

소프트웨어 신뢰도 품질 평가 메트릭에 대한 연구

On the Study of Software Reliability in Quality Testing Metrics

정 혜 정*
Hye-Jung Jung

요 약

소프트웨어 품질 평가 시스템과 소프트웨어 품질 평가 범위는 소프트웨어 신뢰도 측정을 위해서 정의되어야 한다. 본 연구에서 소프트웨어 신뢰도 품질 평가 메트릭을 개발하고 소프트웨어 신뢰도를 측정하기 위한 적당한 범주의 테스트에 대해서 조사한다. 연구를 성공적으로 수행하기 위해서 소프트웨어의 버그수를 이용해서 소프트웨어의 실패시간을 예측했다. 소프트웨어의 신뢰도는 소프트웨어의 품질 평가에 있어서 중요하다. 국제 표준 ISO/IEC 9126의 소프트웨어 신뢰성 평가 메트릭을 기초로 하여 소프트웨어 신뢰도를 평가할 수 있는 메트릭을 제안하고 이것을 중심으로 하여 평가방안을 제안한다.

Abstract

Software quality evaluation system and software evaluation criterion have to be defined to evaluate software reliability. In this study, we develop software reliability quality metrics for evaluating of software reliability and research for testing of suitable criteria of software reliability. To accomplish a study successfully, we have to estimate the failure time of software using the number of software bugs. Software reliability is very important in software quality. We propose the estimation of software reliability. We propose the software reliability metrics and the software testing method that the basis of international standard ISO/IEC 9126 in software testing metrics.

⇨ Keyword : 소프트웨어 신뢰도(Software Reliability), 품질평가(Quality Evaluation), 실패시간(Failure Time), 포아송분포(Poisson Distribution), ISO/IEC 9126-2

1. 서 론

국내의 소프트웨어 개발 업체에서는 개발 제품의 인증에 대한 인식이 확산되어지고 있고 국내 제품의 질적 향상을 위한 노력을 기울이고 있다. 제품자체의 품질관점이란 측면에서 개발방법과 절차가 정확해도 제품자체의 품질을 보증하지 못하면 좋은 품질의 소프트웨어를 개발할 수 없다. 이러한 측면으로 대표적인 적용모델로는 ISO/IEC 12119(Quality Requirement and Testing), ISO/IEC 9126(Software Product Quality, External/Internal Metrics), ISO/IEC 14598(Software Product Evaluation, Acquirers,

Process for Development)이 있다.[1,2,3] 소프트웨어의 인증에 대한 정착화를 위해서 ISO/IEC 9126-2에서 요구하는 평가항목을 시험을 통해서 정확히 평가해야 하며 특히 산업용 소프트웨어의 경우 6개의 품질 특성 중에서 신뢰성에 대한 평가는 상당히 중요한 요소이다. 국내에서 소프트웨어 인증에 대한 관심과 연구가 진행되어지면서 국가 기관에서 우리나라 제품에 대한 질적향상을 기하기 위하여 소프트웨어의 제품 평가를 담당하고 있다. 이와 같은 노력으로 인하여 국내 소프트웨어의 제품에 대한 질적향상에 상당한 변화가 일기 시작하였으며 많은 개발 업체에서 제품에 대한 평가에도 상당한 관심을 가지게 되었다. 산업용 소프트웨어의 경우 지난 2년간 개발 업체와 이 제품을 사용하는 사용업체를 중심으로 하여 설문조사를 하여본 결

* 정 혜 원: 평택대학교 정보통계학과 부교수
jhjung@ptu.ac.kr

12005/08/20 투고 - 2005/09/12 심사 - 2006/01/05 심사완료

과 산업용 소프트웨어의 경우는 6개의 품질특성인 기능성(Functionality), 신뢰성(Reliability), 사용성(Usability), 유지보수성(Maintainability), 효율성(Efficiency), 이식성(Portability) 중에서 신뢰성이 가장 중요하다는 응답을 하였다. 그러나 ISO/IEC 9126-2에서 제시하고 있는 소프트웨어의 신뢰성에 대한 품질 평가 메트릭은 제품이 개발되어지고 개발된 제품에 대하여 시험과정을 거쳐서 조사된 소프트웨어의 실패시간에 대한 자료를 이용하여 신뢰성을 평가하게 되어진다. 그러므로 신뢰성에 대한 평가가 상당히 힘들다고 하겠다. 본 연구에서는 ISO/IEC 9126에서 제시하는 평가메트릭을 기준으로 하여 신뢰성을 평가할 수 있는 평가 방안을 개발 한다. 소프트웨어에 대한 신뢰도 평가를 위해서 소프트웨어 신뢰도 성장 모델에 적용하는 방안을 마련하여 좀더 안정화된 평가방법을 개발하고 소프트웨어 시험 평가에 있어서도 안정성을 기하려고 한다. 여러 가지 자료에 대한 환경적인 요인을 고려하여 소프트웨어 신뢰도 성장 모델에 적용하고 적용된 결과를 이용하여 신뢰도를 평가하여야 한다. 본 연구에서는 소프트웨어 신뢰도를 평가하기 위한 실패시간의 자료를 예측하고 예측된 실패시간 자료를 이용하여 신뢰도를 평가하는 방안에 대하여 제안하려 한다. 본 연구를 위해서 조사한 산업용 소프트웨어의 제작 업체를 대상으로 실시한 설문조사를 통해서 얻어진 자료를 이용해서 소프트웨어 신뢰성 성장 모델에 이용할 수 있는 방안을 연구한다. 국내 소프트웨어의 질적 향상을 기할 수 있는 인증 사업은 계속적으로 진행 되어져야 하며 계속적인 연구를 통해서 시험의 안정화와 정착화를 이루어야 할 것이다. 국내외적으로 소프트웨어의 질적 향상을 기대하는 현 시점에서 평가 방안에 대한 개발 방법의 노력과 연구는 계속적으로 진행되어져야 할 것이다. 본 연구에서는 ISO/IEC 9126의 국제 표준에 의한 품질평가 메트릭의 6가지 품질 특성중에서 특히 신뢰성을 평가할 수 있는 방법을 연구하려한다. 현재의 시험은 6가지 품질 특성을 모두 똑같은 평가 배점

