Probabilistic Conformance Testing of Protocols with Unobservable Transitions

Steven H. Low
AT&T Bell Laboratories, Murray Hill, NJ 07974
slow@research.att.com

Abstract

We propose a probabilistic approach to conformance testing of protocols containing unobservable transitions. We say that an implementation conforms to its specification if their observable behavior is probabilistically the same, when both are subject to the same random environment simulated by the tester. Under the randomized inputs, faults in unobservable transitions may manifest themselves in certain statistics measurable from the implementation, and hence can be detected by comparing these measurements against the desirable statistics computed from the specification. The sensitivity of the non-conformance criterion to the uncertainty in our knowledge of desirable statistics is also studied. The conventional testing of protocols without unobservable transitions uses mismatch in outputs to detect faults. Here, we rely, in addition, on mismatch in the dynamics of the protocol under input randomization.

1 Introduction

To ensure interoperability of, say, a switch and its peripherals, vendors of the peripherals typically implement a communication protocol according to a specification given by the switch vendor. These implementations are tested by the switch vendor for conformance to the specification before they are deployed. A conformance test applies a sequence of inputs to an implementation and concludes that it is conformant if the observed outputs are as specified. As to be seen, we assume the switch vendor has access to a conformant implementation, whose probabilistic behavior can be measured and used in testing other implementations.

To design an revealing and efficient test sequence, the specification is usually modeled as a deterministic finite state (Mealy) machine (FSM) F', e.g., [1, 2, 3, 4]; see also [5, 6]. An implementation is a 'black box' whose behavior is modeled as another FSM M'. The implementation is said to be conformant if M' and F' are equivalent FSMs. Under suitable assumptions, a test input sequence t can be designed such that F' and M' are equivalent if and only if both produce the same output sequence on t. The design depends critically on the assumptions that both F' and M' are deterministic and that all transitions are observable.

In this paper, we explore conformance testing when the specification is modeled by a FSM F' that contains unobservable inputs, called τ inputs, in addition to external inputs. An implementation is modeled as another FSM M' that shares the same set of inputs and outputs as F'. A tester can select external inputs to apply to the implementation, but it cannot directly control (force or forbid) nor observe τ inputs. If the tester selects an external input when the implementation can either make that input transition or make other τ transitions, the choice may be random; see §2. The nondeterminism introduced by the unobservable transitions makes the testing strategy based on deterministic FSM not directly applicable. We propose a probabilistic approach to conformance testing of such protocols. Unobservable transitions arise when the protocol's behavior depends on features not explicitly modeled, such as data variables. For instance, an external input may represent receipt of a message by the protocol device. It triggers some computations involving internal variables as well as variables in the message. Depending on the result of the computations, the device enters a different state. If it is necessary to abstract out the details of the computation and the variables, result of the computation can be modeled by τ transitions. Unobservable transitions also arise naturally when a protocol is specified and implemented as a collection of communicating FSMs (CFSMs). For instance, the specification F' may be obtained as the composite machine of individual CFSMs, in which τ transitions are inter-CFSM communications. For conformance testing of CFSMs that does not compute the composite machine, see [7].

We make two critical assumptions. We assume that

0-8186-3670-X/93 $03.00

some states are observable in the implementation and that, when randomized external inputs are applied to the specification F', its observable probabilistic behavior (made precise in §2) is known. The first condition is common in digital-circuit designs for testability that is yet to find its way to protocol design. The second assumption is justified if the conformance-test center has access to a correct implementation, whose observable probabilistic behavior can then be measured and used in testing other implementations.

Under these assumptions, our method attempts to detect faults in unobservable as well as observable transitions. Intuitively, we require that the implementation and the specification "look alike" (made precise by π- and P-conformant in §3) when both are subject to the same random environment, simulated by the tester. We assume that when randomized external inputs are applied to the specification F', the probabilistic behavior of F' is known. The basic idea is that, when the same randomization is applied to the implementation M', its observable behavior depends not only on the induced probability law governing how the external inputs in I are chosen, but also on the structure of M'. Faulty unobservable transitions in M' may manifest themselves in certain statistics measurable from the 'black box'. Hence, we may check its conformance by comparing these measurements against the desirable statistics computed from the specification. In comparison, the previous strategy based on deterministic FSM uses mismatch in outputs to detect faults. Here, we rely, in addition, on mismatch in the dynamics of the protocol under input randomization, such as the frequency an observable state is visited or a transition is traversed.

