

A Systematic Literature Review on Software Fault Prediction and Fault Tolerance in Software Engineering

Abhijeet Arora¹, Aditya Sikka², Ramanathan L.³

¹²³ SCOPE - School of Computer Science and Engineering, VIT University, Vellore, India

Abstract: This paper surveys a systematized review of the already published Fault Prediction and Fault tolerance papers in the field of Software Engineering. This literature review contains a collection research papers of both the topics combined. Fault prediction and tolerance techniques for hardware failures has been developed to an usable extent and is an important part of any reliable system. Although, for software components, the strategies are not yet common and not widely developed and used. This paper focuses on the statistical algorithms and machine learning algorithms approaches for Software fault predictions. Also, it reflects fault tolerance techniques such as n-version programming and n-checking programming. This paper will eventually help the researchers to have an complete overview of the various methods, metrics, techniques applied in the respective field. Also, the new techniques /trends and future scope in the area has also been included.

Keywords : Fault prediction, tolerance, software quality, system, Programmer.

Introduction

Software fault prediction is one of the important aspects to be considered while developing a software. It helps to make the system more reliable. Amongst the other software predictions such as cost prediction, security prediction and others, fault prediction is the most important and the reason it has been researched the maximum through all these years. Having a fault prediction model helps in cost efficiency and more importantly time efficiency and also adds to the quality assurance of the software. The measures to tackle the faulty modules can be prepared beforehand in the scenario of any problem. It is very helpful in the development of a larger software where the probability and the frequency of faults can be more. Fault prediction helps in the maintainability of the software. Software fault tolerance on the other hand, comes into play after the fault has occurred. It ensures the continuous working of the software after the fault, adding to the reliability and increasing the dependability of the

system. It adds to the ability of working properly even when some of the internal component goes to a failure state. Fault can happen due to any design issue, functionality error or code error. The tolerance of the errors happening at the runtime is more than the compile time errors. Fault tolerant system are getting more importance today as they ensures the no-halt service mechanism. If the faults are not dealt with for long, then the consequences can be major and may disturb the actual outcome of the software.

Literature Survey :

Performance Evaluation Metrics for Software Fault Prediction Studies by

Cagatay Catal [1]: This paper states that there are only a few software modules that causes faults in the software system. That is the reason that only a few are with faulty labels and others with non-faulty labels, such datasets are termed as “imbalanced” and there are various methods through which we can evaluate their performance. The methods that are widely used are evaluation metrics; different researchers have used different evaluation parameter. Hence, it is very difficult to compare and contrast the work that is done in present to that which is done in the past. The researchers have used various evaluation parameters until now, and to select a common parameter for evaluation is critical in important in software engineering today.

An Efficient Software Fault Prediction Model using Cluster based Classification by Pradeep Singh, Shrish Verma

 [2];

This paper states that Software testing has been given a very limited budget and from that a small portion is being allotted to the fault prediction techniques evaluation. So, because of this reason data mining techniques are used. We take a previous evaluated prediction as benchmark and then for the same dataset perform the fault prediction using cluster classifiers. This new proposed model gives better results with superior probability of detection. This fault prediction method is also used for removing unimportant information

and refining the data. This method has some other characteristics as well like it is efficient, easy to use, comprehensible, effective and inexpensive as well.

Study of Predicting Fault Prone Software Modules by Ritika Sharma, NehaBudhija, Bhupinder Singh^[3] :

This paper reviews about the fault prone areas , if we are aware of the faulty areas beforehand this will save our resources and will also, help us in evaluation in a cost effective manner. Majority of the faults are present in a small area, so if we know about them it will be helpful and will also increase the quality of the software. The designing of fault prone area application is done by object orientation method. The approach of researchers is to just find out the best techniques through which the fault prone areas or modules can be found as early as possible. The methods used are complexity metrics and object orientation method. This will save time and will also, be cost effective.