으로 평가하고 있다. 산업용 소프트웨어의 경우 신뢰성이 가장 중요하다는 설문조사 결과를 참고하여 앞으로는 소프트웨어의 평가에 있어서 평가방법의 이원화에 대한 연구를 통해서 시험 방법을 다양하게 적용할 수 있도록 하여야 할 것이다. 본 연구는 2장에서 설문조사 내용을 간단히 소개하고[14] 3장에서 소프트웨어 신뢰도 성장 모델에 대하여 소개하고[7,11,12] 4장에서는 소프트웨어 신뢰도 성장 모델에 적용하여 신뢰도를 평가하는 방법을 설명하고 직접적인 예제를 통해서 검증하였다. 5장에서는 결론과 앞으로 연구과제에 대하여 소개한다.

2. 설문조사 개요

산업자원부 기술표준원에서 산업용 소프트웨어의 품질 향상을 위하여 제품에 대한 평가를 실시하고 실시결과 우수한 제품에 대하여 인증을 부여하는 제도를 도입하였다.[1,2,3] 인증사업을 제도화 하기 위해서 먼저 2차례에 걸친 시범 사업을 실시하여 일정기간 산업용 소프트웨어의 시험 인증을 희망하는 업체를 대상으로 신청서를 받고 개발 제품에 대해서 시험을 실시하였다. 시험을 통해서 우수 제품에 대해서는 ES (Excellent Software)라는 인증을 부여하였다. 이러한 과정을 통해서 소프트웨어의 제품 평가에 대한 것을 확인하고 현재는 본 인증을 통해서 제품에 대한 평가를 실시하고 있다. 평가 기준은 ISO/IEC 9126[1]의 평가 메트릭을 이용하고 있으며 이 평가 메트릭에서는 소프트웨어 제품의 6개 품질 특성인 기능성, 신뢰성, 사용성, 효율성, 유지보수성, 이식성을 평가하고 있으며 시험을 통해서 정량적으로 평가 결과를 제시하고 있다. 특히 6개의 품질특성 중에서 신뢰성에 대한 평가는 상당히 어려움이 많아 계속적인 연구가 진행되어지고 있다. 본 연구도 6개의 품질특성 중에서 신뢰성에 대한 평가를 위해서 설문조사를 실시한 것이며 앞으로도 신뢰성에 대한 평가 연구는 계속적으로 진행되어질 것이다. ISO/IEC 9126에서 제

시하는 신뢰성은 성숙성(Maturity), 오류허용성(Fault Tolerance), 회복성(Recoverability), 준수성(Compliance)로 구성되어 있다. ISO/IEC 9126은 총 네 개의 부분으로 나누어져 있으며 ISO/IEC 9126-1에서는 전체적인 소프트웨어 품질 평가에 대해서 설명하고 있고 ISO/IEC 9126-2에서는 소프트웨어 외부 품질에 대한 평가 메트릭을 제시하고 있다. ISO/IEC 9126-3의 내부 메트릭은 설계나 코딩 도중에 실행할 수 없는 소프트웨어 제품에 대하여 적용할 수 있는 것으로 소프트웨어 제품을 개발할 경우에 그 중간 제품에 대해서 모의실험으로부터 추출가능한 본질적인 속성을 측정하기 위하여 내부 메트릭을 이용하게 되어 있다. ISO/IEC 9126-4에서는 사용에 있어서의 품질에 대해서 설명하고 있다. 본 연구에서 측정 방법을 제시한 소프트웨어 신뢰성이란 것은 소프트웨어가 특정한 시간동안 주어진 환경 하에서 고장 없이 작동할 확률을 의미하는 것으로 소프트웨어 신뢰도라는 것은 확률적인 의미로 설명되어질 수 있으며 같은 소프트웨어라 하더라도 다른 환경조건에서 사용되어진다면 신뢰도가 다르고 하드웨어와 같이 시간에 의존하는 것으로 구별되어진다. 소프트웨어의 특징은 고장이 발견되었을 경우 그 고장을 분석하고 고장을 제거하고 나면 같은 이유로 고장이 다시 발생되어지지 않

