A DYNAMIC PRIORITIZATION APPROACH TO GENERATE TEST SEQUENCE FOR REGRESSION TESTING

Dheeraj Malhotra*

Abstract:

By using fix test cases for regression testing may decrease the number and type of bug fixes than it may not uncovers all errors. And it may lead organization to spend lot of money and time on testing. The proposed approach is about performing a dynamic test suite as some changes are performed in software. As we know when software is tested a test sequence is generated, but when some changes are performed in the software code or some test case is removed and in such case all the links, coverage area are modified. The approach will decide a new cost effective test sequence for the further testing. It will be done removal of unwanted test cases and assigning a new dynamic prioritized sequence.

Keywords- Regression Testing, Dynamic Prioritization, Fitness Function, Mutation, Cross Over.

* M.Tech [CSE], Lovely Professional University.

1. INTRODUCTION:

In software systems development the chances of injecting human errors is more and these errors must be removed before the system is being in practice. Here testing plays an important role in detecting errors in system by analyzing it.

Software testing has three main purposes: verification, validation, and defect:

- The process of verification confirms that software meets its specifications. It ensures that software correctly implemented for specific function.
- Whereas the process of validation ensures that the software meets the business requirements. It provides the traceable activities to customers.
- A defect is inconsistency among the expected and actual result. The defect's ultimate source may be traced to a fault introduced in the specification, design, or development phases.

Regression Testing is important part of development because software development undergoes number of changes due to different requirement or some bug fix and to ensure that change or fix does not have any adverse effect on system we perform regression testing. Regression testing is the process of validating modified software to assure that changed parts of software behave as intended and unchanged parts of software have not been adversely affected by the modification [1] Regression testing can be performed in three ways [2]:

1) *Regression Test Selection:*- it attempt to reduce the cost of regression testing by selecting some appropriate subset of the existing test suite .Test selection techniques normally use the source code of a program to determine which tests should be executed during the regression testing stage .

2) *Regression Test Prioritization:* - techniques attempt to order a regression test suite so that those tests with the highest priority, according to some established criterion, are executed earlier in the regression testing process than those with lower priority. By prioritizing the execution of a regression test suite, these methods hope to reveal important defects in a software system earlier in the regression testing process.

3) *Regression Test distributed:* - another alternative that can make regression testing more practical by more fully utilizing the computing resources that are normally available to a testing team.

And by using part a part of genetic algorithm-mutation, crossover makes regression testing well for optimization of huge search space.

2. GENETIC ALGORITHM:

Genetic Algorithms are good at taking large, potentially huge search spaces and navigating them, looking for optimal combinations of things, solutions you might not otherwise find in a lifetime. A Genetic Algorithm is a search algorithm that is inspired by the way nature evolves species using natural selection of the fittest individuals.

The possible solutions to the problem being solved are represented by a population of chromosomes. The pseudo code of a basic algorithm for GA is as follows [14]:-

Initialize (population)

Evaluate (population)

While (stopping condition not satisfied)

{

Selection (population)

Crossover (population)

Mutate (population)

Evaluate (population)

}

3. PROPOSED WORK:

In this proposed work we are defining a new approach to assign the priorities to the test cases dynamically while performing the regression testing. The proposed approach is the try to reduce the test cases and assigning a new prioritization sequence. A genetic based approach to find the sequence of test suite [3]. For this approach tool used for development is matlab.

The base of the approach is the genetic approach with fuzzy logic

1. Generate the State flow diagram of some given project or the program
2. Analyze and collect all the possible paths between the initial and the Goal State by using permutation.
3. Repeat Steps 4 & 5 such that most distinct path not occurs.
4. Select Two possible path P1 {p11,p12,p13,….,p1n} and P2 {p21,p22,p23….p2n} on basis of fitness function .
5. Perform Crossover on P1 & P2 based on fault analysis in each module such that Generate a fuzzy rule to elect the node with minimum fault ratio as well as minimum load
    - Generate a new sequence P3={p31,p32,p33…p3n}, The sequence will be elected on the basis of the frequency of a state at index position.
6. Performed the Mutation on P3 to perform the pruning on dead states.
7. Exit

This work is generalized and can be used to perform regression testing on any system. It will provide the best cost effective path.

