Testing Against Non-deterministic FSMs: A Probabilistic Approach for Test Suite Minimization

Natalia Kushik[1(✉)], Nina Yevtushenko[2,3], and Jorge López[4]

[1] Télécom SudParis, Institut Polytechnique de Paris, Palaiseau, France
natalia.kushik@telecom-sudparis.eu
[2] Ivannikov Institute for System Programming of the Russian Academy of Sciences, Moscow, Russia
evtushenko@ispras.ru
[3] Higher School of Economics, Moscow, Russia
[4] Airbus Defence and Space, Issy-Les-Moulineaux, France
jorge.lopez-c@airbus.com

Abstract. The paper is devoted to model based testing against non-deterministic specifications. Such test derivation strategies are well developed, for example against non-deterministic Finite State Machines, however the length of the corresponding test suite can be exponential w.r.t. the number of specification states. We therefore discuss how a test suite can be minimized or reduced when certain level of guarantee concerning its fault coverage is still preserved. The main idea behind the approach is to augment the specification by assigning probabilities for the non-deterministic transitions and later on evaluate the probability of each test sequence to detect the relevant faulty implementation. Given a probability P which is user-defined, we propose an approach for minimizing a given exhaustive test suite TS such that, it stays exhaustive with the probability no less than P.

Keywords: Model based testing · Non-deterministic finite state machines · Guaranteed fault coverage · Probabilistic approach

1 Introduction

Model based testing has been actively developing in the past decades; the interested reader can find various recent works, in particular, when checking the proceedings of related conferences such as the International Conference on Testing Software and Systems (ICTSS), the International Symposium on Software Testing and Analysis (ISSTA), the Workshop on Model-Based Testing (MBT), the International Conference on Software Testing, Verification and Validation (ICST), etc. Finite State Machine (FSM) based testing assumes that the specification of the System Under Test (SUT) and its implementations are given

The work was partially supported by Erasmus program.

Published by Springer Nature Switzerland AG 2022
D. Clark et al. (Eds.): ICTSS 2021, LNCS 13045, pp. 55–61, 2022.
https://doi.org/10.1007/978-3-031-04673-5_4

as FSMs and usually the possible implementations share the same input/output alphabets with this specification FSM. In this paper, we study non-deterministic FSMs as related specifications. We note that various (preset and adaptive) testing strategies have been previously proposed for such machines, considering not only the test suite derivation but also learning the specification, test suite minimization and complexity estimation for the aforementioned tasks (see for example, [2,4,5,10]).

In this paper, we consider a white box testing approach, where all the possible faulty implementations are explicitly enumerated [4,7]. A complete test suite is built in such a way that each faulty implementation is *killed* (detected) by some test case of the test suite (test suite exhaustiveness). Such a test suite can be derived, for example, via adding to the test suite each sequence that distinguishes a potential faulty implementation from the specification machine. However, when the conformance relation is represented by non-separability[1], the length of a separating sequence can be exponential (w.r.t. the number of the specification states) [8], and this makes the approach unpractical, even if the fault coverage can be guaranteed. We propose to preserve the fault coverage up to a given level of certainty through augmenting the specification FSM with probabilities. Indeed, whenever for a given input at a given state two or more outputs are possible, these outputs can appear with certain probability. Note that in this paper, we do not discuss how such probabilities are assigned or obtained; they can be provided due to some additional knowledge of an SUT, or its stochastic behavior which can be revealed, for example, during the system monitoring. We only assume that the augmentation of the specification with probabilities is possible.

Once the specification FSM is augmented with probabilities for each non-deterministic transition, a given complete test suite can be *filtered*, i.e., the sequences that are derived for detecting some faulty implementations can be deleted depending on the likelihood of being detected by other test sequences. The level of such likelihood is determined by a user defined probability P. We propose a method for calculating the related likelihood and also discuss how a given exhaustive test suite can be minimized in such a way that it stays exhaustive at least with probability P. Note that we are not aware of any works for test suite minimization with guaranteed fault coverage against probabilistic non-deterministic FSMs and this is thus, the first attempt.