는다는 것과 소프트웨어를 복사하게 되어지면 동일 소프트웨어를 가지고 사용자가 작업을 처리하게 되어지면 작업 결과는 항상 같은 결과를 가지고 온다는 것이 하드웨어와 다른 특징이다. 본 연구를 위해서 3개 업체의 58명을 대상으로 설문조사를 실시하였다. 2004년 8월 1일부터 일주일간 실시를 하였으며 조사방법은 우편조사와 전자메일을 이용하였다. 참석자는 총 58명으로 남자 56명과 여자 2명이 참석을 하였으며 응답 대상자는 인증을 신청한 업체의 개발자와 인증제품의 사용자를 중심으로 조사를 실시하였다. 응답자의 연령대는 26세에서 30세가 48.3%였으며 31세에서 35세까지가 27.6%를 차지하였다. 본 설문조사에 참여한 응답자의 74.2%가 인증의 필요성에 대한 조사에서 인증이 필요하다는 반응을 보였으며 인증 후 제품에 대한 신뢰도는 어떻게 되리라고 생각하는가 하는 질문에서는 81%의 응답자가 신뢰도가 높아질 것이라는 반응을 보여서 인증을 받고 나면 제품의 신뢰도가 향상되어 지리라는 의견을 보이고 있다. 현재 인증을 신청한 업체의 경우 하나의 제품에 대하여 인증을 받고 나면 계속적으로 추가적인 인증을 신청하는 것으로 조사되었다. 소프트웨어 제품을 구입하는데 있어서 가장 고려하고 있는 사항이 무엇이라는 질문에는 제품의 품질 기능이라고 응답한 응

〈표 1〉개발자와 사용자의 일반사항에 대한 평균비교표

소프트웨어 품질특성의 전반적인 사항에 관한 질문	개발자	사용자
타 소프트웨어와 가격면에서 만족한다	6.14	5.68
작업의 효율면에서 사용자 욕구를 충족한다	6.14	5.68
업무의 효과를 높일수 있다고 할 수 있다	6.14	5.68
운용면에서 변경이 용이하다	5.71	5.81
본 소프트웨어는 익히기 쉽다	5.71	5.75
본 소프트웨어의 사용을 적극 추천할 수 있다	6	5.87
본 소프트웨어는 인증을 받는데 손상이 없다	6.29	6.13
이번 기회에 인증에 대해 중요성을 느꼈다	6	6.13
인증혜택은 점차적으로 확대해 가야한다고 생각한다	6.14	6.25
산업용 소프트웨어는 정부기관의 인증이 중요한 척도이다	6	6.19
본 소프트웨어의 전체적인 만족도를 평가하면 몇점정도인가	89.14	86.5

답자가 전체의 87.9%를 차지하여 대부분이 제품의 품질에 상당히 많은 관심을 가지고 있는 것으로 조사되었으며 6개의 품질 특성 중에서 어떤 품질 특성이 산업용 소프트웨어의 평가에서 중요하다고 생각하는가 하는 질문에는 신뢰성이란 응답자가 전체의 55.2%를 차지하여 6개의 품질 특성 중에서 신뢰성이 가장 중요한 품질 특성이라는 것을 알 수 있었다.

다음은 제품의 일반적인 사항을 알아보기 위하여 몇 개의 항목을 설문문항으로 구성하여 7점 척도의 순위자료로 평가를 실시하였다. 각 문항별로 매우 만족하지 않는다는 1점으로 하고 매우 만족한다는 7점으로 하여 전체적인 반응을 분석하였다.

표 1의 내용을 정리하여 보면 소프트웨어의 품질 특성의 전반적인 사항에 대한 조사결과 업무의 효과면에서 개발자의 만족도가 높은 것으로 조사되었으며 인증에 대한 내용면에서는 개발자보다는 오히려 사용자에게 많은 관심이 있는 것으로 조사되었다. 사용자 관점에서 인증혜택이 점차적으로 확대해 나가야 한다는 의견과 산업용 소프트웨어는 정부기관의 인증이 중요한 척도라는 것에 강한 반응을 보이고 있는 것으로 조사되었다.

표 2는 설문조사에 참여한 3개 업체를 대상으로 제품에 대한 품질 특성중에서 어느 정도를 만족하는지 만족도를 조사한 결과이다. 조사결과는 7점 만점으로 순위척도 형태로 조사한 것이며 조사결과에 대한 테이블의 값은 100점 만점으로 환산한 값이다. A사인 경우는 신뢰성에 가장 만족도가 높으며 유지보수성에 상당히 만족도가 낮은 것으로 조사되었으며 B사의 경우는 신뢰성과 기능성에는 만족도가 높으나 이식성에 만족도가 낮은 것으로 조사되었고 C사의 경우는 기능성에 만족도가 높으나 유지보수성에 만족도가 낮은 것으로 조사되었다. 이것으로 보아 각 회사의 제품별 품질특성에 느끼는 만족도에는 차이가 있는 것으로 분석되어지며 특히 C사의 경우 유지보수성에 상대적으로 낮은 만족도를 보

이고 있어 개선이 요구되어진다. 다음은 각 회

〈표 2〉 회사별 품질 특성에 대한 만족도 비교표

품질특성	A사	B사	C사
신뢰성	83.75	84.45	84.50
기능성	80.14	84.07	85.71
사용성	78.36	78.57	79.75
효율성	79.93	77.07	84.93
유지보수성	75.46	77.54	71.04
이식성	81.25	76.50	76.39

사별로 제품에 대하여 6개의 품질특성 중에서 무엇이 가장 중요한가 하는 중요도에 대한 조사 결과이다. 조사방법은 순위척도 방법으로 7점만점으로 조사를 하였으며 조사결과에 대하여 100점 만점으로 환산한 값이다.

