CHAPTER 4

A Testing Scenario for Probabilistic Automata

> Een kind op een strand pakt een handvol zand en gooit het weg. Hoe groot was de kans dat die zandkorrels ooit nog eens bij elkaar zouden komen? Nul. Maar hoe groot was dan duizend jaar geleden die kans geweest? Ook nul. Toch waren ze bij elkaar gekomen. Zo was het met alles. Een voetstap die werd gezet, een druppel die uit de kraan viel, een vogel die op een tak ging zitten – nooit meer zou het precies zo gebeuren, nooit had het kunnen gebeuren.
>
> Tim Krabbé, *De grot*

Abstract Recently, a large number of equivalences for probabilistic automata has been proposed in the literature. However, to the best of our knowledge, none of these equivalences has been characterized in terms of intuitive testing scenarios, except for the probabilistic bisimulation of Larsen & Skou. In our view, this is an undesirable situation: 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. In this chapter, we propose and study a simple and intuitive testing scenario for probabilistic automata. We prove that the equivalence induced by this scenario coincides with the trace distribution equivalence proposed by Segala. A technical result that we need to establish on the way is an *Approximation Induction Principle (AIP)* for probabilistic processes.

4.1 Introduction

A fundamental idea in concurrency theory is that two systems are deemed equivalent if they cannot be distinguished by observation. Depending on the power of an observer, different notions of behavioral equivalence arise. For 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 on which the name of the action is shown that is currently carried out by the process, and (2) some buttons via which an 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 that is contained in the black box. Trace semantics, for instance, is explained in [Gla01] with the *trace machine*, depicted

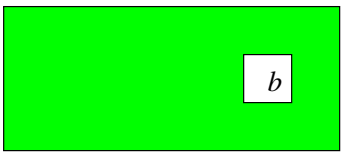

Figure 4.1: The trace machine.

in Figure 4.1. As one can see, this machine has no buttons at all. A slightly less trivial example is the *failure trace machine*, depicted in Figure 4.2, which, apart from the display,

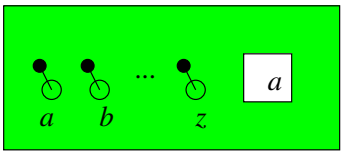

Figure 4.2: The failure trace machine.

contains as its interface to the outside world a switch for each observable action. By means of these switches, an observer may determine which actions are *free* and which are *blocked*. This situation 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 it is prepared to continue with 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 4.3 gives an example of two labeled transition systems

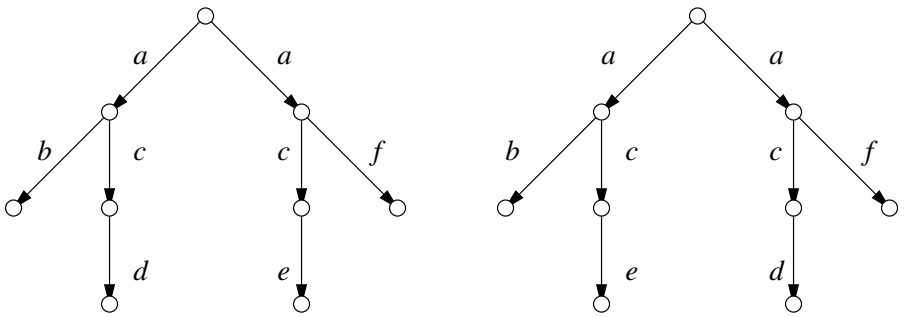

Figure 4.3: Trace equivalent but not failure trace equivalent.

that can be distinguished by the failure trace machine but not by the trace machine. Since both transition systems have the same traces (ε, a, ab, af, ac, 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 when the display becomes empty. In this scenario an observer of the left system may see an e, which is impossible for an observer of the right system. We refer to [Gla01] for an overview of testing scenarios for labeled transition systems.

In this chapter, we propose and study a very simple and intuitive testing scenario for

probabilistic automata: we just add an *on/off* button to the trace machine. The resulting *trace distribution machine* is depicted in Figure 4.4. By turning the machine off and then on again

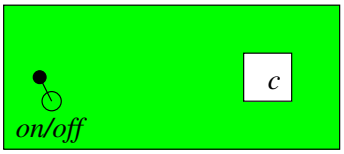

Figure 4.4: The trace distribution machine.

it returns to its initial state and starts again from scratch. In the non-probabilistic case the presence of an *on/off* button does not make a difference[1], but in the probabilistic case it does: we can observe probabilistic behavior by repeating experiments and applying methods from statistics. Consider the two probabilistic automata in Figure 4.5. Here the arcs indicate

Figure 4.5: Probabilistic automata representing a fair and an unfair coin.

