Abstract

For developing successful software, testing is a very important component. In software testing, providing input, executes it and check expected output. Many techniques which automatically produce inputs have been proposed over the years, and today are able to produce test suites with high code coverage. In software testing a common scenario is that test data are generated and a tester manually adds test cases. It is a difficult task to generate test cases manually but it is important to produce small representative test sets and this representativeness is typically measured using code coverage. But there is a fundamental problem with the common approach of targeting one coverage goal at a time. Coverage goals are not independent, not equally difficult, and sometimes infeasible—the result of test generation is therefore dependent
on the order of coverage goals and how many of them are feasible. For solving these problems, propose a novel paradigm which is generation of whole test suite based on search based testing. Instead of evolving each test case individually, evolve all the test cases in a test suite at the same time. At the end, the best resulting test suite is minimized.

References


Index Terms

Computer Science

Algorithms

Keywords

Length Software-based Software Engineering Branch Coverage Memetic Algorithm
Local Search.