Journal of Graphic Era University
Vol. 4, Issue 1, 17-21, 2016
ISSN: 0975-1416 (Print)

Strategies used for State Based Software Testing: A Review

Sanjay Maiduli

Department of Computer Science & Engineering
Uttarakhand Technical University (Main Campus)
Dehradun, 248007, India
Email: smaiduli@gmail.com

Abstract
Testing is one of the most important means to validate the correctness systems, a state based testing is based on state machine, a state based testing approach allows for reuse of the test cases designed for the base program and is therefore suitable for automatic generation of test cases (Binder, 2000). This paper is the review of strategies for state based software testing such as Round Trip Path (RTPT) tests and Sneak Path testing and comparison between two strategies used.

Keywords: Software Testing, Test case, State Based Testing, Round Trip Path Testing (RTPT), Sneak Path Testing (SPT)

1. Introduction

A State Based Software Testing strategy is based on state machines there are several state machine used such as finite state machine, Mealy machine, Moore machine, state chart etc. In a state machine the test cases are executed or simulated with some sequence of event before starting of the actual phase (Khalil et al, 2010).

A state based sequence testing provides a model of testing and captures the dynamic behavior of the system. It has five major components namely State, Event, Action, Transition, and Guard. A system may have several final states. State event and transition are given and usually legal event are associated with transitions (Pressman, 2009).

This paper focuses on the two main methods used for state based testing. Section 2 outlines the components of state based testing. Section 3 outlines the first method Round Trip Path Testing (RTPT). Section 4 outlines the second method Sneak Path Testing (SPT). Section 5 outlines the comparison between the two methods used. Section 6 concludes the paper and finally last section is the outline of references

2. Components of State Based Testing

A state based software testing has five major Components

A State

A state is an abstract situation in the life cycle of a system.

Journal of Graphic Era University
Vol. 4, Issue 1, 17-21, 2016
ISSN: 0975-1416 (Print)

An Event
Sequence of input like method and call.

An Action
Output that follows the event

A Transition
Change in one state to another state when an event is executed.

A Guard Condition
Guarding condition linked with an event which is used to restrict a transition to fire.

3. Round Trip Path Testing (RTPT)

Round Trip Path testing (RTPT) is state based software testing which is use for comparing the prototype model to the actual software execution. An important aspect of RTPT strategy is Mutation Analysis in which a known fault is seeded in the correct system and analysis is done on the basis of mutation score and a system's fault detection effectiveness is measured (Briand et al, 2004).

RTP builds a tree with some initial input from state transition diagram which include all the round trip paths. Figure 1 shows a typical round trip tree.

The main goal of round trip tree is to traverse the graph representing the state machine and generate a transition tree which contain state event and transition, it try to exercise round trip path i.e. path that starts and ends in the same state without any other repeating state (Marciniak, 1994).

This strategy is effective but still suffers with particular faults remain undetected so by using RTPT with combination of other blackbox testing can eliminate undetected faults, so the hybrid approach is helpful to improve the overall fault detection rates.

4. Sneak Path Testing (SPT)

The main drawback of RTPT is certain type of fault remains undetected so Sneak Path Testing is used to eliminate this drawback; the main aim of SPT testing is to test the capability of designed prototype under unspecified condition.

In Sneak Path Testing strategy a sneak path message accepted when it should not accepted can occurs if there is unspecified transition, the transition in SPT occurs even if guard predicate is false (Millar, 2009).

Journal of Graphic Era University
Vol. 4, Issue 1, 17-21, 2016
ISSN: 0975-1416 (Print)

Sneak Path testing also build a sneak path tree similar to RTP round trip tree for determining the unspecified path. Figure 2 shows a typical sneak path tree

Sneak Path Testing (SPT) strategy is important for the testing point of view where unexpected behavior of the system in tested which is difficult by other common state based testing strategies (Marciniak, 1994).

5. Comparison between RTP and SPT Strategies

State based testing (SBT) is focused on comparing the system under test to a test model, in State based testing the test is derived from the finite state machine. It predicts the expected and unexpected behavior of the system. So when a series of events is executed the transitions are checked, if they reached the final state then the system is validated.