probabilistic choice (as opposed to the nondeterministic choice in Figure 4.3), and probabilities are indicated next 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 4.5 by repeatedly running the machine until the display becomes empty and then restart it using the *on/off* 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 4.6. If we repeatedly run the trace distribution machine with this automaton inside, the ratio between the various traces does not need to converge to a fixed value. However, when 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:

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

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

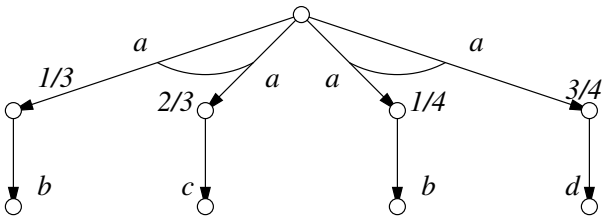

Figure 4.6: The combination of probabilistic and nondeterministic choice.

Even though our testing scenario is trivial, 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 chapter is a proof that the equivalence (preorder) on probabilistic automata induced by our testing scenario coincides with the trace distribution equivalence (preorder) proposed by Segala [Seg95a]. A technical result that we need to establish on the way is an *Approximation Induction Principle (AIP)* (cf. [BK86, BBK87]) for probabilistic processes. This principle says that if two finitely branching processes are equivalent up to any finite depth, then they are equivalent.

Being a first step, this chapter limits itself to a very 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 *on/off* 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 [Chr90a, Seg96, CDSY99, JY01]. These papers all consider testing equivalences and 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 can report success in some way, for instance via success state or success actions. When a test is run on a system, one can compute the probability on success 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 testing equivalent. For instance, two systems $\mathcal{A}$ and $\mathcal{B}$ are testing equivalent in the sense of [JY01] if and only if for all tests T the maximal probability on success in $\mathcal{A} \parallel T$ is less or equal to the maximal probability on success in $\mathcal{B} \parallel T$. The different equivalences and preorders in the mentioned papers arise by considering different kinds of probabilistic systems, by studying tests with different power (purely nondeterministic tests, finite trees or no restrictions) 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 the testing equivalences (preorders, respectively) they introduce in terms of trace distribution equivalence–like relations (trace

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

distribution inclusion–like relations), that is in terms of an extension of the trace equivalence relation (trace inclusion relation) to probabilistic processes.

Thus, the testing relations defined in the mentioned works are button pushing experiments in the sense that a test interacts with a system via synchronization on shared actions. We argue that these relations are not entirely observational because it is not described how the probability on success of a system can be observed.

Our work is most related to the paper by Larsen & Skou [LS91]. This chapter defines a notion of tests for reactive probabilistic processes (that is, processes in which all outgoing transitions of a state have different labels) which allow one to make arbitrary copies of any state. Using 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.)

Each test T in [LS91] gives rise to a set of observations O_T. Tests allow certain properties to be tested with arbitrary confidence $\alpha \in [0,1]$, the so–called level of significance. More precisely, a property Φ is said to be testable if for every level of significance α, there is a test T and a partition of the level of observations O_T into $(E_\Phi, O_T \setminus E_\Phi)$ satisfying the following. If Φ holds in a state s and T is run in s, then it is likely that we observe an element from E_Φ: $\mathbf{P}[E_\Phi] \geq 1 - \alpha$. On the other hand, if Φ does not hold, then the probability to observe an element in E_Φ is small: $\mathbf{P}[E_\Phi] \leq \alpha$. Thus, by checking whether the outcome of the test is in E_Φ or not, we can find out whether s satisfies Φ, where probability that the judgment is wrong is less than α. Using the terminology from hypothesis testing, Φ is the null hypothesis and E_Φ the critical section.

Then is it shown that two states in a system that satisfies the minimal derivation assumption are probabilistically bisimilar if and only if they satisfy exactly the same testable properties. Here the minimal derivation assumption requires that any probability occurring in the system is an integer multiple of some value ε. (Technically, this result is achieved via the probabilistic modal logic PML.) Thus, although not explicitly phrased in these terms, one can say Larsen & Skou present a button pushing scenario for probabilistic processes.

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 probability on a set of observations of a system under a test in [LS91] is uniquely determined. This is not the case for the observations we consider, which therefore give rise to a more generic notion of observation.
- The main result of this chapter, which is the characterization of trace distribution inclusion as a testing scenario, is established for all finitely branching systems, which is much less restrictive 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 chapter.

Overview Even though readers may not expect this after our informal introduction, the rest of this chapter is actually quite technical. We start in Section 4.2 with some mathematical

preliminaries concerning functions, sequences and probability theory. In Section 4.2.3 we recall the definitions of probabilistic automata and their behavior. Section 4.3 is entirely devoted to the proof of the AIP for probabilistic processes. Section 4.4, finally, presents the characterization of the testing preorder induced by the trace distribution machine as trace distribution inclusion.