〈표 3〉 회사별 품질특성에 대한 중요도 비교표

품질특성	A사	B사	C사
신뢰성	87.63	86.75	89.06
기능성	84.82	80.07	83.14
사용성	81.46	79.32	79.57
효율성	83.07	77.36	81.36
유지보수성	80.57	76.18	75.39
이식성	79.25	76.04	76.39

각 회사별로 품질특성에 대한 중요도 평가에는 평균차이가 있음을 알 수 있다. 즉 A사 제품의 경우 신뢰성이 가장 중요하다는 평가를 하였으며 이식성이 중요도가 가장 낮은 것으로 평가되었다. B사의 경우는 신뢰성이 가장 중요하고 유지보수성과 이식성이 중요도가 낮은 것으로 조사되었고 C사의 경우는 신뢰성이 가장 중요하고 유지보수성이 가장 중요도가 낮은 것으로 평가하였다. 이러한 중요도에 대한 평가 조사는 앞으로 계속적으로 진행할 계획이다. 현재 산업자원부 기술표준원에서 실시하고 있는 산업용 소프트웨어의 품질평가에서는 내장형 소프트웨어, 패키지 소프트웨어, 시스템 소프트웨어로 나누어

분류하고 있으나 각 품질 특성에 대한 시험을 통해서 얻어진 결과를 품질 특성 별로 다르게 가중치를 적용하는 방법을 고려해야 함을 연구 중에 있다. 각 분류별 특징을 고려하여 품질특성에 맞는 평가방법을 고려하기 위하여 가중치에 대한 연구는 계속적으로 진행되어질 것이다. [14]

3. 소프트웨어 신뢰성 성장 모델

소프트웨어 신뢰도 성장 모델에 대한 연구와 모수 추정에 대한 연구는 1970년 이후로 상당히 많은 진행이 이루어졌다. 확률적 모델을 기초로 하여 여러 환경에서 소프트웨어 신뢰도를 평가하는 방법에 대한 연구가 진행되었다. 하드웨어 신뢰성에 대한 기본적인 가정과 상당히 다른 형태에서 출발되어진 소프트웨어 신뢰도 성장 모델에 대한 연구는 모델을 추정하고 추정 모델에 대하여 모수를 추정하고 추정된 모수를 이용하여 소프트웨어의 신뢰도를 평가하였다. 본 장에서는 소프트웨어 신뢰도 성장 모델에 대하여 소개하면 아래와 같다.

3.1. Jelinski-Moranda 모델

Jelinski-Moranda에 의해서 연구되어진 소프트웨어 신뢰성 성장 모델은 지수함수를 기초로 하여 이루어졌으며 이러한 지수함수 모델에서 최우추정법을 이용하여 모수를 추정하였다. 아래의 식 (1)에서 N_0 는 소프트웨어가 개발 당시 보유하고 있는 소프트웨어 총 오류수이고 ϕ 는 소프트웨어 실패 강도이다.

$$f(t) = (N_0 - i + 1)\phi \exp(-(N_0 - i + 1)\phi t) \quad (1)$$

위의 식 (1)에 제시된 Jelinski-Moranda(JM) 모델은 소프트웨어 신뢰성 성장 모델의 가장 시초라고 할 수 있으며 모델에 대한 기본 가정은 아래와 같다.

- (1) 초기 소프트웨어 실패의 개수는 알려지지 않는 상수이다
- (2) 발견된 오류는 즉시 제거되어지며 다시 새로운 오류를 만들어 내지 않는다
- (3) 실패사이의 시간은 독립이며 지수함수 분포를 따른다
- (4) 모든 남아있는 소프트웨어의 오류는 항상 같은 실패강도를 가진다

위의 4가지 가정을 살펴보면 JM 모델은 고장의 크기가 고장 발생 순서와 관계없이 모두 같다고 하였으며 고장 발생사이의 시간이 독립이라는 가정 하에 모델을 제안하고 연구하였다. 위의 모델에 대하여 실패율을 구하면 아래와 같다.

$$\lambda(i) = \frac{df(t)}{dt} = (N_0 - i + 1)\phi \quad (2)$$

위의 JM 모델의 신뢰도 함수를 구하면 아래와 같다.

$$R(t) = \exp(-(N_0 - i + 1)\phi t) \quad (3)$$

JM 모델의 모수추정은 최우추정법을 이용하였으며 추정 결과는 아래의 식 (4)와 같다.

