A Testing Scenario for Probabilistic Automata*

Mariëlle Stoelinga[1] and Frits Vaandrager[2]

[1] Dept. of Computer Engineering, University of California, Santa Cruz
marielle@soe.ucsc.edu
[2] Nijmegen Institute for Computing and Information Sciences,
University of Nijmegen, The Netherlands
fvaan@cs.kun.nl

Abstract. Recently, a large number of equivalences for probabilistic automata has been proposed in the literature. Except for the probabilistic bisimulation of Larsen & Skou, none of these equivalences has been characterized in terms of an intuitive testing scenario. In our view, this is an undesirable situation: in the end, the behavior of an automaton is what an external observer perceives. In this paper, we propose a simple and intuitive testing scenario for probabilistic automata and we prove that the equivalence induced by this scenario coincides with the trace distribution equivalence proposed by Segala.

1 Introduction

A fundamental idea in concurrency theory is that two systems are deemed to be equivalent if they cannot be distinguished by observation. Depending on the power of the observer, different notions of behavioral equivalence arise. For systems modeled as labeled transition systems, this idea has been thoroughly explored and a large number of behavioral equivalences has been characterized operationally, algebraically, denotationally, logically, and via intuitive "testing scenarios" (also called "button pushing experiments"). We refer to Van Glabbeek [Gla01] for an excellent overview of results in this area of *comparative concurrency semantics*.

Testing scenarios provide an intuitive understanding of a behavioral equivalence via a machine model. A process is modeled as a black box that contains as its interface to the outside world (1) a display showing the name of the action that is currently carried out by the process, and (2) some buttons via which the observer may attempt to influence the execution of the process. A process autonomously chooses an execution path that is consistent with its position in the labeled transition system sitting in the black box. Trace semantics, for instance, is explained in [Gla01] with the *trace machine*, depicted in Figure 1 on the left. As one can see, this machine has no

Fig. 1. The trace machine (left) and the failure trace machine (right).

buttons at all. A slightly less trivial example is the *failure trace machine*, depicted in Figure 1

* Research supported by PROGRESS Project TES4199, Verification of Hard and Softly Timed Systems (HaaST). A preliminary version of this paper appeared in the PhD thesis of the first author [Sto02a].

on the right. Apart from the display, this machine contains as its interface to the outside world a switch for each observable action. By means of these switches, an observer can determine which actions are *free* and which are *blocked* and may be changed at any time during a run of a process. The display becomes empty if (and only if) a process cannot proceed due to the circumstance that all actions are blocked. If, in such a situation, the observer changes her mind and allows one of the actions the process is ready to perform, an action will become visible again in the display. Figure 2 gives an example of two labeled transition systems that can be distinguished by the failure trace

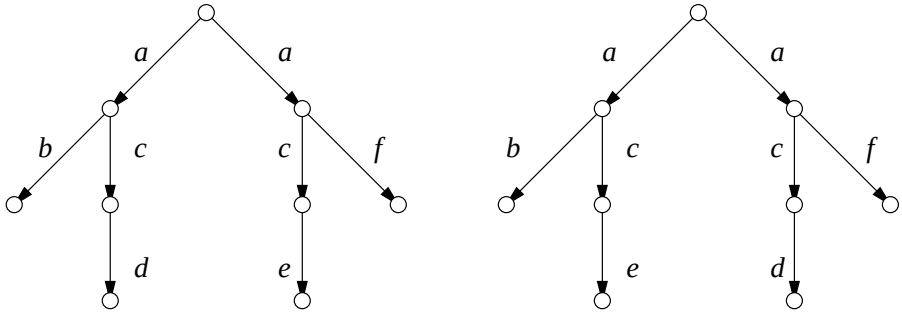

Fig. 2. Trace equivalent but not failure trace equivalent.

machine but not by the trace machine. Since both transition systems have the same traces (ε, a, ab, ac, af, acd and ace), no difference can be observed with the trace machine. However, via the failure trace machine an observer can see a difference by first blocking actions c and f, and only unblocking action c if the display becomes empty. In this scenario an observer of the left system may see an e, whereas in the right system the observer may see a d, but no e. We refer to [Gla01] for an overview of testing scenarios for labeled transition systems.

Probabilistic automata have become a popular mathematical framework for the specification and analysis of probabilistic systems. They have been developed by Segala [Seg95b,SL95,Seg95a] and serve the purpose of modeling and analyzing asynchronous, concurrent systems with discrete probabilistic choice in a formal and precise way. We refer to [Sto02b] for an introduction to probabilistic automata, and a comparison with related models. In this paper, we propose and study a simple and intuitive testing scenario for probabilistic automata: we just add a *reset* button to the trace machine. The resulting *trace distribution machine* is depicted in Figure 3. By resetting

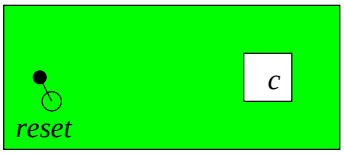

Fig. 3. The trace distribution machine.

the machine it returns to its initial state and starts again from scratch. In the non-probabilistic case the presence of a *reset* button does not make a difference[1], but in the probabilistic case it

[1] For this reason, the *reset* button does not occur in the testing scenarios of [Gla01]. An obvious alternative to the *reset* button would be a *on/off* button.

does: we can observe probabilistic behavior by repeating experiments and applying methods from statistics. Consider the two probabilistic automata in Figure 4. Here the arcs indicate probabilistic

Fig. 4. Probabilistic automata representing a fair and an unfair coin.

choice (as opposed to the nondeterministic choice in Figure 2), and probabilities are indicated adjacent to the edges. These automata represent a fair and an unfair coin, respectively. We assume that the trace distribution machine has an "oracle" at its disposal which resolves the probabilistic choices according to the probability distributions specified in the automaton. As a result, an observer can distinguish the two systems of Figure 4 by repeatedly running the machine until the display becomes empty and then restart it using the *reset* button. For the left process the number of occurrences of trace ab will approximately equal the number of occurrences of trace ac, whereas for the right process the ratio of the occurrence of the two traces will converge to $1:2$. Elementary methods from statistics allow one to come up with precise definitions of distinguishing tests.

