

테스트 단계를 고려한 소프트웨어 신뢰성 평가에 관한 연구

유 창 열* 권 대 곤**

A Study on an Evaluation of Software Reliability with Test

Chang-Yeul Yoo* Dae-Gon Kweon**

요 약

소프트웨어 개발 과정에서 신뢰성 평가는 대단히 중요하다. 테스트 단계를 구분하지 않은 소프트웨어의 신뢰도 평가·분석의 결과는 신뢰성이 결여될 수 있다. 따라서 본 논문에서는 이러한 문제점을 해결하기 위해 테스트 단계를 구분하여 신뢰성 평가에 관한 연구를 하였다. 이를 위해 단위 테스트(Unit Test), 통합 테스트(Integration Test), 확인 테스트(Validation Test), 시스템 테스트(System Test)에서 발견된 결함 데이터를 이용하여 소프트웨어 신뢰도 성장 모델(Software Reliability Growth Model : SRGM)인 지수형, 지연 S자형, 습속 S자형 SRGM에 적용하였다. 그 결과 단위 테스트에서는 습속 S자형, 통합 테스트에서는 지연 S자형, 확인 테스트에서는 지연 S자형, 시스템 테스트에서는 지수형 SRGM이 가장 적합한 것임을 입증하였으며, 신뢰성 평가 척도인 파라미터 추정, 편차 자승합, 기대잔존 결함수 등에서 본 연구의 결과가 기존의 방법보다 우수함을 보였다.

Abstaract

The evaluation of reliability is very important in the development process of software. There may be lack of trustfulness on the results that come from the analysis and evaluation of reliability of softwares which do not divide the test phases. At this point, this article studies how to evaluate the reliability dividing the test phases in order to settle the these problems. In doing so, I apply the fault data to be found in Unit Test, Integration Test, Validation Test and System Test to SRGM(Software Reliability Growth Model), Exponential SRGM, Delayed S-shaped SRGM, and Inflection S-shaped SRGM. The result is that Inflection S-shaped is best suitable with Unit Test, Delayed S-shaped is best suitable with Integration and Validation Test, and Exponential SRGM is best suitable with System test. In this respect, I can show that the results of this study on parameter estimation, difference square summation, number of fault remained is superior to the established methods.

* 남해전문대학 사무자동화과 조교수

** 남해전문대학 사무자동화과 전임강사

논문접수 : 98.4.10 심사완료 : 98.6.12

1. 서 론

지금까지 소프트웨어 신뢰도 모델들의 대부분은 『테스트 초기에 많은 결함이 발견되고 테스트 시간이 경과됨에 따라 발견되는 결함의 수는 감소한다』 [1]라는 시간적 결함 발생을 중심으로 연구되어 왔다. 이러한 관점에서 『테스트 초기에는 결함 발견율이 높고 테스트의 진행 시간이 점차 증가할수록 결함 발견율이 낮아진다』 [2, 3, 4, 5]고 가정하여 다양한 신뢰성 평가 모델들이 제안되었다. 대부분 이러한 모델들은 테스트의 시작에서 종료까지 소프트웨어 결함 발견수 또는 고장 시간 데이터를 포괄적으로 적용하고 있다. 그러나 실제 테스트 단계는 단위 테스트, 통합 테스트, 확인 테스트, 시스템 테스트로 구분되어 순서적으로 병행하여 진행된다.

본 논문에서는 기존의 신뢰성 평가 방법과는 다르게 테스트 단계별로 결함 데이터를 수집한다. 여기서 수집된 데이터를 가지고 각 단계에 가장 적합한 모델을 선정하고, 이에 따른 신뢰성을 분석한다.

그리고 본 연구의 내용 구성은 다음과 같다. 2장에서는 관련 연구로서 소프트웨어 테스트와 신뢰도에 대해 고찰하고, 3장은 실제 수집된 결함 데이터를 사용한 신뢰성을 평가한다. 4장은 연구 결과에 대해 요약한다.

2. 관련 연구

2.1 소프트웨어 테스트

테스트란 소프트웨어 개발과정 중에서 노력과 비용이 가장 집중적으로 투자되는 어려운 과정으로 기능과 성능을 확인하는 작업이다.

