International Journal of Computer Applications Technology and Research
Volume 4– Issue 5, 355 - 358, 2015, ISSN:- 2319–8656

Test-Case Optimization Using Genetic and Tabu Search Algorithm in Structural Testing

Tina Belinda Miranda
Department of Computer Science and Engineering
Panimalar Institute of Technology, Chennai, India

M. Dhivya
Department of Computer Science and Engineering
Panimalar Institute of Technology, Chennai, India

K. Sathyamoorthy
Department of Computer Science and Engineering
Panimalar Institute of Technology, Chennai, India

Abstract-- Software test-case generation is the process of identifying a set of test cases. It is necessary to generate the test sequence that satisfies the testing criteria. For solving this kind of difficult problem there were a lot of research works, which have been done in the past. The length of the test sequence plays an important role in software testing. The length of test sequence decides whether the sufficient testing is carried or not. Many existing test sequence generation techniques uses genetic algorithm for test-case generation in software testing. The Genetic Algorithm (GA) is an optimization heuristic technique that is implemented through evolution and fitness function. It generates new test cases from the existing test sequence. Further to improve the existing techniques, a new technique is proposed in this paper which combines the tabu search algorithm and the genetic algorithm. The hybrid technique combines the strength of the two meta-heuristic methods and produces efficient test- case sequence.

***Keywords*:** Test sequence, testing criteria, test case generation, genetic algorithm, tabu search algorithm.

1. INTRODUCTION

Software engineering is a discipline concerned with all aspects of software right from development to its retirement.[18] Software testing plays a prime role in software development life cycle[7]. It is aimed at discovering the faults in software to provide software quality. In white box testing it is necessary to design a set of test cases that satisfy testing criteria[9]. A test case executes software with a set of input values and then compares the expected output with the obtained output to see whether the test has passed or failed. In this paper we focus on branch coverage. As software testing consumes about 50% of software development effort, test data generation plays an important role[8].

Various approaches for test data generations have been developed. These can be classified into three broad categories: random, static and dynamic techniques. Some of the dynamic methods of test data generation using meta-heuristic techniques treat testing problem as search space or optimization problem. Due to the difficulty and complexity in the testing process, these techniques have to search a large space. Some of the meta-heuristic techniques suffer from the problems of local optimum, when software testing is done.

The solution that is best within neighboring space and not globally is local optimal solution. The search algorithms have a tendency to converge immaturely to local optimum. Because of this, test data generated will not satisfy the testing criteria. Particularly, Genetic algorithm has problems like slow convergence, blind search and risk of getting stuck into local optimum solution.

This paper analyzes test-sequence generation technique based on genetic and tabu search algorithms. Genetic algorithm generates new test data from previously generated good candidates. The tabu search is added to the mutation step of genetic algorithm to reduce the time of search.

The rest of this paper is organized as follows: The Section 2 deals with the related work. The Section 3 deals with the search algorithm. The Section 4 deals with the proposed solution. Section 5 deals with the experimental validation. The section 6 deals with the conclusion.

2. RELATED WORK

There are many search based meta heuristic algorithms that have been proposed to generate the test data. The main characteristic of meta heuristics is to find better solutions at each step by adjusting the sub solutions. Genetic algorithm is an important population-based algorithm. The way in which genetic algorithm was applied to testing the object-oriented software was done by Tonella[4]. A population of test sequences was evolved using evolutionary techniques. The main disadvantage of the paper was that in case of complex conditions in the code the evolutionary search was reduced to random search. Later many researchers used the genetic algorithm for test data generation. Ahmed used the genetic algorithm to generate the test data when path coverage was used as the test criteria[1]. This method covered more paths in one run thus improving structural coverage. The basic concepts of tabu search algorithm were explained by Glover [2]. The main concept of Tabu search to reduce the cost by providing maximum structural coverage. Many researchers used the tabu search algorithm for lot scheduling problems.

3. SEARCH ALGORITHM

There are several search algorithms. A search algorithm will not find a global optima in a fair amount of time. Therefore, it is common to put premature stopping criteria based on the available computational resources. In this paper two search algorithms are analyzed.