In §2, we explain through an example our model, assumptions and approach. In §3, we formally define two notions of conformance – π-conformance and P-conformance – and propose probabilistic tests for them. Faults in unobservable transitions are detected by comparing the statistics of the observed behavior against the desirable statistics. Our approach hence depends critically on the assumption that we know the statistical behavior of F', against which the measured behavior of M' is compared. In §4, we investigate the sensitivity of the non-conformance criterion for π-conformance to uncertainty in such knowledge and derive a non-conformance criterion that is robust against this uncertainty. A test for π-conformance is simpler than one for P-conformance, but is also less informative. In §5, we show how we may estimate a finer structure of the implementation using the measurements from a test for π-conformance. We conclude in §6 with remarks on the limitations of this work. All proofs are omitted and can be found in [8].

Several previous papers on protocol testing or verification adopt a probabilistic approach [9, 10, 11, 7]. Unlike these researchers, who use randomization primarily to circumvent the state-explosion problem, we use it to tackle the unobservability problem. As will become clear, our emphasis is on detecting faults in *unobservable* transitions by exploiting the protocol's dynamic behavior.

2 Protocol model

A FSM is a 4-tuple $A = (S, I, O, \delta)$, where S is a finite set of states, I is a finite set of input labels, O is a finite set of output labels, and $\delta : S \times I \rightarrow 2^{S \times O}$ is a transition function. $\delta(s, a) = (s', nil) \in S \times O$ if A produces no output in transition from state s to state s' on input a. By a transition label, we mean the input/output pair associated with the transition.

For our purposes, a protocol *specification* is a FSM $F' = (S_{F'}, I \cup \{\tau\}, O, \delta_{F'})$ that contains unobservable inputs labelled by $\tau \notin I$. We call inputs in I external inputs. We assume the specification is deterministic on I, i.e. $\delta_{F'}(s, a)$ is a singleton for all state s if $a \neq \tau$, though $\delta_{F'}(s, \tau)$ can be more than a singleton. We require the specification to be completely specified on I in the following sense. F' can either accept all external inputs in I (and possibly unobservable inputs as well) or no external input in I, i.e. for all state s, either $\delta_{F'}(s, a)$ is defined for all $a \in I$ (and possibly for τ as well), or it is defined only for τ. We will assume for simplicity that F' produces no output on unobservable inputs, i.e. for all s for which $\delta_{F'}(s, \tau)$ is defined, $\delta_{F'}(s, \tau) = (s', nil)$ for some s'. An implementation $M' = (S_{M'}, I \cup \{\tau\}, O, \delta_{M'})$ shares the same set of transition labels as the specification F'. We similarly assume that M' is deterministic and completely specified on I, and that it produces no output on unobservable inputs. Finally, we assume both F' and M' are strongly connected.

Following [7], we assume that a *state* is *observable* if, in that state, the implementation can accept external inputs from the tester. Under this assumption, we construct from F' another FSM $F = (S_F, I \cup \{\tau\}, O, \delta_F)$ called the observable specification. S_F is the set of observable states in F'. There is a transition in F from state s to state s' 1) if F' can accept an input a in s, produce an output b, and follow a sequence of τ transitions to reach s', or 2) if F' can follow from s a sequence of τ transitions to reach s'. In the former case, the transition is labelled

in F by a/b i.e. $\delta_F(s,a) = (s',b)$, where b is possibly *nil*. In the latter case, the transition is labelled by τ, i.e. $\delta_F(s,\tau) = (s', nil)$. Derive in the same way the observable implementation $M = (S_M, I \cup \{\tau\}, O, \delta_M)$.