4.2 Preliminaries

4.2.1 Functions

If f is a function, then we denote the domain of f by $Dom(f)$. The *range* of f, notation $Ran(f)$, is the set $\{f(x) \mid x \in X\}$. If X is a set, then the *restriction* of f to X, notation $f \restriction X$, is the function g with $Dom(g) = Dom(f) \cap X$ satisfying $g(x) = f(x)$ for each $x \in Dom(g)$. We say that a function f is a *subfunction* of a function g, and write $f \subseteq g$, if $Dom(f) \subseteq Dom(g)$ and $f = g \restriction Dom(f)$. A function f is called *finite* if $Dom(f)$ is finite.

4.2.2 Sequences

Let X be any set. A *sequence* over X is a function σ from a downward closed subset of the natural numbers to X. So the domain of a sequence is either the set $\mathbb{N}$ of natural numbers, or of the form $\{0, \ldots, k\}$, for some $k \in \mathbb{N}$, or the empty set. In the first case we say that the sequence is infinite, otherwise we say it is finite. The sets of finite and infinite sequences over X are denoted by X^* and X^∞, respectively. We will sometimes write σ_n rather than $\sigma(n)$. The symbol ε denotes the empty sequence, and the sequence containing one element $x \in X$ is denoted by x. Concatenation of a finite sequence with a finite or infinite sequence is denoted by juxtaposition. We say that a sequence σ is a *prefix* of a sequence ρ, denoted by $\sigma \sqsubseteq \rho$, if $\sigma = \rho \restriction Dom(\sigma)$. Thus $\sigma \sqsubseteq \rho$ if either $\sigma = \rho$, or σ is finite and $\rho = \sigma\sigma'$ for some sequence σ'. If σ is a nonempty sequence then $first(\sigma)$ denotes the first element of σ and, if σ is also finite, then $last(\sigma)$ denotes the last element of σ. Finally, $length(\sigma)$ denotes the length of a finite sequence σ. A *subsequence* of an infinite sequence σ is an infinite sequence ρ that is obtained by removing finitely or infinitely many elements from σ. Formally, ρ is a subsequence of σ if there is an *index function*, that is a function $\iota : \mathbb{N} \to \mathbb{N}$ such that (a) ι is strictly monotone (i.e., $n < m$ implies $\iota(n) < \iota(m)$), and (b) $\rho = \sigma \circ \iota$.

An elementary (but fundamental) result from Analysis is the following theorem from Bolzano–Weierstraß.

Theorem 4.2.1 (Bolzano–Weierstraß) *Every bounded infinite sequence in $\mathbb{R}^n$ has a convergent subsequence.*

Let f_0, f_1, f_2, ... be an infinite sequence of functions in $X \to [0,1]$, where X is a finite set. Then this sequence can be seen as a sequence over $[0,1]^n$, where n is the cardinality of X. Applying the Bolzano–Weierstraß Theorem to f_0, f_1, f_2 ... yields that this sequence has a convergent subsequence, i.e. there exists an index function ι such that $f_{\iota(0)}$, $f_{\iota(1)}$, $f_{\iota(2)}$, ... has a limit (in $[0,1]^n$).

4.2.3 Probabilistic Automata

In this chapter, we use the PA model introduced in Chapter 2. However, we want to consider only systems with a unique initial state and with external actions only, taken from a fixed, finite set Act. Moreover, we find it for technical reasons convenient to assume a special element $\delta \in Act$, referred to as the *halting action*. Hence, the automata in this chapter have the following structure. They are triples $\mathcal{A} = (S, s^0, \Delta)$ such that

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

It is clear that the PAs in this section are special cases of the ones introduced previously and, hence, all terminology from Chapter 2 also applies to the PAs in this chapter. The remainder of this section introduces some additional notation.

Definition 4.2.2 A PA $\mathcal{A}$ is *finitely branching* if for each state s, the set $\{(a, \mu, t) \mid s \overset{a,\mu}{\rightsquigarrow} t\}$ is finite.

Thus, each state in a finitely branching PA has finitely many outgoing transitions and the target distribution of each transition has a finite support.

Definition 4.2.3 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.

The adversaries we consider in this chapter differ slightly from those in Chapter 2. Rather than yielding the value $\bot$, the adversaries here schedule a δ–transition to indicate that the execution is interrupted. Technically, this has the advantage that the maximal paths of these adversaries are exactly their infinite paths. Moreover, these adversaries allow an easy definition of what it means to halt after k steps.

Definition 4.2.4 A *(partial, randomized) 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. Adversary E *halts* on a path π if it extends π with the halting transition, i.e.,

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

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 call E *finite* if it halts after k steps, for some $k \in \mathbb{N}$.