테스트 방법은 단위, 통합, 확인, 시스템 테스트로 나눌 수 있다. 단위 테스트는 프로그램의 기본 단위인 각 모듈이 내부 설계 명세에 맞도록 작동하는지를 테스트한다. 복잡한 시스템일 경우 여러 모듈들이 모여서 서브시스템을 형성하게 되는데 이 때 이들 간의 인터페이스를 중심으로 테스트하는 통합 테스트가 이루어진다. 확인 테스트는 요구 분석에

의해 작성된 요구를 소프트웨어가 만족하는가를 확인하는 과정으로 이 때의 요구 명세는 외부 설계 명세를 의미한다. 마지막으로 시스템 테스트는 다른 시스템 요소(하드웨어, 인력, 데이터베이스)들과 연결된 소프트웨어가 시스템 기능 및 성능 등의 사용자 요구를 만족하는지 여부를 테스트한다.

2.2 소프트웨어 신뢰도

소프트웨어 신뢰도 성장 곡선은 검출한 소프트웨어 결함의 누적치와 경과 시간을 양축으로 하여 좌표에 점을 찍은 것이다. 경과 시간은 카렌더 시간 또는 컴퓨터 실행 시간이 있지만 일반적으로는 카렌더 시간이 이용되고 있다. SRGM은 이러한 성장 곡선의 특성을 정식화하여 검출 결함 데이터로부터 잔존 결함 수와 그 검출에 요하는 시간을 추정하는 것이다.

소프트웨어 신뢰성 평가, 특히 잔존 결함수의 추정에는 소프트웨어 개발 관리 공정에서 주요 항목의 한가지이다. 이 추정에 이용되고 있는 신뢰성 모델의 대부분은 현재 SRGM이라고 말한다[6]. 소프트웨어 규모가 큰 경우 통상 폭포수 모델[7]을 베이스로 개발이 진행된다. 이 폭포수 모델은 통합 테스트 단계부터 SRGM이 적용 가능하다.

본 논문에서는 대표적인 NHPP(Nonhomogeneous Poisson Process) 모델 중에서 지수형, 지연 S자형, 습속 S자형 SRGM의 신뢰도 성장곡선을 단위시간당 결함 검출 수를 이용해서 비교·분석한다. 그리고 실측된 결함 데이터를 사용해서 적용 방법 및 잔존 결함 수 추정 정도를 나타낸다. 그 분석 결과를 통하여 모든 테스트 단계에서 가장 적합한 SRGM을 찾아낸다.

2.2.1 지수형 SRGM

지수형 SRGM은 NHPP의 가장 기본적인 모델로서 Goel and Okumoto[8, 9]에 의해 제안되었다. 이 모델의 기본적인 개념은 단위 시간당 발견된 에러수가 그 시각의 잔존 에러수에 비례하는 것으로 정의된다. 모델의 평균치 함수는

$$M_e(t) = a(1 - e^{-bt}), \quad a > 0, \quad b > 0 \quad (1)$$

로 된다. 여기서 a 는 테스트 개시 전에 잠재하는 에러수이며, b 는 에러 발견율을 나타내는 비례상수이다.

2.2.2 지연 S자형 SRGM

Yamada et al[10]가 제안한 지연 S자형 SRGM은 소프트웨어 고장 발생 시간에서부터 그 고장의 원인인 에러를 제거하는 시간까지의 시간 지연을 고려하고 있으며 가장 일반적으로 사용되는 모델이다. 즉, 테스트에 의해 에러를 발견하기 위해서는 먼저, 고장 현상을 확인한 후에 원인 분석을 해야한다는 개념이다.

에러 발견과정에서 시각 t 까지 발견되는 총기대 에러수가 $M(t)$ 일 때 평균치 함수는 식 (2)가 된다.

$$M(t) = a[1 - (1 + bt)e^{-bt}], \quad a, b > 0 \quad (2)$$

2.2.3 습속 S자형 SRGM

습속 S자형 SRGM은 소프트웨어 결함이 다른 결함과 상관없이 발견 가능한 것과 하나의 결함이 발견되기 위해서는 다른 결함이 그 이전에 발견되어 제거되어야 하는 것으로 분류한다. 그리고 테스트 담당자는 소프트웨어에 대한 결함 발견 능력을 나타내는 습속도를 고려하여 결함을 발견하여 제거한다면 남아있는 결함이 발견될 가능성은 높다.

Ohba[11]에 의해 제안된 이 모델의 평균치 함수는

$$M_e(t) = a(1 - e^{-bt}) / (1 + ce^{-bt}), \quad a > 0, \quad b > 0, \quad c > 0 \quad (3)$$

을 얻는다. 여기서 c 는 결함발견 능력에 관한 습속 계수(Inflexion rate)이다.