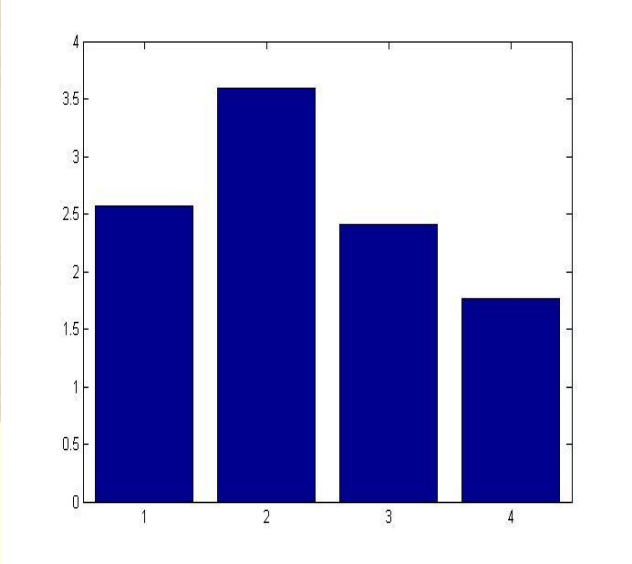

Figure1. Test Cases Vs CPU Time in Regression Testing

4. CONCLUSION:

The proposed approach will reduce the cost of testing by dynamically prioritizing the test cases using genetic approach. We will try to perform regression testing on Hard Real Time System in future work.

5. REFERENCES:

- M. J. Harold, J. A. Jone, T. Li and D. Liang, "Regression test selection for java software," in *Proc. of the ACM Conference, 2001*.
- Srinivasan Desikan, "A test methodology for an effective regression testing", 2006.
- Dheeraj Malhotra," A DYNAMIC TEST CASE UPDATION APPROACH TO REDUCE TEST CASES", IJRIM Volume 2, Issue 2 (February 2012) (ISSN 2231-4334).
- Baikuntha Narayan Biswal Soubhagya Sankar Barpanda Durga Prasad Mohapatra," A Novel Approach for Optimized Test Case Generation Using Activity and Collaboration Diagram", 2010
- Bo Yang and Ji Wu, "A Regression Testing Method for Composite Web Service",2010

- Emelie Engstrom, "Regression Test Selection and Product Line System Testing",2010 Third International Conference on Software Testing, Verification and Validation 2010 IEEE
- King Chun Foo, Zhen Ming Jiang, Bram Adams, "Mining Performance Regression Testing Repositories for Automated Performance Analysis", 2010
- Hyunsook Do, Member, Siavash Mirarab, "The Effects of Time Constraints on Test Case Prioritization:A Series of Controlled Experiments", TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 36, NO. 5, SEPTEMBER, 2010 IEEE
- B. Steece, S. Chulani, and B. Boehm, "Determining Software Quality Using COQUALMO," in Case Studies in Reliability and Maintenance, W. Blischke and D. Murthy, Eds.: Wiley, 2002.
- L. Chung. B. Nixon, E. Yu, J. Mylopoulos," Non-Functional Requirements in Software Engineering", Kluwer, 1999.
- B. Boehm, J. Brown, H. Kaspar, M. Lipow, G. MacLeod, M. Merritt," Characteristics of Software Quality", TRW Report to National Bureau of Standards, November 1973; TRW Software Series Report, TRW-73-09, also published by North Holland, 1978.
- A. Avezienis, J. Laprie and B. Randell, "Fundamental Concepts of Computer System Dependability", IARP/IEEE-RAS Workshop on Robot Dependability: Technological Challenge of Dependable Robots in Human Environments, Seoul, korea, May 21-22, 2001.
- I. Rus, S. Komi-Servio, P. Costa, "Software Dependability Properties: A Survey of Definitions, Measures and Techniques". Fraunhofer Technical Report 03-110, January 2003.
- Sangeeta,ritu,chayanika"Prioritization of test case scenarios derived from activity diagram using genetic algorithm". ICCCT, pp.481-485 IEEE (2010).