

Combinatorial Testing Using SSO Algorithm

Subhash B Tatale¹, Aishwarya Shingar², Shruti Kulkarni³, Diksha Nikore⁴, Nikhil Ahire⁵

Department of Computer Engineering, Vishwakarma Institute of Information Technology, Pune,
Maharashtra, India

Abstract- Software testing is one of the phase of Software Engineering. The main contribution of this paper is to perform testing using SSO Algorithm. Recently, there are different type of testing technique available which can be used to test different user interface element. To facilitate the interaction between the user and the different computing devices the Graphical User Interface (GUI) testing can be performed. Simplified Swarm Optimization as a strategy is used for GUI Functional Testing. For generating an optimized test suite SSO is used with the help of Event-Interaction Graph (EIG).

Key words- GUI functional Testing, Model based Testing, Combinatorial Testing

I. INTRODUCTION

Today the computer is very important device because it is very much accurate, fast and completes many tasks very easily. Even though some tasks take more time such as manual task at that point computer is main requirement. Computer can do very big calculations in just a fraction of seconds. And the best thing is it can store huge amount of data and in various format.

Millions of users are depend on the internet for day to day work and this increases the importance of the website design and its usability. And it is nothing but the User Interface (UI). The UI is the communication medium between the user and the website. They include the basic design elements which are necessary to be present in order to make GUI efficient and more effective. GUI develops a relationship between person and the system. It includes the information about the way of how users will interact with the system. To ensure about the accuracy and problems in the website testing is performed.

There are different type of testing are available. The GUI testing is nothing but the verification of the functional testing .Functional testing of GUI deals with the information that specify the software components under tests. By using the GUI testing technique the defects in the developed software can be discovered. Test case generation is multi-objective approach or better solution to overcome the problem.

Basically the interaction between the user and system is done through some components of the UI such as images, test buttons, checkboxes, radio buttons, etc. In contrast way to non-functional testing, functional testing of GUI checks the proper communication between the user and the application interface without facing the coding internals. Using this technique the software is proposed which increases the productivity and quality of software program.

Through the literature various inspiring algorithms are found but the best is SSO Algorithm. The strategy for GUI functional testing using SSO Algorithm is described and SSO is optimized for the test suite with the help of interaction and making the combinations of pairs. Also, SSO have illustrated the classification problems in which databases are multiclass. Hence, SSO proves the efficiency against other algorithms.

II. RELATED WORK

A. Bestoun S. Ahmed^{*}, Mouayad A. Sahib, Moayad Y. Potrus [1]

In this paper they have presented a strategy to be used for GUI functional testing. The strategy generates the required test cases for the GUI under test using the combinatorial design and then it removes the unwanted test cases. The Simplified Swarm Optimization theory is used to optimize the test cases considering the combinatorial design concepts.

B. Qi XF, Wang ZY, Mao JQ et al [2]

In this paper, they have presented an algorithm called pairwise testing with constraints (PTC) to fully test forms. This PTC algorithm uses pairwise coverage and handles the issues of semantic constraints and illegal values. They have implemented a prototype tool ComjaxTest, which is capable of systematically exploring the state space of a web application.

C. Yodong Zhnag, Shuihua Wang and Genlin Ji [3]

In this paper, they have presented an algorithm called pairwise testing with constraints (PTC) to fully test forms. Our PTC algorithm uses pairwise coverage and handles the issues of semantic constraints and illegal values. We implemented a prototype tool ComjaxTest, which is capable of systematically exploring the state space of a web application.

*D. Dr. B. Narendra Kumar Rao,
M.L.Madhuri [4]*

In this paper, they have presented the performance of resultant test suites obtained from the IPOG and SSO algorithm are observed on the two state mode of the GUI and the best test cases are selected based on the highest coverage ratio.

E. Awdren Fontão, Arilo Claudio, Dias Neto [7]

This paper proposes an approach to support automated database-driven application testing integrated to functional GUI testing approaches. This approach was instantiated to the scenario of web-based application testing. Thus, they have extended a very popular functional GUI testing tool (Selenium RC) to implement the proposed approach, supporting database testing in web-based applications. We called this solution *SeleniumDB*

F. Imran Ali Qureshi and Aamer Nadeem [8]

This paper proposes a survey of existing GUI test case generation techniques; they have thoroughly examined the existing GUI test case generation techniques and given a brief analysis for each technique. For a comprehensive and detailed analysis, they have identified various analysis

parameters and with the help of these parameters, they have compared all the techniques.

G. Isabella and Emi Retna [9]

In this paper, some of the various test case generation methods and various types of GUI testing adapted for different GUI applications and techniques are studied. Different approaches are being used under various testing environment. This study helps to choose the test case generation technique based on the requirements of the testing and it also helps to choose the type of GUI test to perform based on the application type such as open source software, industrial software and the software in which changes are checked in rapidly and continuously.