$$\begin{aligned} L(n_1, \dots, n_k) &= I_{i=1}^k \phi(N_0 - i + 1) \exp(-\phi(N_0 - i + 1)t_i) \\ \ln L(n_1, \dots, n_k) &= \ln I_{i=1}^k \phi(N_0 - i + 1) \exp(-\phi(N_0 - i + 1)t_i) \\ &= k \ln \phi + \sum_{i=1}^k \ln(N_0 - i + 1) - \phi \sum_{i=1}^k (N_0 - i + 1)t_i \\ \frac{\partial \ln L}{\partial N_0} &= \frac{\partial \ln b}{\partial \phi} = 0 \\ \hat{\phi} &= \frac{n}{\sum_{i=1}^k (N_0 - i + 1)t_i} \\ \sum_{i=1}^k \frac{1}{(N_0 - i + 1)} - \frac{k \sum_{i=1}^k t_i}{\sum_{i=1}^k (N_0 - i + 1)t_i} &= 0 \end{aligned} \quad (4)$$

일반적으로 위의 값은 수치해석 절차에 의해서 구해지며 모수를 정확히 추정하기 위해서는 초기치의 값을 잘 주어야 하며 이 경우 초기치는 아래의 식 (5)의 조건을 만족하여야 한다.

$$\frac{\sum_{i=1}^k (i-1)t_i}{\sum_{i=1}^k t_i} > \frac{n-1}{2} \quad (5)$$

3.2. NHPP 모델

가장 보편적으로 사용하고 있는 소프트웨어 신뢰성 성장 모델의 한 예로 S-Shaped NHPP 모델의 예제를 살펴보면 아래와 같다.

$$m(t) = a[1 - (1 + bt)\exp(-bt)], b > 0$$

S-Shaped NHPP 모델에서 고장률과 남아 있는 오류 수에 대한 함수는 아래 식 (6)과 같다.

$$\begin{aligned} \lambda(t) &= \frac{dm(t)}{dt} = ab^2 t e^{-bt} \\ \overline{EM(T)} &= m(\infty) - m(t) \\ &= a - a(1 - (1 + bt)e^{-bt}) = a(1 + bt)e^{-bt} \end{aligned} \quad (6)$$

위에 제시된 S-shaped NHPP 모델에서 t 시험시간 뒤에 남아있는 오류의 수($\overline{M(t)}$)가 k 일 확률은 다음과 같다.

$$P(\overline{M(t)} = k) = \frac{\exp(-((m(\infty) - m(t))) (m(\infty) - m(t))^k)}{k!} \quad (7)$$

제시한 모델의 신뢰도 함수는 식 (8)과 같다.

$$\begin{aligned} R(t,s) &= \exp(-[m(t+s) - m(s)]) \\ R(t,s) &= \exp(-a((1+bs)e^{-bs} - (1+b(s+t))e^{-b(s+t)})) \end{aligned} \quad (8)$$

제시된 모델에 대한 모수추정결과는 최우추정법을 사용하였으며 결과는 아래의 식과 같다.

일반적으로 위의 b 값은 a 값의 결과를 대입하여 얻어진 수치해석 절차에 의해서 구해진다.

$$\begin{aligned} L(n_1, \dots, n_k) &= \prod_{i=1}^k \frac{[a(1 + bs_{i-1})e^{-bs_{i-1}} - a(1 + bs_i)e^{-bs_i}]^{n_i} \exp(-a(1 + bs_{i-1})e^{-bs_{i-1}} - a(1 + bs_i)e^{-bs_i})}{n_i!} \\ \ln L(n_1, \dots, n_k) &= \ln \left(\prod_{i=1}^k \frac{[a(e^{-bs_{i-1}} - e^{-bs_i})]^{n_i} \exp a(e^{-bs_{i-1}} - e^{-bs_i})}{n_i!} \right) \\ &= \sum_{i=1}^k n_i \ln a + \sum_{i=1}^k n_i \ln((1 + bs_{i-1})e^{-bs_{i-1}} - (1 + bs_i)e^{-bs_i}) \\ &\quad - \sum_{i=1}^k (a(1 + bs_{i-1})e^{-bs_{i-1}} - a(1 + bs_i)e^{-bs_i}) - \ln(n_i!) \\ n &= \sum_{i=1}^k n_i \\ \frac{\partial \ln L}{\partial a} &= \frac{\partial \ln b}{\partial b} = 0 \\ a &= \frac{n}{1 - (1 + bs_k)e^{-bs_k}} \\ \sum_{i=1}^k \frac{n_i (bs_i^2 e^{-bs_i} - bs_{i-1}^2 e^{-bs_{i-1}})}{((1 + bs_{i-1})e^{-bs_{i-1}} - (1 + bs_i)e^{-bs_i})} - \frac{ns_k^2 e^{-bs_k}}{1 - (1 + bs_k)e^{-bs_k}} &= 0 \end{aligned} \quad (9)$$

4. 모델적용 실험결과

ISO/IEC 9126의 국제 표준 규격에서 신뢰성 (Reliability)에 관련된 평가항목은 4개의 부특성 즉, 성숙성(Maturity), 결함허용성(Fault Tolerance), 회복성(Recoverability), 준수성(Compliance)으로 나누어져 있다. 이러한 부특성은 다시 부특성별 신뢰성을 평가할 수 있는 평가 메트릭으로 구성되어 있다. 특히 부특성 중 성숙성의 경우 소프트웨어 신뢰성 성장 모델을 이용하여 결과를 예측하도록 메트릭이 구성되어 있다. 본 장에서는 이와 같은 메트릭을 통한 시험 결과를 얻기 위해서 모의시험 결과를 제시한다.