Fault Prediction Modeling for Software Quality Estimation: Techniques by Taghi M. Khoshgoftaar , Naeem Seliya^[4] :

This paper reviews about software reliability. If the fault prone module prediction technique is fast and efficient it will bring high reliability and will also increase the quality of the software. It will also increase the accuracy of the software. There are various dependent and independent variables that are used as parameters of software metrics. Such parameters are decided differently by different researchers , hence, there is no way that a comparison can be done between any two methods but the aim is to have a technique that can detect faults in an efficient and inexpensive way and increase the accuracy and quality of the software.

Software Fault Prediction Using Single Linkage Clustering Method by K.C. Sujitha, S. Leninisha^[5] : This paper reviews about the acceptance testing. As a programmer is more concerned about his work getting passed through the acceptance test, which looks into aspects like quality and reliability. The models that are newly designed for fault prediction tells about the areas in which faults are present and then, it requires very less efforts and resources to locate them and also, to remove them. The methodology that is used in the new model is of simulation. But the new models should not compromise with the accuracy of the prediction and with the quality of the software. In future as well, we should try to invent technique that will stand successful on these grounds.

Data Diversity: An Approach to Software Fault Tolerance Paul E. Ammann And John C. Knight^[6] :

In this paper, the approach used for software tolerance is based on data diversity. This is on account that software faults generally occur under some exceptional cases, and for a few software a program might express its information and interior state in countless proportional ways. These perceptions propose getting a related data points in the data space, executing a similar code on these focuses, and afterward applying a decision algorithm to calculate the output of the system. Such methods utilise data diversity to endure the remaining software faults. Data diversity is orthogonal to the design diversity. The methodologies are not totally unrelated and different mixes are possible.

Software Fault Tolerance in the Application Layer By Yennun Huang And Chandra Kintala^[7] :

This research paper discusses about the fault tolerance happening the application layer. This type of fault tolerance means to detect and prevent the software faults in the application layer only because such faults are not handled by the operating system or the hardware layer of the system. We take only the flaws which cause the application procedure to hang or crash; they incorporate application software faults and the errors which go undetected in the other layers of hardware and operating system.

N-Version Programming : A Fault-Tolerance Approach To Reliability Of Software Operation by Liming CHEN^[8] :

In this research paper, the N-version programming is used as a technique for fault tolerance. In N-version programming, the different versions of the same software are made using the same specification which was initially used. N-version programming technique is characterised as the independent generation of $N > 2$ practically comparable projects, called "versions", from a similar initial specification. N variants of the program are autonomously created concerning the underlying specification. Wherever conceivable, distinct algorithms and languages of programming that all N adaptations of the program begin from a similar starting detail, which is adequately the "hard-core" of this technique. Its accuracy, culmination, and unambiguity must be guaranteed before the N-version programming.

Implementing design diversity to achieve fault tolerance by J.P.J. Kelly^[9] :

In this paper, they discussed about the faults in the software that are especially critical in an ongoing simultaneous framework which are recognized, and the utilization of

design is done to keep their occurrence inspected. There are Two ways to deal with implemented diversity, recovery-block and multi-version programming which are discussed in this research paper. The recovery-block method consolidates N differing versions organized in consecutive order, however the versions may execute concurrently. The multiversion-programming approach pardons all N forms in parallel, exploiting the repetitive processors prone to be accessible in any system which must tolerate all the faults related to hardware and software. Though distinctive, both methodologies require adequately various development situations and the software faults in the particular don't prompt to the similar mistakes in the software.

Exception Handling and Software Fault Tolerance by F. Cristian ^[10] :

This research paper discusses another technique of software fault tolerance using exceptional handling called Backward recovery. This paper discusses all definitions for the necessary ideas that can be make use of to make a base model for software exception handling and default exception handling in projects organized as chains of data abstractions. The cause and effect connection between design flaws and unsuccessful events are investigated and a class of design flaws for which backward recovery can give fault-tolerance efficiently is characterized.

System structure for fault tolerance by B. Randell ^[11] :

The paper presents, and discusses the reasons behinds technique to structure complex software by the method we call Recovery blocks, Fault-Tolerant techniques , interfaces , conversations. The point is to encourage the provision of known error detection and correction facilities which can adapt to the errors due to the design deficiencies, especially in the system software instead of the hardware malfunctioning of the components occasionally. The point is to encourage the provision of known error detection and correction facilities which can adapt to the errors due to the design deficiencies, especially in the system software instead of the hardware malfunctioning of the components occasionally.

