Test Data Generation Based on Test Path Discovery Using Intelligent Water Drop

Praveen Ranjan Srivastava, Birla Institute of Technology and Science, Pilani, India
Amikumar Patel, Birla Institute of Technology and Science, Pilani, India
Kunal Patel, Birla Institute of Technology and Science, Pilani, India
Prateek Vijaywargiya, Birla Institute of Technology and Science, Pilani, India

ABSTRACT

Automatic test data generation is required to generate test cases dynamically for a specific software program. Manual generation of test data is too tedious and a time consuming task. This paper proposes a technique using Intelligent Water Drop (IWD) for automatic generation of test data. Correctly generated test data helps in reducing the effort while testing the software. This paper discusses different algorithms based on IWD to generate test data and path coverage over Control Flow Graph. Test data is generated keeping in mind all of the programming constraints like “if,” “while,” “do while,” etc., available in the program.

Keywords: Control Flow Graph (CFG), Intelligent Water Drop (IWD), Software Testing, Test Data Generation, Test Path Discovery

INTRODUCTION

In software engineering area, software testing is a crucial phase (Kewen, Zilu, & Wenying, 2009) to assess the quality of the software. Software testing contains almost 50% of total cost of the software development (Edvardsson, 1999). Aim of the software testing is to uncover errors and faults present in the program, so that customer requirement can be properly fulfilled. Testing phase includes in the review of specification, analysis, design, and implementation part of the Software Development Life Cycle (SDLC). Manual generation of test data for testing the program, results in low reliability and high cost (Li & Lam, 2005; Pressman, 2010). Due to the lack of cost and reliability, automation of testing process is necessary, so that the cost of testing can be reduced.

Software engineering is a knowledge-intensive activity, which requires access and manipulation of large quantity of information about the project domain (Naaz Raza, 2009). By managing these facts manually, cost becomes very high. Artificial Intelligence (AI) based techniques can help in removing this situation.
AI based technique helps in solving the problem by using fast and proper judgments rather than using step by step deduction (Naaz Raza, 2009). From the last decade various techniques like genetic algorithm, ant colony optimization and other metaheuristic techniques have been introduced for generating test path and test data (Korel, 1990; Michael et al., 1997; Michael & McGraw, 1998; McMinn, 2004; Kewen et al., 2009; Srivastava, Baby, & Raghurama, 2009).

The main issues of various metaheuristic techniques (McMinn, 2004) are optimal test data generation and complete software coverage. Another new optimization technique, Intelligent Water Drop (IWD) (Shah-Hosseini, 2009), an algorithm which is based on swarm optimization techniques, can be used for generation of test data. IWD algorithm works similar to the natural water drop of a river bed.

This paper applies IWD algorithm to generate test data and also discover existing paths in the program. This paper is structured as follows; first, we describe the background work of software testing. Then, we present the basic IWD optimization approach. The next section applies IWD approach to generate test path and sequence generation. We then describe the case study of the suggested approach for test data generation, while the next section discusses the analysis part. Finally we conclude the paper along with future scope of the applied approach.

BACKGROUND

Various techniques (Korel, 1990; Pedrycz & Peters, 1998; Briand, 2002; McMinn, 2004; Harman, 2007; Kewen et al., 2009; Srivastava et al., 2009; Srivastava & Baby, 2010) have been proposed for automated testing to reduce efforts to a remarkable extent. Swarm optimization techniques are widely used for test data generation. Ant Colony Optimization (ACO) is one of them (Li & Lam, 2005; Kewen et al., 2009; Srivastava et al., 2009). Automatic test data generation based on ACO (Kewen et al., 2009) has introduced a model to generate test data using the branch function technique. It has solved the problem of local optimization, but this approach is applicable only for numeric data types and also the model is not suitable for object oriented programs and other types of input behavior. Software test data generation using ACO (Li & Lam, 2005) has also proposed a solution for state-based software testing but complete coverage is not possible by the proposed approach.

Genetic algorithm has been applied for dynamic test data generation (Michael et al., 1997) and the results are pretty impressive over random or exhaustive test data generation, but this model was not applicable for boolean and string type of variables.

Another approach is suggested for generation of test data using genetic algorithm and hamming distance concept (Srivastava et al., 2010) but optimal test data are uncertain, and stuck in the local optima and explore more repetitive paths. An approach for software testing has been recommended (Srivastava et al., 2009) in which ACO procedure is used to discover effective paths which are promising for full path coverage.

Genetic algorithm and tabu search have been applied on control-dependence graph (Rathore et al., 2011). This approach uses only the number of common predicates with the target node for finding objective value, but few important factors such as distance from initial and final node, criticality, etc., that it fails to take into account.

Many papers (Korel, 1990; McMinn, 2004; Srivastava et al., 2009) have suggested techniques for full path coverage and other have suggested approaches for optimal test data generation (Li & Lam, 2005; Srivastava et al., 2008; Kewen et al., 2009; Srivastava et al., 2010), but none of them have suggested combined approaches for tackle both the problems together though they are very closely related. This paper discusses the newly emerging approach for combined solutions to both of these problems together.

IWD algorithm (Shah-Hosseini, 2007; Shah-Hosseini, 2009) is a swarm-based optimization algorithm, simulated from observing
Related Content

Dynamic Document Clustering Using Singular Value Decomposition
[www.igi-global.com/article/dynamic-document-clustering-using-singular-value-decomposition/79915?camid=4v1a](www.igi-global.com/article/dynamic-document-clustering-using-singular-value-decomposition/79915?camid=4v1a)

Using Fuzzy Song Sets in Music Warehouses
François Deliège and Torben Bach Pedersen (2010). *Scalable Fuzzy Algorithms for Data Management and Analysis: Methods and Design* (pp. 54-83).
[www.igi-global.com/chapter/using-fuzzy-song-sets-music/38565?camid=4v1a](www.igi-global.com/chapter/using-fuzzy-song-sets-music/38565?camid=4v1a)

Application of Meta-Heuristic Optimization Algorithms in Electric Power Systems
[www.igi-global.com/chapter/application-meta-heuristic-optimization-algorithms/69898?camid=4v1a](www.igi-global.com/chapter/application-meta-heuristic-optimization-algorithms/69898?camid=4v1a)
Utilizing the Modified Self-Adaptive Differential Evolution Algorithm in Dynamic Cellular Manufacturing System


[www.igi-global.com/article/utilizing-modified-self-adaptive-differential/67330?camid=4v1a](www.igi-global.com/article/utilizing-modified-self-adaptive-differential/67330?camid=4v1a)