The situation becomes more interesting when both probabilistic and nondeterministic choices are present. Consider the probabilistic automaton in Figure 5. If we repeatedly run the trace

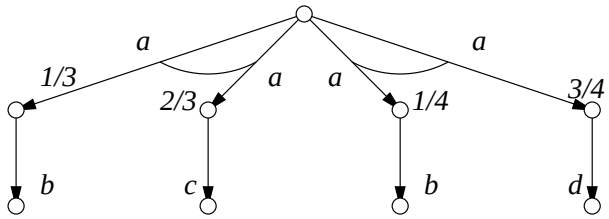

Fig. 5. The combination of probabilistic and nondeterministic choice.

distribution machine with this automaton inside, the ratio between the various traces does not need to converge to a fixed value. However, if we run the machine sufficiently often we will observe that a weighted sum of the number of occurrences of traces ac and ad will approximately equal the number of occurrences of traces ab. Restricting attention to the cases where the left transition has been chosen, we observe $\frac{1}{2}\#[ac] \approx \#[ab]$. Restricting attention to the cases where the right transition has been chosen, we observe $\frac{1}{3}\#[ad] \approx \#[ab]$. Since in each execution either the left or the right transition will be selected, we have:

$$\frac{1}{2}\#[ac] + \frac{1}{3}\#[ad] \approx \#[ab].$$

Even though our testing scenario is simple, the combination of nondeterministic and probabilistic choice makes it far from easy to characterize the behavioral equivalence on probabilistic automata

which it induces. The main technical contribution of this paper is a proof that the equivalence on probabilistic automata induced by our testing scenario coincides with the trace distribution equivalence proposed by Segala [Seg95a].

Being a first step, this paper limits itself to a simple class of probabilistic processes and to observers with limited capabilities. First of all, only *sequential* processes are investigated: processes capable of performing at most one action at a time. Furthermore, we only study *concrete* processes in which no internal actions occur. Finally, observers can only interact with machines in an extremely limited way: apart from observing termination and the occurrence of actions, the only way in which they can influence the course of events is via the *reset* button[2]. It will be interesting to extend our result to richer classes of processes and more powerful observers, and to consider for instance a probabilistic version of the failure trace machine described earlier in this introduction.

Related work Several testing preorders and equivalences for probabilistic processes have been proposed in the literature [Chr90,Seg96,GN98,CDSY99,JY01]. All these papers study testing relations (i.e. testing equivalences or preorders) in the style of De Nicola and Hennesy [DNH84]. That is, they define a test as a (probabilistic) process that interacts with a system via shared actions and that reports success or failure in some way, for instance via success states or success actions. When a test is run on a system, the probability on success is computed, or if nondeterminism is present in either the test or the system, a set of these. By comparing the probabilities on success, one can say whether or not two systems are in the testing equivalence or preorder. For instance, two systems $\mathcal{A}$ and $\mathcal{B}$ are in the testing preorder of [JY01] if and only if for all tests T the maximal probability on success in $\mathcal{A} \parallel T$ is less than or equal to the maximal probability on success in $\mathcal{B} \parallel T$. The different testing relations in the mentioned papers arise by considering different kinds of probabilistic systems, by studying tests with different power (purely nondeterministic tests, finite trees or unrestricted probabilistic processes) and by using different ways to compare two systems under test (e.g. may testing versus must testing). All of the mentioned papers provide alternative characterizations of their testing relation in terms of trace–based relations.

Thus, these testing relations are button pushing experiments in the sense that a test interacts with a system via synchronization on shared actions. However, in our opinion these relations are not entirely observational, because it is not described how the *probability* on success can be observed. In our view, this is an undesirable situation: in the end, the behavior of an automaton is what an external observer perceives. Therefore, we believe that any behavioral equivalence should either be characterized via some plausible testing scenario, or be strictly finer than such an equivalence and be justified via computational arguments.

The only other paper containing a convincing testing scenario for probabilistic systems is by Larsen & Skou [LS91]. They define a notion of tests for *reactive* probabilistic processes, that is, processes in which all outgoing transitions of a state have different labels. Furthermore, the observer is allowed to make arbitrary many copies of any state. For those tests, a fully observable characterization of probabilistic bisimulation based on hypothesis testing is given. (We note that copies of tests can both serve to discover the branching structure of a system – as in the nondeterministic case – and to repeat a certain experiment a number of times.) Our work differs from the approach in [LS91] in the following aspects.

