Volume 5, Issue 6, June 2015 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

MTCPA: Multi-Objective Test Case Prioritization Algorithm Using Genetic Algorithm

Anita Saini, Dr.Sanjay Tyagi
Department of Comp. Sc.& Applications,
Kurukshetra University, Kurukshetra, Haryana, India

Abstract-- The main objective of regression testing is to test the modified software during maintenance phase of Software Development Life Cycle. An easiest strategy to regression testing is to retest all test cases in a test suite, but due to limitation of resources and time constraint, it is not efficient to implement. Test case prioritization technique increases the effectiveness of regression testing by ordering the test cases according to specified criteria. It intends to prioritize test cases in such a way that higher priority test cases execute before lower priority test cases. This paper proposedmulti-objective test case prioritization algorithm (MTCPA)to prioritize test cases based on two objectives that is statement coverage and testing time of test cases using genetic approach. The proposed approach is compared with different existing prioritization techniques to find the optimal one.

Keywords---Regression testing, test case,test case prioritization,genetic algorithm, APFD and MTCPA.

I. INTRODUCTION

Testing is important activity in Software Development Life Cycle (SDLC) to ensure the quality of the software. It is more complex process during software development process. Testing consumes 50% of the total development cost of the software [8]. In SDLC, the different software testing techniques are used to produce best quality software product with low development cost and time. In the field of software development, test case prioritization technique of regression testing techniques is of practical importance and relevance to industry for producing best quality software products.To re-execute all the test cases of the test suite is the simplest strategy of regression testing, but due to certain constraints (resources), it is inefficient to implement the all test cases [6]. For example, to execute all test cases of test suite for a product or program having approximately 20,000 lines of code, consumes seven weeks [2]. Therefore, it is necessary to discover the technique with the goal of increasing the effectiveness of regression testing by arranging the test cases of test suites according to the specified objectives.

Test case prioritization is defined as finding a permutation of test cases to maximize an award function, which reflects the prioritization goal [1]. In this technique, each test case is assigned a priority. Priority is assigned according to specified criterion and test cases with highest priority are scheduled first. The added advantage to previous regression testing techniques is that it doesn't permanently remove the test cases from test suite. The proposed approach uses more than one objective to prioritize test cases.

This paper is organized as follows: section 2 discusses the overview of existing test case prioritization techniques. In section 3, the proposed multi-objective test case prioritization algorithm (MTCPA)is proposed and discussed.Section 4 presents an experiment& results and section 5 concludes the paper.

II. RELATED WORK

In the frequent paragraphs, some of the studies have been discussed, which were previously undertaken in the field of Test Case Prioritization. Yoo et al. [14] and Mohanty et al.[13] described the existing techniques of test case prioritization, which mostly uses code and functional coverage criteria to prioritize the test cases with respect to the system code executed by test cases. In the literature, test case prioritization techniques are based on coverage [9][2], requirement coverage [10] and on fault exposing potential of the test case [7][12]. In literature, some test case prioritization techniques are used to prioritized the test case such asAnt colony optimization (ACO) [4] or Particle swarm optimization (PSO) [17][11].AmrAbdelFatah Ahmed et al.[15] presented software testing suite prioritization using multi-criteria fitness function. The approach used multiple control flow coverage metrics for prioritization. Mahfuzul Islam et al [16] identified multi-objective techniques to prioritize test cases based on latent semantic indexing. The proposed technique counts the coverage of source code & application requirements and the cost to execute test cases. Xiaolin Wang and HongweiZeng [18] proposed a method to calculate the prioritization values of test case separately based on five criteria namely coverage, fault, requirement, history information & time and calculated the weighted sum of their optimization results, then it sorted the test suite according to the prioritization value that was obtained by weighted sum. Test case prioritization using multi objective PSO (Particle Swarm Optimizer) was proposed by ManikaTyagi and SonaMalhotra [19]. The proposed algorithm has used the maximum fault coverage and minimum execution time for

prioritization of test cases. Mitrabinda Ray et al.[20] was developed Multi-objective test prioritization via a genetic algorithm. This related study provides a brief review on multi-objective based test case prioritization for regression testing. It has been found that the best optimal order of test cases can be obtained by using the concept of genetic algorithm with multiple fitness functions.

III. PROPOSED APPROACH

The proposed Multi-objectiveTest Case Prioritization Algorithm (MTCPA) is based on statement coverage and testing time information to prioritize test cases using multi objective genetic algorithm. The main aim of proposed approach is to prioritize the test cases in any sequence that covers maximum statement coverage of program along with minimum testing time.

