Pair-Wise Time-Aware Test Case Prioritization for Regression Testing

Prem Parashar\textsuperscript{1}, Arvind Kalia\textsuperscript{1}, and Rajesh Bhatia\textsuperscript{2}

\textsuperscript{1}Computer Science Department, Himachal Pradesh University, Shimla, India
\textsuperscript{2}Computer Science Department, Deen Bandhu Chotu Ram University, Murthal, India
\{prem.parashar, arvkalia, rbhatiapatiala\}@gmail.com

Abstract. After maintenance, software requires regression testing for its validation. Prioritization of test cases for regression testing is required as software is tested under strict time and other constraints. A Pair-wise time-aware Test Case Prioritization (PTCP) technique has been proposed in this paper that determines the effectiveness of a test case on the basis of total number of faults present in software, number of faults detected till time, and the time of execution of different test cases. It selects that test case which determines maximum new faults, not yet detected, within minimum time. Thus prioritized test suite contains those test cases which are effective and tend to minimize repetitive faults detection. Through two comparative studies, it has been observed that with least wastage of time, the proposed technique performed equally well as other two parallel prioritizing techniques, Average Percentage of Fault Detection (APFD) based prioritization, and Optimal Test Case Prioritization (OTCP).

Keywords: Regression testing, prioritization, fault detection, redundancy, random selection.

1 Introduction

Software testing is one of the very expensive stages of software development life cycle [22]. It is an important part of the software development irrespective of the programming paradigm used. It is a broad term containing a wide spectrum of different activities. It starts from the testing of small unit of software to the post implementation and maintenance of software. The main activities involved in software testing are preparing test cases and their respective test oracles [12]. Since both these tasks are very tedious, therefore even after preparing test cases and test oracles with utmost care, some bugs remain uncovered. Software may malfunction due to uncovered bugs. The consequences of such errors may be nominal or catastrophic depending upon the type of software application [13, 14, 15, 22]. In order to avoid such situations, the system is tested with sufficient number of test cases and the gap between expected value and actual value is observed. Any difference between these values declares the system as erroneous.
Regression testing has always remained a challenge for maintenance team. As maintenance is not a regular or periodic activity for most of the software[13], it is very difficult to make prediction about software maintenance time. For some software systems like business applications, frequency of maintenance may be very high whereas in other types like scientific application, it may be low. When software is brought for maintenance, it is very difficult to reconstitute the same team which had developed and tested it. The formation of new maintenance team makes this process complex.

One of the most important constraints in regression testing is time budget [6]. With the maintenance of software, the size of test cases usually grows making it difficult to execute all the existing test cases and new test cases with in the specified time limit. Prioritization of test cases is even significant if the time budget and total time for the execution of a test suite are equal. After modification, when new changes are incorporated in existing software, software need to be tested well by using a test suite that covers change impact set (all modified components of software and the other components affected by this change). Generally, due to limited time constraints, it is impossible to execute all test cases of a test suite [11, 20, 22]. Under such circumstances, a test case prioritization technique is required which can generate an optimum subset of test cases that covers all or almost all changes. It is not an easy task to construct such subset of given test suite, especially if the number of changes made is very large. Researchers have developed different types of test case prioritization techniques. In this paper it is evident that those techniques also prove to be effective under given time constraints. Pair-wise Test Case Prioritization (PTCP) technique has been presented in this paper, which prioritizes test cases such that test cases selected from test suite (i) will always run within given time limit, (ii) will have the highest potential for fault detection, and (iii) will have the minimum wastage of time.

2 Review of Literature

Rothermel [16,17] et al. described different test case prioritization techniques and compared their relative results through empirical studies. The average percentage of fault detection (APFD) and total fault-exposing potential (TFEP) measures proposed by them have become benchmarks for making comparisons with other techniques. Dalal [3, 4] et al. proposed that automatic test cases that are generated by using combinatorial approach of pair wise interaction between the input fields are highly effective in detecting the failures as compared to traditional approach of software testing. By using this approach, number of test cases required for testing software can be reduced considerably.

Zang [23] et al. used integer linear programming (ILP) method for prioritizing test cases. They represented the test case prioritization problem as an integer linear programming of operations research constructing objective function with the help of number of faults detected by test cases and constraints by taking time of execution of test cases into consideration. The optimum solution of ILP gives the set of feasible test cases for the software and they are prioritized by implementing any of the