Software Reliability Testing using Monte Carlo Methods

Harnam Singh
(Student)
Lovely Professional University

Preet Pal
(Asst. Prof.)
Lovely Professional University

ABSTRACT

Monte Carlo method is used in this paper to test the software reliability. The Monte Carlo method is a technique that can be used to solve mathematical or statistical problems. Monte Carlo simulation uses frequent sampling to determine the properties of some phenomenon. There have been very rare uses of such simulation methods for software testing. This paper provides an accurate algorithm for testing software based on Monte Carlo Methods. The results would decide the percentage reliability of the software. This scheme has a number of applications in financial analysis, Econometric, statistics, software testing, fault detection in circuits and many more.

Keywords—Monte Carlo method, Software testing, Document characterization.

1. INTRODUCTION

According to IEEE the definition of software reliability is as follows “the ability of a system or component to perform its required functions under stated conditions for a specified period of time”. Monte Carlo method was given by S. Ulam and Nicholas Metropolis in reference to games of chance, a well-liked attraction in Monte Carlo, Monaco [1] [2]. Computer simulation has to make use of computer models to imitate real life or make predictions. When you make a model with a spread sheet like Excel, you have a certain number of input parameters and a few equations that use those inputs to give you a set of outputs (or response variables). Monte Carlo simulation is a method for iteratively evaluating a deterministic model using sets of random numbers as inputs. This method is frequently used when the model is complex, nonlinear, or involves more than couple uncertain parameters. A simulation can typically involve in 10,000 evaluations of this model, a task which in the past was only practical using super computers. The Monte Carlo method is one of the methods for analysing uncertainty propagation, where the aim is to determine how random disparity, lack of knowledge, or error affects the sensitivity, performance and reliability of the system that is being modelled [3].

The feature of the Monte Carlo method makes it perfect for estimating the reliability of software systems. Different from many other mathematical models, system complexity is not relevant to the method. Along with exact structure of software we also have a software structure that is dynamic. Instead, system components need only be tested for failure during operation, which ensures that the components which are used more often contribute proportionally more to the overall reliability estimate.

In computerized adaptive testing, which includes the selection algorithm, controls of item exposure and criteria of termination Monte Carlo method used developed a new assessment technique for the computerized adaptive testing with the help of the available research methodologies [10].

Monte Carlo testing system used for the T-Rex planner which was developed at MBARI for AUV Control. Monte Carlo method is also used to test the constraint based planning system especially those which are used for the real time applications which large possible states which make it really hard to test by any other mathematical method. The harness of the test runs the planner iteratively, making the each run to goes through a series of random states just to find out the any uncover error or the bug in the system which are not discovered by the any manual methods and after finding those bugs it generate the report of the bugs which it find and send it to the programmers so that they can remove this bugs by changing the code according to the requirement [7].

Monte Carlo and Multilevel Monte Carlo is used for the fault tolerance of multilevel Monte Carlo Methods. These methods help to find out the partial differential equations with the inputs which are random [8].

Monte Carlo method can be used effectively into the direct calculations of the free energy. To calculate the energy over the wide range of temperature random walk in space of both particle coordinates and temperature is used [9].

Software testing is significant to verify reliability of software system. It is important to consider how testing can be performed more effectively and at a lower cost through the use of systematic and automated methods. Since exhaustive testing, the checking of all possible inputs is usually difficult and expensive, it is important for testers to make best use of their limited testing resources and generate good test cases which have the high probability of detecting undiscovered errors.

Instead of attempting to solve integrals analytically, the method estimates an integral by firing random points at the function. The law of large numbers predicts that if more random points are selected, the ratio of points below to the points above the function will approximately be equal to the ratio of the area beneath the function to total area of the sample space from which the random points are drawn [4].

Software reliability is the probability of failure free software operation for a specified period of time. Predicting the software system reliability at some point in the future based on data is already available. It is one of the important challenges of the software projects. The implicit objective of the management is to achieve the software system reliability goals with minimal costs and schedule of the projects. Therefore, prediction in this sense is very useful in supporting software project management to achieve this objective.

For testing purposes, character recognition system is implemented. This system works on a textual data and will provide the category of the document with the help of a dictionary. Dictionary will act as the back bone of whole implementation. It is to be made very sure that the character recognition system implemented should work absolutely fine. Next, we will provide certain percentage of input from the input text document. Based on that percentage, the system will calculate a suitable name for the document. At higher percentages, the software will provide more accuracy and reliability where as at lower percentages, accuracy and reliability will decrease. The proposed algorithm successfully computes document's category at higher percentages.