In this approach, first step calculates the statement coverage and testing time of each test case in test suite. Next step is prioritization of test cases by using MTCPA.The last step is to compare it with other prioritization techniques such as reverse, random and previous work of Xiaolinwang et al.[18] prioritization techniques on the basis of total statement coverage and testing time of complete test suite. The diagrammatic representation of proposed approach is shown in Figure 1.

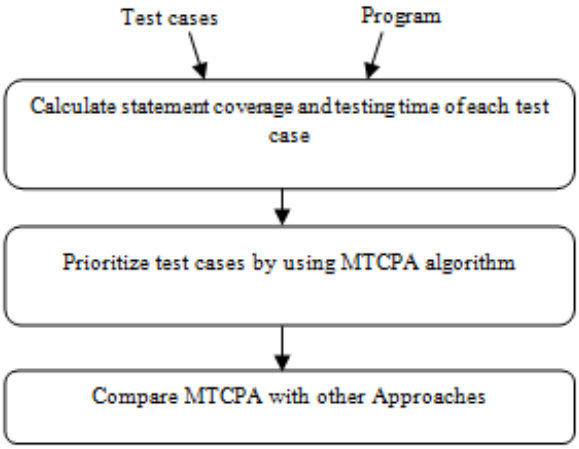

Figure 1 Fundamental steps of proposed approach

A. Objective Function

This paper has two objectives to prioritize the test cases. These objectives are discussed as follows:

i. Statement coverage

Statement coverage means total number of statement in a program covered by test cases. Let $T = \{T_1, T_2, T_3 \ldots T_n\}$ indicates a set of n test cases and $S = \{S_1, S_2, S_3 \ldots S_m\}$ indicates a set of m statements of program P. Let Statement_Coverage (ST_i) be a function that return the average number of statements S covered by the individual test case T_i. Then statement coverage of a solution $T = \{T_1, T_2, T_3 \ldots T_n\}$ is given as:

Statement_Coverage $(ST_i) = \sum S(T_i) / m$

Where, $\sum S(T_i)$ = Sum of no. of statements that are covered by test case T_i

m = Total no. of statements of program.

i = 1, 2… n, where n is number of test cases.

ii. Testing time

The testing time is the time taken to run a test suite which is the summation of running time of all the test cases. Let Testing_Time (TT_i) be a function that returns the execution time of individual test cases T_i. Testing time is to be minimized and it can be defined as:

Testing_time$(TT_i) = Tt_i$

$Tt_i = (Tt_1, Tt_2, \ldots\ldots, Tt_n)$ indicates a set of n test case's testing time.

B. Multi-objective Test Case Prioritization Algorithm(MTCPA)

The MTCPA algorithm uses the genetic approach with Pareto front concept to find the optimal order of test cases. Genetic algorithm is a multi-objective optimization algorithm which deals with the task of optimizing simultaneously two or more conflicting objectives with respect to a set of certain constraints [3]. Genetic algorithm uses a set of operators to iteratively evolve an initial population of candidate solution (i.e. test case ordering). The evolution is guided by an objective function (fitness function) that evaluates the quality of each candidate solution alongwith the considered criteria. The Pareto front [3] finds the best optimal solution by using both fitness functions such as statement coverage and testing time of test cases. The algorithm terminates when the population converges toward the optimal solution.

Algorithm: Multi-objectiveTest Case Prioritization Algorithm (MTCPA)

Input: Randomly generated numbers of population of test cases sequence.

Output: Optimal order of test cases.

Step 1: Create Initial population

Step 2: For each generation do
i) For each individual do
Compute multi objective fitness functions.
End for
ii) Find out non-dominated solution by using Pareto front.
iii) Updating global best fitness solution.
Step3: Perform selection and mutation to generate new population
Step 4: Add new population to the next generation
End for
Step 5: Return all non-dominated solution.

IV. EXPERIMENT AND RESULTS

The proposed algorithm MTCPA is implemented on a problem described byXiaolin Wang et al.[18]. The MTCPA algorithm is implemented on MATLAB. For experiment, the proposed algorithm considered the same test suite as used in Xiaolin Wang et al.[18]. First, statement coverage and testing time of each test case is calculated. Next, the MTCPA algorithmfinds the optimal order of test cases byimplemented fitness functions which are statement coverage and testing time of test cases. In MTCPA, the Pareto front function is used to find the best alternative solutions from evolved population (random generation order of test cases).The MTCPA algorithm is compared with otherprioritization techniques such as reverse prioritization, random prioritization andXiaolin Wang et al.[18] prioritization. The proposed algorithm shows that theoptimal order of test cases cover the maximum statement coverage and minimum testing time as compared to other prioritization techniquesas represented in Figure 2 and Figure 3 respectively.

