# Coherent Resolutions of Nondeterminism

Marco Bernardo

Dipartimento di Scienze Pure e Applicate, Università di Urbino, Italy
marco.bernardo@uniurb.it

**Abstract.** We study the impact that different ways of resolving nondeterminism within probabilistic automata have on the properties of probabilistic behavioral equivalences. Firstly, we provide a uniform definition of structure-preserving and structure-modifying resolutions of nondeterminism, respectively generated by different families of schedulers. Secondly, we exhibit a number of anomalies arising from the excessive power of the various families of schedulers, which affect the discriminating power, the compositionality, and the backward compatibility of probabilistic trace equivalence. Thirdly, we propose to remove those anomalies by enforcing coherency within resolutions of nondeterminism. This ensures that a scheduler cannot select different continuations in equivalent states of an automaton, so that also the states to which they correspond in any resolution of the automaton have equivalent continuations.

**Keywords:** Probabilistic automata · Schedulers · Equivalences.

## 1 Introduction

Quantitative models of computing systems describe the order in which activities are executed – possibly admitting nondeterminism in case of concurrency phenomena or to support implementation freedom – and include information about the probabilities or the timing of the activities themselves. A particularly expressive model is given by *probabilistic automata* [22], as they encompass fully nondeterministic models like labeled transition systems [18], fully probabilistic models like action-labeled variants of discrete-time Markov chains [19], and reactive probabilistic models like Markov decision processes [11].

Behavioral relations play a fundamental role in the analysis of quantitative models. They formalize observational mechanisms that permit relating models that, despite their different representations in the same mathematical domain, cannot be distinguished by external entities when abstracting from details deemed unimportant for specific purposes. Moreover, they support system modeling and verification by providing a means to relate system descriptions expressed at different levels of abstraction, as well as to reduce the size of a system representation while preserving specific properties to be assessed later.

In the case of fully nondeterministic models, from the first comparative work [8] to the elaboration of the full spectrum [13], a number of equivalences have emerged that range from the branching-time – i.e., (bi)simulation-based –

endpoint [21] to the linear-time – i.e., trace-based – endpoint [7] passing through testing relations [9]. The spectrum becomes simpler when considering fully probabilistic models [17,14,1], whereas as shown in [4] it is much more variegated in the case of models with nondeterminism and probabilities like probabilistic automata. The reason is that the probability of equivalence-specific events can be calculated only after removing nondeterminism. Examples of such events are reaching via given actions certain sets of equivalent states (bisimulation semantics) or executing specific action sequences (trace semantics), with states/traces being possibly decorated with additional information.

In this paper, we study the impact on the discriminating power, the compositionality, and the backward compatibility of behavioral equivalences for nondeterministic and probabilistic models, due to the different ways of resolving nondeterminism. We restrict ourselves to *simple* probabilistic automata [22], i.e., state-transition graphs where each transition is labeled with an action and goes from a state to a probability distribution over states. In this model, nondeterminism is expressed by the presence of *several* transitions departing from the same state. A *resolution of nondeterminism* is obtained by applying a *scheduler* that decides which activity has to be performed next, where by activity we mean executing a transition or stopping the execution altogether.

The first contribution of this paper is a discussion of different families of schedulers, with the result of providing a uniform way, based on *correspondence functions*, of defining the resolutions induced by those schedulers.

We divide resolutions into *structure preserving* and *structure modifying*, depending on whether they respect or alter the structure of the automaton from which they are obtained. A structure-preserving resolution is produced by a *deterministic scheduler*, which selects at the current state one of the transitions departing from that state or no transitions at all. A structure-modifying resolution is derived via a *randomized scheduler* [22], which probabilistically combines the transitions departing from the current state, or an *interpolating scheduler* [10], which splits the current state into copies, each having at most one outgoing transition, whose probabilities sum up to the probability of the original state. We formalize any resolution as a fully probabilistic automaton, which we equip with a correspondence function from the acyclic state space of the resolution to the possibly cyclic state space of the original automaton, as done for the first time in [15] for deterministic schedulers.

The second contribution of this paper is the presentation of a number of anomalies affecting probabilistic behavioral equivalences, mostly arising under deterministic schedulers, together with a proposal for avoiding them based on limiting the excessive power of schedulers.

We focus on probabilistic trace equivalence by showing that it does not contain probabilistic bisimilarity, it is not a congruence with respect to action prefix, and it is not backward compatible with its version for fully probabilistic models. The reason is that schedulers have the freedom to make *different* decisions in *equivalent* states occurring in the target distribution of a transition, with these decisions being not necessarily replicable in equivalent distributions of distinct

automata. This is especially true for deterministic schedulers, as the resolutions they induce must be structure preserving.