In [5], the author proposes a software random testing scheme based on Markov chain Monte Carlo (MCMC) method. There are many issues that happen in software testing like, while making test cases we have to use the prior knowledge of experienced testers and the information obtained from the preceding test outcomes. The concept of Markov chain Monte Carlo random testing (MCMCRT) is based on the Bayes approach to parametric models for testing of software, and can use the earlier knowledge and the information on preceding test outcomes for their parametric estimation. In numerical experiments, we observe the effectiveness of MCMCRT with ordinary random testing and adaptive random testing. In this paper, we propose a new software random testing method; Markov chain Monte Carlo random testing (MCMCRT) based on the statistical model using the prior knowledge of program semantics. The main advantage of MCMCRT is that it allows the use of statistical inference measures to compute probabilistic aspects of the testing process. The test case generation procedure is done by using Markov chain Monte Carlo (MCMC) method which generates new test case from previously generated test cases based on the construction of software testing model like input domain model.

This paper is organised as follows. Section I is "the introduction concerning Monte Carlo and software testing" .Section II talks about "software reliability and issues related to it". Section III is all about "the prerequisite knowledge". Section IV describes the "proposed algorithm for computation". Section V is "results and conclusions" and references are at the last.

2. SOFTWARE RELIABILITY

There are various views for reliability of software. If you want to judge software from the point of view of a manager, various issues like customer satisfaction, repeated costumers, ethicality of the software and its legal liability comes into mind. If you judge software from a costumer's view, the software should save time, should increase the efficiency, should be correct and should be reliable. Software reliability is difficult to calculate because without considering the program evolution, failure rate is statistically non-existent. There are many possible causes for design defects to fail. Errors can occur without prior warnings. One cannot improve software quality if identical software components are used .Although periodic restarts can sometimes help to fix problems. Errors also occur by incorrect statements, incorrect logic, or incorrect input data. For which the software may require infinite testing. There are many models to choose from when calculating software reliability and Monte Carlo methods is one of them.

Software reliability testing is a field of testing which deals with checking the ability of software to function under given environmental conditions for a particular amount of time, taking into account the precision of the software. In software reliability testing, problems are discovered regarding software design and functionality and assurance is given that the system meets all requirements. Software reliability is the probability which determines that software will work properly in a specified environment for a given time. Using equation (1), the probability of failure is calculated by testing a sample of all available input states.

$$Probability = \frac{Number\ of\ cases\ failed}{Total\ number\ of\ cases\ taken\ under\ consideration}$$
......(1)

The set of all possible input states is called as input space. To find reliability of software, we need to look out for output space from given input space and software [6].

3. PREREQUISITE KNOWLEDGE

Software Reliability Testing requires checking features provided by that software, the load that software can manage, and regression testing [11].

A. Feature Test

Feature testing checks the attributes provided by the software which are conducted in the following steps [11]:

1) Every operation in the software is executed once.

2) Interaction between the two operations is minimised

3) Each operation is verified for its proper execution

B. Load Test

Under Maximum work condition this test is performed to check system. Any software performs better up to some amount of workload, software degradation starts after a resonant value. For example, a web site can be tested to see how many simultaneous users it can support without performance degradation. Testing is mainly helpful for Database and Application server's applications as .in the case above. Load testing also requires testing of software's performance which checks how well some software works under workload [7].

C. Regression Test

Regression testing is used to ensure if any new bugs had been introduced through previous bug fixes. Regression testing is held after every change or update in the features of the software. This testing is periodic, depending on the features of the software [11].

4. PROPOSED ALGORITHM

The algorithm developed so far is a mix of software reliability computation and Monte Carlo simulations. Random inputs are taken from a word file and then document categorization is employed on these random inputs. When the extent of these inputs is increased, the system tends to behave more accurately. Thus using the concept of Monte Carlo methods at a point where the system incorporates 100% randomized inputs, system works at its 100% capacity to provide the most accurate result. Hence, finally the implemented software is tested at different percentages of random inputs provided to it and shows better results when the percentage of random inputs is increased.

The proposed algorithm is being implemented on document characterization system. Following are the steps involved for its computation:

1) Implement a document characterization system

2) This system is able to work efficiently on any text inputted. This system will provide the category of the text document inputted on the basis of a dictionary file saved in excel format.

