Analysing the Impact of Change on State Machine Test Sets

Sarah Salahuddin
Department of Computer Science,
University of Sheffield, UK.
s.salahuddin@sheffield.ac.uk

Kirill Bogdanov
Department of Computer Science
University of Sheffield, UK.
k.bogdanov@sheffield.ac.uk

Abstract

Software is continuously evolving and a part of the way to adapt to change is to be able to predict and assess the impact of change on test suites. Finite state machines (FSM) are one way to model systems. FSM-based testing methods have been used in protocol testing for years. In this short paper we describe the experiments conducted and their results used to analyse the impact of change on test sets and their components using the W method. The four types of changes considered are adding a transition, deleting a transition, re-labelling a transition and re-directing a transition.

1. Introduction

On the whole, a change to a system leads to a new system; a test suite needs to evolve too. Many of the regression testing approaches focus on testing of new and modified portions of a program, assuming that re-running the whole test suite will be prohibitively expensive. This is typically accomplished by analysing dependencies between new/modified parts of a system and the original parts, in order to identify the scope for testing and subsequently choose the appropriate tests from the original test suite and add new tests for the functionality which cannot be tested reusing the tests [1-2]. The newly added tests and what remains of the original test suite is then expected to form a new test suite, which could potentially be used to completely retest the new version of a system.

FSM and EFSM-based testing aims to demonstrate conformance of an implementation to a model represented in an FSM or EFSM form. X-machines are EFSMs, where each label on the transition is associated with a transition function. This makes it possible to perform EFSM-based testing by separate testing of a transition diagram and the behaviour of labels [3]. The idea of separate testing makes it possible to test more complex systems because transition diagrams tend not to contain large numbers of states and testing of labels does not have to involve exploration of the whole state space of the system which can be helpful when changes are made to the model. Testing of labels makes it easier to track changes and the impact of those changes in generating test sets.

The focus of the current work is to analyse whether one can reuse the existing test data for parts of a test suite and identify how to build test suites so as to make them tolerant to changes. To analyse how test suite parts are affected by changes, a number of pseudo-random state machines (with specific properties) are built and modified in four of ways. Test suite parts such as the characterization set *(W)* and the state cover *(C)* (discussed in section 2) are generated for original and modified machines and we identify the likelihood that test suite parts have to be modified when the machines they are built from are modified.

The main contributions of this work is:

i. Development of an experimental framework to allow FSM generation and experimentation;
ii. Investigation and identification of characteristics of small FSM models (such as singleton W sets);
iii. The investigation and identification of analysing change impact on test sets of FSM models using the test set components;
iv. Demonstration of the accuracy of the change impact analysis supported by experiments conducted and the corresponding theoretical formulae.

2. FSM Generation and Experiments

A number of authors [4-5] have used random FSM generation to evaluate the effectiveness of FSM-based testing methods however, there are few details of the FSMs used.

Parameters for FSM generation are the number of states, number of labels and the degree of completeness. The degree of completeness refers to the ratio of transitions to be generated to the maximum number of transitions possible from each state. Therefore, a lower number represents fewer transitions while a higher number represents a machine with more transitions and 1 corresponds to a complete machine where each state has a transition for each label.

Papers [6-7] provide details of the machines used for experiments and the parameters for generation are similar to those identified for these machines.

An FSM Generation framework has been developed for the generation of machines to be used in experiments to analyse the impact of change. The framework generates different types of FSMs which can be categorised into three types, balanced state machines, machines with sun-like states and random machines. Balanced machines contain the same number of incoming/outgoing transitions for every state. Sun-like states are states with several times the in/out degree of the ordinary states. The details of these can be found in [8].

The four types of changes used in the experiments are adding a transition with an existing input, removing a transition, re-labelling a transition with an existing input and redirecting a transition. After a change was made to the original machine the test sets for the changed machines were generated and then compared to the test sets for the original machines.

The impact on test set has been analysed in terms of the number of sequences added or removed in case of a particular change.

The change between the test sets components (characterisation set and state cover) has been analysed in terms of the maximum number of cases in which the C or the W set would be required for a machine to be changed. The resistance of larger W sets (containing more sequences) to change was also analysed using two implementations of the W method where the optimal W was a smaller W set.

The results obtained by the experiments have been verified against the theoretical formulae. The experiments have also shown that there is a lesser change to the contents of the test set if original W and C Sets are applicable and are used. This has a greater significance since it makes the test set less brittle if the original C and W are applied and used.

3. Results

In the experiments performed, the W-set changes more frequently than the C set except for the re-direction changes. The pattern followed by the changes needed for both implementations of the W-set are similar but the values vary, for FSMs with 20 transitions 6% change is needed with the optimal W and 4% change is needed with a bigger W set for the change involving adding a transition. A marked increase is seen for machines with a higher degree of completeness (more transitions) since with increased number of transitions the likelihood that a sequence distinguishes a pair of states decreases.

The number of times C set had to be modified for each type of change have shown conformance to the predicted values.

4. Conclusions and Future Work

The results of the experiments conducted show that it is useful to analyse the impact of change on separate parts of the test set (C and W set). If these components can be applied to a changed machine then the test set can be built using the original components and far less test data will have to be generated anew.

The results also show that W sets with more sequences than needed are more resistant to change than the shorter W sets. Another characteristic of the W sets which has been investigated is the singleton elements of W sets, which is more representative of the software examined and has not been mentioned in FSM testing literature as the literature primarily focuses on protocol testing.

5. References

1. Leung, H.K., and White,L., *Insights into testing and regression testing global variables.* Journal of Software Maintenance: Research and Practice, 1990. **2**(4): p. 209-22
2. Xie, B., *Requirement-Based Regression Test Suite Reduction Using Dependence Analysis*. 2005, University of Ottawa, Ottawa, Canada.
3. Holcombe, M., and Ipate,F., *Correct systems: building a business process solution*. 1998: Springer-Verlag. .
4. Petrenko, A., and Bochman,G., *On fault coverage of tests for finite state specifications.* Computer Networks and ISDN Systems, 1996. **29**(1): p. 81-106.
5. Sidhu, T.P., and Leung,T.K., *Formal Methods for Protocol Testing: A detailed study.* IEEE TSE, 1989. **15**(4): p. 413-426.
6. Dorofeeva, R., Yevtushenko,N., El-Fakih,K., and Cavalli,A.R. *Experimental Evaluation of FSM-Based Testing Methods* in *Proceedings of the Third IEEE international Conference on SEFM* 2005. Washington: IEEE Computer Society.
7. Simao, A., Petrenko,A., and Maldonado,J.C. *Experimental Evaluation of Coverage Criteria for FSM-based Testing*. in *Proceedings of XXI SBES 2007*. Joao Pessoa, Brazil.
8. Salahuddin, S., Bogdanov,K., and Walkinshaw, N. *Balanced FSM Generation for Empirical Studies* in *TESTBENCH'08 co-located with ICST 2008*. 2008. Lillehammer, Norway.