- We present our results in the more general PA model, whereas [LS91] considers the reactive model. As a consequence, the composition of a system and a test in [LS91] is purely probabilistic, that is, it does not contain nondeterministic choices, and theory from classical hypothesis testing applies. In contrast to this, the probabilistic automata that we consider do contain nondeterministic choices. To distinguish between likely and unlikely outcomes in these automata,

[2] This ensures that our testing scenario truly is a "button pushing experiment" in the sense of Milner [Mil80]!

we have to extend (some parts of) hypothesis testing with nondeterminism, which is technically quite involved.
- The main result of this paper, which is the characterization of trace distribution inclusion as a testing scenario, is established for all finitely branching systems, which is much more general than the minimal derivation assumption needed for the results in [LS91].
- The possibility in the testing scenario of Larsen & Skou to make copies of processes in any state (at any moment), is justified for instance in the case of a sequential system where one can make core dumps at any time. But for many distributed systems, it is not possible to make copies in any but the initial state. Therefore, it makes sense to study scenarios in which copying is not possible, as done in this paper.

Overview Even though readers may not expect this after our informal introduction, the rest of this paper is actually quite technical. Section 2 recalls the definitions of probabilistic automata and their behavior and Section 3 presents the characterization of the testing preorder induced by the trace distribution machine as trace distribution inclusion. Sketches of some of the proofs are included in Appendix A. For complete proofs of all our results we refer to the full version of this paper [SV03].

2 Probabilistic Automata

We first recall a few basic notions from probability theory and introduce some notation.

Definition 1. *A* probability distribution *over a set X is a function $\mu : X \to [0,1]$ such that $\sum_{x \in X} \mu(x) = 1$. We denote the set of all probability distributions over X by* $\mathrm{Distr}(X)$*. The probability distribution that assigns 1 to a certain element $x \in X$ and 0 to all other elements, is called the* Dirac distribution *over x and is denoted by $\{x \mapsto 1\}$.*

Definition 2. *A* probability space *is a triple $(\Omega, \mathcal{F}, \mathbf{P})$, where*

- *Ω is a set, called the* sample space*,*
- *$\mathcal{F} \subseteq 2^\Omega$ is σ-*field*, i.e. a collection of subsets of Ω which is closed under countable[3] union and complement, and which contains Ω,*
- *$\mathbf{P} : \mathcal{F} \to [0,1]$ is a* probability measure *on $\mathcal{F}$, which means that $\mathbf{P}[\Omega] = 1$ and for any countable collection $\{C_i\}_i$ of pairwise disjoint subsets in $\mathcal{F}$ we have $\mathbf{P}[\cup_i C_i] = \sum_i \mathbf{P}[C_i]$.*

Now, we recall the notion of a probabilistic automaton from Segala and Lynch [Seg95a,SL95]. Basically, a probabilistic automaton is a non-probabilistic automaton with the only difference that, rather than a single state, the target of a transition is a probability distribution over next states. We consider systems with only external actions, taken from a given, finite set Act. For technical reasons, we assume that Act contains a special element δ, referred to as the *halting* action.

Definition 3. *A* probabilistic automaton (PA) *is a triple $\mathcal{A} = (S, s^0, \Delta)$ with*

- *S a set of states,*
- *$s^0 \in S$ the initial state, and*
- *$\Delta \subseteq S \times Act \times \mathrm{Distr}(S)$ a transition relation.*

We write $s \xrightarrow{a} \mu$ for $(s, a, \mu) \in \Delta$ and $s \overset{a,\mu}{\rightsquigarrow} t$ if $s \xrightarrow{a} \mu$ and $\mu(t) > 0$. We refer to the components of $\mathcal{A}$ as $S_\mathcal{A}, s^0_\mathcal{A}, \Delta_\mathcal{A}$. Moreover, $\mathcal{A}$ is finitely branching *if for each state s, the set $\{(a, \mu, t) \mid s \overset{a,\mu}{\rightsquigarrow} t\}$ is finite, i.e. if every state in $\mathcal{A}$ has finitely many outgoing transitions and the target distribution of each transition assigns a positive probability to finitely many elements.*

[3] In our terminology, countable objects include finite ones.

For the remainder of this section, we fix a PA $\mathcal{A} = (S, s^0, \Delta)$ and assume that Δ contains no transition labeled with δ.

As in the non-probabilistic case, an execution of $\mathcal{A}$ is obtained by resolving the nondeterministic choices in $\mathcal{A}$. This choice resolution is described by an adversary, a function which in each state of the system determines the next transition to be taken. Adversaries are (1) randomized, i.e. make their choices probabilistically, (2) history-dependent, i.e. make choices depending on the path leading to the current state, and (3) partial, i.e. they may choose to halt the execution at any point in time. For technical simplicity, we prefer adversaries that only produce infinite sequences, even if the execution is halted. Therefore, we define the adversaries of a PA $\mathcal{A}$ via its halting extension.

