

Empirical evaluation of the fault detection effectiveness and test effort efficiency of the automated AOP testing approaches.

ABSTRACT

Testing process is a time-consuming, expensive, and labor-intensive activity in any software setting including aspect-oriented programming (AOP). To reduce the testing costs, human effort, and to achieve the improvements in both quality and productivity of AOP, it is desirable to automate testing of aspect-oriented programs as much as possible. In recent past years, a lot of research effort has been devoted to testing aspect-oriented programs but less effort has been dedicated to the automated AOP testing. This denotes that the current research on automated AOP testing is not sufficient and is still in a stage of infancy. In order to advance the state of the research in this area and to provide testers of AOP-based projects with a comparison basis, a detailed evaluation of the current automated AOP testing approaches in a thorough and experimental manner is required. Thus, the objective of this paper is to provide such evaluation of the current approaches. In this paper, we carry out an empirical study based on mutation analysis to examine four (namely Wrasp, Aspectra, Raspect, and EAT) existing automated AOP testing approaches, particularly their underlying test input generation and selection strategies, with regard to fault detection effectiveness. In addition, the approaches are compared in terms of required effort in detecting faults as part of efficiency evaluation. The experimental results and comparison provided insights into the effectiveness and efficiency of automated AOP testing with their respective strengths and weaknesses. Results showed that EAT is more effective than the other automated AOP testing approaches but not significant for all approaches. EAT was found to be significantly better than Wrasp at 95% confidence level (i.e. $p < 0.05$), but not significantly better than Aspectra or Raspect. Concerning the test effort efficiency, Wrasp was significantly ($p < 0.05$) efficient with requiring the lowest amount of test effort compared to the other approaches. Whereas, EAT showed to be not very efficient by recording the highest amount of test effort. This implies that EAT can currently be the most effective automated AOP testing approach but perhaps less efficient. More generally, search-based testing (as underlying strategy of EAT approach) might achieve better effectiveness but at the cost of greater test effort compared to random testing (as underlying strategy of other approaches).

Keyword: Aspect-oriented programming; Software testing; Automated testing; Aspect testing; Empirical study.