As an example, consider the specification and its implementation in Figure 1 where $I = \{0,1\}$ and $O = \{a,b,c,d\}$. Ignore for the moment the bracketed probability associated with a transition. Note that the outgoing transition from state 2 on input 1 is not shown, with the interpretation that the tester will not select that input. Similarly for input 0 in state 1. The implementation has two faults (dash transitions): an extra τ transition from state 3 to state 0, and the incorrect destination state for the τ transition out of state 2. For this example, states 0, 1, 2 are observ-

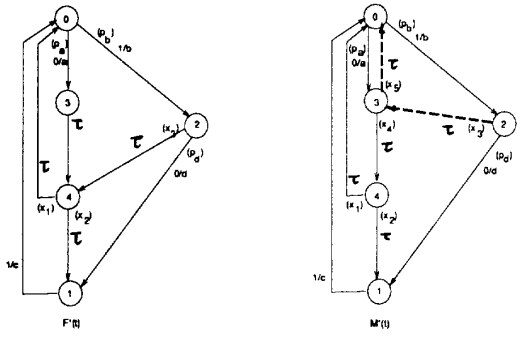

Figure 1: The specification and its implementation

able, and others are not. The observable counterparts of the specification and the implementation are shown in Figure 2.[1]

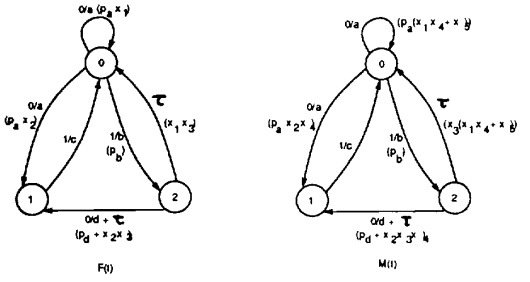

Figure 2: $F = M$ but $F(t) \neq M(t)$

We say that the implementation M' has an *external* fault if the *observable* implementation M contains a transition that has an incorrect output label or an incorrect destination state, or both. Note that this includes the case in which M has an τ transition whose destination state is incorrect. Faulty unobservable transitions may sometimes manifest themselves as external faults that can be detected by simply noting the output and (observable) destination state of each transition in the observable implementation M. Other times, however, they manifest themselves *only* in the dynamic properties of the observable implementation. For our example, despite faults in τ transitions, the observable implementation M is identical to the observable specification F (Figure 2). This kind of faults can be detected using our knowledge of probabilistic behavior of the protocol under input randomization, as explained next.[2]

We assume that the tester can generate external inputs, when the implementation is in an observable state, in such a way that all transitions in the specification F' are traversed with fixed and known probabilities. For our example, in state 0, F' traverses the transitions $0/a$ and $1/b$ with respective probabilities p_a and p_b, satisfying $p_a + p_b = 1$; in state 1, F' traverses the transition $1/c$ with probability 1; in state 2, F' traverses the transitions $0/d$ and τ with probabilities p_d and x_3, satisfying $p_d + x_3 = 1$, etc. (Transition probabilities except 1 are shown in parentheses in figure 1.) Let $F'(t)$ be the state of F' after t transitions, counting τ transitions. Then the state process $F'(t)$ is a Markov chain. It describes the desirable behavior when the implementation M' is under test. Suppose, however, that the two τ transitions out of state 3 in M' are selected with fixed but *unknown* probabilities x_4 and x_5, and that other transition probabilities are as for F'. Then the state process $M'(t)$ of M' is also a Markov chain.

The Markov process $F'(t)$ induces a state process $F(t)$ for the observable specification F. $F(t)$ is the value of $F'(t)$ when $F'(t)$ visits the set S_F of observable states for the t-th time. $F(t)$ is also a Markov chain whose transition matrix P_F can be easily computed from the transition matrix $P_{F'}$ of $F'(t)$. Similarly, $M'(t)$ induces a Markov chain $M(t)$ for the observable implementation M, which is the value of $M'(t)$ when $M'(t)$ visits the observable states for the

[1] A transition labelled by $\sigma + \sigma'$ between a state pair is an abbreviation for two transitions labelled by σ and σ' between the same state pair.

[2] To assess how common this kind of faults are, we performed a simple experiment on an adaptation of full duplex alternating bit protocol originally specified as CFSMs. We randomly introduced faults in τ transitions between unobservable states. We found that in 79% of the (130) randomly generated faulty implementations, the faults manifest themselves as external faults, and in the remaining 21%, the faults only manifest themselves in the observable probabilistic behavior.

t-th time. The key to our approach is to observe that, even though M and F are identical, their dynamic behavior can be quite different. Specifically, their state process $M(t)$ and $F(t)$ may have different transition probabilities.

