Kernel Density Adaptive Random Testing

CS-569 Project Proposal

Muhmmad Bangash (932-676-646)

Background:

Random testing is not only a beneficial testing technique but also plays a core role in many other testing methods. Significant improvement to random testing has an impact throughout the software testing community. Newly Adaptive Random Testing (ART) was proposed as an active substitute to random testing. Mutation inquiry is used to measure the effectiveness of a test data generation technique and finding its faults. When a mutant is killed the decisions must be made whether to spread or strengthen the succeeding test inputs. Meanwhile diversification delivers an extensive range of test inputs with the purpose to increase the chances of killing new mutants. By differentiating or seeing it by intensification it selects test inputs which are similar to those previously shown to be successful and by taking the advantage of overlaps in the conditions in which mutants can be killed.

Introduction:

Technique for generating test data to kill mutants is based upon Random Testing (RT), which aims to reveal as many faults as possible by executing a wide range of random values from the input domain of the program under test [1]. Although this idea is simple, research and industry suggest RT can be effective and scalable [2], [3]. KD–ART, a new approach towards improving the effectiveness of ART by taking advantage of run-time test execution information. KD–ART has two stages: it starts by applying the traditional ART algorithm to generate test inputs that cover the input domain as evenly as possible, whilst collecting information about the run-time properties of these tests; then it applies Kernel Density Estimation (KDE) to generate new test inputs according to the distribution of tests that were found to have useful run-time properties. The intensification strategy favors new test inputs that are close to the tests previously shown to be useful. It is based on the intuition that effective regions of the input domain for finding faults tend to overlap and/or be in close proximity to each other. Kernel Density ART (KD–ART). KD–ART enhances the effectiveness of ART by taking advantage of additional white box information to guide the selection of test cases towards those that are likely to be more effective. This information may be gathered using feedback from the people doing the testing or derived from test adequacy metrics such as branch coverage and mutation score. KD–ART uses this information to diversify or intensify the search for additional test inputs.

Project Plan:

My project will demonstrate the trade-offs between diversification and intensification by augmenting Adaptive Random Testing (ART) to estimate the Kernel Density (KD–ART) of input values which are found to kill mutants. During the whole project my aim will be that to show the intensification is typically more effective at finding faults than diversification. KD–ART (intensify) achieves 7.24% higher mutation score on average than KD–ART (diversify). Additionally, KD–ART is computationally less expensive than ART. I will start it by surveying and then will look into how TSTL supports my idea and how it will be implemented using AVL tree. In the end code coverage and execution time will be compared with TSTL tester.

References:

[1]. P. S. Loo and W. K. Tsai. "Random testing revisited". Information and Software Technology, Vol.30. No 7, pp. 402-417, 1988.

[2]. T. Y. Chen, H. Leung, and I. K. Mak, "Adaptive random testing" in Proceedings of the 9th Asian Computing Science Conference, LNCS 3321, pp. 320-329, 2004.

[3]. T.Y. Chen, F.-C. Kuo, R.G. Merkel, and T.H. Tse, "Adaptive random testing: the ART of test case diversity," *Journal of Systems and Software*, vol. 83, no. 1, pp. 60–66, 2010.

[4]. J. H. Andrews, L. C. Briand, and Y. Labiche, "Is Mutation an Appropriate Tool for Testing Experiments?" in *Proceedings of the 27th International Conference on Software Engineering (ICSE'05)*, St Louis, Missouri, 15-21 May 2005, pp. 402 – 411.

[5]. A. J. Offutt, A. Lee, G. Rothermel, R. H. Untch, and C. Zapf, "An Experimental Determination of Sufficient Mutant Operators," *ACM Transactions on Software Engineering and Methodology*, vol. 5, no. 2, pp. 99–118, April 1996.