Such anomalies can be avoided by employing *coherent resolutions* in the definition of probabilistic trace equivalence. The idea is that, if several states in the target distribution of a transition are equivalent, then the states to which they correspond in a resolution must be equivalent as well. This constraint can be formalized by reasoning on trace distributions, i.e., families of sets of traces each endowed with its execution probability in a given resolution.

This paper is organized as follows. In Sect. 2, we recall simple probabilistic automata. In Sect. 3, we discuss different notions of resolution usable in probabilistic behavioral equivalences and provide a uniform way of defining all of them. In Sect. 4, we illustrate the aforementioned anomalies of probabilistic trace equivalence caused by the excessive power of schedulers. In Sect. 5, we show how to avoid those anomalies by forcing resolutions to be coherent. Finally, in Sect. 6 we present some concluding remarks.

## 2 Nondeterministic and Probabilistic Models

We formalize systems featuring nondeterminism and probabilities through a variant of simple probabilistic automata [22], in which we do not distinguish between external and internal actions.

**Definition 1.** *A* nondeterministic and probabilistic labeled transition system*, NPLTS for short, is a triple* $(S, A, \longrightarrow)$ *where* $S \neq \emptyset$ *is an at most countable set of states,* $A \neq \emptyset$ *is a countable set of transition-labeling actions, and* $\longrightarrow \subseteq S \times A \times Distr(S)$ *is a transition relation with* $Distr(S)$ *being the set of discrete probability distributions over* $S$*.* ■

A transition $(s, a, \Delta)$ is written $s \xrightarrow{a} \Delta$. We say that $s' \in S$ is not reachable from $s$ via that $a$-transition if $\Delta(s') = 0$, otherwise we say that it is reachable with probability $p = \Delta(s')$. The reachable states form the support of $\Delta$, i.e., $supp(\Delta) = \{s' \in S \mid \Delta(s') > 0\}$. An NPLTS can be depicted as a directed graph in which vertices represent states and action-labeled edges represent transitions, with states in the same support being linked by a dashed line and decorated with the respective probabilities (see the forthcoming Figs. 1 to 9).

An NPLTS represents (i) a *fully nondeterministic* process when every transition has a target distribution with a singleton support, (ii) a *fully probabilistic* process when every state has at most one outgoing transition, or (iii) a Markov decision process when for each action any state has at most one outgoing transition labeled with that action implying the absence of *internal nondeterminism*.

**Definition 2.** *Let* $\mathcal{L} = (S, A, \longrightarrow)$ *be an NPLTS and* $s, s' \in S$*. We say that the finite sequence:*

$$c \equiv s_0 \xrightarrow{a_1} s_1 \xrightarrow{a_2} s_2 \ldots s_{n-1} \xrightarrow{a_n} s_n$$

*is a* computation *of* $\mathcal{L}$ *of length* $n \in \mathbb{N}$ *from* $s = s_0$ *to* $s' = s_n$ compatible *with trace* $\alpha = a_1 a_2 \ldots a_n \in A^*$*, written* $c \in \mathcal{CC}(s, \alpha)$*, iff for all* $i = 1, \ldots, n$ *there exists in* $\mathcal{L}$ *a transition* $s_{i-1} \xrightarrow{a_i} \Delta_i$ *such that* $s_i \in supp(\Delta_i)$*, with:*

- $\Delta_i(s_i)$ *being the execution probability of step* $s_{i-1} \stackrel{a_i}{\longmapsto} s_i$ *conditioned on the selection of transition* $s_{i-1} \stackrel{a_i}{\longrightarrow} \Delta_i$ *at state* $s_{i-1}$*, or simply the execution probability of that step if* $\mathcal{L}$ *is fully probabilistic;*
- $prob(c) = \prod_{1 \leq i \leq n} \Delta_i(s_i)$ *being the execution probability of* $c$ *if* $\mathcal{L}$ *is fully probabilistic, assuming that* $prob(c) = 1$ *when* $n = 0$*;*
- $prob(C) = \sum_{c \in C} prob(c)$ *if* $\mathcal{L}$ *is fully probabilistic, provided that none of the computations in* $C$ *is a proper prefix of one of the others.* ■

## 3 An Overview of Resolutions of Nondeterminism

When several transitions depart from the same state $s$ of an NPLTS $\mathcal{L}$, they describe a nondeterministic choice among different behaviors. Eliminating these choices is necessary to perform the calculations required by probabilistic behavioral equivalences. A *resolution* of $s$ is the result of a possible way of resolving nondeterministic choices starting from $s$, as if a *scheduler* were applied that decides which activity has to be performed next. A resolution of nondeterminism can thus be formalized as a *fully probabilistic* NPLTS $\mathcal{Z}$ with a *tree-like structure*, whose branching points correspond to target distributions of transitions deriving from those of $\mathcal{L}$.

