Name: Huipu Xu

Class : CS569

Email: xuhui@oregonstate.edu

Project Proposal

Adaptive Random Testing On TSTL

Introduction

As we know, random testing is a common and useful method on TSTL, In addition, it is not only a useful testing technique in itself, but also plays a core role in many other testing methods. However, Random testing is not a perfect testing method, which is inefficient that means random testing can not detect all of failures. Recently, Adaptive Random Testing (ART) was proposed as an effective alternative to random testing. This paper presents a synthesis of the most important research results related to ART. ART is an attempt to improves the errors/failures detection effectiveness of Random Testing (RT). ART is based on various empirical observations showing that many program faults result in failures in contiguous areas of the input domain, known as failure patterns.[1] The ART method, the Fixed Size Candidate Set ART algorithm (FSCS-ART)

(Chen et al., 2004), is proposed by Chen in 2004. “Essentially, to choose a new test case, k candidates are randomly generated. For each candidate ci, the closest previously executed test is located, and the distance di is determined. The candidate with the largest di is selected, and the other candidates are discarded. The process is repeated until the desired stopping criterion, be it the exhaustion of testing resources or the detection of enough failures, is reached.”[1]

In my proposal, I plan to apply this method, ART, into TSTL to instead of RT, since this method is better than depth First Search (DFS) and Breadth First Search (BFS), because it can find spot faults quickly in AVL tree, which method is to greatly decrease the testing time and the number of test cases to detect faults within the AVL tree.

Based on empirical observations that contiguous failure regions are common, adaptive random testing combines random candidate selection with a filtering process to encourage an even spread of test cases throughout the input domain. Experimental studies have shown that ART can detect failures using up to 50% fewer test cases than random testing.[1]

Plan

The basic plan of this final project is below like this:

First of all, from now to 4/29, I will collect some papers about RT, ART and some related works.

Then from 5/1-5/6, revise and rewrite basic ART in python and implement on TSTL.

Next step, from 5/9-5/31, continue to test the ART and make sure it works well on TSTL. The details: test the ART whether it can detect and find faults quickly, and how is its coverage, and the other specification.

Final step, from 5/31-6/6, Test everything in my project, then finish all of paper work..

References

[1] Chen, T.Y., Leung, H., Mak, I.K., 2004, Adaptive random testing, In: Proceedings of the Ninth Asian Computing Science Conference (ASIAN'04), Lecture Notes in Computer Science, Vol 3321, 320–329.

[2] Chen, T.Y., Leung, H., Mak, I.K., 2004, Adaptive random testing, In: Proceedings of the Ninth Asian Computing Science Conference (ASIAN'04), Lecture Notes in Computer Science, Vol 3321, 320–329.