2 Preliminaries

When testing against FSMs, guaranteed fault coverage can be achieved when a corresponding fault model is properly defined. A *fault model* [6] is a triple $\langle \mathcal{S}, @, FD \rangle$ where $\mathcal{S}$ is the specification of the system behavior, @ represents the conformance relation between an implementation $\mathcal{I}_j$ under test and the specification $\mathcal{S}$, while FD is a fault domain which limits the set of possible implementations, i.e., $\mathcal{I}_j \in FD$. We are interested in an *exhaustive* test suite, i.e., a test suite

[1] There exists an input sequence such that output responses of the specification and an implementation to this sequence do not intersect.

that detects each implementation $\mathcal{I}_j \in FD$ that is not conforming to $\mathcal{S}$ ($\mathcal{I}_j \not@ \mathcal{S}$). Moreover, we work under the white box testing methodology, which means that the implementations from FD are explicitly enumerated, $FD = \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_k\}$. Usually, each $\mathcal{I}_j \in FD$ corresponds to a potential faulty implementation and thus, represents a *mutant* (transfer and/or output) of the specification $\mathcal{S}$. In this work, @ is the non-separability relation ($\cong$) which we further adjust to *probabilistic* non-separability while the specification is represented by an initialized complete non-deterministic observable FSM $\mathcal{S}$.

An *FSM* is a 5-tuple $\mathcal{S} = \langle S, I, O, h_S, s_0 \rangle$ where S is a finite nonempty set of states with the designated initial state $s_0 \in S$, I and O are finite input and output alphabets, and $h_S \subseteq S \times I \times O \times S$ is a *transition relation*. The FSM $\mathcal{S}$ is *non-deterministic* if for some pair $(s, i) \in S \times I$, there exist several pairs $(o, s') \in O \times S$ such that $(s, i, o, s') \in h_S$; otherwise, the FSM is *deterministic*. The FSM $\mathcal{S}$ is *observable* if for every two transitions (s, i, o, s_1), $(s, i, o, s_2) \in h_S$ it holds that $s_1 = s_2$; otherwise, the FSM is *non-observable*. The FSM $\mathcal{S}$ is *complete* if for every pair $(s, i) \in S \times I$, there exists a transition $(s, i, o, s') \in h_S$; otherwise, the FSM is *partial* (partially specified).

Let each $\mathcal{I}_j \in FD$ and $\mathcal{S}$ share the same input alphabet I. We say that $\mathcal{I}_j \not@ \mathcal{S}$ if there exists a *separating* sequence $\alpha \in I^*$ for $\mathcal{I}_j$ and $\mathcal{S}$, i.e., the set of output reactions of $\mathcal{I}_j$ and $\mathcal{S}$ to α do not intersect, i.e., $out(\mathcal{I}_j, \alpha) \cap out(\mathcal{S}, \alpha) = \emptyset$, where $out(\mathcal{I}_j, \alpha)$ (resp. $out(\mathcal{S}, \alpha)$) is the set of output responses on α at the initial state of the FSM $\mathcal{I}_j$ (resp. FSM $\mathcal{S}$). Otherwise, $\mathcal{I}_j$ is non-separable from the specification machine $\mathcal{S}$. Note that TS is an exhaustive test suite w.r.t. the fault model $\langle \mathcal{S}, \cong, FD = \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_k\} \rangle$, if for each $\mathcal{I}_j \in FD$, there always exists such $\alpha_j \in TS$ that separates $\mathcal{I}_j$ from $\mathcal{S}$.

Deriving such complete test suite TS is possible and this problem has been well studied previously (see, for example [4,10]). However, the length of the corresponding separating sequence (even for a given mutant) can be exponential w.r.t. the number of states of the specification FSM $\mathcal{S}$. Therefore, an iterative test suite derivation even for the white box testing approach can return a test suite of exponential length. Correspondingly, in this paper, we discuss how such test suite (length) can be reduced via introducing probabilities to the specification FSM and a given level of certainty about the TS exhaustiveness.

