

Chapter 32

Test Suite Optimization Using Chaotic Firefly Algorithm in Software Testing

Abhishek Pandey

*School of Computer Science and Engineering, University of Petroleum and Energy Studies,
Dehradun, India*

Soumya Banerjee

Department of Computer Science and Engineering, Birla Institute of Technology Mesra, Ranchi, India

ABSTRACT

Software testing is time consuming and a costly activity. Effective generation of test cases is necessary in order to perform rigorous testing. There exist various techniques for effective test case generation. These techniques are based on various test adequacy criteria such as statement coverage, branch coverage etc. Automatic generation of test data has been the primary focus of software testing research in recent past. In this paper a novel approach based on chaotic behavior of firefly algorithm is proposed for test suite optimization. Test suite optimization problem is modeled in the framework of firefly algorithm. An Algorithm for test optimization based on firefly algorithm is also proposed. Experiments are performed on some benchmark Program and simulation results are compared for ABC algorithm, ACO algorithm, GA with Chaotic firefly algorithm. Major research findings are that chaotic firefly algorithm outperforms other bio inspired algorithm such as artificial bee colony, Ant colony optimization and Genetic Algorithm in terms of Branch coverage in software testing.

1. INTRODUCTION

Software testing is laborious and costly activity. It claims nearly 50% of the total cost of software projects (Myers, 1979). Automation of software testing processes could reduce the time and cost of development of software (Miller, & Spooner, 1976). Many of the software testing problems require solutions which satisfies conflicting objectives such as minimum set of test data which achieves maximum coverage

DOI: 10.4018/978-1-7998-3016-0.ch032

criteria. Reformulating software engineering tasks as an optimization problem is referred as search based software engineering (Harman, & Jones, 2001).

Bio inspired computational intelligence algorithms are gaining its popularity by the fact that these algorithms performed well on various parameters and give the converged optimized solution. Here it will be worth noting that many conventional methods such as newton's method or gradient descent method, simplex method requires lot of assumption in order to model the problem and thus unable to model the real world problem correctly and precisely. Linear programming methods are also unable to model the real world problem because many of the real world problems had been proved as a nonlinear problem.

Software Test data generation problem is a non- linear problem, refers to the generation of valid input data that satisfy the certain testing goal. Test data generation problem can be reformulated as an optimization problem where search space is the values of input variables and fitness function is designed for test adequacy criteria such as code coverage. Test suite optimization refers to optimize test data either in structural software testing or in regression testing.

Automatic software testing and test data optimization has been the area of interest in the research community since last four decades. Previously two approaches of the problem emerged. The first one came to be known as symbolic execution (King, 1976) and the second method that reformulated the problem as an optimization problem, which latter known as search based approach (Harman et al., 2003).

There exist various optimization algorithms which are used to optimize software test data. The application of evolutionary algorithms to test data generation is called as evolutionary testing in the literature (Xanthakis et al., 1992). Search based software engineering (Harman et al., 2003) reformulated the software engineering problems as an optimization problem. In this area of research various software metrics are taken as a fitness function. The major area research in SBSE is in the field of software testing.

Test suite optimization is required in various software applications including cloud computing and smart cities. Search based software testing is also used with hadoop for generating test data using genetic algorithm (Geronimo, Ferrucci, Murolo, & Sarro, 2012). Strong mutation testing using hill climbing of C programs yield better performance than random search according to recent paper (Souza, Papadakis, Le Traon, & Delamaro, 2016). Considering the broader aspect of search based software engineering (SBSE), many applications and models have been proposed still enhancement of quality of test cases and measuring the efficiency of test cases is yet to achieve the best results.

The principal reason of such scope of results is due to the presence of different set of constraints during the output phase for different software applications. Evolutionary methods as mentioned earlier could be a major motivation due to its heuristic properties of search towards different test cases.

Firefly algorithms are swarm intelligence based meta-heuristics so it has all the advantages of other swarm algorithms. Firefly algorithms can save around 78% computational cost when compared to Genetic Algorithms and 67% computational cost when compared to PSO (Xin-She Yang, & Xingshi He., 2013). Parameters can be tuned to control the randomness as iteration proceed. This speed up the convergence process.

There are basically two advantages of applying firefly algorithm that are automatic subdivision and ability to control multi-modality. In this paper a novel approach of Chaotic behavior of the firefly algorithm has been utilized for modeling the test suite optimization problem. Methodology for carrying out this research work follows exhaustive literature review from software engineering and Swarm intelligence algorithms. It has been found that though there are good number of publications exists in the field of software testing, test data generation and test suite optimization as well as on bio inspired optimization algorithm, their exists less publication exists when these two domains overlapped.

16 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/test-suite-optimization-using-chaotic-firefly-algorithm-in-software-testing/261051

Related Content

Formalization Studies in Functional Size Measurement

Baris Özkanand Onur Demirörs (2012). *Computer Engineering: Concepts, Methodologies, Tools and Applications* (pp. 1758-1778).

www.irma-international.org/chapter/formalization-studies-functional-size-measurement/62543

Enhanced Speech-Enabled Tools for Intelligent and Mobile E-Learning Applications

S- A. Selouani, T-H. Lê, Y. Benahmedand D. O'Shaughnessy (2012). *Computer Engineering: Concepts, Methodologies, Tools and Applications* (pp. 1913-1932).

www.irma-international.org/chapter/enhanced-speech-enabled-tools-intelligent/62553

Calling Police Using SMS

Mohammad Shirali-Shahrezaand M. Hassan Shirali-Shahreza (2012). *Computer Engineering: Concepts, Methodologies, Tools and Applications* (pp. 914-923).

www.irma-international.org/chapter/calling-police-using-sms/62487

Extended Time Machine Design using Reconfigurable Computing for Efficient Recording and Retrieval of Gigabit Network Traffic

S. Sajan Kumar, M. Hari Krishna Prasadand Suresh Raju Pilli (2012). *Computer Engineering: Concepts, Methodologies, Tools and Applications* (pp. 699-709).

www.irma-international.org/chapter/extended-time-machine-design-using/62473

Business Model Development for Stability, Sustainability, and Resilience

Beata Maria Staszewska (2020). *Disruptive Technology: Concepts, Methodologies, Tools, and Applications* (pp. 240-264).

www.irma-international.org/chapter/business-model-development-for-stability-sustainability-and-resilience/231190