3. 신뢰도 성장곡선의 분석

본 논문에서는 실시간 항공 관제 시스템 개발 소

프트웨어에서 수집된 결함 데이터에 대해 누적 결함수와 신뢰도 성장 곡선을 단위, 통합, 확인, 시스템 테스트 단계를 구분하여 추정·분석한다.

3.1 기존의 방법

기존에는 소프트웨어 신뢰성을 평가하는데 모든 테스트 단계에 걸쳐 수집된 결함 데이터를 소프트웨어 신뢰도 성장 모델에 적용하였다. 이 결함 데이터 자료는 실시간 항공 관제 시스템에서 수집된 것으로 그림 1과 같으며 좀 더 객관적인 자료를 이용하기 위해 채택하였다.

그 과정은 테스트 데이터 수집에서부터 소프트웨어 신뢰도 성장 모델 선택, 모델 파라미터 추정, 평균치 함수, 편차 자승합을 통한 최적 모델 선택까지의 단계이다. 결과물로 테스트 종료 시점에서 기대 잔존 결함수와 신뢰도를 추정하였다.

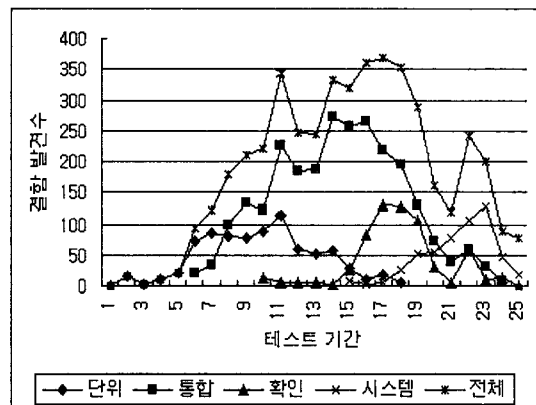


그림 1 테스트 단계별 결함 데이터

추정 결과 수집된 결함 데이터에 대해서는 어느 한 모델이 적합함을 보였다. 본 논문에서 참고한 결함 데이터에 적용한 과정은 표 1과 같다.

표 1에 의하면 본 논문에서 이용한 결함 데이터에 대해서는 편차 자승합이 가장 작은 지수형 SRGM이 최적의 모델이며, 이 모델을 사용하여 기대 잔존 에러수와 신뢰도를 추정해 볼 수 있다.

표 1 신뢰성 데이터 해석 과정

모델 과정	지수형	지연 S자형	습속 S자형
파라 미터 추정	a=5078.375 b=0.094053	a=4547.827 b=0.462723	a=4550.072 b=0.400100 c=9.000000
평균치 함수	$a(1 - e^{-bt})$	$a(1 - (1 + bt)e^{-bt})$	$\frac{a(1 - e^{-bt})}{(1 + ce^{-bt})}$
편차 자승합	52,235,168	125,051,488	80,537,632
기대 잔존 결함수	531	0	0

표 2 단위 테스트에서의 결함 데이터 이용

모델 과정	지수형	지연 S자형	습속 S자형
파라 미터 추정	a=801.000012 b=0.999000	a=818.779602 b=0.318747	a=832.61005 b=0.307717 c=9.000000
평균치 함수	$a(1 - e^{-bt})$	$a(1 - (1 + bt)e^{-bt})$	$\frac{a(1 - e^{-bt})}{(1 + ce^{-bt})}$
편차 자승합	3,867,930	703,624	161,853
기대 잔존 결함수	0	0	3

3.2 본 논문에서의 분석

본 논문에서는 기존의 모든 테스트 공정에 걸쳐 수집된 데이터를 가지고 신뢰성 평가를 하는 것이 아니고 테스트의 단계를 구분하여 소프트웨어 신뢰성을 평가한다. 즉, 총 테스트 시간에 대해 평균적으로 적합한 모델을 찾아내는 것이 아니라 테스트를 각 단계별로 나누어 각각 어떤 모델이 적합한지 분석함으로써 신뢰도 척도 추정에 보다 정확한 결과를 얻을 수 있다. 그 방법은 다음과 같다.

먼저, 단위 테스트를 하여 검출된 결함 데이터를 바탕으로 기존의 방법대로 파라미터 추정에서 적합도 검정까지 과정을 행한다.

그 다음은 적용한 모델 중에서 최적의 모델을 찾아내기 위하여 각 모델에 대해 편차 자승합을 계산한다. 적합 모델 중 편차 자승합이 최소인 모델이 단위 테스트 단계에서 가장 적합한 모델이다. 이 과정은 표 2와 같으며, 세 가지 모델 중 편차 자승합이 가장 적은 161,853을 가지는 습속 S자형 SRGM이 최적의 모델이다.

