



**ARAB ACADEMY FOR SCIENCE, TECHNOLOGY AND
MARITIME TRANSPORT**

College of Computing and Information Technology

Information Systems Department

**SOFTWARE TESTING SUITE PRIORITIZATION
USING MULTI-CRITERIA FITNESS FUNCTION**

By:

AMR ABDELFATAH AHMED

Thesis submitted in partial fulfillment of the MASTER degree requirements

In:

INFORMATION SYSTEMS

Supervised by:

Dr. Mohamed Shaheen

Dr. Essam Kosba

College of Computing and Information Technology
ARAB ACADEMY FOR SCIENCE, TECHNOLOGY AND MARITIME TRANSPORT
Alexandria, Egypt
2012

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

وَإِذْ قَالَ رَبُّكَ لِلْمَلَأِكَةِ إِنِّي جَاعِلٌ فِي الْأَرْضِ خَلِيفَةً
قَالُوا أَتَجْعَلُ فِيهَا مَنْ يُفْسِدُ فِيهَا وَيَسْفِكُ الدِّمَاءَ وَنَحْنُ
نُسَبِّحُ بِحَمْدِكَ وَنُقَدِّسُ لَكَ قَالَ إِنِّي أَعْلَمُ مَا لَا تَعْلَمُونَ ﴿٣٠﴾
وَعَلَّمَ آدَمَ الْأَسْمَاءَ كُلَّهَا ثُمَّ عَرَضَهُمْ عَلَى الْمَلَأِكَةِ
فَقَالَ أَنْبِئُونِي بِأَسْمَاءِ هَؤُلَاءِ إِنْ كُنْتُمْ صَادِقِينَ ﴿٣١﴾ قَالُوا
سُبْحٰنَكَ لَا عِلْمَ لَنَا إِلَّا مَا عَلَّمْتَنَا إِنَّكَ أَنْتَ الْعَلِيمُ الْحَكِيمُ
﴿٣٢﴾ قَالَ يَكَادُمُ أَنْبِئُهُمْ بِأَسْمَائِهِمْ فَلَمَّا أَنْبَأَهُمْ بِأَسْمَائِهِمْ قَالَ
أَلَمْ أَقُلْ لَكُمْ إِنِّي أَعْلَمُ غَيْبَ السَّمٰوٰتِ وَالْأَرْضِ وَأَعْلَمُ مَا
تُبْدُونَ وَمَا كُنْتُمْ تَكْتُمُونَ ﴿٣٣﴾

DEDICATION

To my Father

To my Mother

To my family

To my teachers

To my colleagues

To everyone seeking truth

Acknowledgements

By the Name of Allah, the Most Gracious and the Most Merciful

I wish to thank Dr. Mohamed Shaheen for his thorough guidance and support throughout the research and his whole-hearted efforts in providing the necessary help and many useful discussions and suggestions for the research project. With gratitude I would also acknowledge Dr. Essam Kosba especially for his help during the research time and for the encouragement provided. Many thanks go to Dr. Mohamed kholief and Dr. Ayman Adel for valuable discussions and suggestions at meetings.

I am especially indebted to my colleagues especially Eng. Mohamed Nagi for his great efforts in helping me.

In loving Memory, I pray to Allah for the Heavens to be the home of Dr. Medhat Fkhary. As he always supported us with his knowledge and effort to be better students. Last but not least, I would like to thank my parents, my sister and my grandparents, without them I would not be here.

Abstract

Regression testing is the process of validating modifications introduced in a system during software maintenance. It is an expensive, yet an important process. As the test suite size is very large, system retesting consumes large amount of time and computing resources. Unfortunately, there may be insufficient resources to allow for the re-execution of all test cases during regression testing. Testcase prioritization techniques aim to improve the effectiveness of regression testing, by ordering the test cases so that the most beneficial are executed first with higher priority. The objective of test case prioritization is to detect faults as early as possible. An approach for automating the test case prioritization process using genetic algorithm with Multi-Criteria fitness function is presented. It uses multiple control flow coverage metrics. These metrics measure the degree of coverage of conditions, multiple conditions, and statements that the test case covers. These metrics are weighted by the number of faults revealed and their severity. The proposed Multi-criteria technique uses more coverage items compared to the other related work.

Table of Contents

	Page
Acknowledgments	i
Abstract	ii
Table of Contents	iii
List of tables	vi
List of figures	vii
Nomenclatures	ix
Chapter 1: Introduction	1
1.1 Overview	2
1.2 Research Problem	3
1.3 Objectives and aims of the research project	3
1.4 Methodology	4
1.5 Structure of Thesis	4
Chapter 2: Background of Software Testing	6
2.1 The Importance of software testing	6
2.2 Terminology of Testing.....	6
2.3 Testing techniques.....	7
2.4 Six Types of Software Testing.....	8
2.4.1 Unit testing	8
2.4.2 Integration testing.....	9
2.4.3 Functional and system testing	9
2.4.4 Acceptance testing.....	11
2.4.5 Regression Testing	11
2.4.6 Beta Testing.....	13
2.5 Deriving Test Cases	13
2.5.1 Basis Path Testing	13
2.5.2 Control-flow/Coverage Testing.....	15
2.5.3 Method Coverage	16
2.5.4 Statement Coverage.....	16
2.5.5 Branch Coverage	17
2.5.6 Condition Coverage.....	18

