Information Sciences 370–371 (2016) 63–78

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

A heuristic transition executability analysis method for generating EFSM-specified protocol test sequences

Ting Shu[a,*], Zuohua Ding[a], Meihwa Chen[b], Jinsong Xia[a]

[a] School of Information Science and Technology, Zhejiang Sci-Tech University, Hangzhou 310018, China
[b] Computer Science Department, University at Albany-State University of New York, Albany, NY 12222, USA

ARTICLE INFO

Article history:
Received 3 May 2016
Revised 12 July 2016
Accepted 25 July 2016
Available online 26 July 2016

Keywords:
EFSM
Conformance testing
Executable test sequences
Test generation

ABSTRACT

Automatic executable test sequence generation is a challenging task for protocol test based on the Extended Finite State Machine (EFSM) model. Transition executability analysis (TEA) can guarantee the executability of generated test sequences. However, as a reachability analysis technique, it often suffers from the state explosion problem. To remedy the situation, this paper proposes a heuristic transition executability analysis (HTEA) method for compressing the explored state space during test sequences derivation. In our approach, a transition feasibility guidance matrix is first constructed through extracting the inherent constraint relationship between each adjacent transitions in the EFSM under testing. Then, relying on the built matrix and related runtime feedback, we define an objective function to dynamically evaluate state nodes in the HTEA expanding tree. Finally, the proposed objective function is utilized to heuristically guide the state space traversal in the HTEA tree for executable test sequences generation. Consequently, the executable test generation problem can be reformulated into a multi-objective search problem based on a HTEA tree. Experimental results impressively show the potential of the novel method for avoiding the state explosion during test generation, compared with the classic TEA technique in the breath-first way.

1. Introduction

Undetected minor faults in the protocol implementation could lead to disastrous consequences. To guarantee the communication quality, protocol conformance testing plays an integral role in the protocol development lifecycle. It is an important experimental activity to validate if the protocol implementation under test (IUT) is consistent with its specification or not. In the protocol conformance testing, test sequences that can be described as a series of input/output pairs are usually generated and stimulated to observe the external behavior of the IUT under test. Manual test sequence generation is expensive and error-prone [7]. Thus, automating the generation of test sequences from model-based specifications has become a hot issue for both industry and academic research, in the field of protocol testing.

Finite State Machine (FSM) and Extended Finite State Machine (EFSM) have been widely used to formally describe the protocol specifications under test [25,28]. A protocol specification usually comprises of a control portion and a data portion. The FSM is only suitable for modeling the control portion. Since the EFSM model extends the FSM model by introducing

[*] Corresponding author.
E-mail addresses: shuting@zstu.edu.cn (T. Shu), zouhuading@hotmail.com (Z. Ding), mchen@albany.edu (M. Chen), js_xia@126.com (J. Xia).

http://dx.doi.org/10.1016/j.ins.2016.07.059
0020-0255/© 2016 Elsevier Inc. All rights reserved.

context variables and parameters, it is enhanced to represent both control and data portions [13,24]. In the past decade, substantial test generation methods based on FSMs have been proposed such as UIO method [14], D method [17] and W method [26]. Unfortunately, when testing from EFSM, these developed approaches are unable to be directly utilized. The main obstacle is that some generated test sequences may be infeasible, due to the existence of variables and associated predicate guards in transitions. That is to say, in terms of certain context variables, condition conflicts between transitions limit the executablility of the generated test sequences. However, automatic feasible test sequence derivation for an EFSM is NP-hard and remains an open research problem [13].

Although transition executability analysis (TEA) technique [18] provides a promising solution to overcome the infeasibility of generated test sequences, it often suffers from the state explosion problem. The reason stems from the fact that the TEA is one of the reachability methods that generate test sequences by exploring a TEA tree's reachable state space. However, the state space of an expanding TEA tree for a specific IUT is usually too huge to be traversed exhaustively. Thus, the classic TEA by means of systematic state space search, such as the breath-first way, may lead to state explosion. To partially alleviate this problem, a pruning strategy for TEA tree is introduced to limit the search space to some extent. But the remaining search space is still large. In addition, in our previous works, we found that it brings about a new potential thread for test generation. That is, corresponding to some specific test coverage criteria, the application of an unsuitable pruning strategy may result in the failure of test generation.