Definition 4. *A* path *of $\mathcal{A}$ is an alternating, finite or infinite sequence*

$$\pi = s_0 a_1 \mu_1 s_1 a_2 \mu_2 s_2 \dots$$

of states, actions, and distributions over states such that (1) π starts with the initial state,[4] *i.e. $s_0 = s^0$, (2) if π is finite, it ends with a state, (3) $s_i \overset{a_{i+1},\mu_{i+1}}{\rightsquigarrow} s_{i+1}$, for each nonfinal i. We set the* length *of π, notation $|\pi|$, to the number of actions occurring in it and denote the set of all finite paths of $\mathcal{A}$ by $Path^*(\mathcal{A})$. If π is finite, then $last(\pi)$ denotes its last state. We define the associated* trace *of π, notation $trace(\pi)$, by $trace(\pi) = a_1 a_2 a_3 \dots$.*

Definition 5. *The* halting extension *of $\mathcal{A}$ is the PA $\delta\mathcal{A} = (S \cup \{\bot\}, s^0, \Delta')$, where Δ' is the least relation such that*

1. $s \xrightarrow{\delta}_{\delta\mathcal{A}} \{\bot \mapsto 1\}$,
2. $s \xrightarrow{a}_{\mathcal{A}} \mu \implies s \xrightarrow{a}_{\delta\mathcal{A}} (\mu \cup \{\bot \mapsto 0\})$.

Here we assume that $\bot$ is fresh. The transitions with label δ are referred to as halting *transitions.*

Definition 6. *A* (partial, randomized, history-dependent) adversary *E of $\mathcal{A}$ is a function*

$$E : Path^*(\delta\mathcal{A}) \to \mathrm{Distr}(Act \times \mathrm{Distr}(S_{\delta\mathcal{A}}))$$

such that, for each finite path π, if $E(\pi)(a, \mu) > 0$ then $last(\pi) \xrightarrow{a}_{\delta\mathcal{A}} \mu$.

We say that E is deterministic *if, for each π, $E(\pi)$ is a Dirac distribution. An adversary E* halts *on a path π if it extends π with the halting transition, i.e.,*

$$E(\pi)(\delta, \{\bot \mapsto 1\}) = 1.$$

For $k \in \mathbb{N}$, we say that the adversary E halts after k steps *if it halts on all paths with length greater than or equal to k. We denote by $Adv(\mathcal{A}, k)$ the set of all adversaries of $\mathcal{A}$ that halt after k steps and by $Dadv(\mathcal{A}, k)$ the set of deterministic adversaries in $Adv(A, k)$. Finally, we call E* finite *if $E \in Adv(\mathcal{A}, k)$, for some $k \in \mathbb{N}$.*

The probabilistic behavior of an adversary is summarized by its associated probability space. First we introduce the function $\mathbf{Q}^E$, which yields the probability that E assigns to finite paths.

Definition 7. *Let E be an adversary of $\mathcal{A}$. The function $\mathbf{Q}^E : Path^*(\delta\mathcal{A}) \to [0, 1]$ is defined inductively by $\mathbf{Q}^E(s_0) = 1$ and $\mathbf{Q}^E(\pi a \mu s) = \mathbf{Q}^E(\pi) \cdot E(\pi)(a, \mu) \cdot \mu(s)$.*

Definition 8. *Let E be an adversary of $\mathcal{A}$. The* probability space *associated to E is the probability space given by*

[4] Here we deviate from the standard definition, as we do not need paths starting from non-initial states.

1. $\Omega_E = Path^\infty(\delta\mathcal{A})$,
2. $\mathcal{F}_E$ *is the smallest* σ*-field that contains the set* $\{C_\pi \mid \pi \in Path^*(\delta\mathcal{A})\}$, *where* $C_\pi = \{\pi' \in \Omega_E \mid \pi \text{ is a prefix of } \pi'\}$,
3. $\mathbf{P}_E$ *is the unique measure on* $\mathcal{F}_E$ *such that* $\mathbf{P}_E[C_\pi] = \mathbf{Q}^E(\pi)$, *for all* $\pi \in Path^*(\delta\mathcal{A})$.

The fact that $(\Omega_E, \mathcal{F}_E, \mathbf{P}_E)$ *is a probability space follows from standard measure theory arguments, see for instance [Coh80].*

As for non-probabilistic automata, the visible behavior of $\mathcal{A}$ is obtained by removing the non-visible elements (in our case, the states) from an execution (adversary). This yields a trace distribution of $\mathcal{A}$, which assigns a probability to (certain) sets of traces.

Definition 9. *The* trace distribution *H of an adversary E, denoted $trd(E)$, is the probability space given by*

1. $\Omega_H = Act^\infty$,
2. $\mathcal{F}_H$ *is the smallest* σ*-field that contains the sets* $\{C_\beta \mid \beta \in Act^*\}$, *where* $C_\beta = \{\beta' \in \Omega_H \mid \beta \text{ is a prefix of } \beta'\}$,
3. $\mathbf{P}_H$ *is the unique measure on* $\mathcal{F}_H$ *such that* $\mathbf{P}_H[X] = \mathbf{P}_E[trace^{-1}(X)]$.

Standard measure theory arguments [Coh80] ensure again that that $trd(E)$ is well-defined. The set of trace distributions of adversaries of $\mathcal{A}$ is denoted by $trd(\mathcal{A})$ and $trd(\mathcal{A}, k)$ denotes the set of trace distributions that arise from adversaries of $\mathcal{A}$ halting after k steps. We write $\mathcal{A} \equiv_{\mathrm{TD}} \mathcal{B}$ if $trd(\mathcal{A}) = trd(\mathcal{B})$; $\mathcal{A} \sqsubseteq_{\mathrm{TD}} \mathcal{B}$ if $trd(\mathcal{A}) \subseteq trd(\mathcal{B})$ and $\mathcal{A} \sqsubseteq^k_{\mathrm{TD}} \mathcal{B}$ if $trd(\mathcal{A}, k) \subseteq trd(\mathcal{B}, k)$.