The adversaries in this chapter are special cases of those in Chapter 2. Hence, each of the adversaries in this chapter can be assigned a probability space $(\Omega_E, \mathcal{F}_E, \mathbf{P}_E)$ and a trace distribution $(\Omega_H, \mathcal{F}_H, \mathbf{P}_H)$ as before, see Definition 2.3.13 on page 46 and Definition 2.3.16 on page 48 respectively.

This gives rise to a second notion of trace distribution for PAs: to define the set of trace distributions in a PA, one can either use the adversaries or the adversaries defined in this chapter. However, it is easy to see that they induce exactly the same notion of trace distribution inclusion. In the sequel, also the function $\mathbf{Q}^E(\pi)$, yielding the probability on the path π in the adversary E, plays an important role. Hence, the reader may have another look at Definition 2.3.10 on page 45.

In the next section, we focus on trace distributions generated from adversaries (those from Definition 4.2.4) that halt after k steps, for a given k. Hence, we introduce the following notation.

Notation 4.2.5 For $k \in \mathbb{N} \cup \{\infty\}$, denote the set of all paths of $\mathcal{A}$ of length k by $Path^k(\mathcal{A})$. The set of trace distributions of adversaries of $\mathcal{A}$ which halt after k steps is denoted $trd(\mathcal{A}, k)$. If $trd(\mathcal{A}, k) \subseteq trd(\mathcal{B}, k)$, then we write $\mathcal{A} \sqsubseteq^k_{\mathrm{TD}} \mathcal{B}$.

It is important to realize that for adversary E that halts after k steps, $\mathbf{P}_E$ is fully determined by $\mathbf{Q}^E \restriction Path^k(\mathcal{A})$, i.e., the probability on the paths of length k.

Lemma 4.2.6 *Let $k \in \mathbb{N}$, let $\mathcal{A}$ be a finitely branching PA and let E be an adversary of $\mathcal{A}$ that halts after k steps. Then E can be written as a convex combination of deterministic adversaries that halt after k steps, i.e., there exists a probability distribution ν over the set $\mathcal{D}$ of deterministic adversaries of $\mathcal{A}$ that halt after k steps such that, for all π, a and μ,*

$$E(\pi)(a, \mu) \quad = \quad \sum_{D \in \mathcal{D}} \nu(D) \cdot D(\pi)(a, \mu).$$

PROOF: Since $\mathcal{A}$ is finitely branching, each adversary that halt after k steps is in fact a finite adversary. Hence, this result is a reformulation of Proposition 5.3.24(1). □

4.3 The Approximation Induction Principle

This section is entirely devoted to a proof of an *Approximation Induction Principle (AIP)* (cf. [BK86, BBK87]) for probabilistic processes. We need this result to characterize the equivalence on probabilistic automata induced by the trace distribution machine in Section 4.4. The result and an informal proof sketch have also in appeared in [Seg96]. The proof below provides an explicit construction of the adversary needed and shows that this adversary meets the desired properties, thus filling in several nontrivial details which were not proven in [Seg96].

Theorem 4.3.1 (Approximation Induction Principle) *Let $\mathcal{A}$ and $\mathcal{B}$ be PAs and let $\mathcal{B}$ be finitely branching. Then*

$$\forall k[\mathcal{A} \sqsubseteq^k_{\mathrm{TD}} \mathcal{B}] \implies \mathcal{A} \sqsubseteq_{\mathrm{TD}} \mathcal{B}.$$

PROOF: Assume that $\mathcal{A} \sqsubseteq^k_{\mathrm{TD}} \mathcal{B}$, for all k. In order to prove $\mathcal{A} \sqsubseteq_{\mathrm{TD}} \mathcal{B}$, let H be a trace distribution of $\mathcal{A}$ and let E be an adversary of $\mathcal{A}$ with $H = trdistr(E)$. Via a number of sublemma's, we prove that $H \in trdistr(\mathcal{B})$.

For each $k \in \mathbb{N}$, define E_k by

$$E_k(\pi) = \begin{cases} E(\pi) & \text{if } |\pi| < k, \\ \{(\delta, \{\bot \mapsto 1\}) \mapsto 1\} & \text{otherwise.} \end{cases}$$

Clearly, E_k is an adversary of $\mathcal{A}$ that equals E on paths with length less than k and halts after k steps. Therefore $E_k \in \mathit{trdistr}(\mathcal{A}, k)$. By assumption, there is an adversary F_k of $\mathcal{B}$ such that $\mathit{trdistr}(E_k) = \mathit{trdistr}(F_k)$. We view F_k as a function in

$$\mathit{Path}^*(\delta\mathcal{B}) \times \mathit{Act} \times \mathrm{Distr}(S_{\delta\mathcal{B}}) \to [0,1].$$