4.1. 본 연구 결과의 적용과정

$$f(t) = (N_0 - i + 1)\phi \exp(-(N_0 - i + 1)\phi t)$$

본 연구는 위의 Jelinski-Moranda 모델에 적용하여 신뢰도를 평가하여 보았으며 먼저 위의 모델에서 적용방법은 다음에 의해서 실시하여 보았다.[15]

- (1) 시험단계에서 각 날짜 별로 오류수를 측정한다.
- (2) 발견된 오류는 즉시 제거되어지며 다시 새로운 오류를 만들어 내지 않고 오류 발생 사이의 시간은 독립임을 가정한다.
- (3) 정확한 오류시간의 예측이 불가능하므로 단위 시간당 발생한 오류수를 중심으로 하여 포아송분포에서 평균값을 주고 실패시간을 구해낸다.
- (4) 실패시간을 이용하여 총 오류수와 실패강도를 예측한다.
- (5) 미지의 모수에 대한 값을 신뢰도 함수에 대입하여 신뢰도를 측정한다.
- (6) 총 남아있는 오류 수에 대비하여 다음에 발생할 평균오류시간을 구해낸다.

첫 번째 예제의 시험데이터는 6일간 제품을 시험하면서 발견한 오류 수에 대한 자료를 이용하여 시험시간을 예측한 자료이다. 본 데이터는 시험기관에서 실제 시험을 의뢰한 소프트웨어 제품의 시험기간동안 시험을 통해서 발견된 오류에 대한 자료이다.

〈표 4〉 시험날짜별 발견된 오류 수(첫번째 예)

시험	1	2	3	4	5	6
오류수	8	12	19	11	5	2

이 시험자료는 6일 모두 하루에 8시간씩 시험하여 얻어진 자료이다. 단위시간당 발견된 오류의 수를 중심으로 포아송과정으로 난수를 발생시켜서 발생시간에 대한 자료를 구하였다. 6일간 얻어진 고장 데이터의 시간에 대한 자료를 난수를 통해서 발생한 예측 고장시간에 대한 자료를 신뢰도를 측정하기 위한 제품에 대한 고장시간의 자료로 활용한다.

일반적으로 시험을 하면서 오류 발생 시간에 대한 자료를 오류 발생 시간마다 기록하기 힘들다. 이와 같은 이유 때문에 각 회사에서 자체 시험을 실시하고 있거나 시험을 담당 기관에서 시험을 하고 있는 동안에 발생한 오류 수를 조사하였다. 이와 같이 조사된 오류 수는 단위 시간당 발생한 오류수를 중심으로 포아송 분포를 이용하여 시험의 오류 시간에 대한 자료를 모의 실험하여 구해내었다. 구해진 결과의 시간을 소프트웨어 신뢰성 성장 모델에 적용하여 신뢰도를 평가하였다. 총 57개의 고장 시간에 대한 자료를 이용하여 소프트웨어 신뢰도 성장 모델에 적용하여 모수를 추정하고 추정된 모수를 이용하여 신뢰도를 예측하였다. 확률적 성질을 기반으로 제안된 모델로는 크게 고장발생시간을 중심으로 하는 경우와 발생된 고장수를 중심으로 연구되어진 모델로 나누어 볼 수 있다. 고장이 발생된 시간을 중심으로 하여 소프트웨어 신뢰도 성장 모델을 연구한 대표적인 확률 모

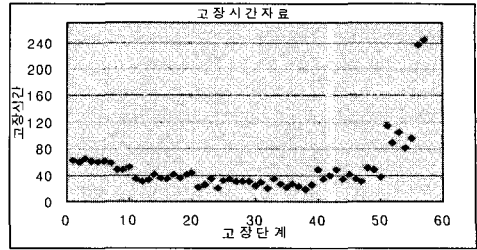
〈표 5〉 고장시간자료(오류수가 증가하다 감소하는경향)

단위: 분

시험날짜	오류발생시간
첫째날	63, 60, 65, 61, 60, 61, 59, 49
둘째날	48, 52, 35, 31, 33, 41, 36, 35, 41, 36, 41, 43
세째날	22, 25, 34, 21, 32, 34, 31, 31, 31, 24, 29, 20, 35, 27, 22, 27, 23, 19, 25
네째날	49, 35, 39, 48, 34, 41, 34, 30, 51, 48, 37
다섯째날	115, 89, 105, 82, 96
여섯째날	237, 245

텔로는 Markov 소프트웨어 신뢰도 성장 모델을 들 수 있으며 고장수를 중심으로 하여 모델을 연구한 대표적인 것으로는 포아송 분포의 성질을 이용한 비동질적 포아송과정 (Non-Homogeneous Poisson Process:NHPP)의 소프트웨어 신뢰도 성장 모델을 들 수 있다. 또한 이러한 모델에서 사전의 경험을 이용하여 모델과 모수를 추정하는 베이지안적 입장의 모델에 대한 연구가 발표되었으며 이러한 모델들은 모두 확률적인 접근 방법에 따라서 다양하게 연구 되어졌다. 제시한 6일간의 시험 자료를 각 시스템별로 변화를 보기 위해서 산점도를 그려보았다. 가로축은 고장이 발생되어진 단계를 나타내었고 세로축은 고장발생 시간(단위:분)을 나타내었으며 그래프의 제시된 형태를 살펴보면 고장 발생에 있어서 고장 발생 시간은 서서히 감소하다가 증가추세를 나타내고 있다. 대체적으로 고장 발생 시간의 형태는 모델을 결정짓는 중요한 역할을 하게 되므로 고장발생시간에 대한 플롯을 하여 보고 그 형태에 맞는 소프트웨어 신뢰성 성장 모델에 적용하는 것이 신뢰도를 정확히 예측하는 방법이 될 것이다. 그러기 위해서는 소프트웨어 신뢰도를 측정할 수 있는 고장 시간에 대한 자료를 이용하여 소프트웨어 신뢰도 성장곡선의 다양한 형태를 고려하여 고장시간을 가장 잘 설명할 수 있는 모델을 개발하기 위한 연구가 필요하다.