3 Characterization of Testing Preorder

This section characterizes the observations of a trace distribution machine. That is, we define the set $Obs(\mathcal{A})$ of sequences of traces that are likely to be produced when the trace distribution machine operates as specified by the PA $\mathcal{A}$. Then, our main characterization theorem states that two PAs have the same observations if and only if they have the same trace distributions.

Define a *sample O of depth k and width m* to be an element of $(Act^k)^m$, i.e., a sequence consisting of m sequences of actions of length k. A sample describes what an observer may potentially record when running m times an experiment of length k on the trace distribution machine. Note that if, during a run, the machine halts before k observable actions have been performed, we can still obtain a sequence of k actions by attaching a number of δ actions at the end. We write $freq(O)$ for the function in $Act^k \to \mathbb{Q}$ that assigns to each sequence β in Act^k the frequency with which β occurs in O. That is, for $O = \beta_1, \beta_2, \ldots, \beta_m$ let

$$freq(O)(\beta) = \frac{\#\{i \mid \beta_i = \beta, 1 \leq i \leq m\}}{m}.$$

Note that $freq(O)$ is a probability distribution over $(Act^k)^m$. We base our statistical analysis on $freq(O)$ rather than just O. This means we ignore some of the information contained in samples, which more advanced statistical methods may want to explore. If, for instance, we consider the sample O of depth one and width 2000 that consists of 1000 *head* actions followed by 1000 *tail* actions, then it is quite unlikely that this will be a sample of a trace distribution machine implementing a fair coin. However, the frequency function $freq(O)$ can very well be generated by a fair coin.

Assume that the process sitting in the black box is given by the PA $\mathcal{A}$. This means that, when operating, the trace distribution machine chooses a trace $\mathcal{A}$ according to some trace distribution H

of $\mathcal{A}$. Thus, when running m experiments on the trace distribution, we obtain a sample O length m generated by a sequence of m trace distributions in $trd(\mathcal{A}, k)$.

For a trace distribution $H \in trd(\mathcal{A}, k)$, we denote by $\mu_H : Act^k \to [0, 1]$ the probability distribution given by $\mu_H(\beta) = \mathbf{P}_H[C_\beta]$. Since H halts after k steps, $\mu_H(\beta)$ yields the probability that the sequence β is picked when we generate a trace according to H. In other words, $\mu_H(\beta)$ yields the probability that during a run, the trace distribution machine produces the action sequence β, when it resolves its nondeterministic choices according to an adversary E with $trd(E) = H$. Now, we generate a sample of width m by independently choosing m sequences according to distributions $H_1, \ldots H_m$ respectively. Then, the probability to pick the sample $O = \beta_1, \beta_2, \ldots, \beta_m$ is given by

$$\mathbf{P}_{H_1,\ldots,H_m}[O] \;=\; \prod_{i=1}^{m} \mu_{H_i}(\beta_i).$$

Finally, the probability that an element from the set $\mathcal{O} \subseteq (Act^k)^m$ is picked equals

$$\mathbf{P}_{H_1,\ldots,H_m}[\mathcal{O}] \;=\; \sum_{O \in \mathcal{O}} \mathbf{P}_{H_1,\ldots,H_m}[O].$$

Given $H_1, H_2, \ldots, H_m$, we want to distinguish between samples that are likely to be generated by $H_1, H_2, \ldots, H_m$, and those which are not. To do so, we first fix an $\alpha \in (0, 1)$ as the desired level of significance. Our goal is to define the set $\mathcal{K}_{H_1,H_2,\ldots,H_m}$, of likely outcomes in such a way that

1. $\mathbf{P}_{H_1,\ldots,H_m}[\mathcal{K}_{H_1,H_2,\ldots,H_m}] > 1 - \alpha$,
2. $\mathcal{K}_{H_1,H_2,\ldots,H_m}$ is, in some sense, minimal.

Condition (1) will ensure that, most likely, $H_1, \ldots, H_m$ generate an element in $\mathcal{K}_{H_1,H_2,\ldots,H_m}$. The probability that we reject O as a sample generated by $H_1, \ldots, H_m$ while it is so, is at most α. Condition (2) will ensure that $\mathbf{P}_{H'_1,\ldots,H'_m}[\mathcal{K}_{H_1,H_2,\ldots,H_m}]$ is as small as possible for sequences $H'_1, \ldots, H'_m$ different from $H_1, \ldots, H_m$. (How small this probability is highly depends on which H'_i's we take.) Therefore, the probability that we consider O to be an execution while it is not, is as small as possible. In terminology from hypothesis testing: our null hypothesis states that O is generated by $H_1, \ldots, H_m$ and condition (1) bounds the probability on false rejection and (2) minimizes the probability on false acceptance. The set $\mathcal{K}_{H_1,H_2,\ldots,H_m}$ is the complement of the critical section. Note that in classical hypothesis testing all subsequent experiments $\beta_1, \ldots, \beta_m$ are drawn from the same probability distribution, whereas in our setting, each experiment is governed by a different probability mechanism given by H_i.