*H. Mark TIMMER, Ed BRINKSMA, and Mariëlle
STOELINGA[10]*

This paper contains the hard core of that successful framework that represents about the well-established understanding of the desired relation between useful implementation relations for dynamic behaviour on the one hand, and test generation and evaluation on the other hand.

*I. Muhammad Imran, Rathiah Hashim and Noor
Elaiza Abd Khalid [11]*

From above study, they came to know that after proposal of PSO many researchers are working to improve its performance. In original PSO, there was no inertia weight but to improve the performance researchers introduced the inertia weight. Then they tried to improve the performance by trying the different initialization methods. Researchers also work on the global best particle to escape it from the local minima. For this purpose they introduce the different mutation operators to improve the performance of PSO.

J. Beston S Ahmed [12]

This paper discusses the adaptation, implementation, and evaluation of the new

dragonfly algorithm (DFA) for the use of pairwise test suite generation. The algorithm is implemented successfully for this purpose. Experimental evaluation is conducted for the algorithm to evaluate its efficiency against its counterpart PSO algorithm as well as other pairwise tools.

III. SSO ALGORITHM

SSO (Simplified Swarm Optimization) is simplified version of PSO and it is implemented by eliminating the personal influence term in the velocity modification equation.

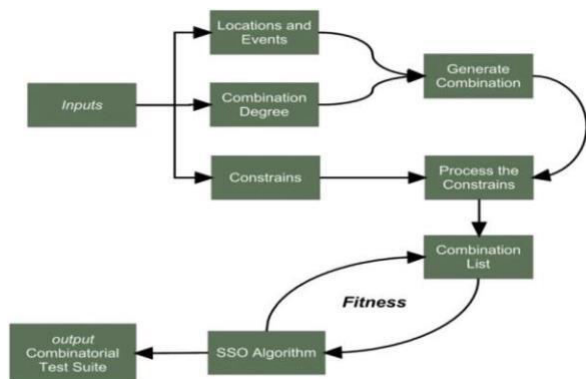


Figure 1. SSO Algorithm

In SSO Algorithm, initially the number of swarm population size, number of maximum pairs of generation and three specified parameters are determined. In every generation the position of particle and value in each dimension will be taken or be updated by its best value or by gbest value or be replaced by new random value. It gives the generate time and the best value.

The steps involved in SSO algorithm are as follows:

- Initialize randomly the positions X_i and velocities V_i of each particle in the swarm. (The initial values must fall within the boundaries defined to the positions and velocities)

- Calculate the fitness value of each particle in the swarm using the problem's objective function.
- Based on the results of step (b) modify the best particle position among all the swarm particles P_g
- Update the particle's velocity and position vectors using Equations (3) and (2) respectively.
- Repeat steps (b), (c), and (d) iteratively until the best fitness value or the maximum generation is met.

IV. CONCLUSION

The survey paper includes an approach to perform combinatorial testing using SSO Algorithm using GUI components on model of Railway Reservation System. The SSO Algorithm also overcomes the limitations of the PSO Algorithm. The complexity of our algorithm is lesser than the other algorithms. The future work of our study is to design a better approach to test all the other GUI components available.

REFERENCES

- [1] Bestoun S. Ahmed^{*}, Mouayad A. Sahib, Moayad Y. Potrus, "Generating Combinatorial test cases using Simplified Swarm Optimization (SSO) for automated GUI functional Testing"
- [2] Qi XF, Wang ZY, Mao JQ *et al.* Automated testing of web applications using combinatorial strategies. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 32(1): 199–210 Jan. 2017. DOI 10.1007/s11390-017-1699-x
- [3] Yodong Zhnag, Shuihua Wang and Genlin Ji, "A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications"
- [4] Dr. B. Narendra Kumar Rao¹, M.L.Madhuri², "Coverage MetricOn Combinatorial Test Cases Of GUI Functional Testing."
- [5] Awdren Fontão, Arilo Claudio, Dias Neto, "An Approach to Perform Automated Functional Testing in Database-Driven Applications"
- [6] Imran Ali Qureshi and Aamer Nadeem, "GUI Testing Techniques: A Survey".

- [7] Isabella and Emi Retna, “Study Paper on Test Case Generation Of GUI based Testing”
- [8] Mark TIMMER, Ed BRINKSMA, and Mariëlle STOELINGA, “Model -Based Testing.”
- [9] Muhammad Imran, Rathiah Hashim and Noor Elaiza Abd Khalid, “An overview of Particle Swarm Optimization Variants”.
- [10] Beston S Ahmed, “Generating Pairwise Combinatorial Interaction Test suites Using Objective Dragonfly Optimization Algorithm”.