3) Now, Monte Carlo will be implemented on top of this system

4) User will be confirm to input a percentage values *P*

5) Now, *P*% random words will be selected from the text document, to be tested into the document categorization system.

6) The accuracy of document characterization is directly proportion to the value of p entered.

7) At each value of *P*, user will get a randomized display of words from each of the topic from the dictionary, based on which the category of the input text can be determined.

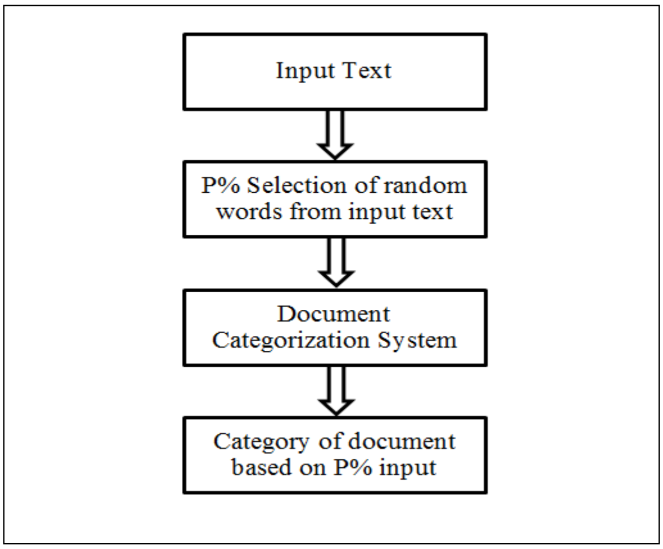
Input Text
P% Selection of random words from input text
Document Categorization System
Category of document based on P% input

Fig 1: Proposed algorithm for computation

5. RESULTS AND CONCLUSIONS

The whole system was implemented on MATLAB. Figure 2 shows the results when the data was selected at 77% of the random text of input text document. Here, the input text document was of "Engineering" topic. Now 77% random words were selected from this document and were fed to the document categorization system. Based on the 77% random data availability, the system was able to calculate the correct category for the document. Figure 4 shows results for 7 topics from the dictionary. The more the white portion in an image, the more number of words are present from that topic and hence the image with most white portion will be the final category of the input text. Figure 3 shows the percentage of document belonged from different categories and figure 5 shows document maximum belonged to engineering category. When we'll increase the percentage value, categorization will tend to be more accurate. Hence the system is able to efficiently calculate the category of the document inputted.The topics whose no words are included in input text remainscomplete black.

There may be cases when the random input will contain words sufficient amount of words belonging to the desired topic, that even at lower percentages the system calculates the correct topic for the input text document.