The idea behind the definition of $\mathcal{K}_{H_1,\ldots,H_m}$ is as follows. The *expected frequency* of a sequence β in a sample generated by $H_1, \ldots, H_m$ is given by

$$\mathbf{E}_{H_1,\ldots,H_m}(\beta) = \frac{1}{m} \sum_{i=1}^{m} \mu_{H_i}(\beta).$$

Since fluctuations around the expected value are likely, we allow deviations of at most ε from the expected value. Here, we choose ε as small as possible, but large enough such that the probability on a sample whose frequency deviates at most ε from $\mathbf{E}_{H_1,\ldots,H_m}$ is bigger than α. Then, conditions (1) and (2) above are met. Formally, define the *ε-sphere $B_\varepsilon(\mu)$ with center μ* as

$$B_\varepsilon(\mu) = \{\nu \in \mathrm{Distr}(Act^k) \mid dist(\mu, \nu) \leq \varepsilon\},$$

where *dist* is the standard distance on $\mathrm{Distr}(Act^k)$ given by $dist(\mu, \nu) = \sqrt{\sum_{\beta \in Act^k} |\mu(\beta) - \nu(\beta)|^2}$.

Definition 10. *For a sequence $H_1, H_2, \ldots H_m$ of trace distributions in $trd(\mathcal{A}, k)$, we define $\mathcal{K}_{H_1,\ldots H_m}$ as the smallest*[5] *sphere $B_\varepsilon(\mathbf{E}_{H_1,\ldots H_m})$ such that*

$$\mathbf{P}_{H_1,\ldots,H_m}[\{O \in (Act^k)^m \mid freq(O) \in B_\varepsilon(\mathbf{E}_{H_1,\ldots H_m})\}] > 1 - \alpha.$$

We say that O is an observation of $\mathcal{A}$ (of depth k and width m) *if*

$$O \in \mathcal{K}_{H_1,\ldots,H_m}.$$

We write $Obs(\mathcal{A})$ for the set of observations of $\mathcal{A}$.

Example 1. We take $\alpha = 0.05$ as the level of significance. First, consider the leftmost PA in Figure 4 and samples of depth 2 and width 100. This means that the probabilistic trace machine is run 100 times and each time we get a trace of length 2.

Then any sample O_1 in which the sequence ab occurs 42 times and ac 58 times is an observation of $\mathcal{A}$, but samples in which ab occurs 38 times and ac 62 times are not. Let E be the adversary that, in each state of $\mathcal{A}$, schedules with probability one the unique transition leaving that state, if there is such a transition. Otherwise, E schedules the halting transition with probability one. For $H = trd(E)$, we have $\mu_H(ab) = \mu_H(ac) = \frac{1}{2}$ and $\mu_H(\beta) = 0$ for all other sequences. Let $H^{100} = (H_1, \ldots H_{100})$ be sequence of adversaries with $H_i = H$. Then $\mathbf{E}_{H^{100}} = \mu_H$ and, since μ_H assigns a positive probability only to ab and ac, we have that $\mathbf{P}_{H^{100}}[B_\varepsilon(\mu_H)] = \mathbf{P}_{H^{100}}[\{O_1 \mid \frac{1}{2} - \varepsilon \leq freq(O_1)(ab) \leq \frac{1}{2} + \varepsilon\}]$. One can show that then smallest sphere such that $\mathbf{P}_{H^{100}}[B_\varepsilon(\mu_H)] > 0.95$ is obtained by taking $\varepsilon = \frac{1}{10}$. Since $freq(O_1) \in B_\varepsilon(\mu_H)$, O_1 is an observation.

Then, a sample O_2 containing with 20 $\delta\delta$'s, 42 ab's and 58 ac's is an observation of depth 2 and width 120. It arises from taking 100 times adversary E as above and 20 adversaries that halt with probability one on every path. Now, consider the automaton in Figure 5. Consider the scheduler E_3 that in the initial state, schedules both a transitions with probability $\frac{1}{2}$. In the other states, E_3 schedules with probability one the unique outgoing transition if avaible and halts otherwise. Let $H_3 = trd(E_3)$ and let H_3^{120} be the sequence consisting of 120 times the adversary H_3. The expected frequency of H_3^{120} is $\frac{7}{24}$ for ab, $\frac{8}{24}$ for ac, and $\frac{9}{24}$ for ad. Then $\mathcal{K}_{H_3^{120}} = B_{\frac{1}{11}}(\mathbf{E}_{H_3^{120}})$ and for instance, the sequence with 40 ab's, 40 ac's and 40 ad's is an observation of the mentioned PA.

We can now state our main characterization theorem.

Theorem 1. *For all finitely branching PAs $\mathcal{A}$ and $\mathcal{B}$*

$$Obs(\mathcal{A}) = Obs(\mathcal{B}) \iff \mathcal{A} \equiv_{\mathrm{TD}} \mathcal{B}.$$

Acknowledgement The ideas worked out in this paper were presented in preliminary form at the seminar "Probabilistic Methods in Verification", which took place from April 30 – May 5, 2000, in Schloss Dagstuhl, Germany. We thank the organizers, Moshe Vardi, Marta Kwiatkowska, Christoph Meinel and Ulrich Herzog, for inviting us to participate in this inspiring meeting.

References

[BBK87] J.C.M. Baeten, J.A. Bergstra, and J.W. Klop. On the consistency of Koomen's fair abstraction rule. *Theoretical Computer Science*, 51(1/2):129–176, 1987.

[BK86] J.A. Bergstra and J.W. Klop. Verification of an alternating bit protocol by means of process algebra. In W. Bibel and K.P. Jantke, editors, *Math. Methods of Spec. and Synthesis of Software Systems '85, Math. Research 31*, pages 9–23, Berlin, 1986. Akademie-Verlag.