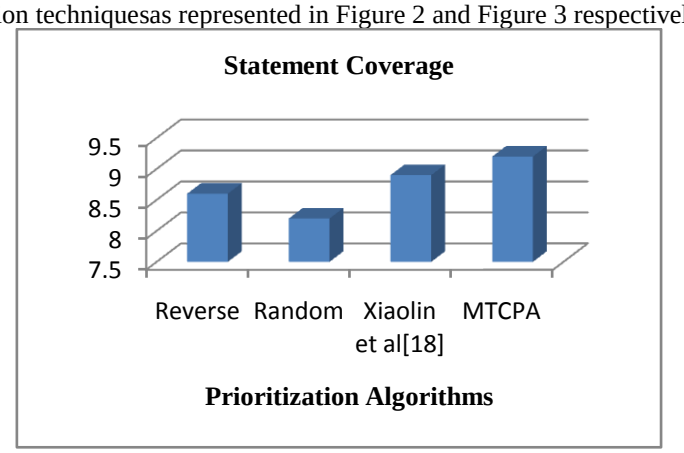

Figure2 Comparison based on Statement Coverage

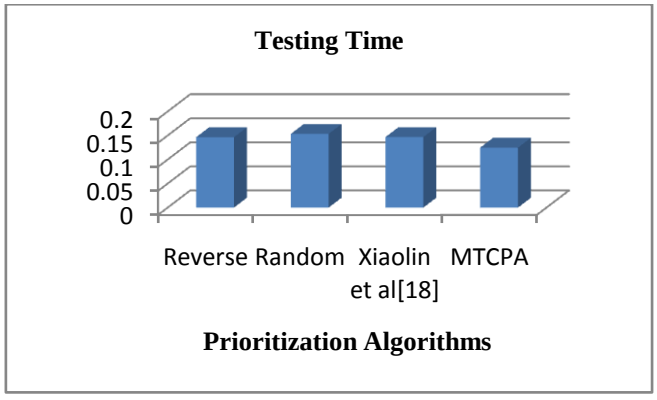

Figure 3 Comparison based on Testing Time

The Average Percentage Fault Detection (APFD) is measures the weighted average of the percentage of faults detected over the test suite. APFD values ranges from 0 to 100 and higher value of APFD imply better faults detection rates [5].
Let T be a test suite containing n test cases and F be a collection of m faults revealed by T. Let TF_i be the first test case in ordering T' of T which reveals fault i. The APFD for test case suite T' is given by the equation:
$APFD=1- ((TF_1 + TF_2 + \ldots + TF_m) / nm) + (1 / (2n))$
The aim of APFD quantifies to increase the rate of fault detection of the test suite. Calculate the fault detection using the Table I.

TABLE I NUMBER OF FAULTS DETECTED BY TEST CASES

Faults/Test cases	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10
T1	*	*					*	*	*	*

T2	*	*	*	*						
T3	*	*	*		*	*	*			*
T4	*	*					*			*
T5	*	*	*		*	*	*			*

The proposed optimal order of test cases is compared with reverse order, random order and Xiaolin Wang et al.[18] order of test cases by calculating the average percentage fault detection (APFD) for each order as represented in Figure 4.

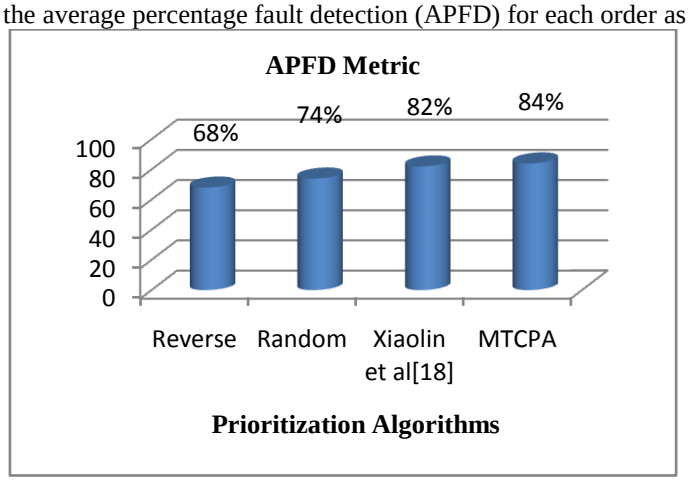

Figure 4 APFD for different prioritization

The results show that MTCPA algorithm is more efficient than rest of the three prioritization algorithms (Reverse, Random and Xiaolin Wang et al.[18] prioritization) in terms of statement coverage, testing time and APFD values.