The transition probabilities for $F(t)$ and $M(t)$ in our example are shown in parentheses in Figure 2. For concreteness, suppose the tester generates random inputs in such a way that $p_a = p_b = p_d = 0.5$, $p_c = 1$ and $x_i = 0.5$, $i = 1, \ldots, 5$. Then the transition matrices of $F(t)$ and $M(t)$ are

$$P_F = \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{2} \\ 1 & 0 & 0 \\ \frac{1}{4} & \frac{3}{4} & 0 \end{bmatrix}, \quad P_M = \begin{bmatrix} \frac{3}{8} & \frac{1}{8} & \frac{1}{2} \\ 1 & 0 & 0 \\ \frac{3}{8} & \frac{5}{8} & 0 \end{bmatrix}$$

and the unique stationary distributions are (see (1) below)

$$\pi_F = [\frac{8}{17} \quad \frac{5}{17} \quad \frac{4}{17}], \quad \pi_M = [\frac{16}{31} \quad \frac{7}{31} \quad \frac{8}{31}]$$

Moreover, the fraction of time $M(t)$ visits state i, $i = 0, 1, 2$, converges to $\pi_M(i) \neq \pi_F(i)$ as the test length increases (see §3). Were M' equivalent to F', it would converge to $\pi_F(i)$. Hence we can measure π_M and detect the faults for this example by counting the frequency each observable state is visited in the implementation.

In practice, the transition probabilities of the specification F' under input randomization may be difficult to obtain. If the conformance-test center has access to a correct implementation, it can measure its transition probabilities among *observable* states under the same randomization strategy and use these measurements as P_F to test other implementations.

3 Probabilistic conformance testing

As in §2, a protocol specification is a FSM $F' = (S_{F'}, I \cup \{\tau\}, O, \delta_{F'})$. An implementation is another FSM $M' = (S_{M'}, I \cup \{\tau\}, O, \delta_{M'})$. In addition to the general setup described in §2, we make the following assumptions:

A1: States of M' in which it can accept inputs are observable, and M' starts in a known observable state.

A2: Randomized external inputs can be generated in such a way that F' and M' behave as Markov chains. Moreover, the transition probabilities in F' among the observable states are known.

As in §2, F and M denote the observable specification and the observable implementation, respectively, and the Markov chains $F(t)$ and $M(t)$ are their state processes. The test we propose below will take the observable implementation M through all its states. If M and F do not have the same number of states, extra or missing observable states will be discovered at the conclusion of a test. We are concerned with the interesting case when M and F indeed have the same number of states. Hence, for ease of exposition, we restrict ourselves to the situation when the following *nonessential* assumption is satisfied:

A3: M' and F' have the same number of *observable* states.

Finally, we make the following technical assumption, which will be satisfied if M and F each contain a self-loop.

A4: The Markov chains $M(t)$ and $F(t)$ are aperiodic [12, pp.65].

Let $S = \{1, \ldots, n\}$ be the state space of F and M. Suppose a test suite satisfying Assumption A2 is applied to F' and M'. Under the assumptions, $F(t)$ and $M(t)$ are two independent discrete-time Markov chains on S with known and unknown transition matrices P_F and P_M, respectively.

Proposition 1 *Under Assumptions A1-A4, the observable process $F(t)$ ($M(t)$) admits a unique stationary distribution $\pi_F > 0$ ($\pi_M > 0$).*[3] *Moreover, $F(t)$ ($M(t)$) is asymptotically stationary and ergodic.*

Proposition 1 provides the basis for our probabilistic approach. It implies that we can measure π_M or P_M from the 'black box' and compare them against the desirable value π_F or P_F.

We can now formally define our notion of conformance. Roughly, it says that implementation M' conforms to specification F' if their observable behavior is probabilistically the same. Note that in the tests proposed below, every transition in M will be traversed, and therefore an external fault will be detected with probability one, as the test length increases.

Definition 1 *Suppose M' has no external faults. Under Assumptions A1-A4, M' is said to be*

1. **π-conformant** *(to F') if the observable processes $M(t)$ and $F(t)$ have the same unique stationary distribution, i.e., $\pi_M = \pi_F$.*

[3] We use the convention that a probability distribution is always a *row* vector. The notation $x > 0$ means that every component x_i of the vector x is strictly positive; similarly for $x \geq 0$.

2. **P-conformant** *(to F') if the observable processes $M(t)$ and $F(t)$ have the same transition matrix, i.e., $P_M = P_F$ (elementwise).*

By Proposition 1, $P_M = P_F \Rightarrow \pi_M = \pi_F$. Hence, P-conformance is stronger than π-conformance: the set of implementations that are P-conformant is a subset of implementations that are π-conformant to a specification.