In this paper, we present a heuristic transition executability analysis (HTEA) method for generating feasible test sequences from a class of EFSMs, which are normalized, fully executable and untrapped. These specific constraints of EFSMs are prerequisites for application of TEA-based methods [18]. The purpose of our proposed method is to significantly compress the explored state space for feasible test sequences derivation and to eventually avoid the state explosion problem. In addition to the reduction of explored state space, minimizing the length of a generated test sequence is another objective. It is obvious that exploring the state space in breath-first way tends to find the shortest path satisfying a specific test criterion. Therefore, the proposed method makes a trade-off between the two aims.

In the HTEA approach, we first construct a transition feasibility guidance matrix by extracting the inherent relationship between adjacent transitions from EFSM under test. Then, based on the constructed matrix and the run-time information towards the test coverage criterion, an objective function f_{obj} is defined and computed to heuristically guide the state space traversal for test generation. For a state s_i in the TEA tree, the value of $f_{obj}(s_i)$ represents an approximate estimation of the probability that there exists an optimal feasible path from s_i to the objective transitions in the current coverage set. During test generation, each time a state node with the maximal function value is singled out from the explored nodes as the new root node for continuing to traverse the state space in the TEA tree. As a result, the feasible test generation problem is reformulated into a multi-objective optimal search problem based on a TEA expanding tree.

To evaluate the effectiveness and efficiency of the HTEA, we conduct a detailed case study on four popular protocol EFSM models: INRES[18], Network Monitor [19], SCP [8] and Class II transport protocol [29]. Experimental results indicate that the HTEA significantly outperforms the traditional TEA method based on breath-first search in terms of search state space. The statistical data impressively show our method's potential in avoiding the state explosion problem during test generation.

The remainder of this paper is organized as follows. Section 2 briefly discusses related work. In Section 3, some preliminaries are provided, including an overview of TEA-based methods for test generation. The proposed approach to generating feasible test sequence is presented in Section 4. Section 5 provides experimental results. In Section 6, we draw concluding remarks.

2. Related work

In terms of EFSM-based test generation, many approaches were proposed in the literatures [2,4,5,32,34]. Hierons et al.[16] introduced a two-phase method to overcome the infeasible path problem for a class of EFSM models, on whose structure some restrictions were placed. The method is required to transform the EFSM into a new model called EEFSM without containing any infeasible path. Based on the expanded EFSM, feasible test sequences may then be generated. Uyar and Duale [13,30] attempted to enable the automatic test sequences generation for EFSMs where all operation and guards are linear. Algorithms were represented to detect and eliminate all conflicts among transitions from EFSM model. Therefore, all paths in the resulting EFSM are feasible and FSM-based methods can be directly employed to derive the feasible test sequences. However, the severe assumptions imposed on the considered EFSM limits its general application.

Similarly, some researchers also apply FSM-based testing approaches for testing from EFSM by expanding an EFSM to a FSM [11,23,27]. Unfortunately, model transformation often comes with the state explosion. The main reason is that the size of resultant state space is exponential increasing with the number of variables in EFSM model. Furthermore, such transformations may produce semantic losses, which will complicate test sequences generation [15,21].

Chanson et al. [9] tackled the test sequence executability problem by using constraint satisfaction problem method. When testing from EFSMs containing the influencing self-loops, transition self-loop analysis technique can be utilized to make the generated path feasible. Koh et al. [22] and Bourhfir et al. [6] also produced executable test sequences based on control flow and data flow criterion for an EFSM-specified protocol using the similar technique. Zhang et al. [37] investigated a test generation method and developed a toolkit using the combination of symbolic execution and constraint solving. It was used to deal with the infeasible path problem for testing an SDL specification [31]. But this class of methods is not always applicable for general EFSM models.