 평가", 경남대학교 대학원 석사학위논문, 1995.12.
- [3] P.Zeephongsekul, G.Xia, S.Kumar : "Software-Reliability Growth Model : Failures Generate Secondary-Faults Under Imperfect Debugging", IEEE Trans Reliability, Vol.43, No.3, 1994.
- [4] 山田 茂, 高橋宗雄 著 : "ソフトウェアマネジメントモデル 入門", 共立出版株式会社, 1993.
- [5] 山田 茂 : "ソフトウェアの信頼性評価技術", HBJ出版局, 1989.
- [6] S.Yamada, J.Hishitani and S.Osaki : "Software Reliability Growth with a Weibull Test-Effort : A Model & Application", IEEE Trans., Reliability, Vol.42, NO.1, 1993.
- [7] G.J.Schick and R.W.Wolverton : "An analysis of competing Software reliability models", IEEE Trans., Software Engineering, Vol.SE-4, No.2, 1978.
- [8] A.L.Goel and K.Okumoto : "A Markovian model for reliability and other performance measures of software systems", Proc. National Computer Conf., 1979.

- [9] Capers Jones : "Applied Software Measurement : Assuring Productivity Quality", 1991
- [10] J.D.Musa and A.Iannino and K.Okumoto : "Software Reliability : Measurement, Prediction, Application", McGraw-Hill, 1987

신 경 애



1986 한국방송대학교 경영학과 졸업(경영학사)  
1991 한국방송대학교 전자계산학과 졸업(이학사)  
1989 계명대학교 교육대학원 전자계산전공(교육학석사)  
1999 경남대학교 대학원 컴퓨터공학과(공학박사)  
1983~1989 동주여자상업고등학교 재직  
1990~현재 동주대학 컴퓨터정보계열 부교수

관심분야: 소프트웨어 비용예측, 결합허용, 신뢰도분석, 소프트웨어 신뢰도 성장모델, CAI