We will construct an adversary G of $\mathcal{B}$ with $\mathit{trdistr}(G) = H$ from the sequence of functions $F = F_0, F_1, F_2 \ldots$. The idea is that, since only the paths of length k matter, F_k is essentially a finite function and we can use the Bolzano–Weierstraß Theorem to obtain G as the limit of a convergent subsequence of F. However, this theorem cannot be applied immediately, because the finite domains of these functions are growing. Therefore, we will operate in several stages. The basic idea is to construct at stage $n+1$ a convergent subsequence with index function ι_{n+1} of $F_0^n, F_1^n, F_2^n \ldots$, where F_k^n is the restriction of F_k to paths of length n. This sequence consists of finite functions with the same, finite domain and a bounded range (viz. $[0,1]$) and has therefore a convergent subsequence. We define G_n as the limit of ι_n. Thus, we will obtain a sequence of increasing subfunctions $G_1 \subseteq G_2 \subseteq G_3 \ldots$ and we take G to be its limit. We will need several technical lemma's to ensure that everything is as expected and to prove finally that $\mathit{trdistr}(G) = \mathit{trdistr}(E)$.

Throughout this proof, we use the following notations.

$$\begin{aligned} P_n &= \mathit{Path}^n(\delta\mathcal{B}) \\ D_n &= \{\mu \in \mathrm{Distr}(S_{\delta\mathcal{B}}) \mid \mu \text{ occurs in some } \pi \in P_{n+1}\} \end{aligned}$$

$$\begin{aligned} P &= \mathit{Path}^*(\delta\mathcal{B}) \\ D &= \mathrm{Distr}(S_{\delta\mathcal{B}}) \end{aligned}$$

Note that $P_n \subseteq P_{n+1}$, $D_n \subseteq D_{n+1}$, $P = \cup_n P_n$ and $D \supseteq \cup_n D_n$. In fact, D may contain distributions that are not contained in any D_n. Observe also that $\pi \in P_n$ and $\pi \overset{a,\mu}{\rightsquigarrow} s$ implies $\mu \in D_n$. Since $\mathcal{B}$ is finitely branching, there are only finitely many paths of length at most n and hence P_n and D_n are both finite. Recall that Act is finite by definition. Therefore, the following function F_k^n is finite:

$$\begin{aligned} F_k^n &: P_n \times \mathit{Act} \times D_n \to [0,1] \\ F_k^n &= F_k \restriction P_n \times \mathit{Act} \times D_n. \end{aligned}$$

Claim 1 $F_k^n \subseteq F_k^{n+1}$ for all k, n.

PROOF: Easy verification. ⊠

For each n, let ρ_n be the sequence

$$\rho_n \;=\; F_0^n \;\; F_1^n \;\; F_2^n \;\; F_3^n \ldots$$

and let ι_n be the index function defined inductively as follows:

- ι_0 is the identity function.
- Let ι be the index function of a convergent subsequence of $\rho_n \circ \iota_n$ (such a subsequence exists by the Bolzano–Weierstraß Theorem). Then $\iota_{n+1} = \iota_n \circ \iota$.

We define function $G_n : P_n \times Act \times D_n \to [0,1]$ by

$$G_n = \lim(\rho_n \circ \iota_{n+1}),$$

i.e. , G_n is the limit of the convergent subsequence just defined. Note that G_n is in fact the point wise limit of a sequence of functions. (It needs not to be an adversary).

Claim 2 $G_n \subseteq G_{n+1}$.

PROOF: Clearly, $\mathrm{Dom}(G_n) \subseteq \mathrm{Dom}(G_{n+1})$. Let $(\pi, a, \mu) \in \mathrm{Dom}(G_n)$. Then

$$\begin{aligned}
G_n(\pi)(a,\mu) &= \lim_{k\to\infty} F^n_{\iota_{n+1}(k)}(\pi)(a,\mu) && \{\mathrm{Ran}(\iota_{n+2}) \subseteq \mathrm{Ran}(\iota_{n+1}\} \\
&= \lim_{k\to\infty} F^n_{\iota_{n+2}(k)}(\pi)(a,\mu) && \{\text{Claim 1}\} \\
&= \lim_{k\to\infty} F^{n+1}_{\iota_{n+2}(k)}(\pi)(a,\mu) \\
&= G_{n+1}(\pi)(a,\mu).
\end{aligned}$$

⊠

Let $G' = \cup_n G_n$, i.e. for $\pi \in P_n$ and $\mu \in D_n$, $G'(\pi)(A,\mu) = G_n(\pi)(a,\mu)$. Then G' is a function in $\cup_n P_n \times Act \times D_n \to [0,1]$. We extend G' to a function G in $P \times Act \times D \to [0,1]$ as follows

$$G(\pi)(a,\mu) = \begin{cases} G'(\pi)(a,\mu) & \text{if } \exists n : \pi \in P_n \wedge \mu \in D_n, \\ 0 & \text{otherwise.} \end{cases}$$

The rest of this proof is concerned with showing that G is an adversary with $trdistr(G) = H$, which is exactly what we are after.