다음으로 통합 테스트 단계에 대해서도 단위 테스트 단계에서 분석한 신뢰도 데이터 해석과 같은 과정을 따른다. 적용된 세 가지 모델 중 최소의 편차 자승합을 가지는 모델을 선택한다.

표 3에서는 지연 S자형 SRGM이 편차 자승합이 가장 적은 750,174를 가지므로 최적의 모델이다.

표 4는 확인 테스트의 추정·분석 결과이다. 확인

표 3 통합 테스트에서의 결함 데이터 이용

모델 과정	지수형	지연 S자형	습속 S자형
파라 미터 추정	a=3979.8113 b=0.054063	a=3604.5645 b=0.130590	a=3935.7383 b=0.1563350 c=9.000000
평균치 함수	$a(1 - e^{-bt})$	$a(1 - (1 + bt)e^{-bt})$	$\frac{a(1 - e^{-bt})}{(1 + ce^{-bt})}$
편차 자승합	1,713,133	750,174	2,409,895
기대 잔존 결함수	301	1,424	201

표 4 확인 테스트에서의 결함 데이터 이용

모델 과정	지수형	지연 S자형	습속 S자형
파라 미터 추정	a=930.143754 b=0.068644	a=1092.62939 b=0.119011	a=645.742126 b=0.343020 c=9.000000
평균치 함수	$a(1 - e^{-bt})$	$a(1 - (1 + bt)e^{-bt})$	$\frac{a(1 - e^{-bt})}{(1 + ce^{-bt})}$
편차 자승합	271,041	96,512	146,551
기대 잔존 결함수	310	162	2

테스트 단계에 대해서도 단위 테스트 단계에서 분석한 신뢰도 데이터 해석과 같은 과정을 따른다. 세 가지 모델 중 편차 자승합이 가장 적은 96,512를 가지는 지연 S자형 SRGM이 최적의 모델이다.

마지막으로 시스템 테스트 단계에 대해서도 단위

테스트 단계에서 분석한 신뢰도 데이터 해석과 같은 과정을 따른다. 적용된 세 가지 모델 중 최소의 편차 자승합을 가지는 모델을 선택한다.

표 5에서는 지수형 SRGM이 편차 자승합이 가장 적은 113,769를 가지므로 최적의 모델이다.

표 5 시스템 테스트에서의 결합 데이터 이용

모델 관점	지수형	자연 S자형	습속 S자형
파라미터 추정	a=4811.95268 b=0.010502	a=525.106323 b=0.999000	a=573.123169 b=0.427395 c=9.000000
평균치 함수	$a(1 - e^{-bt})$	$a(1 - (1 + bt)e^{-bt})$	$\frac{a(1 - e^{-bt})}{(1 + ce^{-bt})}$
편차 자승합	113,769	891,957	114,518
기대 잔존 결함수	4,286	0	5

3.3 신뢰성 평가의 비교 분석

전 절에서 기존의 방법과 본 논문에서 제안한 방법으로 각각 테스트 단계에서 수집된 데이터를 이용하여 신뢰성을 평가하였다. 이들을 비교한 결과는 표 6과 같다.

표 6 신뢰성 데이터 해석 비교

구분	선택 S	파라미터 추정	편차 자승합	기대잔존 결함수
기존방법	자연S자형	a=5078.375 b=0.094053	52,235,168	531
단위테스트	습속S자형	a=832.61005 b=0.307717 c=9.000000	161,853	3
통합테스트	자연S자형	a=3604.5645 b=0.130590	750,174	1,424
확인테스트	자연S자형	a=1092.62939 b=0.119011	96,512	162
시스템테스트	지수형	a=4811.95268 b=0.010502	113,769	4,286

분석 결과를 종합하여 정리하면 다음과 같다. 기존의 방법은 지수형 SRGM을 이용하여 소프트웨어 신뢰성을 평가하였다. 그런데 본 연구의 경우 단위 테스트 단계에서는 습속 S자형, 통합 테스트 과정에

서는 자연 S자형, 확인 테스트에서는 자연 S자형, 시스템 테스트에서는 지수형 SRGM이 선택되어 소프트웨어 신뢰성을 평가하였다.

이상과 같이 표 6의 신뢰성 데이터 해석을 비교해 보면 본 논문에서 제안한 방법이 모든 평가척도에서 타당함이 입증되었다. 따라서 테스트 단계를 구분하여 신뢰성을 평가해 간다면 보다 양질의 소프트웨어 개발이 가능하다.

