Abstract:

Regression testing is a critical activity which occurs during the maintenance stage of the software lifecycle. However, it requires large amounts of test cases to assure the attainment of a certain degree of quality. As a result, test suite sizes may grow significantly. To address this issue, *Test Suite Reduction* techniques have been proposed. Such techniques permanently remove some test cases from the test suite that, due to code modifications over time, have become redundant with respect to testing requirements for which they were generated. However, suite size reduction may lead to significant loss of fault detection efficacy. Since fault detection is the main purpose of software testing, it is necessary to establish a proper tradeoff between suite size and its fault detection effectiveness. In other words, an appropriate reduced test suite should exercise all different execution paths within the program while retaining the fault detection capability of the suite at an acceptable level. To achieve this, three algorithms for test suite reduction have been presented in this thesis. In the first two algorithms, clustering of the test cases' execution profiles is utilized to distinguish redundant test cases from essential ones. When sampling from clusters, a test case which satisfies the maximum number of testing requirements is selected, because the coverage of testing requirements can provide a reasonable estimation of faults behavior. The third algorithm attempts to select a test case which satisfies the maximum number of testing requirements while having minimum overlap in requirements coverage with other test cases.

In order to evaluate the proposed algorithms, experiments similar to those used in prior studies have been conducted on the *Siemens suite* and the *Space program.* The experiments have investigated the effectiveness of the algorithms by reducing 6000 test suites in 6 different suite size ranges with varying number of test cases for each program. The results demonstrate the effectiveness of the proposed algorithms by retaining the fault detection capability of the suites while achieving significant suite size reduction.

Keywords: Software regression test, testing criteria, test suite reduction, test suite minimization, fault detection effectiveness.