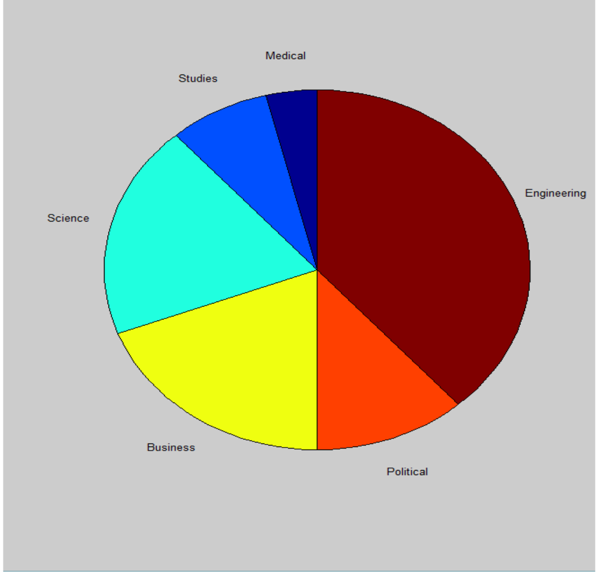

Fig 2: Shows result from different categories

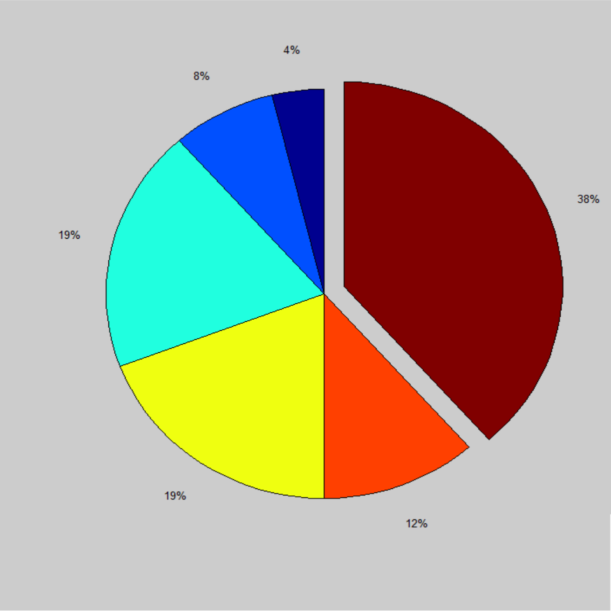

Fig 3: Shows percentage result from different categories

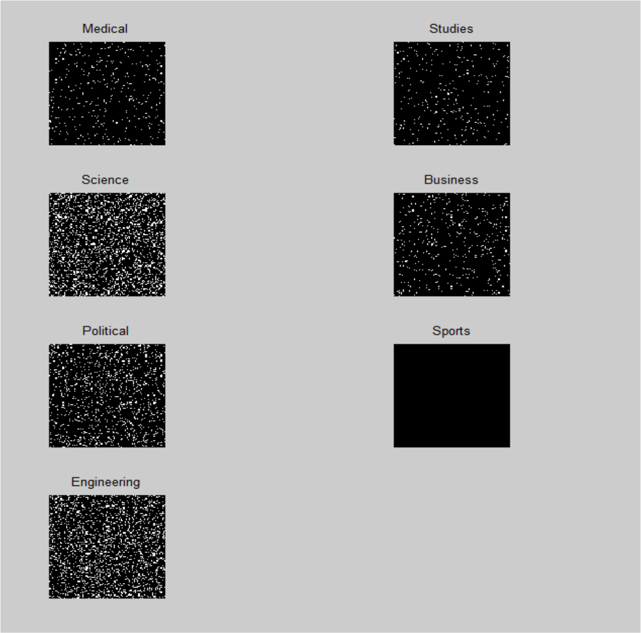

Fig 4: shows dataset belonged from different categories

Fig 5: shows maximum data sets is from engineering

6. REFERENCES

[1] Paul Coddington. "Monte Carlo Simulation for Statistical Physics." Northeast Parallel Architectures Center at Syracuse University

[2] Brain Korver 1994 “The Monte Carlo Method for Software Reliability Theory”, TR 94-1

[3] Bo Zhou, Hiroyuki Okamura and Tadashi Dohi “Markov Chain Monte Carlo Random Testing”Software Reliability. Hoang Pham.

[4] John D. Musa. Software reliability engineering: more reliable software faster and cheaper. McGraw-Hill. ISBN 0-07-060319-7

[5] Bo Zhou, Hiroyuki Okamura, and Tadashi Dohi(2010) “Markov Chain Monte Carlo Random Testing” T.H. Kim and H. Adeli (Eds.): AST/UCMA/ISA/ACN 2010, LNCS 6059, pp. 447–456, 2010.Springer-Verlag Berlin Heidelberg 2010

[6] Kazuhiko Hayakawa, M. Hashem Pesaran(2012) “Robust Standard Errors in Transformed Likelihood Estimation of Dynamic Panel Data Models” University of Cambridge ESRC Grant No. ES/1031626/1

[7] Zeyn A. Saigol(2008) “Monte-Carlo Testing for AUV Planning Software” University of Birmingham..

[8] Stefan Pauli, Peter Arbenza , Christoph Schwab “Intrinsic Fault Tolerance of Multi Level Monte Carlo Methods” Journal of Parallel and Distributed Computing.

[9] A. P Lyubartsev, A. A. Martsinovski, S. V. Shevkunov, and P. N. Vorontsov-Velyaminov(1991) “New approach to Monte Carlo calculation of the free energy:Method of expanded ensembles” J. Chem. Phys. 96 (3), 1 February 1992 1992 American Institute of Physics.

[10] Nathan A. Thompson, David J. Weiss(2011) presented a system “A Framework for the Development of Computerized Adaptive Tests” Practical Assessment, Research & Evaluation ISSN 1531-7714 Volume 16, Number 1, January 2011

[11] T.B.Whitaker,J.W.Dickens(1975) “Monte Carlo Technique to simulate Aflatoxin Testing Programs for Peanuts” Journal of the American Oil Chemists Scoiety,Vol.53,No.8,Pages:545-547.