Claims 3 and 4 show that G is an adversary of $\mathcal{B}$. Claim 3 proves that G respects the transition relation of $\delta\mathcal{B}$, and Claim 4 establishes that G has the required type, i.e.

$$G : Path^*(\delta\mathcal{B}) \to \mathrm{Distr}(Act \times \mathrm{Distr}(S_{\delta\mathcal{B}})).$$

Claim 3 Suppose $\pi \in P$, $a \in Act$, $\mu \in D$ and $G(\pi)(a,\mu) > 0$. Then $last(\pi) \xrightarrow{a} \mu$ is a transition of $\delta\mathcal{B}$.

PROOF: Since $G(\pi)(a,\mu) > 0$, it follows from the definition of G that $G(\pi)(a,\mu) = G'(\pi)(a,\mu)$. Hence, by definition of G', there is an n such that $G'(\pi)(a,\mu) = G_n(\pi)(a,\mu)$. Then

$$\begin{aligned}
0 < G_n(\pi)(a,\mu) & && \{\text{def. } G_n\} \\
= \lim_{k\to\infty} F^n_{\iota_{n+1}(k)}(\pi)(a,\mu) & && \{\text{def. } F^n_i\} \\
= \lim_{k\to\infty} F_{\iota_{n+1}(k)}(\pi)(a,\mu)
\end{aligned}$$

This implies that $F_{\iota_{n+1}(k)}(\pi)(a,\mu) > 0$ for large k. Since $F_{\iota_{n+1}(k)}$ is an adversary of $\mathcal{B}$, $last(\pi) \xrightarrow{a} \mu$ is a transition of $\mathcal{B}$. ⊠

Claim 4 For all $\pi \in P$, $\sum_{a \in Act, \mu \in D} G(\pi)(a,\mu) = 1$.

PROOF: Choose $\pi \in P$ and let $n = |\pi|$. Then $\pi \in P_n$ and

$$\begin{aligned}
&\sum_{a \in Act, \mu \in D} G(\pi)(a,\mu) && \{\text{def. } G\}\\
&= \sum_{a \in Act, \mu \in D_n} G'(\pi)(a,\mu) && \{\text{def. } G'\}\\
&= \sum_{a \in Act, \mu \in D_n} G_n(\pi)(a,\mu) && \{\text{def. } G_n\}\\
&= \sum_{a \in Act, \mu \in D_n} \lim_{k\to\infty} F^n_{\iota_{n+1}(k)}(\pi)(a,\mu) && \{\text{def. } F^n_i\}\\
&= \sum_{a \in Act, \mu \in D_n} \lim_{k\to\infty} F_{\iota_{n+1}(k)}(\pi)(a,\mu) && \{\text{finite sum}\}\\
&= \lim_{k\to\infty} \sum_{a \in Act, \mu \in D_n} F_{\iota_{n+1}(k)}(\pi)(a,\mu) && \{\text{def. } Act, D_n\}\\
&= \lim_{k\to\infty} \sum_{a \in Act, \mu \in D} F_{\iota_{n+1}(k)}(\pi)(a,\mu) && \{F_i \text{ adversary}\}\\
&= \lim_{k\to\infty} 1\\
&= 1
\end{aligned}$$

⊠

Note that the following claim concerns G and F_i, which are adversaries. In contrast, G_n and F^n_k are just functions, not adversaries.

Claim 5 $\mathbf{Q}^G(\pi) = \lim_{k\to\infty} \mathbf{Q}^{F_{\iota(n)(k)}}(\pi)$ for all $\pi \in Path^*(\delta\mathcal{B})$ with $|\pi| = n$.

PROOF: By induction on n.

- Then case $n = 1$ follows immediately from the fact that $\mathbf{Q}^E(s_0) = 1$ for all adversaries E.
- Case $n+1$. Let π' be a path of length $n+1$ and write $\pi' = \pi \overset{a,\mu}{\rightsquigarrow} s$. Then $\pi \in P_n$, $a \in Act$, $\mu \in D_n$ and

