Combinatorial Testing using Bio-Inspired Algorithms

Dr. Priyanka Chawla¹, Rohit Chawla²

¹Professor, School of Computer Science and Engineering, Lovely Professional University, Punjab, India.
²Apeejay College of Fine Arts, Jalandhar Punjab, India.

ABSTRACT

Combinatorial Testing is a specification-based testing mechanism that generates test suites by picking values for input parameters as well as by blending the parameter values. It is based on high strength interaction combinations that can detect more faults but it suffers from combinatorial explosion problem in the case of large system possessing many parameters. This problem can be handled effectively by bio-inspired algorithms based combinatorial test data generation methodology that can provide an effective mechanism to cope up with the computational cost associated with the coverage of all combinations for particular interaction strength (s). Such test set generation should be able to detect faults at a faster rate in minimum time and cost that can be achieved by using cloud computing as it provides virtualized commodity hardware. The outcome of this work would enable software industry to deliver good quality software products at a lower cost.

I. INTRODUCTION

Software testing is considered as high-priced and time exhausting activity and is usually limited by budget of project. The National Institute for Standards and Technology (NIST) states that bugs in the software costs economy of U.S close to $60 billion in a year and by utilizing an effective testing methodology $22 billion can be resurrected [23]. With increase in the complexity of the software that should be functional in a distributed, networked and complicated environment, the design of modern software is required to be highly configurable that can be executed in a heterogeneous environment. This scenario has necessitated the utilization of modern software development methodologies such as component-based software and the service oriented software and that has led to rise in the number of parameters of a software system and interactions among parameters may result in rise of failures. Such kind of tendencies mandates the need of intelligent test data selection procedures to identify defects caused due to interaction among parameters.
The need for innovative software testing methodology is of utmost importance that extends a significant cost-benefit ratio while detecting faults in the software. Combinatorial testing is a model-based testing methodology that generates test sets by picking values for input parameters as well as by blending the its values. This strategy has been applied in various verticals such as application of design of experiments to compiler testing, evaluation of multifactor dependent system performance, software fault interaction and implications etc. [24] [25]. Combinatorial Testing has the ability to detect hard to find bugs effectively than other test case selection methods. It can be used in complement with other testing techniques such as equivalence partitioning and boundary value analysis. However, several issues need to be addressed are listed below:

1. Importance of interactions should be considered and the test should be prioritized.

2. The strength of different features should be kept variable instead of being fixed.

3. Combinatorial explosion problem that raises the need of expensive computational procedure and resources.

Although, finding a smallest test suites is NP-complete problem, in this work problem of finding a t-way test set would be formulated as a search problem and bio-inspired algorithm would be used to solve it Thus, an effective cloud-based framework would be designed and developed to build optimal and cost-effective t-way combinatorial test suites using bio-inspired algorithm. The advantages of using cloud as execution platform are enumerated below:

1. Cloud computing provides a model that can be used to for address massive computational problems by using virtualized commodity hardware

2. The very high level of parallelism provided by a cloud helps in achieving speed up of test execution at a lower cost.

3. It helps in reduction in the cost of acquisition and maintenance of the test environment.

4. Software can be tested effectively in the highly heterogeneous environment provided by cloud that helps in achieving improvement in the reliability of the software.
II. LITERATURE REVIEW

The generalized form of pair-wise testing is referred as t-way testing and it corresponds to testing methodology that ensures the coverage of each and every combination of ‘t’ input parameters of a system by at least one test case. T-way test strategy is an efficient specification-based test data generation methodology as it detects errors at very low cost without requiring knowledge about the implementation details. The study carried out by Kuhn et al. reported that all the known faults can be detected by up to 6-way interactions [1].

Yu Lei et al. devised deterministic strategy combinatorial testing by combining IPOG [2][3] and IPOG-D[3]. Renee C. Bryce et al. [4] proposed strategy based upon greedy strategy to generate prioritized interaction test suite. In the similar way, Xiang Chen et al. [5] proposed test data generation algorithm based on priority by using ant colony optimization. In the work [6] interaction testing has been described and evaluated w.r.t parameters such as accuracy, execution time, etc.

