Ranking of Software Reliability Growth Models using Greedy Approach

Neha Miglani¹ and Poonam Rana²

¹M Tech. Research Scholar, ²Senior Lecturer
Department of Computer Science & Engineering
Ambala College of Engineering & Applied Research, Devsthal
(near Mithapur), Ambala, India
E-mail- neha.miglani27@gmail.com, rana.poonaml@gmail.com

Abstract

A large number of software reliability growth models (SRGMs) have been proposed during the past thirty years to estimate software reliability measures such as the number of residual faults, software failure rate, and software reliability. Selection of optimal SRGM for use in a particular case has been an area of interest for researchers in the field of software reliability. Tools and techniques for software reliability model selection found in the literature cannot provide high level of confidence as they use a limited number of model selection criteria. There is therefore a need for evolving more efficient techniques. An effort has been made in this paper to review some of the well known techniques of this area and the possibility of developing a more efficient technique.

Introduction

In recent years software systems such as operating systems, control programs, and application programs have become more complex and larger than ever. It is quite natural to produce reliable software systems efficiently since the breakdown of the computer system, which is caused by software errors, results in a tremendous loss and damage for social life. Then, software reliability is one of the key issues in modern software product development. Although advances have been made towards the production of defect free software, any software required to operate reliably must still undergo extensive testing and debugging. This can be a costly and time consuming process, and managers require accurate information about how much software reliability is achieved as a result of a particular process in order to effectively manage their budgets and projects. A process, by which it is hoped that software can be made
more reliable may be modeled using Software Reliability Growth Models (for e.g., Generalized Goel Model, Goel-Okumoto Model, Gomperts Model, Inflection S-Shaped Model, Logistic Growth Model and so on).

Applying the SRGM’s to the observed software error data, the important software reliability measures, such as the number of errors remaining in the system and the software reliability function, can be estimated. These models enable software reliability practitioners to make predictions about the expected future reliability of software under development. Such techniques allow managers to accurately allocate time, money and human resources to a project, and assess when a piece of software has reached a point where it can be released with some level of confidence in its reliability [3]. An error made by a human being and results in a fault in the project. The manifestation of a fault, which means departure from what the software is supposed to do, is referred to as a failure (IEEE standard 782[9]). There is difference between reliability and fault content. A product may have a number of Faults, but these may be locked in paths that are seldom executed; then this product is considered to be reliable. Faults considered by reliability models are those that effect reliability under prevalent conditions, and not necessarily the total faults contents of the software.

Techniques and tools are needed for keeping track of the fault content and the reliability, as long as fault free software cannot be guaranteed. The customer, who buys the software system, need to know if the product fulfils the quality constraints put on it. The tools available for this are mainly software reliability models.

**Literature Review**

Today the number of existing models exceeds hundred with more models developed every year. Still there does not exist any model that can be applied in all cases. Models that are good in general are not always the best choice for a particular data set, and it is not possible to know in advance what model should be used in any particular case [6]. Over the past thirty years, many SRGMs have been proposed for estimation of reliability growth of products during software development process [1], [5], [7], [8], [10].

Many researchers like Musa et al. [4] have shown that some families of models have, in general, certain characteristics that are considered better than others.

Goel[2] and others[11],[12] started describing processes for which each model would be tested to see how well the model fits the data and predicts the future events. The assertion was that different models predict well only on certain data sets.

The power of several of these statistical tests has been evaluated for a variety of reliability models including those based on a non homogeneous Poisson process, and the Moranda model. Power of these tests has also been compared later.

**Proposed Approach**

In the present study we are considering the effectiveness of greedy search approach in ranking reliability models. A **greedy algorithm** is any algorithm that follows the problem solving heuristic of making the locally optimal choice at each stage with the
Greedy algorithms are characterized as being 'short sighted', and as 'non-recoverable'. They are ideal for problems which have 'optimal substructure'. Despite this, greedy algorithms are best suited for simple problems.

Designing of greedy algorithm is based on finding out the shortest path by using suitable algorithms and calculating its weight or distance from the origin, naming it as OPTIMUM and then comparing the values of different models with the optimum value.

Assume x is a candidate model and its objective value is Alt(x). Our aim is to find x which minimizes value of Alt(x).

In Mathematical terms, it can be represented as:

Minimize s |OPT-Alt(x)|
Subject to x.

Where Alt(x) represents SRGM alternative and s represent distance from optimum value ‘OPT’, OPT is desired optimal value.

Figure.1 describes how ranking of the models would occur by calculating the distances of different models from the optimum value OPT. All the models would lie in a feasible region named ACTIVE. By calculating the distance of these models from OPT, we will attain the objective values for different models and hence, models can be ranked on this basis.

The flowchart of the proposed technique is shown in figure2.

Example

We present here for illustrations Sharma et.al [3]. It targeted testing the suitability of the developed DBA method so that a comprehensive ranking of the alternative SRGMs could be made combining various attributes relevant to SRGMs for a data set.

The paper included NHPP SRGMs namely,

- Generalized Goel Model
- Goel-Okumoto Model
- Gomperts Model
- Inflection S-Shaped Model
- Logistic Growth Model
A dataset was taken from the open literature for evaluation, optimal selection and ranking of the NHPP SRGMs based on criteria named Bias, MSE, and MAE and so on. The dataset was collected from a subset of products for four separate software releases at Tandem Computers Company as shown in Table 1. The value of the comparison criteria are calculated using Least Square Estimation. Then, estimated and optimal values are used to compare the rankings of all the models based on values of comparison criteria.

Work in hand
Currently we are trying to rework ranking of these models using greedy approach and compare the results with those of Sharma et.al [3] and others. The work is still in progress and we hope to present the results in the conference.

Figures and Tables

Figure 1. Greedy Approach
Conclusions
The objective of this work is to develop a ranking technique which is based on greedy approach to rank different types of software reliability models and compare its performance with the currently available algorithms.

We also intend to apply this approach to some specific case studies of softwares.
References