2.6 Determining Test Adequacy.....	19
2.7 Evaluating Test suites for Regression testing	20
2.7.1 Test Case Prioritization	21
2.7.2 Prioritized Test Suite Effectiveness.....	21
Chapter 3: Related Work	23
3.1 Overview on genetic Algorithm.....	23
3.1.1 History	23
3.1.2 Biological Background.....	24
3.1.3 Basic Genetic Algorithm	24
3.1.3.1 Working principals	25
3.1.3.2 Encoding.....	25
3.1.3.3 Operators of genetic algorithm	28
3.2 Test case prioritization using structural testing criteria	30
3.2.1 Genetic algorithm for test case prioritization using code coverage	30
3.2.2 A Bee Colony Optimization Algorithm for Fault Coverage Based Regression Test Suite Prioritization	32
3.2.3 Hybrid Particle Swarm Optimization for Regression Testing.....	33
3.2.4 Test Case Prioritization Using Average Faults Found Per Minute	35
3.2.5 Test Case Prioritization for Regression Testing based on Severity of Fault.....	37
3.3 Testing prioritization using functional test criteria	39
3.3.1 Factors Oriented Test Case Prioritization Technique in Regression Testing using Genetic Algorithm	39
3.3.2 Requirements based Test Case Prioritization using Genetic Algorithm	42
Chapter 4: Proposed Multi-Criteria Prioritization Technique	45
4.1 Introduction	45
4.2 Proposed Genetic Algorithm for Test Cases Prioritization	45
4.2.1 The Proposed Multi-Criteria Fitness Function	45
4.2.2 The encoding method	49
4.2.3 The algorithm operators	49
4.3 Proposed Multi-Criteria Prioritization Technique.....	51
Chapter 5: Implementation and Experiment Results.....	54
5.1 Phase 1: Find all possible independent paths	55
5.2 Phase 2: Assign each path to be a test case	57

5.3 Phase 3: Generate a Random Test Suites	58
5.4 Phase 4: Proposed Multi-Criteria Genetic Algorithm	60
5.5 Experimental Results and Analysis	60
5.6 Results Discussion.....	64
Chapter 6: Conclusion and Future work.....	65
6.1 Conclusion.....	67
6.2 Future Work	68
References.....	70
Appendix A.....	72

List of Tables

	Page
Table 2.1: Decision Coverage.....	17
Table 2.2: Condition coverage	18
Table 2.3: Condition Coverage Continued	19
Table 5.1: Assign each path to be a test case	58
Table 5.2: Test case Fault Matrix case study 1	61
Table 5.3: Test case Fault Matrix case study 2.....	62
Table 5.4: Comparison between test case prioritization Techniques (case study 1) ..	63
Table 5.5: Comparison between test case prioritization Techniques (case study 2) ..	63

List of figures

	Page
Figure 2.1: Regression Testing	12
Figure 2.2: Regression testing at different software testing levels	13
Figure 2.3: Flow graph of purchasing property	14
Figure 2.4: Sample Code for Coverage Analysis	16
Figure 2.5: Fragment of code	17
Figure 2.6: General Classification framework for testing criteria	20
Figure 3.1: Basic genetic algorithm flow chart	26
Figure 3.2(a): Tree encoding example 1	28
Figure 3.2(b): Tree encoding example 2	28
Figure 3.3: Test case prioritization technique	31
Figure 3.4: bee colony optimization algorithm	33
Figure 3.5: Hybrid Particle Swarm Optimization algorithm	34
Figure 3.6: Test Case Prioritization algorithm (Srivastava, 2008)	36
Figure 3.7: Test case prioritization algorithm based on Severity of Fault	38
Figure 3.8: regression test suite prioritization algorithm	41
Figure 3.9: Requirements based Test Case Prioritization approach	43
Figure 4.1: The applied genetic algorithm approach	48
Figure 4.2: Encoded Chromosome Example	49
Figure 4.3: crossover operation flowchart	50
Figure 4.4(a): Apply mutation operation Example	51
Figure 4.4(b): mutation operation result	51
Figure 4.5: The proposed Multi- criteria technique workflow block diagram ...	52
Figure 5.1: The Proposed Multi-Criteria Prioritization Technique in phases	54
Figure 5.2: Triangle problem source code	56
Figure 5.3: The Generated Control Flow Graph	57
Figure 5.4: Generate Randomly the Test Suites	59
Figure 5.5: The Random Generated Test Suites That Acts As Chromosomes	59
Figure 5.6: The Implementation of the Test Case Fault Matrix (case study 1) ..	61
Figure 5.7: The Implementation of the Test Case Fault Matrix (case study 2) ..	62
Figure 5.8: The relation between Genetic Algorithm Proposed Fitness Function Value and its Iterations	64

Figure 5.9: The relation between the crossover rate, mutation rate and the iteration number 65

Nomenclatures

Symbol	Nomenclatures
APFD	Average Percentage Of Fault Detected
CC	Condition Coverage
CFG	Control Flow Graph
CN	Cyclomatic Number
DFS	Depth First Search
F(x)	Fitness Function
FI	Fault Impact
GA	Genetic Algorithm
IDE	Integrated Development Environment
Max(s)	Maximum Severity Among All Test Cases
MCC	Multiple Condition Coverage
O	Order
RFT	Rate Of Fault
S	Severity Of Test Case
SC	Statement Coverage
SV	Severity Of Fault
TCW	Test Case Weight
V(G)	Cyclomatic Complexity