$$\begin{aligned}
&\mathbf{Q}^G(\pi')\\
&= \mathbf{Q}^G(\pi \overset{a,\mu}{\rightsquigarrow} s) && \{\text{def. } \mathbf{Q}\}\\
&= \mathbf{Q}^G(\pi) \cdot G(\pi)(a,\mu) \cdot \mu(s) && \{\text{IH, } |\pi| = n\}\\
&= \lim_{k\to\infty} \mathbf{Q}^{F_{\iota(n)(k)}}(\pi) \cdot G_n(\pi)(a,\mu) \cdot \mu(s) && \{\text{def. } G_n\}\\
&= \lim_{k\to\infty} \mathbf{Q}^{F_{\iota(n)(k)}}(\pi) \cdot \lim_{k\to\infty} F_{\iota(n)(k)}(\pi)(a,\mu) \cdot \mu(s)\\
&= \lim_{k\to\infty} \mathbf{Q}^{F_{\iota(n)(k)}}(\pi) \cdot F_{\iota(n)(k)}(\pi)(a,\mu) \cdot \mu(s) && \{\text{def. } \mathbf{Q}\}\\
&= \lim_{k\to\infty} \mathbf{Q}^{F_{\iota(n+1)(k)}}(\pi \overset{a,\mu}{\rightsquigarrow} s)\\
&= \lim_{k\to\infty} \mathbf{Q}^{F_{\iota(n+1)(k)}}(\pi')
\end{aligned}$$

$\boxtimes$

Claim 6 $\mathbf{Q}^E(\pi) = \lim_{k\to\infty} \mathbf{Q}^{E_{\iota(n)(k)}}(\pi)$ for all n and π.

PROOF: Since $\iota(n)$ is an index function, we have $\lim_{k\to\infty} \iota(n)(k) = \infty$ and therefore $E_{\iota(n)(k)}[\pi] = E[\pi]$ for $\iota(n)(k) \geq |\pi|$. So, $\lim_{k\to\infty} \mathbf{Q}^{E_{\iota(n)(k)}}(\pi) = \mathbf{Q}^E(\pi)$. $\boxtimes$

The following is an immediate consequence of the previous claim.

Claim 7 $\mathbf{P}_E[C_\alpha] = \lim_{k\to\infty} \mathbf{P}_{E_{\iota(n)(k)}}[C_\alpha]$, for all α.

Claim 8 $trdistr(G) = trdistr(E)$.

PROOF: Let $H_1 = trdistr(G)$ and $H_2 = trdistr(E)$. It suffices to show that $\mathbf{P}_{H_1}[C_\alpha] = \mathbf{P}_{H_2}[C_\alpha]$ for all $\alpha \in Act^*$. Let $n = |\alpha|$.

$$
\begin{aligned}
\mathbf{P}_{H_1}[C_\alpha] &= \sum_{\pi\,|\,trace(\pi)\in C_\alpha} \mathbf{P}_G[\pi] \\
&= \sum_{\pi\,|\,trace(\pi)=\alpha\wedge|\pi|=n} \mathbf{P}_G[C_\pi] && \{\text{def. } C_\pi\} \\
&= \sum_{\pi\,|\,trace(\pi)=\alpha\wedge|\pi|=n} \mathbf{Q}^G(\pi) && \{\text{Claim 5, } |\pi|\text{=n}\} \\
&= \sum_{\pi\,|\,trace(\pi)=\alpha\wedge|\pi|=n} \lim_{k\to\infty} \mathbf{Q}^{F_{\iota(n)(k)}}(\pi) && \{\text{finite sum}\} \\
&= \lim_{k\to\infty} \sum_{\pi\,|\,trace(\pi)=\alpha\wedge|\pi|=n} \mathbf{Q}^{F_{\iota(n)(k)}}(\pi) && \{\text{def. } C_\alpha, \mathbf{P}_{F_i}\} \\
&= \lim_{k\to\infty} \mathbf{P}_{F_{\iota(n)(k)}}[C_\alpha] && \{trdistr(F_i) = trdistr(E_i)\} \\
&= \lim_{k\to\infty} \mathbf{P}_{E_{\iota(n)(k)}}[C_\alpha] && \{\text{Claim 7}\} \\
&= \mathbf{P}_E[C_\alpha]
\end{aligned}
$$

Note that the set $\{\pi \mid trace(\pi) = \alpha \wedge |\pi| = n\}$ is finite, because $\mathcal{A}$ is finitely branching and hence the summations above are all finite. $\boxtimes$

$\square$

4.4 Characterization of Testing Preorder

The operational behavior of trace distribution machines described in Section 4.1 is specified accurately by the notion of a (partial, randomized) adversary, introduced in Definition 4.2.4. A trace distribution machine has to choose an execution path within some probabilistic automaton $\mathcal{A}$. For a given state, the choice which transition to take may depend on the execution history. Also, a probabilistic mechanism may be used to resolve the choice between the various transitions that are available. At any point during execution, there is a certain probability

that the environment (the person doing the experiments) turns the machine off, i.e., aborts the execution via a halting transition.