[5] This minimum exists, because there are finitely many samples.

[CDSY99] R. Cleaveland, Z. Dayar, S. A. Smolka, and S. Yuen. Testing preorders for probabilistic processes. *Information and Computation*, 154(2):93–148, 1999.

[Chr90] I. Christoff. Testing equivalence and fully abstract models of probabilistic processes. In J.C.M. Baeten and J.W. Klop, editors, *Proceedings CONCUR 90,* Amsterdam, volume 458 of *Lecture Notes in Computer Science*. Springer-Verlag, 1990.

[Coh80] D.L. Cohn. *Measure Theory*. Birkhaeuser, Boston, 1980.

[DNH84] R. De Nicola and M. Hennessy. Testing equivalences for processes. *Theoretical Computer Science*, 34:83–133, 1984.

[Gla01] R.J. van Glabbeek. The linear time — branching time spectrum I. The semantics of concrete, sequential processes. In J.A. Bergstra, A. Ponse, and S.A. Smolka, editors, *Handbook of Process Algebra*, pages 3–99. North-Holland, 2001.

[GN98] C. Gregorio-Rodrígez and M. Núñez. Denotational semantics for probabilistic refusal testing. In M. Huth and M.Z. Kwiatkowska, editors, *Proc. ProbMIV'98*, volume 22 of *Electronic Notes in Theoretical Computer Science*, 1998.

[JY01] B. Jonsson and W. Yi. Compositional testing preorders for probabilistic processes. *Theoretical Computer Science*, 2001.

[LS91] K.G. Larsen and A. Skou. Bisimulation through probabilistic testing. *Information and Computation*, 94:1–28, 1991.

[Mil80] R. Milner. *A Calculus of Communicating Systems*, volume 92 of *Lecture Notes in Computer Science*. Springer-Verlag, 1980.

[Seg95a] R. Segala. Compositional trace–based semantics for probabilistic automata. In *Proc. CONCUR'95*, volume 962 of *Lecture Notes in Computer Science*, pages 234–248, 1995.

[Seg95b] R. Segala. *Modeling and Verification of Randomized Distributed Real-Time Systems*. PhD thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, June 1995. Available as Technical Report MIT/LCS/TR-676.

[Seg96] R. Segala. Testing probabilistic automata. In *Proc. CONCUR'96*, volume 1119 of *Lecture Notes in Computer Science*, pages 299–314, 1996.

[SL95] R. Segala and N.A. Lynch. Probabilistic simulations for probabilistic processes. *Nordic Journal of Computing*, 2(2):250–273, 1995.

[Sto02a] M.I.A. Stoelinga. *Alea jacta est: verification of probabilistic, real-time and parametric systems*. PhD thesis, University of Nijmegen, the Netherlands, April 2002. Available via `http://www.soe.ucsc.edu/~marielle`.

[Sto02b] M.I.A. Stoelinga. An introduction to probabilistic automata. In G. Rozenberg, editor, *EATCS bulletin*, volume 78, 2002.

[SV03] M.I.A. Stoelinga and F.W. Vaandrager. A testing scenario for probabilistic automata. Technical Report NIII-R0307, Nijmegen Institute for Computing and Information Sciences, University of Nijmegen, 2003. Available via `http://www.soe.ucsc.edu/~marielle`.

A Appendix

This appendix proves the main characterization theorem of this paper, which says that the testing equivalence induced by the trace distribution machine coincides with the trace distribution equivalence. Our proof uses various auxiliary results which are stated, but the reader is referred to [SV03] for their proofs.

The first result we need states that each finite adversary in a finitely branching PA can be written as a convex combination of deterministic adversaries.

Lemma 1. *Let $k \in \mathbb{N}$, let $\mathcal{A}$ be a finitely branching PA and let E be an adversary in $Adv(\mathcal{A}, k)$. Then E can be written as a convex combination of deterministic adversaries in $Dadv(\mathcal{A}, k)$, i.e., there exists a probability distribution ν over $Dadv(\mathcal{A}, k)$ such that, for all π, a and μ,*

$$E(\pi)(a, \mu) = \sum_{D \in Dadv(\mathcal{A},k)} \nu(D) \cdot D(\pi)(a, \mu) \quad \textit{and} \quad \mathbf{Q}^E(\sigma) = \sum_{D \in Dadv(\mathcal{A},k)} \nu(D) \cdot \mathbf{Q}^D(\sigma).$$

A crucial result needed to characterize the testing equivalence is the *Approximation Induction Principle (AIP)* (cf. [BK86,BBK87]). This result is interesting in itself and was first observed in [Seg96]. A proof can be found in [SV03].

Theorem 2 (Approximation Induction Principle). *Let $\mathcal{A}$ and $\mathcal{B}$ be PAs and let $\mathcal{B}$ be finitely branching. Then*

$$\forall k.\ \mathcal{A} \sqsubseteq^k_{\mathrm{TD}} \mathcal{B} \quad \Longrightarrow \quad \mathcal{A} \sqsubseteq_{\mathrm{TD}} \mathcal{B}.$$

By Chebychev's Inequality, one easily derives the following.