3.1 Test for π-conformance

The test procedure is:

1. Design test generation so that Assumption A2 is satisfied. Compute π_F using P_F and (1) below.
2. Apply the probabilistic test to M', and count the frequency each observable state is visited.
3. Terminate the test when either an external fault is detected, or (3) below is satisfied, whichever comes first.
4. Conclusion: M' is π-conformant if no external fault is detected and (2) below holds.

Suppose M' has no external faults. Recall from Proposition 1 that both $M(t)$ and $F(t)$ admit unique stationary distributions π_M and π_F, respectively. π_F can be computed from

$$\pi_F P_F = \pi_F, \qquad \sum \pi_F(i) = 1 \tag{1}$$

Since $M(t)$ is ergodic, regardless of the initial state, the fraction of time $M(t)$ visits state i converges to $\pi_M(i)$ as $t \to \infty$. The ergodicity of $M(t)$ provides the basis for our probabilistic testing.

Let the row vector $\pi(t)$ denote the fraction of time each state in M is visited by t, as measured in Step 2 of the test procedure. By Proposition 1, $\lim_t \pi(t) = \pi_M$, and hence M' is π-conformant if and only if

$$\pi(t) \to \pi_F \tag{2}$$

In practice, we may choose two testing parameters $\epsilon > 0$ and $\delta > 0$. We terminate the test when

$$||\pi(t) - \pi(t-1)|| < \epsilon \tag{3}$$

and declare a fault if

$$||\pi(t) - \pi_F|| > \delta$$

where $||\cdot||$ denotes a vector norm.

3.2 Testing for P-conformance

The test procedure for P-conformance is similar to that in §3.1, except that the tester also counts the frequency each transition is traversed:

1. Design test generation so that Assumption A2 is satisfied.
2. Apply the probabilistic test to M'. Count the frequency each observable state is visited, and the frequency the transition between each observable state pair is traversed.
3. Terminate the test when either an external fault is detected, or (5-6) below are satisfied, whichever comes first.
4. Conclusion: M' is P-conformant if no external fault is detected and (4) below holds.

Suppose M' has no external faults. Let $\pi(t)$ be as defined in §3.1. For each state pair $(i,j) \in S^2$, let $P(i,j)(t)$ be the fraction of time M transits from i to j by t, as measured in Step 2 of the test procedure. Since $M(t)$ is ergodic by Proposition 1,

$$P_M(i,j) = \lim_t \frac{P(i,j)(t)}{\pi(i)(t)}$$

Hence M' is P-conformant if and only if

$$\frac{P(i,j)(t)}{\pi(i)(t)} \to P_F(i,j) \tag{4}$$

for all $(i,j) \in S^2$, where $P(t)$ and $\pi(t)$ are measured during the test. In practice, we may choose testing parameters $\epsilon_1 > 0$, $\epsilon_2 > 0$, and $\delta > 0$. We terminate the testing when

$$||\pi(t) - \pi(t-1)|| < \epsilon_1 \tag{5}$$

$$|P(i,j)(t) - P(i,j)(t-1)| < \epsilon_2, \quad (i,j) \in S^2 \tag{6}$$

and declare a fault if

$$\left|\frac{P(i,j)(t)}{\pi(t)} - P_F(i,j)\right| > \delta$$

for some $(i,j) \in S^2$.

4 Sensitivity

Our conformance test detects faults by comparing the measured stationary distribution π_M against the desirable π_F computed from P_F according to (1), and

declare non-conformance if they differ, i.e. if $||\pi_M - \pi_F|| > 0$. In this section, we investigate the effect of uncertainty in P_F on detecting non-conformance.

Specifically, suppose that, due to modeling error (or measurement error if measurements of some other correct implementation is used as P_F), the actual transition matrix of $F(t)$ is

$$P_F(E) = P_F + E$$

for some error matrix E, instead of the nominal P_F. We seek a tolerance $\delta > 0$ such that

$$||\pi_M - \pi_F|| > \delta \tag{7}$$

guarantees the true non-conformance condition

$$||\pi_M - \pi_F(E)|| > 0 \tag{8}$$

where $\pi_F(E)$ is the stationary distribution of the actual matrix $P_F(E)$. We will assume throughout that E is "small enough" so that Assumptions A1-A4 are in force.