V. CONCLUSION AND FUTURE SCOPE

In this paper, MTCPA (Multi-objective Test Case Prioritization Algorithm) is proposed for test case prioritization. This algorithm uses two criteria to prioritize test cases, these are statement coverage and testing time. The MTCPA uses the concept of multi objective optimization technique in which more than one objective is used to obtain optimal solution (order of test cases). This paper shows thatthe proposed algorithm schedules the test cases in a sequence that covers the maximum statement coverage and minimum testing timealongwith maximum APFD (Average Percentage FaultDetection). This proposed algorithm,as found from results is more efficient as compared to the other prioritization techniques such reverse, random and Xiaolin Wang et al.[18] prioritization.Future perspective of this work is to add other objective (like requirement coverage and risk) for test case prioritization using multi objective genetic algorithm.

REFERENCES

[1] Wong W.E., Horgan J.R., Londonm S. and Agrawal H., "A study of effective regression testing in practice", In: Proc. 8th Int'l Symp. Soft.Reliability Eng., pp. 230–238, 1997.

[2] Rothermel G., Untch R., Chu C. and Harrold M., "Test case prioritization: An empirical study", IEEE International conference of Software Maintenance, pp. 179-188, 1999.

[3] Deb K., "Multi-Objective Evolutionary Algorithms: Introducing basic Among Pareto-Optimal Solutions", Kanpur Genetic Algorithms Laboratory Report, 1999.

[4] Elbaum S., Malishevsky A. and Rothermel G., "Prioritizing test cases for regression testing", Proc. The ACM SIGSOFT International Symposium on Software Testing and Analysis, Portland, Oregon, U.S.A., pp. 102–112, 2000.

[5] Elbaum S., Malishevsky A. and Rothermel G., "Test case prioritization: A family of empirical studies", IEEE Transactions on Software Engineering, vol. 28, no. 2, pp. 159-182, 2002.

[6] Roger S. Pressman, Software Engineering: a practitioner's approach, 6th edition, McGraw-Hill, 2005.

[7] Srikanth, H., Williams L. and Osborne J. "System Test Case Prioritization of New and Regression Test Cases", International Symposium on Empirical Software Engineering, 2005.

[8] Aggarwal K.K. and Yogesh Singh, "Software Engineering", 3th edition, New Age International Publishers, pp. 459-470, 2007.

[9] Zheng Li, Harman M. and Hierons R.M. "Search Algorithms for Regression Test Case Prioritization", IEEE Trans. on Software Engineering, vol. 33, no. 4, pp. 225-237, 2007.

[10] Xiaofang Zhang, Nie C., Xu B. and Qu B., "Test Case Prioritization based on Varying Testing Requirement Priorities and Test Case Costs", 7th International Conference on Quality Software, pp. 15-14, 2007.

[11] KhinHla, Choi Y. and ParkJ.S.. "Applying Particle Swarm Optimization to Prioritizing Test Cases for Embedded Real Time Software Retesting", 8th IEEE Int. Conf. on Computer and Information Technology Workshops, pp. 527-532, 2008.

[12] Smith A.M. and Kapfhammer G.M. "An Empirical Study of Incorporating Cost into Test Suite Reduction and Prioritization", Proceedings of the ACM symposium on Applied Computing, pp.461-467, 2009.

[13] Mohanty S., Acharya A. and Mohapatra D., "A survey on model based test case prioritization", International Journal ofComputer Science and Information Technologies, pp. 1042-1047, 2011.

[14] Yoo S. and Harman M., "Regression testing minimization, selection and prioritization: a survey", SoftwareTesting, Verification and Reliability, pp. 67-120, 2012.

[15] AmrAbdelFatah Ahmed, Dr. Mohamed Shaheen and Dr.EssamKosba, "Software testing suite prioritization using multi criteria fitness function", IEEE ICCTA, Alexandria, Egypt, pp. 160-166, 2012.

[16] Islam M.M., Marchetto A., Susi A. and Scanniello G., "A Multi Objective Technique to Prioritize Test Cases Based on Latent Semantic Indexing", Proc. 16th European Conference Software Maintenance and Reeng., pp. 21-30, 2012.

[17] Ashraf E., Rauf A., and Mahmood K., "Value based Regression Test Case Prioritization", Proceedings of the World Congress on Engineering and Computer Science, pp. 24-26, 2012.

[18] Xiaolin Wang and HongweiZeng, "Dynamic test case prioritization based on Multi-Objective", 15th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), pp. 1-6, 2014.

[19] ManikaTyagi and SonaMalhotra, "Test Case Prioritization using Multi Objective Particle Swarm Optimizer", IEEE International Conference on Signal Propagation and Computer Technology (ICSPCT), pp. 390-395, 2014.

[20] Mitrabinda Ray and Durga Prasad Mohapatra, "Multi-objective test prioritization via a genetic algorithm", Innovations System Software Eng Springer, pp. 261-270, 2014.