Future Work

Software fault prediction has been studies more than a decade now but still needs some implementational changes to meet the complex software requirements for the

commercial datasets. It is in the top of the research areas and has to be continuously developed. Whereas software fault tolerance is still in its early deveoplment phase. The present technology or techniques do not fulfill the fault tolerance quality. As the complexities of the software increases, the tolerance techniques needs to get evolved to tackle the various issues.

Conclusion

A review on software fault prediction and tolerance is presented in this paper. The main objective of this systematic review is to identify various techniques used in software fault prediction and tolerance. The techniques used in the review helps to increase and improve the software quality. The software fault prediction and tolerance are significant processes since they help in reducing the directing test effort, reduces the budget cost, increases the reliability along with the quality of the software. The fault tolerance schemes which depend on the single or multiple design are covered here. The Single design programming techniques such as system structuring and exception handling are discussed. Multiple design methods depend on the presumption that the system built in a different way ought to fail and hence, if one of the similar version fails, at least one of them must give a successful result. In conclusion, a considerable measure of strategies and techniques have been created for successfully predicting or removing the faults from the software. The use of these techniques is moderately new to the range of fault prediction and tolerance. Besides, every technique should be unique for each application. This must also be according to the cost of the fault prediction and tolerance the customer wants. As a result of our present inability to create software without errors, the software fault prediction and tolerance will be significant processes . in the software systems.

References

1. Performance Evaluation Metrics for Software Fault Prediction Studies Cagatay Catal Istanbul Kultur University, Department of Computer Engineering, Atakoy Campus, 34156, Istanbul, Turkey, c.catal@iku.edu.tr
Newcastle upon Tyne Published in: · Journal IEEE Transactions on Computers
2. An Efficient Software Fault Prediction Model using Cluster based Classification ,Pradeep Singh and Shrish Verma. Article: An Efficient Software Fault Prediction Model using Cluster based Classification. International Journal of Applied Information Systems 7(3):35-41, May 2014
3. Study of Predicting Fault Prone Software Modules Ritika Sharma, NehaBudhija, Bhupinder singh IGCE,Abhipur(Mohali) IGCE,Abhipur(Mohali) IGCE,Abhipur(Mohali)
4. Fault Prediction Modeling for Software Quality Estimation: Comparing Commonly Used Techniques, Taghi M. KhoshgoftaarNaem Seliya, Florida Atlantic UniversityBoca Raton
5. Software Fault Prediction Using Single Linkage Clustering Method K.C. Sujitha, S. Leninisha PG Student, Dept. of IT, Easwari Engineering College, Chennai, Tamilnadu, India , Assistant Professor, Dept. of IT, Easwari Engineering College, Chennai, Tamilnadu, India
6. Data Diversity: An Approach to Software Fault Tolerance Authors: Paul E. Ammann Univ. of Virginia, Charlottesville John C. Knight Univ. of Virginia, Charlottesville, IEEE Transactions on Computers - Fault-Tolerant Computing archive Volume 37 Issue 4, April 1988
7. Software Fault Tolerance in the Application Layer YENNUN HUANG and CHANDRA KINTALA AT&T Bell Laboratories
8. N-VERSION PROGRAMMING : A FAULT-TOLERANCE APPROACH TO RELIABILITY OF SOFTWARE OPERATION Liming CHEN Algirdas AVIZIENIS University of California Los Angeles, CA 90024 USA Xerox Corporation Computer Science Department El Segundo, CA 90245 U
9. Implementing Design Diversity to Achieve Fault Tolerance Authors: John P. J. Kelly Thomas I. McVittie Wayne I. Yamamoto Published in: · Journal IEEE Software archive Volume 8 Issue 4, July 1991
10. Exception Handling and Software Fault Tolerance Author: F. Cristian Computing Laboratory, University of
11. System structure for software fault tolerance ,Author: B. Randell University of Newcastle upon Tyne, U.K.