Genetic Algorithm:

Genetic algorithm is a famous meta-heuristic search based algorithm [10]. It has been demonstrated that the test cases generated by genetic algorithm are more efficient than the random search algorithm. Genetic algorithm generates new test data from already generated good candidates. This algorithm is inspired by Darwin's Theory. The algorithm uses evaluation, selection, crossover point and the mutation operators to generate new test cases from the existing test sequence. The evaluation procedure measures the fitness of each individual solution also known as chromosome in the population and assigns a value based on the optimizing criterion. The selection procedure selects individuals randomly in the current population for development of the next generation. The selection procedure chooses the individual solutions to be recombined and mutated out of the initial population. Recombination procedure reproduces the selected individuals and exchanges the information for generating new individuals. The information that is exchanged is called crossover. The crossover procedure chooses the two selected individuals and then combines them, thereby creating two new individuals. Mutation creates a small change to newly created individual. The resulting individuals are then evaluated through the fitness function. The fitness procedure measures how well chromosome satisfies the testing criteria. These concepts have been explained earlier in[5,6].

```
Choose population N uniformly at random
    from S(l)
While global optimum not found
Copy best a solutions from N to
N' While N' is not completely
filled
Select 2 parents from N according to selection
criterion
Generate two offspring that are same as their
parents
Apply crossover on
offspring Mutate each
offspring
Copy the 2 new offspring into
N' N=N'.
```

Pseudo code of Genetic Algorithm

There is an issue in using the genetic algorithm for generation of test cases because it suffers from problems like slow convergence, blind search and the risk of getting stuck to the local optimum solution. Local optimum is a solution that is best within the neighboring space but not globally.

Tabu Search Algorithm:

Tabu search is a meta heuristic approach which is used to solve the optimization problems[2,3]. It is designed in such a way to guide other methods to move away from the local optima. It provides memory to avoid falling into the local optima.

The main characteristics of Tabu Search are its flexible memory structure which is designed so that criteria as well the information regarding the search are exploited thoroughly. Tabu maintains two different types of memory a short term memory and the long term memory. The recent moves are captured in the short term memory and the related moves are captured in the long term memory. The intensification and the diversification strategies help the search process to give optimal results. The intensification strategies help to reinforce previous solutions that are found good. And the diversification strategies help to search new areas that have not been explored earlier. To avoid getting stuck in the local optima or searching the same solution, a list is created to maintain the most recently visited solutions. This list is called as the tabu list. The tabu list consists of a set of forbidden moves to prevent cycling and avoids getting stuck to the local optima. The tabu search will search for better solutions until the testing criteria are met.

```
Create an intial solution n
While the stopping criteria is not met
  Create a set of solutions K that are
the neighbors of n and that are not in
   tabu list
Choose a best solution n* in K
Update the tabu list based on
n*
Let n=n*
End
```

Pseudo code for Tabu Search Algorithm

4. PROPOSED SOLUTION

A test suite is used to test the software. There is an issue in selecting the appropriate test cases for testing. The inappropriate and redundant test case selection will increase the test sequence length. The genetic algorithm suffers from local optima, in order to avoid this situation; The hybrid algorithm is proposed which is a combination of the genetic algorithm and the tabu search algorithm. The tabu search algorithm is added to the mutation step of genetic algorithm to reduce the randomness and the execution time of search and this enhances the quality of the end result. The genetic algorithm is used initially and its result is passed to the tabu search algorithm and this deals with repeated individuals by forbidding it from being chosen. This helps in generating new individuals in the next generation which is not present in the tabu list.

Initially a group of test cases are generated. Then a set of test cases are selected randomly. The selected test cases are set as the population size. Recursively select a number of best solutions from the population size. Then select two parent test cases according to the selection criteria. Generate two offspring that are the replica of their parents. Use crossover on the offspring with specified probability. Then mutate the offspring based on the long term and short term tabu list in order to avoid the unwanted new offspring generation. Then generate new offspring and put into the solution. This gives the optimized test sequence.

1. Generate Random Test cases and set population size
2. Define the Initial Population Size called PopSize
3. Generate the Random Population Set to represent the possible test sequences

   Define Fitness Function called Maximum Coverage
4. For i=1 to MaxIterations
   [Repeat Steps 6 to 10]
5. Select two Random Parents called P1 and P2 from Population Set
6. Perform crossover to generate new Child
7. Perform Mutation Operation Child=Mutation (Child) using tabu search list
8. Population=Population U Child
9. Return Optimized Test Sequence

Algorithm for Hybrid Technique

5. EXPERIMENTAL VALIDATION

To evaluate the performance of this algorithm an experiment was conducted to analyze the test suite. Initially a sample voter validation form was created and then structural testing had to be done. So, a set test cases were generated for the form. Then the genetic algorithm was used to optimize the test cases.

