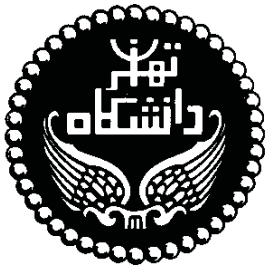

University of Tehran
School of Engineering
Faculty of Electrical and Computer Engineering

Title

Evolutionary Computing of Software Design and Test Cases

Author

Mehdi Amoui Kalareh

Advisor

Dr. Caro Lucas

Co-Advisor

Dr. Marjan Sirjani

Master Thesis in
Computer Engineering
Artificial Intelligence and Robotics

February 2006

ABSTRACT

High quality software needs to satisfy a wide range of requirements. Therefore software errors and violation of software requirements shape a huge set of bugs. As many of these bugs share similar characteristics, we can categorize them from different aspects. A candidate clustering may be categorizing common software tests and errors followed by the effort on fixing the errors through general solutions for each test/error pair. Our approach to address this issue is based on Christopher Alexander's pattern and pattern language concept. The patterns in this language are grouped into three major sections and connect the three concepts of test, error, and debug. These patterns and their hierarchical relationship shape a pattern language that introduces a solution to solve software errors in a known testing context.

To address the solutions proposed in this Pattern Language, we have developed a framework called ACE as a sample implementation to support several patterns of this language by automating the whole process of evolving software via evolutionary methods. By using this framework, user will be able to define, customize and map his problem defined in a particular pattern as an evolutionary problem, and automatically evolve his design problem to the optimized solution.

Moreover, optimizing an object-oriented design to ameliorate its quality is a notoriously difficult, but necessary, activity in software development process. Throughout this thesis, we will propose a GA approach to find the best sequence of valid high-level design transformations to improve software reusability while trying to preserve other aspects of software quality. This evolutionary multi-objective optimization problem is implemented, in the developed ACE framework, using two approaches. In the first approach, by using GOF design patterns and several design metrics, we implement an automated environment to improve the design quality with design level transformations. In the second approach, we tend to automate design transformations, based on meta-patterns, at source code level to find the fittest sequence of valid transformations according to the related design metrics. The results of applying these two methods on several open source applications show that these methods can effectively optimize the primary metric while preventing other metrics to deteriorate. We conclude that this framework can be used as a decision support system to suggest the best feasible sequence of design transformations.