Define an *observation 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. An observation describes what an observer potentially may 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 observation O a function which assigns to each sequence of actions of length k the number of times it occurs in O divided by m. Note that $freq(O)$ is a probability distribution . We will base our statistical analysis on $freq(O)$ rather than just O. This means we ignore some of the information contained in observations, information that more advanced statistical methods may want to explore. If, for instance, we consider the observation 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 an observation of a trace distribution machine implementing a fair coin. However, the frequency function $freq(O)$ can very well be generated by a fair coin.

Now choose a level of significance $\alpha \in (0,1)$. Let $E_1, \ldots, E_m$ be adversaries of $\mathcal{A}$ that halt after k steps. Let H_0 be the null hypothesis, which states that O has been generated using adversaries $E_1, \ldots, E_m$. We define the critical region $K_{E_1,\ldots,E_m}$ to be the complement of the largest hypercube K around the expected value for $freq(O)$, assuming H_0 and with $\mathbf{P}_{H_0}[K] \leq \alpha$.[3] Now we come to one of the key definitions of this chapter.

Definition 4.4.1 We say that O is an *observation of* $\mathcal{A}$ if

$$\exists E_1, \ldots, E_m \in Adv(\mathcal{A}, k).freq(O) \notin K_{E_1,\ldots,E_m}$$

Here $Adv(\mathcal{A}, k)$ denotes the set of adversaries of $\mathcal{A}$ that halt after k steps. We write $Obs(\mathcal{A})$ for the set of observations of $\mathcal{A}$ (for some depth k and width m).

Thus we only include those observations for which it is likely (given the chosen level of significance) that they are generated by some adversaries of $\mathcal{A}$. If O is an observation of $\mathcal{A}$ then its frequency distribution is contained in the hypercube with probability mass α surrounding the expected value of the frequency distribution for a (well-chosen) series of adversaries of $\mathcal{A}$. We can now prove our main characterization theorem.

Theorem 4.4.2 $Obs(\mathcal{A}) \subseteq Obs(\mathcal{B}) \;\Leftrightarrow\; \mathcal{A} \sqsubseteq_{\mathrm{TD}} \mathcal{B}$.

PROOF: (Sketch)
"$\Rightarrow$" Assume that $\mathcal{A} \not\sqsubseteq_{\mathrm{TD}} \mathcal{B}$. We show that $Obs(\mathcal{A}) \not\subseteq Obs(\mathcal{B})$. By Theorem 4.3.1, there exists a k such that $\mathcal{A} \not\sqsubseteq^k_{\mathrm{TD}} \mathcal{B}$. This means $trd(\mathcal{A}, k) \not\subseteq trd(\mathcal{B}, k)$. Let H be a trace distribution in $trd(\mathcal{A}, k)$ that is not a trace distribution in $trd(\mathcal{B}, k)$ and which is generated by an adversary E of $\mathcal{A}$ that halts after k steps. Using Lemma 4.2.6 we may assume, without loss of generality that E is a deterministic adversary. Also, by the same lemma, we may assume that adversaries that generate traces in $trd(\mathcal{B}, k)$ can be written as convex combinations of (a finite number of) deterministic adversaries. This means that there

[3] This choice of using hypercubes is actually quite impractical, but serves well the theoretical aims we have for this chapter.

is minimal distance d between H and *any* trace distribution in $trd(\mathcal{B}, k)$. By choosing m sufficiently large we obtain, using Chebychev's inequality, $\bar{K}_{E_1,\ldots,E_m} \cap \bar{K}_{F_1,\ldots,F_m} = \emptyset$, for $E_i = E$ and $F_1, \ldots, F_m$ arbitrary adversaries from $Adv(\mathcal{B}, k)$. In addition, we may assume that there exists an observation O with $freq(O) \in \bar{K}_{E_1,\ldots,E_m}$. Clearly, O is an observation of $\mathcal{A}$. However, by construction, O is not an observation of $\mathcal{B}$.

"$\Leftarrow$" Assume that $\mathcal{A} \sqsubseteq_{\mathrm{TD}} \mathcal{B}$. Suppose O is an observation of $\mathcal{A}$ with depth k and width m. Then, by definition, there exist adversaries $E_1, \ldots, E_m \in Adv(\mathcal{A}, k)$ such that $freq(O) \notin K_{E_1,\ldots,E_m}$. By the assumption, we can choose adversaries $F_1, \ldots, F_m \in Adv(\mathcal{B}, k)$ with, for all i, $trdistr(E_i) = trdistr(F_i)$. Since the critical region $K_{E_1,\ldots,E_m}$ can actually be computed from $trdistr(E_1), \ldots, trdistr(E_m)$ (and similarly for $K_{F_1,\ldots,F_m}$), it follows that $K_{E_1,\ldots,E_m} = K_{F_1,\ldots,F_m}$. Hence it follows that $O \in Obs(\mathcal{B})$, as required. □

Acknowledgement The ideas worked out in this chapter 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.