3 Introducing the Probabilities in the Specification

Given the specification machine $\mathcal{S} = \langle S, I, O, h_S, s_0 \rangle$, we augment each non-deterministic transition $(s, i, o, s') \in h_S$ with the probability p. The probabilistic specification is thus the FSM $\mathcal{S} = \langle S, I, O, h_S, s_0, pr \rangle$, where pr is the function that defines the probability for the output o to be produced at state s under input i, $pr : S \times I \times O \longrightarrow [0, 1]$. Note that, we restrict the assignation of pr in such a way that $\forall s \in S \ \forall i \in I \sum_{o \in O} pr(s, i, o) = 1$. The function pr can be extended over input/output sequences from $(IO)^*$; given an input/output sequence $\alpha/\beta = (\alpha'/\beta').(i/o)$, $pr(s_0, \alpha, \beta) = pr(s_0, \alpha', \beta') * pr(s, i, o)$, where s is the α'/β'-successor of the state s_0 of the specification FSM $\mathcal{S}$; if the trace α'/β'

is not defined at state s_0 then this probability equals 0. Note also, that for the defined sequence γ such a successor is unique due to the observability of the specification FSM $\mathcal{S}$. Note as well, that as usual $pr(s, \varepsilon, \varepsilon) = 1$.

As an example, consider the FSM in Fig. 1 where transitions at states 1 and 2 are non-deterministic and augmented with probabilities.

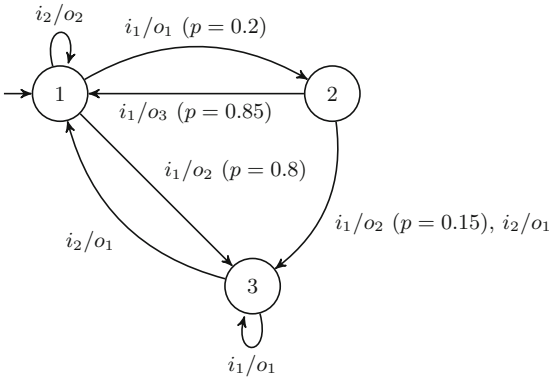

Fig. 1. An example probabilistic FSM $\mathcal{S}$

The notion of a probabilistic FSM has been introduced before, as well as the notion of distinguishability (as non-equivalence) for such machines (see for example, [1,3,9]). However, in this work, we consider the non-separability conformance relation that we adjust, having such an augmented probabilistic specification FSM $\mathcal{S}$.

For a fault model $\langle \mathcal{S}, \cong, \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_k\}\rangle$ we thus define a *probabilistic separability* for a given implementation $\mathcal{I}_j$ from the specification $\mathcal{S}$. Given P as a user defined probability[2], a sequence $\alpha \in I^*$ is a *P-probably separating* sequence for $\mathcal{I}_j$ and $\mathcal{S}$, if $\sum_{\beta \in out(\mathcal{I}_j, \alpha) \cap out(\mathcal{S}, \alpha)} pr(s_0, \alpha, \beta) \leq 1 - P$. Note that $\mathcal{I}_j$ is not probabilistic, and $pr(s_0, \alpha, \beta)$ is the probability to observe β when α is applied at the initial state s_0 of $\mathcal{S}$. For the considered example FSM $\mathcal{S}$ (shown in Fig. 1), and a potential implementation $\mathcal{I}_1$ shown in Fig. 2, by direct inspection one can observe that $\alpha = i_1 i_2$ is a 0.8-probably separating sequence.

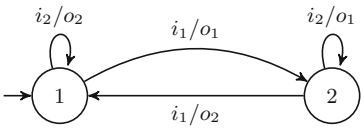

Fig. 2. An implementation FSM $\mathcal{I}_1 \in FD$

[2] A level of certainty that a sequence separates the specification and an implementation.

For a fault model $\langle \mathcal{S}, \cong, FD = \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_k\}\rangle$, we say that the test suite $P\text{-}TS$ is *P-probably exhaustive* if $\forall \mathcal{I}_j \in FD\ \exists \alpha \in P\text{-}TS$ such that α is a P-probably separating sequence for $\mathcal{I}_j$ and $\mathcal{S}$. We aim at deriving such test suites for user defined probabilities via *filtering* a given exhaustive test suite TS for the fault model $\langle \mathcal{S}, \cong, FD = \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_k\}\rangle$.

