Abstract

Intense global competition in the dynamic environment has lead to up-gradations of software product in the market. The software developers are trying very hard to project themselves as organizations that provide better value to their users. One major way to increase the market charisma is by offering new functionalities in the software periodically. But these intermittent add-ons in the software lead to an increase in the fault content. Thus, for modelling the reliability growth of software with these up-gradations, we must consider the failures of the upcoming release and the faults that were not debugged in the previous release. Based on this idea, a mathematical modelling framework for multiple releases of software products has been proposed. The model uniquely identifies the faults left in the software when it is in operational phase during the testing of the new code. The model has been validated on real data set.
Software Up-gradations and Optimal Release Planning

Now, since the proposed structure is dependent only on time, it can be categorized under one dimensional modelling outline. But the need of the hour is to consider other factors (available resources; coverage, etc) simultaneously. Therefore, using a Cobb Douglas production function we have extended our own modelling framework and developed a two dimensional software reliability growth model for multi releases which concurrently takes into consideration testing time and the available resources. Another major concern for the software development firms is to plan the release of the upgraded version. In a Software Development Life Cycle the testing phase is given a lot of importance. But testing cannot be done indefinitely, hence it is pertinent to find the optimal release time during testing phase. Too late an entry is likely to lead to significant loss of opportunity and on the other hand early release of any software product might hinder its growth due to lack of receptiveness of users towards new expertise. Therefore, timing plays a very important role. In software world we term this problem as Release Time Problem. Many release time problems with optimization criteria like cost minimization, reliability maximization and budgetary constraints etc. have been discussed in the literature. We have formulated an optimal release planning problem which minimizes the cost of testing of the release that is to be brought into market under the constraint of removing a desired proportion of faults from the current release. The problem is illustrated using a numerical example, and is solved using Genetic Algorithm. Further, we have also discussed the release time problem based on a new concept of Multi-Attribute Utility Theory that takes into consideration two conflicting attributes simultaneously. This framework has also been illustrated using a numerical example.

References

- Wood. &apos; Predicting software reliability &apos;, IEEE Computer, Vol. 9, pp. 69-77,
1996.

Index Terms

Computer Science Software Engineering
Keywords
Software Reliability  Multi Up-gradation  Multi Attribute Utility Theory  Software Reliability Growth Model

Cobb Douglas Production Function

Optimal Release Time