There after the same experiment was carried out using the proposed hybrid algorithm and similarly the test cases were optimized . And then by comparing the test cases produced by genetic algorithm and hybrid algorithm it was found that test cases produced by the hybrid algorithm was more efficient than the test cases produced by the genetic algorithm.

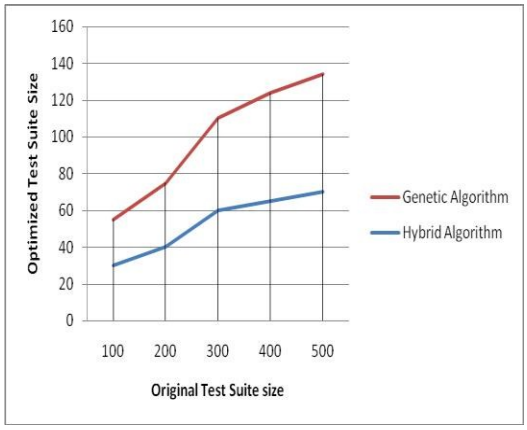

Fig.1. Test Suite size versus Optimized Test Suite Size

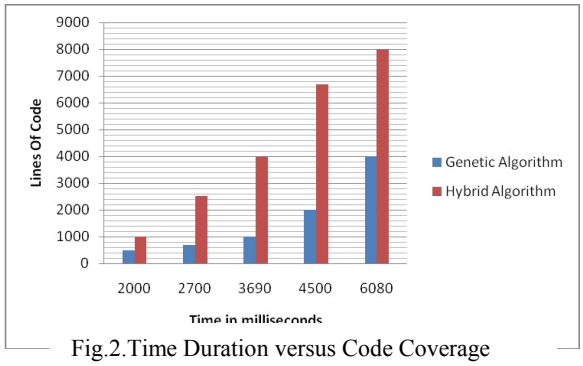

Fig.2.Time Duration versus Code Coverage

6. CONCLUSION

In this paper we have the analyzed the way in which the genetic algorithm and the tabu search algorithm are used for optimizing the test cases.

The disadvantages of the genetic algorithm are analyzed, the problem of getting stuck in the local optima are overcome by using the hybrid algorithm. The Hybrid algorithm proposed in this paper generates the test cases that satisfies the given test criteria. The results of optimized test case and statistical data support claim that this algorithm performs better than other related strategies.

7. REFERENCES

[1] Ahmed, M.A. and I. Hermadi, "GA-based multiple paths test data generator," Computers and Operations Research, 2008. 35(10): p. 3107-3124.

[2] F. Glover and M. Laguna, “Tabu Search,” Kluwer Academic Publishers, 1997.

[3] F. Glover, “Tabu Search Part I,” *ORSA Journal on Computing*, Vol. 1, No. 3, 1989, pp. 190-206.

[4] Tonella, P, "Evolutionary testing of classes," ACM SIGSOFT Software Engineering Notes.Boston, MA, United states: Association for Computing Machinery, p.119-128, 2004.

[5] B Jones et al. “Automatic Structural Testing Using Genetic Algorithms”, Software Engineering Journal, Vol.11, No.5, 1996.

[6] P McMinn, “Search-Based Software Test Data Generation: A Survey”, Software Testing, Verification and Reliability, Vol.14, No.2, pp. 105—156, 2004.

[7] I. Sommerville, “Software Engineering, Pearson Education,”7th Edition, Tata Mc-Graw Hill, India, 2005.

[8] B. Beizer, “Software Testing Techniques,” 2nd Edition,van Nostrand Reinhold, New York, 1990.

[9] G. Myers. *The Art of Software Testing.* Wiley, New York, 1979.

[10] M. Harman, S.A. Mansouri, and Y. Zhang,“Search Based Software Engineering: A Comprehensive Analysis and Review Of Trends Techniques and Applications,”Technical Report TR-09-03, King’s College, 2009.