위의 자료를 이용하여 현재 까지 발견된 오류



〈그림 1〉 고장시간과 고장단계에 대한 그래프 (단위:시험단계)

수가 57개라는 자료를 이용하여 예측한 총 오류수는 88개로 예측되어지고 고장률은 0.000355로 추정되어진다. 미지의 모수를 이용하여 신뢰도를 측정하여 보면 이 단계에서 시험을 종료하였을 경우 0.499로 낮은 신뢰도를 보이고 있다. 이와 같은 결과는 총 57개의 데이터에서 초기 단계에서부터 계속적으로 고장이 빈번히 발생하다가 마지막 단계에서만 고장이 적게 발생되어지는 시간 분포이므로 이러한 현상이 발생되어지는 것으로 분석된다.

〈표 6〉시험날짜별 발견된 오류 수(두번째 예)

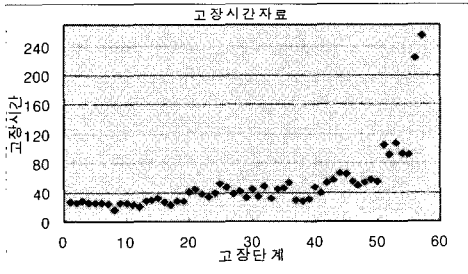
시험	1	2	3	4	5	6
오류수	19	12	11	8	5	2

〈표 7〉 고장시간자료(고장 시간이 점점 커지는 형태)

단위: 분

시험날짜	오류발생시간
첫째날	27, 25, 28, 25, 25, 26, 24, 17, 26, 25, 23, 20, 28, 29, 32, 27, 23, 28, 28
둘째날	41, 45, 38, 34, 39, 52, 47, 38, 42, 33, 44, 35
세째날	48, 32, 44, 46, 53, 29, 28, 30, 47, 39, 53
네째날	57, 66, 65, 55, 50, 53, 57, 55
다섯째날	104, 92, 107, 93, 92
여섯째날	224, 255

6일간 고장시간에 대한 자료를 이용하여 각 단계별 고장시간에 대한 분포를 이용하여 그래프를 그려보았다.



〈그림 2〉 고장시간과 고장단계에 대한 그래프
(단위: 시험단계)

위의 시험자료는 첫 번째 예제와 같이 총 발견된 오류 수는 57개로 동일하고 자료의 형태 변화만 차이가 있는 경우이다. 본 예제의 자료를 이용하여 모수를 추정하면 소프트웨어에 남아있는 총 고장수는 67개로 예측되어지며 예측된 고장률은 0.000661로 예측되어지고 이 경우에 신뢰도는 0.65로 추정되어진다. 위의 첫 번째 예제와 같이 총 오류의 수가 57개로 같은 상황이지만 시험날짜의 변화에 따른 오류 수에 따라서 예측되어지는 고장률과 신뢰도에는 차이가 있음을 알 수 있다. 두 개의 시험자료를 통해서 신뢰도를 측정하여 본 결과 같은 고장시간의 자료라 하더라도 고장시간의 분포에 따라서 신뢰도에 많은 영향을 준다는 것을 쉽게 파악할 수 있으며 신뢰도의 측정에 있어서는 고장시간의 분포와 고장시간의 간격에 상당히 많은 영향을 받음을 예측할 수 있었다.

본 연구는 소프트웨어 신뢰도 성장 모델에 적용하는 방법론에 대한 연구를 중심으로 하였으며 가장 보편적으로 알려진 모델에 적용하여 적용결과를 확인하였다.

5. 결론 및 향후 연구 과제

본 연구는 소프트웨어 신뢰도 성장 모델에 적용하여 신뢰도를 측정하는 방안에 대한 연구이다. ISO/IEC 9126에 적용하여 신뢰성에 관련된 품질 평가 메트릭을 측정하기 위해서는 소프트웨어 개발업체에서 개발 과정이 완료되고 시험단계