By the Perron-Frobenius theorem [13], we can write, possibly after rearranging the columns and rows,

$$I - P_F = \begin{bmatrix} v & a \\ U & b \end{bmatrix} \tag{9}$$

where v is an $(n-1)$-dimensional row vector, U an $(n-1)\times(n-1)$ nonsingular matrix, a a scalar, and b an $(n-1)$-dimensional column vector. Here, n is the number of observable states. Similarly, write

$$I - P_F(E) = \begin{bmatrix} v & a \\ U & b \end{bmatrix} + \begin{bmatrix} e' & a' \\ E' & b' \end{bmatrix} \tag{10}$$

The next proposition provides a tolerance δ for non-conformance that does not require solving for $\pi_F(E)$. We use the l_1 norm $||y|| = \sum |y_i|$ for vectors, and the induced norm for matrices, i.e. $||A|| = \max_j \sum_i |A(i,j)|$ for matrix A.

Proposition 2 *Suppose* $||E'|| < 1/||U^{-1}||$. *Then (7) implies true non-conformance (8) provided*

$$\delta \geq \frac{2\kappa(U)}{1-\kappa(U)||E'||/||U||}\left(\frac{||E'||}{||U||} + \frac{||e'||}{||v||}\right)$$

where $\kappa(U) = ||U||\,||U^{-1}||$.

The tolerance δ bounds the error in using $||\pi_M - \pi_F||$ in place of $||\pi_M - \pi_F(E)||$ to detect non-conformance. It depends on the relative error ($\frac{||E'||}{||U||}$ and $\frac{||e'||}{||v||}$) in the nominal P_F. The proposition says that, if $\kappa(U)$ is small, then a small relative error in P_F induces a small error in using $||\pi_M - \pi_F||$. Note that different v, U in (9), and corresponding e', E' in (10), can be used, and one should select a combination that gives a small δ.

For the example in §2, suppose there is an error of ϵ in the nominal values (0.5) of x_1, x_2, x_3 and p_d. Application of Proposition 2 to the non-conformance criterion

$$M' \text{ is not } \pi\text{-conformant if } ||\pi_M - \pi_F|| > \delta(\epsilon)$$

gives $\delta(\epsilon) = \frac{7\epsilon(7+6\epsilon)}{3(4-7\epsilon^2)}$. We found that (see [8]) we can have an relative error (2ϵ) of up to 60% in our knowledge of the values of x_1, x_2, x_3, p_d and still detect the faults. For instance, when $\epsilon = 0.3$, $||\pi_M - \pi_F|| = 0.74 > 0.66 = \delta(\epsilon)$. □

5 Estimating P_M

Step 2 of the test procedure for π-conformance (§3.1) is less expensive than that for P-conformance (§3.2), because it has to keep track of n^2 fewer counts, n being the number of observable states. Moreover, the termination criterion (3) for the former is less stringent than the termination criterion (5-6) for the latter, possibly leading to a shorter test length. Hence the test for π-conformance is easier to perform, though it is also less informative since it measures only π_M, not P_M. When a test for π-conformance shows that M' is not π-conformant, we may wish to estimate, using the measured π_M, the transition matrix P_M to help identify faults in M'. Let $\mathsf{m}(\pi_M)$ be the set of all irreducible and aperiodic matrices whose unique stationary distribution is π_M. We propose to find among the infinitely many transition matrices in $\mathsf{m}(\pi_M)$ (see [8, Proposition 2]) one which is "closest" to P_F, subject to the condition that no external faults are observed. We hence seek the solution to the following problem:

$$\min_P \quad ||P - P_F||^2 \tag{11}$$

$$\text{subject to} \quad P \in \mathsf{m}(\pi_M), \quad P \prec P_F \tag{12}$$

where for a matrix P, $||P|| = \sqrt{\sum_{i,j}[P(i,j)]^2}$ is the Frobenius norm [14]. Here, '$P \prec P_F$' denotes '$P_F(i,j) = 0 \Rightarrow P(i,j) = 0$'. If this condition is violated, then an external fault will indeed be observed, provided the test length is sufficiently long. In this section, we derive the solution for (11-12).