Proposition 1. *Let $\alpha, \varepsilon > 0$. Then there exists an $m' \in \mathbb{N}$ such that the following holds. For all $m \geq m'$, and all sequences $X_1, X_2, \ldots, X_m$ of m independent random variables, where X_i has a Bernoulli distribution with parameter p_i, for some $p_i \in [0, 1]$ (i.e. $\mathbf{P}[X_i = 1] = p_i, \mathbf{P}[X_i = 0] = 1 - p_i$), we have that*

$$\mathbf{P}[|Z_m - \mathbf{E}[Z_m]| > \varepsilon] \leq \alpha.$$

Here, $Z_m = \frac{1}{m} \sum_{i=1}^{m} X_i$ yields the frequency of the number of times that a 1 has been drawn in $(X_1, \ldots, X_m)$.

One can reformulate this proposition as follows.

Corollary 1. *Let $\alpha, \varepsilon > 0$ and $k \in \mathbb{N}$. Then there exists an $m' \in \mathbb{N}$ such that for all $m \geq m'$ and all trace distributions $H_1, H_2, \ldots, H_m \in trd(\mathcal{A}, k)$*

$$\mathbf{P}_{H_1,\ldots,H_m}[\{O \in (Act^k)^m \mid freq(O) \in B_\varepsilon(\mathbf{E}_{H_1,\ldots,H_m})] > 1 - \alpha.$$

The following results are elementary. The second part follows from Lemma 1.

Proposition 2. *1. $H = K \iff \mu_H = \mu_K$.*

2. For every $H \in trd(\mathcal{A}, k)$, μ_H can be written as a convex combination of distributions μ_{H_i}, where H_i is generated by a deterministic adversary. That is, there exists a probability distribution ν over the set $Dadv(A, k)$ such that, for all $\sigma \in Act^k$, $\mu_K(\sigma) = \sum_{D \in \mathcal{D}} \nu(D) \cdot \mu_{trd(D)}(\sigma)$.

Now, we can prove our main theorem.

Theorem 3. *For all finitely branching PAs $\mathcal{A}$ and $\mathcal{B}$*

$$Obs(\mathcal{A}) = Obs(\mathcal{B}) \iff \mathcal{A} \equiv_{\mathrm{TD}} \mathcal{B}.$$

Proof: The "$\Longleftarrow$" follows immediately from the definitions. To prove " $\Longrightarrow$ " assume that $\mathcal{A} \not\sqsubseteq_{\mathrm{TD}} \mathcal{B}$. We show that $Obs(\mathcal{A}) \not\subseteq Obs(\mathcal{B})$.

By Theorem 2, there exists a k such that $\mathcal{A} \not\sqsubseteq^{k}_{\mathrm{TD}} \mathcal{B}$, i.e. $trd(\mathcal{A}, k) \not\subseteq trd(\mathcal{B}, k)$. Let H be a trace distribution in $trd(\mathcal{A}, k)$ that is not a trace distribution in $trd(\mathcal{B}, k)$. Then, Proposition 2(1) concludes that there is no $K \in trd(\mathcal{B}, k)$ such that $\mu_H = \mu_K$. Moreover, Proposition 2(2) states that the set $\{\mu_K \mid K \in trd(\mathcal{B}, k)\}$ is a polyhedron. Therefore, there is minimal distance $d > 0$ between μ_H and *any* μ_K with K in $trd(\mathcal{B}, k)$.

We write H^m for the sequence $(H_1, H_2, \ldots, H_m)$ with $H_i = H$ for all $1 \leq i \leq m$. By Corollary 1, we can find $m_{\mathcal{A}}$ and $m_{\mathcal{B}}$ such that for all $m \geq m_{\mathcal{A}}$ and $m \geq m_{\mathcal{B}}$ and all trace distributions $K_1, K_2, \ldots, K_m$ in $trd(\mathcal{B}, k)$

$$\mathbf{P}_{H^m}[\{O \in (Act^k)^m \mid freq(O) \in B_{\frac{d}{3}}(\mathbf{E}_{H^m})\}] > 1 - \alpha$$

and

$$\mathbf{P}_{K_1,\ldots,K_m}[\{O \in (Act^k)^m \mid freq(O) \in B_{\frac{d}{3}}(\mathbf{E}_{K_1,\ldots,K_m})\}] > 1 - \alpha.$$

Hence, $\mathcal{K}_{H^m} \subseteq B_{\frac{d}{3}}(\mathbf{E}_{H^m}) = B_{\frac{d}{3}}(\mu_H)$. On the other hand, for $1 \leq i \leq m$, let $E_i \in trd(\mathcal{B}, k)$ be such that $K_i = trd(E_i)$ and take $K = trd(()\frac{1}{m}\sum_{i=1}^{m} E_i)$. One easily shows that $\mathbf{E}_{K_1,\ldots,K_m} = \mathbf{E}_{K^m} = \mu_K^m$. Therefore, $\mathcal{K}_{K_1,\ldots,K_m} \subseteq B_{\frac{d}{3}}(\mathbf{E}_{K_1,\ldots,K_m}) = B_{\frac{d}{3}}(\mu_K)$. Since $|\mu_H - \mu_K| \geq d > 0$, we have $B_{\frac{d}{3}}(\mu_H) \cap B_{\frac{d}{3}}(\mu_K) = \emptyset$, and therefore, $\mathcal{K}_{H^m} \cap \mathcal{K}_{K_1,\ldots,K_m} = \emptyset$. Hence, none of the observations in $\mathcal{K}_{H^m}$ is an observation of $\mathcal{B}$, i.e. $Obs(\mathcal{A}) \not\subseteq Obs(\mathcal{B})$. □