Chapter 77
Test Suite Optimization Using Firefly and Genetic Algorithm

Abhishek Pandey
University of Petroleum and Energy Studies, Dehradun & Birla Institute of Technology, Mesra-Ranchi, India

Soumya Banerjee
Conservatoire Nationale des Arts et Metiers & INRIA Paris ex-visiting CNRS INSA de Lyon, Paris, France

ABSTRACT

Software testing is essential for providing error-free software. It is a well-known fact that software testing is responsible for at least 50% of the total development cost. Therefore, it is necessary to automate and optimize the testing processes. Search-based software engineering is a discipline mainly focussed on automation and optimization of various software engineering processes including software testing. In this article, a novel approach of hybrid firefly and a genetic algorithm is applied for test data generation and selection in regression testing environment. A case study is used along with an empirical evaluation for the proposed approach. Results show that the hybrid approach performs well on various parameters that have been selected in the experiments.

INTRODUCTION

Software testing is the most expensive and time-consuming task among all other activities that are performed in software engineering (Myers, Sandler & Badgett, 2011; Korel, 1992). The first idea of software testing is probably due to Turing (Turing, 1949). The first mention of software optimization of any kind is due to Ada Augusta Lovelace in 1842 (Harman, 2010). The first application of optimization techniques in software testing is due to the seminal work of James King (King, 1969). Search based software engineering is an emerging area of research to optimize various software engineering processes.

DOI: 10.4018/978-1-6684-3702-5.ch077

Search-based software engineering reformulates software-testing problem as an optimization problem (Harman & Jones, 2001). Search based methods (SBMs) also applied in various testing problems such as test data generation, test suite minimization, test case selection and test case prioritization in the literature (Harman, Jia & Zhang, 2015). Software module clustering and software refactoring problem are good candidates for the application of search-based techniques (Harman, Mansouri & Zhang, 2012). Regression testing is performed in the software maintenance phase of the software development life cycle. The whole software undergoes retesting whenever any modification occurs during regression testing. Regression test case selection techniques strive to increase the testing quality based on the test adequacy criteria, such as effort, coverage, and fault detection.

Various new software development paradigms impose many restrictions on regression testing. Retesting is necessary in these cases. In this case, regression testing must be performed using the available computing resources judicially. Regression testing problem can thus be seen as a combination of test suite minimization, test suite selection, test suite prioritization problem in order to save computing resources (Rothermel & Harrold, 1996). The present work is an attempt to optimize the regression testing process and to evaluate the performance of newly proposed nature-inspired algorithms such as hybrid firefly and genetic algorithm for test case selection. In this paper, we are proposing a new hybrid algorithm for test case selection problem. Initially, a case study based on the available test suite is performed for regression testing and simulation results are shown. Results shows the better performance of hybrid approaches when compared to some popular swarm intelligence-based algorithms.

Rest of the paper is organized as related work, problem formulation, proposed methodology, experimental evaluation, results and conclusion.

RELATED WORK

Agrawal & Kaur (2018) compares the performance of two metaheuristics namely ant colony and hybrid particle swarm optimization exclusively for test cases selection problem. The quality parameters in this research are execution time and fault coverage. Experiments were performed using Matlab. This article demonstrates the significance of hybrid algorithms for test cases selection problem in software engineering.

De Oliveira Neto et al. (2018) evaluates similarity-based test case selection on integration level tests. The results confirm the existing strong evidence that similarity-based test case selection is the major candidate for test optimization.

Choudhary, Agrawal & Kaur (2018) presents an effective method for test case selection using Pareto based multi-objective harmony research. Fors et al. (2019) present a safe regression test case selection for Modelica using static analysis.

Nogueira et al. (2019) discuss model-based testing using natural language description of use cases. It is important to note that formal description of use cases using mathematical notation poses challenges in test case generation and selection process. To overcome this issues, use cases are described in natural language that is easily understandable to the testing team.

Arrieta et al. (2019) describes a search-based approach for prioritizing the test cases in cyber physical system (CPS). Wang et al. (2019) proposes a location-based test case prioritization for embedded software using law gravitation.

Shin et al. (2019) discusses empirical evaluation of various test case prioritization techniques in a recent research study. Sahoo & Ray (2018) presents a comprehensive review of various search-based