The problem (11-12) is made difficult by the fact that $\mathsf{m}(\pi)$, $\pi > 0$, is in general not closed. Hence the feasible solution set $\mathsf{m}(\pi) \cap \{P | P \prec P_F\}$ is not compact and the objective function $||P - P_F||^2$ may not achieve its minimum on the feasible solution set, as the example below demonstrates. If we remove the restriction of irreducibility and aperiodicity and replace $\mathsf{m}(\pi)$ by the closed superset

$$\mathsf{m}'(\pi) = \{P | P \text{ is stochastic, } \pi P = \pi\}$$

then the solution indeed exists. The following observation is the key to our solution for (11-12).

Proposition 3 *Given any $\pi > 0$, $\mathsf{m}(\pi)$ is dense in $\mathsf{m}'(\pi)$, i.e., $\overline{\mathsf{m}}(\pi) = \mathsf{m}'(\pi)$, where $\overline{\mathsf{m}}(\pi)$ denotes the closure of $\mathsf{m}(\pi)$.*

The proposition suggests solving the following problem instead of (11-12):

$$\min_P \quad ||P - P_F||^2 \tag{13}$$

$$\text{subject to} \quad P \in \mathsf{m}'(\pi_M), \quad P \prec P_F \tag{14}$$

A unique solution to the original problem (11-12) exists if and only if the unique solution P^* for (13-14) lies in $\mathsf{m}(\pi)$. If P^* lies on the boundary $\overline{\mathsf{m}}(\pi_M) \setminus \mathsf{m}(\pi_M)$, then by the proposition, there exist matrices in $\mathsf{m}(\pi_M)$ that are arbitrarily close to the optimal, i.e. given any $\epsilon > 0$, there exists a matrix P satisfying (12) and

$$||P - P_F||^2 < ||P^* - P_F||^2 + \epsilon$$

Such P is not unique. The proof of the proposition in [8] also shows how to construct such a matrix when (11-12) has no solution.

We illustrate the general method by the following example.

Example

Consider the problem (13-14), and suppose that

$$P_F = \begin{bmatrix} 0.375 & 0.625 & 0 \\ 0.250 & 0.500 & 0.250 \\ 0.500 & 0 & 0.500 \end{bmatrix}$$

and that π_M is measured to be $\pi_M = [1/3 \;\; 1/3 \;\; 1/3]$. To satisfy $P \prec P_F$, we restrict ourselves to matrices in $\mathsf{m}'(\pi_M)$ of the form

$$P = \begin{bmatrix} x_1 & x_2 & 0 \\ x_3 & x_4 & x_5 \\ x_6 & 0 & x_7 \end{bmatrix}$$

where $x_1 \geq 0, \ldots, x_7 \geq 0$ are to be estimated. The conditions that P is stochastic and $\pi_M P = \pi_M$ then becomes

$$\begin{aligned} x_1 + x_2 &= x_3 + x_4 + x_5 = x_6 + x_7 = 1 \\ x_1 + x_3 + x_6 &= x_2 + x_4 = x_5 + x_7 = 1 \\ x_i &\geq 0, \quad i = 1, \ldots, 7 \end{aligned}$$

which can be written more compactly in matrix form as

$$\begin{aligned} Ax &= b \\ x &\geq 0 \end{aligned}$$

where A is a 6×7 matrix, $x = [x_1 \; \cdots \; x_7]^T$, $b = [1 \; \cdots \; 1]^T$, and $0 = [0 \; \cdots \; 0]^T$ are 7-dimensional *column* vectors.[4] Let $y = [0.375 \; 0.625 \; 0.25 \; 0.5 \; 0.25 \; 0.5 \; 0.5]^T$ be the *column* vector of corresponding nonzero entries of P_F. Then (13-14) is reduced to

$$\min_x \quad f(x) = ||x - y||^2 \tag{15}$$

$$\text{subject to} \quad Ax = b \tag{16}$$

$$x \geq 0 \tag{17}$$

where in the above $||v|| = \sqrt{\sum v^2(i)}$ denotes the Euclidean norm of a vector v. By Kuhn-Tucker theorem [15, pp. 314], x^* is a solution for (15-17) if and only if (since f is strictly convex) there exists $\mu = [\mu_1 \; \cdots \mu_6]^T \neq 0$ and $\lambda = [\lambda_1 \; \cdots \lambda_7]^T \geq 0$ such that

$$2x^{*T} + \mu^T A + \lambda^T = 2y^{*T} \tag{18}$$

$$\lambda^T x^* = 0 \tag{19}$$

Equations (18-19) define a system of equations in x^*, μ and $\lambda \geq 0$. They can be solved to yield $x^* = [0.5 \; 0.5 \; 0.5 \; 0.5 \; 0 \; 0 \; 1]$, corresponding to the minimizer

$$P^* = \begin{bmatrix} 0.5 & 0.5 & 0 \\ 0.5 & 0.5 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

for the problem (13-14) with the given P_F and π_M.

