IGI PUBLISHING
ITB14918
701 E. Chocolate Avenue, Suite 200, Hershey PA 17033-1240, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.igi-pub.com
This paper appears in the publication, **Advances in Machine Learning Applications in Software Engineering**
edited by **Du Zhang & Jeffrey J.P. Tsa © 2007, IGI Global**

Chapter XII

Application of Genetic Algorithms in Software Testing*

Baowen Xu, Southeast University & Jiangsu Institute of Software Quality, China

Xiaoyuan Xie, Southeast University & Jiangsu Institute of Software Quality, China

Liang Shi, Southeast University & Jiangsu Institute of Software Quality, China

Changhai Nie, Southeast University & Jiangsu Institute of Software Quality, China

Abstract

Genetic algorithms are a kind of global meta-heuristic search technique that searches intelligently for optimal solutions to a problem. Evolutionary testing is a promise testing technique, which utilises genetic algorithms to generate test data for various testing objectives. It has been researched and applied in many testing areas, including structural testing, temporal performance testing, safety testing, specification-based testing, and so forth. Experimental studies have shown that compared with the traditional techniques, evolutionary testing can greatly improve the testing efficiency.

Background of Evolutionary Testing

Motivation of Software Testing

Software engineering is always a complicated domain characterised by a large number of competing and inter-related constraints. With the rapid development of software scale and complexity, more and more software engineering problems, such as initial planning, requirements analysis, cost estimation, system integration, software testing, system maintenance, and so on, become vague and hard to solve, which makes it more difficult to balance these problems in the limited budget and schedule. Since traditional methods may not meet the continuous increasing complexity, promising techniques based on artificial intelligence have been proposed and researched. Nowadays, many software engineering problems have been considered from a new perspective as search problems, which meta-heuristic search techniques can be applied to.

Software testing is one of the significant components of software engineering with many complex and inter-related constraints. Efficient testing requires systematic and automatic test data generation to satisfy pre-defined standards. However because of the increasing complexity of software systems, usual testing techniques have demonstrated their limitation in certain areas.

- In functional testing and structural testing, test data are often generated manually according to some testing standards. Yet, it is hard to automate the generation task, which makes it too resource intensive.
- In safety testing and performance testing, it is always required to generate test data to examine software behaviours in some critical or dynamic situations. Since there are many implicit factors that influence the system's behaviours unpredictably, it may be a hard task for test data generation. In many cases, the manual work can hardly increase the confidence in software correctness.
- Random testing can be completely automated. However, since it is not performed systematically, it may generate excessive test data that have low possibility to satisfy the testing requirements and low efficiency to detect errors in software.

Test data generation can be transformed into a search problem in which meta-heuristic techniques are applied. Meta-heuristic search techniques refer to a set of generic algorithms that search intelligently for optimal or near optimal solutions to a problem within various search space. Since the search space of software testing is usually large, non-linear, and discontinuous, local search-based methods, such as hill climbing, are inefficient to find good solutions. However, genetic algorithms are a kind of global search-based strategy, which has been proved suitable for software testing (Clark et al., 2003; Wegener, 2001).

Evolutionary testing is a promise testing technique, which utilises genetic algorithms to generate test data for various testing objectives. Experimental studies have shown that evolutionary testing can be very efficient in various testing areas, including structural testing, temporal performance testing, safety testing, specification-based testing, and so forth. This