International Journal of Information Technology Project Management
Volume 10 • Issue 1 • January-March 2019

Coverage Criteria for State-Based Testing: A Systematic Review

Sonali Pradhan, Siksha 'O' Anusandhan University, Bhubaneswar, India
Mitrabinda Ray, Siksha 'O' Anusandhan University, Bhubaneswar, India
Srikanta Patnaik, Siksha 'O' Anusandhan University, Bhubaneswar, India

ABSTRACT

State-based testing (SBT) is known as deriving test cases from state machines and examining the dynamic behaviour of the system. It helps to identify various types of state-based faults within a system under test (SUT). For SBT, test cases are generated from state chart diagrams based on various coverage criteria such as All Transition, Round Trip Path, All Transition Pair, All Transition Pair with length 2, All Transition Pair with length 3, All Transition Pair of length 4 and Full Predicate. This article discuses a number of coverage criteria at the design level to find out various types of state-based faults in SBT. First, the intermediate graph is generated from a state chart diagram using an XML parser. The graph is traversed based on the given coverage criteria to generate a sequence of test cases. Then, mutation testing and sneak-path testing are applied on the generated test cases to check the effectiveness of the generated test suite. These two are common methods for checking the effectiveness of test cases. Mutation testing helps in the number of seeded errors covered whereas sneak-path testing basically helps to examine the unspecified behavior of the system. In round trip path (RTP), it is not possible to cover all paths. All transition is not an adequate level of fault detection with more execution time compared to all transition pair (ATP) with length 4 (LN4). In the discussion, ATP with LN4 is the best among all coverage criteria. SBT can able to detect various state-based faults-incorrect transition, missing transition, missing or incorrect event, missing or incorrect action, extra missing or corrupt state, which are difficult to detect in code-based testing. Most of these state-based faults can be avoided, if the testing is conducted at the early phase of design.

KEYWORDS

All Transition Pair, All Transitions, Mutation Testing, Round Trip Path, Sneak-Path Testing, State-Based Faults, State-Based Testing

INTRODUCTION

Testing at the early phase of software development life cycle can able to find the ambiguities and inconsistencies in the design and hence, design should be enhanced before the program is written (Antonio et al., 2002; Sundararajan et al., 2017). Research is going on state-based testing (SBT) to find effective test cases and to minimize cost of the test suite (Agrawal et al., 1989; Holt et al., 2014).

DOI: 10.4018/IJITPM.2019010101

Copyright © 2019, IGI Global. Copying or distributing in print or electronic forms without written permission of IGI Global is prohibited.

For this, Unified Modeling Language (UML) diagrams are used to generate test cases. Initially testers were going for traditional testing, which is also known as code coverage testing. But, state-based coverage cannot be achieved in code-based testing (Binder, 2000). To achieve this, tester generates test scenarios from the state chart diagrams and then test cases are generated from these scenarios. Test cases are generated at design level and coverage analysis is performed from the source code. The diagrams are generated at the design stage of development life cycle. Generating test cases based on UML diagrams come under Model Based Testing (MBT). It is a better testing approach than code-based testing as it detects the error at the early phase which requires less cost to fix it (Chen & Wang, 2014; Dias Neto et al., 2007). MBT are conducted for the following reasons:

- To get an abstract model of the system;
- Validate the model;
- Generate and execute test cases;
- Assigning pass/fail verdict;
- Analyzing the execution result;
- When it is not required to model the full system;
- To prevent fault;
- To reduce cost with updating test cases.

Early testing activities make early fault detection (Binder, 2000; Broy et al., 2005) and more and more articles are referred to as MBT using state-based testing. In a very recent article, we find in MBT, where it elaborates several findings from MBT users in industry, security testing and various MBT challenges (Utting et al., 2016). Utting and Legeard (Briand & Labiche, 2001) have proposed a SBT technique to design black box testing. State-based testing is primarily considered as a black box testing to generate test cases (Briand & Labiche, 2001).

A test case is a document, which has a set of test data, expected results with preconditions and post conditions. Test case is a particular test scenario in order to verify action against a specific requirement. There are different types of software faults that can be found in different ways (y Hernández & Marsden, 2017). A set of test cases is called a test suite. To examine the effectiveness of the test suit, tester goes for the mutation testing and sneak-path testing. Mutation testing technique is applied on the generated test suite to measure its efficiency. In mutation testing, a faulty version of a software system is generated by introducing some mutant in the software, which is known as mutant operators (Chen & Wang, 2014). Some mutants still undetected after conformance testing. So, another way of testing where remaining mutants are killed is known as sneak-path testing (Chen & Wang, 2014; Utting et al., 2016). Conformance testing is seeking to test specified behavior of the model and also it is equally important to test unspecified behavior of the model through sneak-path testing (Chen & Wang, 2014). Complete testing is not cost-effective as it takes longer time to investigate the software. The complete software testing is also known as exhaustive testing (Ostrand & Balcer, 1988). Therefore, testing must be performed on selected subsets, but not the whole domain of testing, which makes effective software testing. In a very recent approach, a non-deterministic action system model is used for generating a number of mutants (Aichernig, 2016). In the paper (Aichernig, 2016), original model is executed and tested for conformance after the mutants are generated, so that conformance check is done efficiently, and the complete testing is not possible. Test cases help us in retrieval of important information. Different types of tests are more effective for different classes in the model (Jacky et al., 2007). Enabling graphical interfaces MBT is a novel technique for graphical user interfaces (GUI) testing (Belli & Beyazıt, 2017; Reed & Angolia, 2018). In this survey paper, first, we discuss the SBT by generating test cases based on various types of state-based coverage criteria. Next, mutation testing and sneak-path testing are discussed and applied on the generated test cases to measure the efficiency.

18 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage: www.igi-global.com/article/coverage-criteria-for-state-based-testing/215011

Related Content

Stakeholder Challenges in Information Systems Project Offshoring: Client and Vendor Perspectives
Peter Haried (2013). *Perspectives and Techniques for Improving Information Technology Project Management (pp. 96-112).*
www.irma-international.org/chapter/stakeholder-challenges-information-systems-project/73230

E Bario Project: In Search of a Methodology to Provide Access to Information Communications Technologies for Rural Communities in Malaysia
Poline Bala, Roger W. Harrisand Peter Songan (2008). *Information Communication Technologies: Concepts, Methodologies, Tools, and Applications (pp. 826-839).*
www.irma-international.org/chapter/bario-project-search-methodology-provide/22703

Strategic Planning for Information Resources: The Evolution of Concepts and Practice
William R. King (1988). *Information Resources Management Journal (pp. 1-9).*
www.irma-international.org/article/strategic-planning-information-resources/50903

Design and Implementation of a Wide Area Network
Rohit Rampal (2002). *Annals of Cases on Information Technology: Volume 4 (pp. 427-439).*
www.irma-international.org/article/design-implementation-wide-area-network/44522

Neural Data Mining System for Trust-Based Evaluation in Smart Organizations
T. T. Wong (2008). *Information Communication Technologies: Concepts, Methodologies, Tools, and Applications (pp. 2950-2967).*
www.irma-international.org/chapter/neural-data-mining-system-trust/22856