RTP and SPT are widely used testing strategies, the SPT testing in the context of state machine based system in terms of test suite size is inexpensive as compared to the RTP however in terms of preparation time RTP beats SPT it takes less preparation time as compare to SPT (Binder, 2000).

Final point of comparison is that these strategies are complementary to each other. They are used with different purpose. Sneak Path Testing can catch different types of faults. SPT catches more fault than Round Trip Path. So sneak path testing is always performed in state based testing while round trip path testing is depend on the requirement of system.

6. Conclusion

This paper's objective is to introduce the two commonly used strategies used for State Based Testing (SBT) and comparison between RTPT and SPT testing strategies which are based on state machine, round trip tree and sneak path tree. They both generate a test suite from the test model automatically which helps reducing test cost.

Journal of Graphic Era University
Vol. 4, Issue 1, 17-21, 2016
ISSN: 0975-1416 (Print)

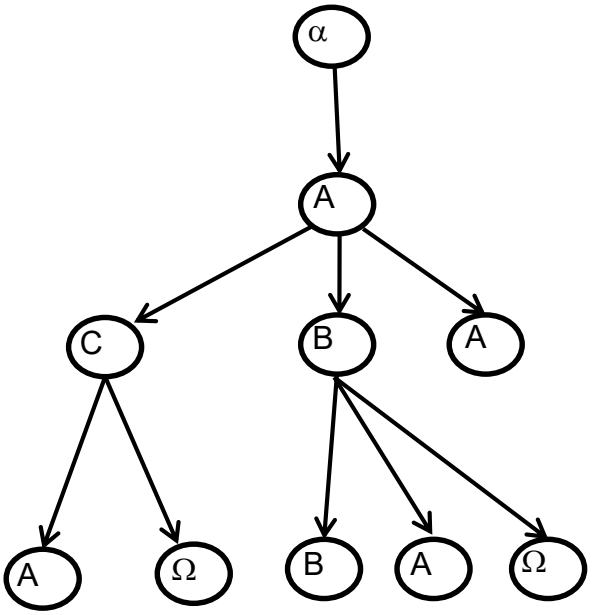

Figure 1. A sample tree of RTPT from α to Ω

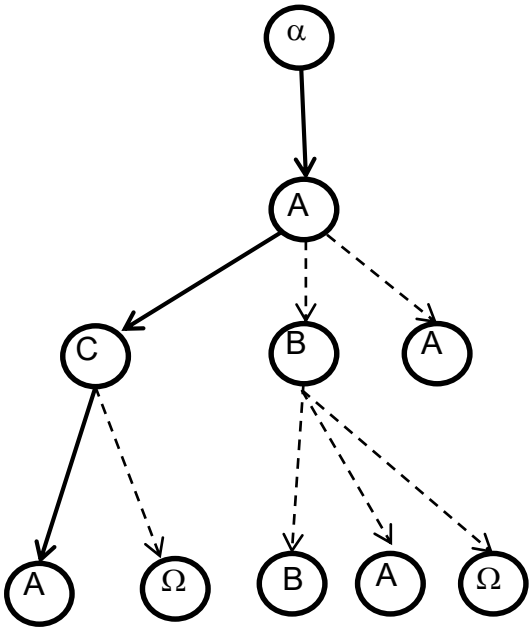

Figure 2. A Sneak path tree from α to Ω

References

Binder, R. (2000). *Testing object-oriented systems: models, patterns, and tools*. Addison-Wesley Professional.

Khalil, M., & Labiche, Y. (2010, November). On the round trip path testing strategy. In *2010 IEEE 21st International Symposium on Software Reliability Engineering* (pp. 388-397). IEEE.

Briand, L. C., Di Penta, M., & Labiche, Y. (2004). Assessing and improving state-based class testing: A series of experiments. *IEEE Transactions on Software Engineering*, *30*(11), 770-783.

Miller, E. F. (2009). Introduction to software testing technology. *Tutorial: Software Testing & Validation Techniques, Second Edition, IEEE Catalog No. EHO*, 180-0, pp. 4-16.

Marciniak, J. J. (1994). *Encyclopedia of software engineering (vol. 2 OZ)*. Wiley-Interscience.

Pressman, Roger S. (2009). *Software engineering: a practitioner's approach* (Fifth edition), McGraw-Hill Higher Education