M.B. Cohen et al [7] proposed variable strength combinatorial testing mechanism in which variable strength interactions has been considered with a restriction that all interactions should be disjoint with higher strength. Wang et al. [8] extended the model by adopting greedy approach and developed test generation mechanisms using “one-test-at-a-time” and in-parameter-order algorithm to generate combinatorial test suite. Xiang Chen et al [9] developed a methodology named as variable strength interaction test suites (VSITs) in which ant colony based one-test-at-a-time strategy is used to generate test data. It has been empirically compared with several tools such TVG [10] and PICT [11]. In work [12] [13], deterministic pairwise testing has been proposed and compared with strategies like AETG, IPO, etc. and outperformed w.r.t size of test suites.

Lei and Tai proposed an easy to use java based optimal framework In-Parameter-Order (IPO) for test suite generation that fulfills pair-wise coverage for horizontal as well as vertical growth of test suite [15][16][17]. Yu Lei and Raghu Kackar developed a tool named as FireEye based on t-way testing by generalizing from pair-wise testing [18].

S. Vilkomir et al. [20][21] devised combinatorial methodology that ensures the coverage of different characteristics of mobile device and applications. Experimental investigation demonstrated the effectiveness of the proposed approach. In the work [22], coverage levels MC/DC and combinatorial t-way testing was investigated.
and experimental results showed increase in the coverage level with the increase in number of random test cases for both for MC/DC and t-way testing.

In India, research work has been done researchers in the labs of IBM, iGATE and TCS Innovation Lab in the field of Combinatorial testing (CT). S. Patel et al. [28] proposed Multi-Perspective Feature Models (MPFM) as an input model for combinatorial testing for achieving better coverage of variability in the product line. M. Mehta et al. [29] utilized combinatorial testing approach in the verticals of product & engineering, banking and insurance. M. Phadke et al. [27] utilized Orthogonal Arrays (OA) for the test data generation in IT systems in the financial services industry and observed an average reduction in total test effort as well as cost. R Krishnan et al. [26] designed and developed web-based application for test data generation using Orthogonal Array Based Testing methodology (OATS). Mohammad Younis et al. [19] developed MIPOG that can generate efficient higher order t-way test suite. This work is further extended and designed in such a way that allows test suites to be distributed over compute nodes on grid and can be executed simultaneously. This design helped in achieving optimized test set size in lesser time. In work [14] existing work was modified in input parameter order generalized (IPOG) and devised parallel t-way test data generation strategy and named as multicore modified input parameter order (MC_MIPOG) for multicore systems. Empirical results demonstrated better performance of MC_MIPOG as compared to other strategies such as IPOG, IPOF, IPOF2, IPOG-D, ICH, TConfig, Jenny and TVG w.r.t test size and execution time. However, its tremendous potential in terms of test optimization and fault detection is yet to be exploited to its fullest.

Following gaps are identified:

1. The tools developed so far require requires software engineers to circumscribe model of the software and equivalence classes for input parameters. Hence there is need for automated test generation methodology.

2. Actual interaction characteristics should be considered necessarily some interactions should be allowed to have variable strengths higher than fixed N. There is need to devise an effective approach for test generation that can be used by software industry for fault detection.

3. Another important issue that needs to be addressed is combinatorial explosion problem. Strategies developed so far has addressed the problem that aimed at minimization of test sets by removing
unsolicited data and control dependencies, pairwise testing etc. but none of the work is able to address it completely.

III. CONCLUSION

Developing Software Testing is considered as imperative aspect of software development that should be performed judiciously to authenticate that the developed software meets approved quality principles. It has been observed and experienced by software industry that expenditure for testing the software is almost 30% of the total cost of the software development and for safety critical software the cost can go even higher. Additionally, application is said to be robust if it has been tested on multiple operating systems and browsers of different versions. Such kind of tendencies mandates the need of intelligent test data selection procedures to detect failures caused due to interaction among parameters. Combinatorial testing is based on high strength interaction combinations can detect more faults. It is able to detect faults at a faster rate in minimum time and cost that can be achieved by using cloud computing as it provides virtualized commodity hardware. The outcome of this work would facilitate researchers as well as software industry to carry out automated testing at reduced cost and time.

REFERENCES


