International Journal of Academic Research and Development
ISSN: 2455-4197
Impact Factor: RJIF 5.22
www.academicsjournal.com
Volume 3; Issue 1; January 2018; Page No. 376-379

A literature survey on the use of particle swarm optimization technique for software testing

Mandeep Kaur
Department of Computer Science, Khalsa College for Women, Civil Lines, Ludhiana, Punjab, India

Abstract
Software testing is the process of finding errors to evaluate and improve the quality of software. It helps in finding the errors in all phases of software development lifecycle like requirement analysis, design, coding, system resources, system environments, etc. With the changing needs of the customer, it has become important to develop and deliver good quality software which conforms to the needs of the end user. So testing ensures that the software developed is free from bugs, is reliable and has good performance. Optimization is the process of obtaining maximum or minimum value of a function and this process is used to generate and prioritize the test cases in order to evaluate the performance of the software. This paper presents a literature review on the use of particle swarm optimization technique in the field of software testing particularly for software test case generation and prioritization.

Keywords: test case generation, test case prioritization, software testing, particle swarm optimization (PSO)

1. Introduction
Software testing is the process of subjecting the software to different inputs in order to find errors and to check whether the given software conforms to expected outputs. It is conducted by executing the program developed with test inputs and comparing the observed output with the expected one. The more number of errors found, the more successful the test suite is. Software testing helps in evaluating the quality and reliability of a software [1]. It focuses on prevention, detection, demonstration of errors and improvement in quality [2]. It is a time consuming and laborious activity and requires about 50% of the cost of the software development [3]. A test case is a single executable test which constitutes input, conditions, testing procedure, output and expected result that are used to check whether the software behaves according to the specified requirements or not. Hence it is important to generate proper test case and prioritize them to evaluate the software efficiently. Particle swarm optimization technique is nature inspired swarm intelligent technique which has been used effectively in the field of software testing. It is used by many researchers for optimizing the process of test case generation and test case prioritization. This paper represents a review on the use of particle swarm optimization for software testing.

2. Software Testing
Software testing involves evaluation of the software for errors, faults, missing requirements as compared to required ones. It is performed by software tester, developer, Project Manager, end user [4]. It is performed in every phase of software development lifecycle. Software testing methodologies are broadly classified into three categories:

a. Black-Box Testing: It is the type of testing where the knowledge of the internal working of the application is not required [5]. It is also called closed box testing. Some techniques used in black box testing are:
1. Equivalence partitioning
2. Boundary Value Analysis
3. Cause Effect Graphing

b. White-Box Testing: It includes detailed knowledge of the internal logic and structure of the software. It is also called open box testing. Some techniques used in white box testing are:
1. Control flow testing
2. Data flow testing
3. Branch testing
4. Statement testing
5. Decision coverage
6. Path testing

c. Grey-Box Testing: It is the combination of black box and white box testing where limited knowledge of the internal workings of an application is required.

2.1 Testing Strategies

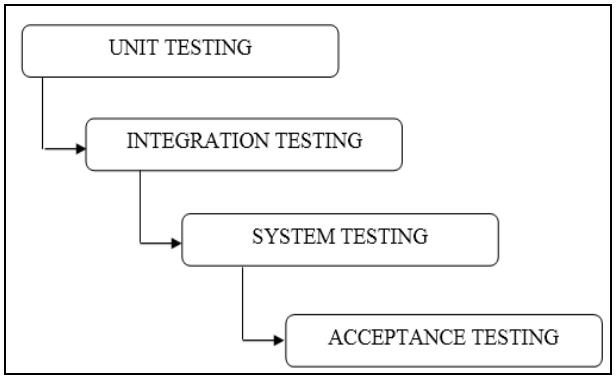

Fig 1: System testing strategies

a) Unit Testing: Unit testing is the testing of individual modules or components of a software.
b) Integration Testing: Integration testing is the testing of two or more combined modules. It occurs after unit testing. It can follow top-down, bottom-up, big bang approach.
c) System Testing: System testing is the testing of the combination of all modules of a software.
d) Acceptance Testing: It is conducted by the quality assurance team to determine whether the software meets the client's requirements and specifications.

2.2 Test Case Life Cycle

a) Test Case Generation

A test case is a set of inputs, conditions or variables under which a system is tested to determine whether it satisfies the specified requirements or works correctly. It helps in finding errors, problems in requirements or design of an application. A collection of test cases is known as test suite. Following are a few characteristics of good test case [6]:

1. Test case should not be complicated.
2. It should test one thing at a time.
3. It should cover all positive scenarios and negative scenarios.
4. It should be written in simple and easy language.
5. It should be accurate, economical, traceable, repeatable and reusable.