Note that P^* is not irreducible and hence is not in $\mathsf{m}(\pi_M)$. Thus the original problem (11-12) has no solution, though one can find a matrix P in $\mathsf{m}(\pi_M)$ that is arbitrarily close to P^*. □

6 Conclusion

We have proposed a probabilistic approach to conformance testing of protocols with unobservable transitions. Unobservable transitions arise naturally when

[4] For any vector or matrix v, v^T denotes its transpose.

a protocol is specified and implemented as a collection of CFSMs, or when its behavior depends on features not explicitly modeled. We have formally defined two notions of conformance and have suggested ways to test for them. We have derived a non-conformance criterion that is robust against uncertainty in our knowledge of the desirable probabilistic behavior. We have also presented a way to estimate the transition matrix of the observable implementation using the measured stationary distribution. The contribution of the paper is to exploit the different observable probabilistic behavior of an implementation to probe the unobservable transitions of a protocol.

Many issues are not addressed in this preliminary work. The previous test strategy based on deterministic model assumes observability of all transitions, but no states need be observable. Our model allows unobservable transitions and the induced nondeterminism, but only at the expense of observability of some states (A1) and the probabilistic assumption on the test generation (A2). The practicality of these assumptions on real protocols needs further study. Our approach uses heavily the limit behavior of an implementation. The rate of convergence to its limit behavior determines the test length required to achieve a certain accuracy. The problem of choosing a good input randomization that is revealing and efficient has not been considered.

Acknowledgements: We are grateful to D. Kristol, D. Lee, N. Maxemchuk, S. Paul, and K. Sabnani for helpful discussions and criticisms.

References

[1] F. C. Hennie. Fault detecting experiments for sequential circuits. *Proc. 5th Annual Symposium Switching Circuit Theory and Logical Design*, pages 95–110, November 1964.

[2] M. P. Vasilevskii. Fault diagnosis of automata. *Kibernetika*, July 1973.

[3] Tsun S. Chow. Testing software design modeled by finite-state machines. *IEEE Trans. on Software Engineering*, SE-4(3), May 1978.

[4] K. Sabnani and A. Dahbura. A protocol test generation procedure. *Computer Networks*, 15(4), 1988.

[5] Zvi Kohavi. *Switching and finite automata theory, 2nd Ed.* McGraw-Hill, 1978.

[6] Gerard J. Holzmann. *Design and Validation of Computer Protocols.* Prentice-Hall, 1991.

[7] D. Lee, K. K. Sabnani, D. M. Kristol, M. U. Uyar, and S. Paul. Conformance testing of protocols specified as communicating FSMs. *The Proceedings of Infocom'93*, 1993.

[8] Steven Low. Probabilistic conformance testing of protocols with unobservable transitions. Technical Memo. 11382-930202-56TM, AT&T Bell Laboratories, February 1993.

[9] Colin H. West. Protocol validation by random state exploration. In B. Sarikaya and G. V. Bochmann, editors, *Protocol Specification, Testing, and Verification, VI*, pages 233–242. Elsevier (North-Holland), 1978.

[10] N. F. Maxemchuk and Krishan Sabnani. Probabilistic verification of communication protocols. *Distributed Computing*, pages 118–129, 1989.

[11] Mihalis Yannakakis and David Lee. Testing finite state machines: Fault detection. to appear in JCSS, 1992.

[12] Jean Walrand. *An Introduction to Queueing Networks.* Prentice Hall, 1988.

[13] Roger A. Horn and Charles R. Johnson. *Matrix Analysis.* Cambridge University Press, 1985.

[14] Ben Noble and James W. Daniel. *Applied Linear Algebra, 3rd Ed.* Prentice-Hall, 1988.

[15] David G. Luenberger. *Linear and Nonlinear Programming, 2nd Ed.* Addison-Wesley Publishing Company, 1984.