4 Minimizing an Exhaustive Test Suite Against $\langle \mathcal{S}, \cong, \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_k\}\rangle$

Given an exhaustive test suite $TS = \{\alpha_1, \ldots, \alpha_l\}$ derived for the fault model $\langle \mathcal{S}, \cong, FD = \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_k\}\rangle$, given also a user defined probability P, we propose to derive a test suite $P\text{-}TS \subseteq TS$, aiming at reducing $|P\text{-}TS|$ (in size), and which is P-probably exhaustive for $\langle \mathcal{S}, \cong, FD = \{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_k\}\rangle$. In order to do so, we propose to build a matrix M whose rows correspond to the test sequences of TS while columns correspond to all the implementations from FD. $m_{i,j}$ contains the maximal guaranteed probability $p_{i,j}$ for the sequence α_i (in lexicographical order) to separate the implementation I_j (also in lexicographical order) from the specification FSM $\mathcal{S}$. This probability is calculated as $p_{i,j} = 1 - \sum_{\beta \in out(\mathcal{I}_j, \alpha_i) \cap out(\mathcal{S}, \alpha_i)} pr(s_0, \alpha_i, \beta)$.

Note that, by construction, each column of the matrix M contains at least one 1, as the test suite TS is exhaustive. After M is derived what is left to do is to build a minimal cover of it, such that a subset $P\text{-}TS$ corresponding to the rows covers all columns $\{\mathcal{I}_1, \mathcal{I}_2, \ldots, \mathcal{I}_k\}$, where each probability $p_{i,j} \geq P$. The latter means that for each potential faulty implementation from FD there exists at least one test sequence from $P\text{-}TS$ that P-probably separates it from the specification $\mathcal{S}$.

We omit the discussion about how such a row cover can be constructed - it can be done through an explicit combinatorial enumeration or various (combinatorial) optimization strategies can be applied. The solution to the problem always exists and in the worst case scenario, when nothing could be minimized, $P\text{-}TS = TS$.

Consider again the example FSM $\mathcal{S}$, and the $FD = \{\mathcal{I}_1, \mathcal{I}_2, \mathcal{I}_3\}$, where $\mathcal{I}_1$ is the mutant from Fig. 2; it is separated from $\mathcal{S}$ via $\alpha = i_1 i_2 i_1 \in TS$. $\mathcal{I}_2$ shown in Fig. 3 is separated from $\mathcal{S}$ via the application of $\alpha = i_1 i_1$, and 0.2-probably separated via $\alpha = i_1 i_2$. Finally, the mutant $\mathcal{I}_3$ is shown in Fig. 4. The corresponding separating sequence is $\alpha = i_1 i_2$.

Assume that the $TS = \{i_1 i_1, i_1 i_2, i_1 i_2 i_1\}$; the matrix M for the example FSM, mutants $\mathcal{I}_1, \mathcal{I}_2$ and $\mathcal{I}_3$ and this test suite is the following: $\begin{pmatrix} 0.97 & 1 & 0.2 \\ 0.8 & 0.2 & 1 \\ 1 & 0.36 & 1 \end{pmatrix}$.

As an example, note that the M cover that only consists of first two rows provides 0.97-probably exhaustive[3] $P\text{-}TS = \{i_1 i_1, i_1 i_2\}$. The last test sequence thus can be omitted, preserving the exhaustiveness with the probability 0.97.

[3] This is just an illustrative example; some other pair of rows can even return an exhaustive test suite, nonetheless longer.

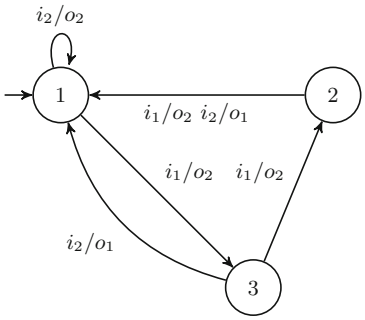

Fig. 3. An implementation FSM $\mathcal{I}_2 \in FD$

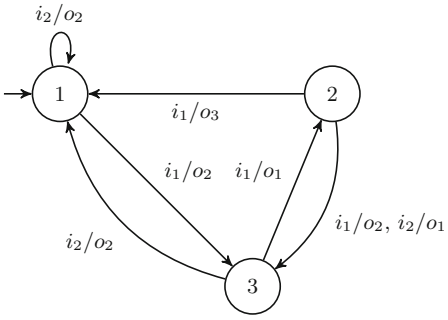

Fig. 4. An implementation FSM $\mathcal{I}_3 \in FD$

5 Conclusion

In this paper, we discussed a possibility of reducing an exhaustive test suite built for a non-deterministic specification, via augmenting this specification with probabilities. The proposed technique relies on a user defined probability P that each potential faulty implementation will be detected (with this probability). The same approach can be applied for a test suite with adaptive separating sequences. In this case, the probability of a test case is the minimum probability of all test case traces. Note also that the proposed approach can also be applied for filtering a non-exhaustive test suite, as long as the sequences left, respect the P-separability relation with the specification.