A test case includes following components:

1. Test case ID
2. Product module
3. Product version
4. Purpose
5. Assumptions
6. Pre-conditions
7. Steps
8. Expected outcome
9. Actual outcome
10. Post-conditions

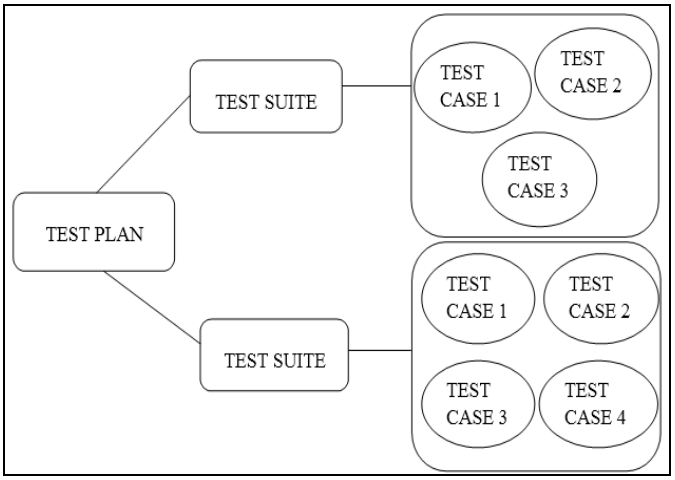

Fig 2: Test Plan, Test Suite, Test Case Hierarchy

b) Test Case Selection and Minimization

Test case selection is the process of selecting specific test cases from a test suite which lead to reduction in time, cost and effort for software testing. Test case minimization is similar to test case selection. Its main purpose is to remove those test cases from the test suite that have become obsolete or redundant [7].

c) Test Case Prioritization and Evaluation

Test case prioritization is the process of ranking the test cases from various test suites of software. There are different approaches to schedule and rank the test cases.

3. Particle Swarm Optimization

Particle swarm optimization (PSO) is an optimization technique developed by Dr. Eberhart and Dr. Kennedy in 1995 [8, 9]. It is inspired by the natural social behavior of flock of birds or fish schooling where a bird flock randomly searches for food in an area. The birds do not know where the food is but they know how far the food is in each iteration. So the best strategy to find the food is to follow the bird which is nearest to the food. In particle swarm optimization, simple software agents, called particles that are birds in natural space, move in the search space of an optimization problem. Every particle has an associated position which represents a candidate solution to optimization problem and velocity which directs the flight of particles. In each iteration, the following two values, called as "best" values are updated:

a) "pbest" value which is the best solution found by a particle so far.
b) "gbest" value is best value obtained so far out of all the best values of particles.

With these two best values, the velocity and position of the particle is updated using the equation 1 and 2.

$$V(it+1) = V(it) + c1 * rand() * (pbest(it) - X(it)) + c2 * rand() * (gbest(it) - X(it)) \quad (1)$$

$$X(it+1) = X(it) + V(it+1) \quad (2)$$

V()is the particle velocity, X()is the current particle position(solution), pbest() and gbest() are particle best and global best position respectively, rand () is a random number between (0,1), c1 and c2 are learning factors.

3.1 Working of Particle Swarm Optimization

a) A group of random particles are initialized with random position and velocity.
b) Particles move in the search space and are evaluated on the basis of fitness criterion.
c) Pbest value is evaluated for each particle. If the current value is better than the pbest value of a particle then the current value is assigned as the new pbest else the previous pbest is kept.
d) Gbest is found on the basis of pbest of all particles.
e) Velocity of each particle is evaluated.
f) This evaluated velocity is used to update the position of particle.
g) This process is repeated till the stop criterion is met.

3.2 Advantages of Particle Swarm Optimization

a) It requires very simple calculation.
b) It has great optimization capability.
c) It has no mutation and overlapping calculation.

d) It can be used for both scientific research and engineering use.

4. Literature Review

Kaur A and Bhatt D [10] used Hybrid Particle Swarm Optimization (HPSO) algorithm which was a combination of Particle Swarm Optimization (PSO) technique and Genetic Algorithms (GA) for Regression Testing where regression testing ensures that the functionality of the software does not change after any change is made to the software. HPSO was used to perform prioritization of test cases which helps in reduction in cost of regression testing. The proposed method was tested on the basis of Maximum fault coverage in minimum execution time. The Genetic Algorithm (GA) operators provided optimized way to perform prioritization in regression testing and Particle Swarm Optimization (PSO) technique provided effective fast solution. The Genetic Algorithm (GA) operator used was mutation operator. The results obtained showed the effectiveness of the technique proposed.

Patidar C [11] proposed a model based on sequence diagram to generate more dynamic test cases. In this research information was extracted from sequence diagram. Initially all sequence diagrams were created for the given problem. Then dependency table was generated followed by dependency graph. Finally test cases were generated from dependency graph which were optimized by Discrete Particle Swarm Intelligence Algorithm.

Windisch A *et al*. [12] implemented particle swarm optimization on 25 small artificial test objects and 13 more complex industrial test objects taken from various development projects to generate test cases. The research compared the results of PSO with that obtained by using Genetic algorithm. It was found that the PSO was more effective and efficient in generating software test cases as compared to genetic algorithms.