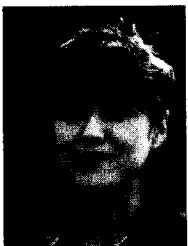
에 들어갔을 경우 시험 기간동안 발생되어진 소프트웨어 오류 시간에 대한 관찰 자료가 있어야 한다. 그러나 이러한 자료를 쉽게 구할 수 없으므로 본 연구에서는 인증을 신청한 업체에 대하여 시험기관에서 시험을 담당할 경우 시험기간 동안 발생한 오류의 수를 중심으로 신뢰도를 측정하는 방안에 대하여 제시하였다. 두 개의 예제를 통해서 확인된 것으로는 소프트웨어 신뢰도라는 것은 오류 발생 시간의 형태에 따라서 많은 영향을 받는다는 것을 알 수 있었다. 여러 가지 여건으로 인하여 동일 조건에서 소프트웨어 개발업체가 시험을 실시 할 수 없을 경우에는 현재 제안된 방법에 의한 신뢰도의 평가가 더욱더 신뢰성이 있을 것이다. 본 연구는 앞으로 시험기관에서 제시하는 오류 발생 데이터 자료를 더욱 확보하여 확보된 자료를 정확히 모델에 적용하는 방안에 대한 연구가 진행되어야 한다. 시험 데이터에 따라서 그 특성을 고려한 소프트웨어 신뢰도 성장 모델의 적용방안에 대한 연구는 지속적으로 이루어져야 한다. 본 연구에서 예제로 제시된 두개의 시험자료에서도 상당히 다른 현상을 보이고 있음을 알 수 있다. 기존에 보유하고 있는 소프트웨어 시험자료에 대하여 그래프를 그려 본 결과 본 연구에서 제시한 첫 번째 예제와 같이 S자 성장 곡선 형태로 시험자료가 주어짐을 확인하였다. 이러한 점을 고려하여 각각의 시험기관에서 얻어진 자료를 이용하여 그래프를 그려 보면 S자 성장 곡선의 형태를 보이고 있는 것들이 많으므로 본 연구는 앞으로 계속적인 모델 개발을 위해서 연구되어야 할 것이다. 소프트웨어 신뢰도 성장 모델에 대한 연구와 모수추정에 대한 연구, 신뢰도를 평가하기 위한 연구가 진행되어야 하며 업체에서 개발한 제품에 대하여 간단히 소프트웨어 신뢰도를 평가할 수 있는 방안에 대한 연구도 진행되어야 할 것이다.

참고 문헌

[1] ISO/IEC 9126, "Information Technology

- Software Quality Characteristics and metrics-Part 1,2,3.
- [2] ISO/IEC 14598, "Information Technology Software Product Evaluation-Part 1,2,3,4,5,6.
- [3] ISO/IEC 12119, "Information Technology Software Package Quality requirement and testing."
- [4] Velleman, P. & Wikinsn. L. "Normal, ordinal, interval, and ratio typologies are misleading, The American Statistician 47(1)," pp. 65-72, 1993.
- [5] Bain. L. J. & Engelhardt. M., 'Statistical Analysis of Reliability and Life-Testing Model, Marcel Dekker, Inc, 1991.
- [6] Bain. L. J. & Engelhardt. M., "Squential Probability Raio Tests for the Shape Parameter of a Nonhomogeneous Poisson Proecess," IEEE Trans on Reliability, R-31, pp. 79-83, 1982.
- [7] Bain. L. J. & Engelhardt. M., "Approximate Distributional Results Based on the Maximum Likelihood Estimators for the Weibull Distribution, J of Quality Technology, Vol-18, pp. 174-181, 1986.
- [8] Johnson. K., "Software Cost Estimation: Metrics and Models," Department of Computer Science University of Calgary, Albreta, Canada, <http://sern.ucalgary.ca/courses/seng/621/W98/Johnsonk/cost.htm>, 1998.
- [9] Keremer. C. F., "An Empirical Validation of Software Cost Estimation Models," Communications of ACM, Vol. 30, No. 5, pp. 416-429, 1987.
- [10] Low. G. C. & Jeffery. D. R., "Function Point in the Estimation of the Software Process," IEEE Trans on Software Eng., Vol. 16, pp. 64-71, 1990.
- [11] Laranjeira. L. A. . "Software size estimation of object-oriented systems." IEEE Trans. Software Eng., Vol 67, pp. 10-18, 1990.
- [12] Meyerhoff. D. & Laibarra. B., & R. V. D. Pouw Kraan, & A. Wallet, 'Software Quality and Software Testing in Internet Times, Springer, 2002.
- [13] Matson. J. E. & Barrett. B. E. & Mellichamp. J. M., "Software Development Cost Estimation Using Function Points," IEEE Trans. on Software Eng., Vol.20, No.4, pp. 275-287, 2000.
- [14] 정혜정., S/W신뢰성 평가기술 품질관리 적용방안, 산업자원부, 2003.
- [15] 정혜정., 소프트웨어 신뢰성 평가를 위한 평가방안, 산업자원부, 2005.
- [16] 황인수., ISO/IEC JTC1/SC7/WG10 WG10 동향조사보고서, 한국표준협회, 2003.

● 저 자 소 개 ●



정 혜 정 (Hye-Jung Jung)

1988년 8월 : 경북대학교 통계학과 조기졸업(이학사)
 1991년 2월 : 경북대학교 대학원 통계학과 졸업(이학석사)
 1994년 8월 : 경북대학교 대학원 통계학과 졸업(이학박사)
 1995년 3월~현재 : 평택대학교 정보통계학과 부교수
 관심분야 : 소프트웨어 신뢰성 공학, 소프트웨어 품질 평가, 소프트웨어 시험, 데이터마이닝 등
 E-mail : jhjung@ptu.ac.kr