

Improving Branch Coverage for White-box Testing Slicing

Myint Myitzu Aung

myintmyitzaung@gmail.com

University of Computer Studies, Mandalay, UCSM

Kay Thi Win

kthiwin11@gmail.com

University of Computer Studies, Mandalay, UCSM

Software testing is a testing technique for the quality of software in which white-box testing brings the complexity because it has disadvantage that the tester must have the knowledge of the program. Moreover, in conventional concolic testing, branch coverage is not high and there is a need for improving performance in measuring the branch coverage. Therefore, this paper is intended to improve performance and reduce the complexity in measuring the branch coverage for white-box testing. An effective branch coverage measuring plan is proposed in which the input program will be divided into all possible positive and negative branches and only executable branches are sliced. And then coverage is measured by using new proposed metric. To reduce complexity, the original input program is transformed into slice program by using Tree Slicing. Then, to improve performance, branch coverage is produced by using bidirectional symbolic analysis that combines symbolic reachability and symbolic execution and new coverage metric. Path-Sensitively Sliced CFG (PSS-CFG), “Tree Slicing” is used. It is a technique to slice and merge different sub-trees under certain conditions to reduce time complexity. This paper will provide more effective branch coverage by using the sliced program, removing the infeasible branches and new coverage metric.