4. 결론

소프트웨어 개발 과정에서 신뢰성 평가는 대단히 중요하다. 기존에는 이러한 분석 과정을 전체 테스트 공장에서 수집된 결합 데이터를 이용하였다. 이와 같이 테스트 단계를 구분하지 않은 소프트웨어의 신뢰도 평가·분석의 결과는 신뢰성이 결여될 수 있다.

따라서 본 논문에서는 이러한 문제점을 해결하기 위해 테스트 단계를 구분하여 신뢰성 평가에 관한 연구를 하였다. 이를 위해 단위, 통합, 확인, 시스템 테스트에서 발견된 결합 데이터를 소프트웨어 신뢰도 성장 모델인 지수형, 자연 S자형, 습속 S자형 SRGM에 적용하였다.

실제로 수집된 테스트 데이터를 사용하여 테스트 단계를 구분하지 않았을 때의 신뢰성 평가 결과와 비교해 본 결과, 본 논문에서처럼 각 테스트 단계로 분할하여 신뢰성을 평가한 것이 보다 정확하며 각 단계별로 자기 다른 모델이 적합함을 확인할 수 있었다.

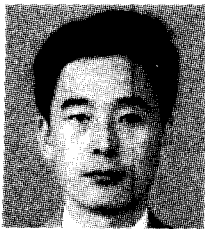
향후 연구 과제는 테스트 단계별 신뢰성 분석을 기반으로 새로운 소프트웨어 신뢰도 선정모델 개발에 대한 연구이다.

References

[1] A. L. Goal, "Software error detection model with applications", J. Systems and Software, Vol. 1, pp. 243-249, 1980.

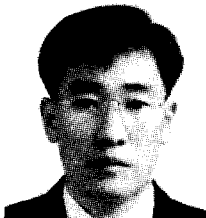
- [2] A. L. Goal, K. Okumoto, "Time-dependent error- detection rate model for software reliability and other performance measures", IEEE Trans. Reliability, Vol. R-28, No. 3, pp. 206-211. 1979.
- [3] B. W. Boehm, "Software Engineering", IEEE Trans. Computers, Vol. C-25, No. 12, pp. 1226-1241, 1976.
- [4] C. V. Ramamoorthy and F. B. Bastani, "Software Reliability-Status and Perspectives", IEEE Trans. Software Eng., Vol. SE-8, No. 4, pp. 354-371, 1982.
- [5] M. Ohba, "Inflection S-shaped software reliability growth model", in Stochastic Models in Reliability Theory, S. Osaki and Y. Hatoyama(eds), pp. 144-162, Springer-Verlag, Berlin, 1984.
- [6] M. Xie, B. Bergman, "On modeling reliability growth for software", Proc 8th IFAC/IFORS Symp. Identification and System Parameter Estimation, pp. 567-570, 1988.
- [7] M. Xie, "On a generalization of the J-M model", Proc Reliability '89, pp. 6Ba/3/1-5Ba/3/7, 1989.
- [8] P. B. Moranda, "Event-altered rate models for general reliability analysis", IEEE Trans Reliability, Vol. R-28, No. 5, pp. 376-381, 1979.
- [9] P. B. Moranda, "Prediction of software reliability during debugging", Proc 1975 Annual reliability and Maintainability Symp., pp. 327-332, 1975.
- [10] S. Yamada, M. Ohba and S. Osaki "S-shaped reliability growth modeling for software error detection, IEEE Trans. Reliability, Vol. R-32, No. 5, pp. 475-478, 1983.
- [11] Z. Jelinski, P. B. Moranda, "Software reliability research", in Statistical Computer Performance Evaluation, Frciberger, W. (ed.), pp. 456-484, Academic Press, 1972.

● 저자소개



유 창 열

1987년 : 경상대학교 전산통계학과(학사)
 1994년 : 경상대학교 대학원 전자계산학과(석사)
 1994년~1997년 : 경상대학교 대학원 박사과정 수료
 1990년~1996년 : 연암공업전문대학 전산소 전임연구원
 1996년~현재 : 남해전문대학 사무자동화과 조교수



권 대 곤

1984년 : 경남대학교 전자계산학전공(학사)
 1996년 : 경남대학교 대학원 소프트웨어공학(석사)
 1996년~현재 : 경남대학교 컴퓨터공학과 박사과정
 1984년~1989년 : 육군군수사령부 전자계산소 프로그램장교
 1997년~현재 : 남해전문대학 사무자동화과 전임강사