As a future work, we plan to extend this short paper by considering other fault models, as the proposed technique only considers the non-separability conformance relation and only relies on the white box testing assumption. At the same time, we plan to investigate the model learning strategies for obtaining the probabilities of interest. Finally, as for test derivation, it is interesting to consider how an augmented specification can be used for choosing input sequences, which are more efficient for distinguishing faulty implementations from the specification.

References

1. Alur, R., Courcoubetis, C., Yannakakis, M.: Distinguishing tests for nondeterministic and probabilistic machines. In: Leighton, F.T., Borodin, A. (eds.) Proceedings of the Twenty-Seventh Annual ACM Symposium on Theory of Computing, 29 May–1 June 1995, Las Vegas, Nevada, USA, pp. 363–372. ACM (1995). https://doi.org/10.1145/225058.225161
2. El-Fakih, K., Hierons, R.M., Türker, U.C.: K-branching UIO sequences for partially specified observable non-deterministic FSMS. IEEE Trans. Softw. Eng. **47**(5), 1029–1040 (2021). https://doi.org/10.1109/TSE.2019.2911076
3. Hierons, R.M., Merayo, M.G.: Mutation testing from probabilistic and stochastic finite state machines. J. Syst. Softw. **82**(11), 1804–1818 (2009). https://doi.org/10.1016/j.jss.2009.06.030
4. Kushik, N., Yevtushenko, N., Cavalli, A.R.: On testing against partial non-observable specifications. In: 9th International Conference on the Quality of Information and Communications Technology, QUATIC 2014, Guimaraes, Portugal, 23–26 September 2014, pp. 230–233. IEEE Computer Society (2014). https://doi.org/10.1109/QUATIC.2014.38
5. Petrenko, A., Avellaneda, F.: Learning and adaptive testing of nondeterministic state machines. In: 19th IEEE International Conference on Software Quality, Reliability and Security, QRS 2019, Sofia, Bulgaria, 22–26 July 2019, pp. 362–373. IEEE (2019). https://doi.org/10.1109/QRS.2019.00053
6. Petrenko, A., Yevtushenko, N., von Bochmann, G.: Fault models for testing in context. In: Gotzhein, R., Bredereke, J. (eds.) Formal Description Techniques IX: Theory, application and tools, IFIP TC6 WG6.1 International Conference on Formal Description Techniques IX/Protocol Specification, Testing and Verification XVI, Kaiserslautern, Germany, 8–11 October 1996. IFIP Conference Proceedings, vol. 69, pp. 163–178. Chapman & Hall (1996)
7. Poage, J.F., McCluskey, E.J.: Derivation of optimum test sequences for sequential machines. In: 1964 Proceedings of the Fifth Annual Symposium on Switching Circuit Theory and Logical Design, pp. 121–132 (1964). https://doi.org/10.1109/SWCT.1964.7
8. Spitsyna, N., El-Fakih, K., Yevtushenko, N.: Studying the separability relation between finite state machines. Softw. Test. Verif. Reliab. **17**(4), 227–241 (2007). https://doi.org/10.1002/stvr.374
9. Vidal, E., Thollard, F., de la Higuera, C., Casacuberta, F., Carrasco, R.C.: Probabilistic finite-state machines-part I. IEEE Trans. Pattern Anal. Mach. Intell. **27**(7), 1013–1025 (2005). https://doi.org/10.1109/TPAMI.2005.147
10. Yenigün, H., Kushik, N., López, J., Yevtushenko, N., Cavalli, A.R.: Decreasing the complexity of deriving test suites against nondeterministic finite state machines. In: 2017 IEEE East-West Design and Test Symposium, EWDTS 2017, Novi Sad, Serbia, September 29–October 2, 2017, pp. 1–4. IEEE Computer Society (2017). https://doi.org/10.1109/EWDTS.2017.8110091