Arora D and Baghel AS [13] used both genetic algorithms and particle swarm optimization for test case generation. After that a comparative analysis of the both techniques was done. It was suggested that particle swarm optimization technique overcame the drawbacks of genetic algorithms.

Ming YD *et al*. [14] proposed the use of particle swarm optimization for improving the efficiency and quality of software test case automatic generation. The optimization was based on the clustering thought where the algorithm divided the population of particles into two types. These were main particle and secondary particles. The results of the research suggested that the proposed method was advantageous and effective than the other contrastive algorithms in the software test case automatic generation.

Huang M [15] used the group self-activity feedback (SAF) operator and Gauss mutation to improve the performance of particle swarm optimization. This improved algorithm was used in software test case generation. It was found that the single path fitness function structure and multi-path fitness calculation of parallel thinking was superior to the standard PSO and more efficient in multi-path test case generation.

Hla KHS *et al*. [16] proposed the use of particle swarm optimization (PSO) algorithm for prioritizing the test cases automatically based on the modified software units in order to select those test cases which have new higher priority in regression testing process. Particle swarm optimization algorithm was found to be effective and efficient in prioritization of the test cases in the test suites.

Singla S [17] introduced a new algorithm based on the combination of the power of Genetic Algorithm (GA) and Particle Swarm Optimization (PSO), called Genetic-Particle Swarm Combined Algorithm (GPSCA) which was used to generate automatic test data for satisfying data-flow coverage criteria. The research compared the results of the proposed model with the Genetic Algorithm and PSO algorithms. It was observed that the new approach GPSCA was better than GA and PSO as it provided higher coverage ratio % than the PSO and GA. Also test case requirement by GPSCA was less.

5. Conclusion

Testing is one of the crucial activities of software development process. Its main purpose is to find as many errors, faults as possible so as to produce a reliable, usable, easily maintainable, user friendly, good quality software. One of the important goals of testing is to spend less amount of money on its test case generation. Testers work rigorously to generate, prioritize and execute a suitable test suite. Particle swarm optimization is a swarm intelligent technique which is effective in optimizing the procedure of test case generation and prioritization. It has been observed from the literature review that particle swarm optimization has been used extensively in the field of software testing and the results obtained were better than many other techniques. This paper presented a literature survey on the use of PSO for software testing and this paper will help many researchers to learn and explore this subject.

6. References

1. Beizer. Software testing techniques, Second Edition, Van Nostrand Reinhold, New York, 1990.
2. Chauhan RS, Singh I. Latest Research and Development on Software Testing Techniques and Tools, International Journal of Current Engineering and Technology, 2014, 2368-2372.
3. Desikan S, Ramesh G. Software testing principles & practices, Pearson Education, 2007.
4. https://www.tutorialspoint.com/software_testing/software_testing_overview.htm
5. Khan ME. Different Approaches to Black Box Testing Technique for Finding Errors, International Journal of Software Engineering & Applications. 2011; 2(4):31-40.
6. http://softwaretestingfundamentals.com/test-case
7. Singh R, Santosh M. Test Case Minimization Techniques: A Review, International Journal of Engineering Research & Technology, 2013, 2 (12): 1048-1056.
8. Bai Q. Analysis of Particle Swarm Optimization Algorithm, Computer and Information Science. 2008; 3(1):180-184.
9. Kennedy J, Eberhart R. Particle Swarm Optimization, Proceedings of IEEE International Conference on Neural Networks. 1995; 4:1942-1948.
10. Kaur A, Bhatt D. Hybrid Particle Swarm Optimization for Regression Testing, International Journal on Computer Science and Engineering. 2011; 3(5):1815-1824.
11. Patidar C. Test Case Generation Using Discrete Particle

Swarm Optimization Algorithm, International Journal of Scientific Research in Computer Science and Engineering. 2013; 1(1):38-42.
12. Windisch A, Wappler S, Wegener J. Applying Particle Swarm Optimization to Software Testing, Proceedings of the 9th annual conference on Genetic and evolutionary computation, 2007, 1121-1128.
13. Arora D, Baghel AS. Application of Genetic Algorithm and Particle Swarm Optimization in Software Testing, IOSR Journal of Computer Engineering. 2015; 17(1):75-78.
14. Ming YD, Ting WY, Hui WD. Particle Swarm Optimization Algorithm for Test Case Automatic Generation Based on Clustering Thought, IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), 2015.
15. Huang M, Zhang C, Liang X. Software Test Cases Generation Based on Improved Particle Swarm Optimization, International Conference on Information Technology and Electronic Commerce (ICITEC), 2014.
16. Hla KHS, Choi Y, Park JS. Applying Particle Swarm Optimization to Prioritizing Test Cases for Embedded Real Time Software Retesting, International Conference on Computer and Information Technology Workshops, 2008.
17. Singla S, Singla P, Rai HM. A Combination of GA and PSO for Automatic Test Data Generation using Data Flow Coverage, International Journal of Advanced Research in Computer Science. 2011; 2(2):488-491.