We now present an overview of various ways of resolving nondeterminism, with the result of providing a uniform technique for defining all of them based on correspondence functions, so to facilitate their comparison. In Sects. 3.1 to 3.3 we address the notions of resolution stemming from two different approaches, respectively preserving or modifying the structure of the original NPLTS. The idea underlying the former approach is to construct a resolution by *importing states and transitions* from the original model. The idea at the basis of the latter approach is that (i) a transition of a resolution can be produced by *probabilistically combining transitions* of the original model, or (ii) a state of a resolution can be obtained by *probabilistically splitting states* of the original model.

### 3.1 Structure-Preserving Resolutions via Deterministic Schedulers

A *deterministic scheduler* selects one of the transitions departing from the current state or no transitions at all thus stopping the execution. As a consequence, the resulting resolution is isomorphic to a submodel of the original model (or of its unfolding, should cycles be present), thereby *preserving* the structure of the original model (or of its unfolding). If the model is fully nondeterministic, each of its resolutions coincides with a computation of the model; if the model is fully probabilistic, its maximal resolution coincides with the entire model.

In [26] a resolution was defined as a maximal subtree of the unfolding of the considered model – with the unfolding yielding a potentially infinite tree – in which every state has at most one outgoing transition. Resolutions were defined as fully probabilistic maximal subtrees also in [16], but the considered models were finite trees in lieu of directed graphs. Subtree maximality was required just because of the focus of those works on testing semantics.

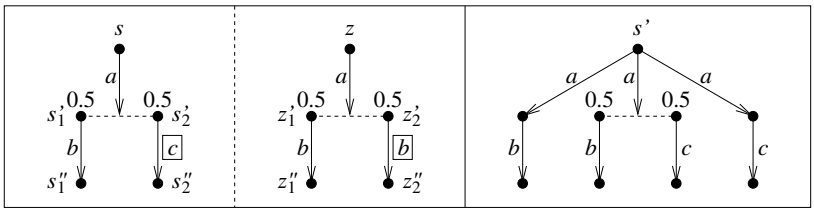


**Fig. 1.** Lack of injectivity breaks structure preservation

The paper [15], instead of reasoning in terms of unfoldings and submodels, introduced for the first time a *correspondence function* $corr_{\mathcal{Z}} : Z \to S$ from the acyclic state space of the resolution $\mathcal{Z} = (Z, A, \longrightarrow_{\mathcal{Z}})$ being built, to the possibly cyclic state space of the considered model $\mathcal{L} = (S, A, \longrightarrow)$. This function had to satisfy the following constraint on transitions: if $z \xrightarrow{a}_{\mathcal{Z}} \Delta$ then $corr_{\mathcal{Z}}(z) \xrightarrow{a} \Gamma$, with $\Delta(z') = \Gamma(corr_{\mathcal{Z}}(z'))$ for all $z' \in supp(\Delta)$.

The correspondence function with its constraint as defined in [15] and reused in [3,4] has the drawback of not being structure preserving in the case that the target distribution of a transition assigns the same probability to several inequivalent states. Let us see for instance the three NPLTS models in Fig. 1. The correspondence function that maps $z$ to $s$, $z'_1$ and $z'_2$ to $s'_1$, and $z''_1$ and $z''_2$ to $s''_1$ causes the central NPLTS to be considered a legal resolution of the leftmost NPLTS, although the former is not isomorphic to any submodel of the latter. This may have no consequences on the discriminating power of testing equivalences, the subject of [15], if all transitions of testing systems are identically labeled. However, it would lead to consider the leftmost NPLTS and the rightmost NPLTS as trace equivalent, because also the leftmost one would have a resolution in which trace $a\,b$ (resp. trace $a\,c$) is executable with probability 1.

The constraint was rectified in [5] by requiring the *injectivity* of $corr_{\mathcal{Z}}$ over $supp(\Delta)$, so that in Fig. 1 $z'_1$ and $z'_2$ can no longer be both mapped to $s'_1$. We also point out that in [2] it was further observed that *bijectivity* between $supp(\Delta)$ and $supp(\Gamma)$, rather than injectivity, is necessary to preserve the overall reachability mass in more general settings like the ULTraS metamodel where, unlike the probabilistic case, there is no predefined value like 1 for the reachability mass of the target of a transition.

Below is the rectified definition of [5] in the style of [15], i.e., based on a correspondence function from the acyclic state space of the resolution to the possibly cyclic state space of the considered model.

**Definition 3.** *Let $\mathcal{L} = (S, A, \longrightarrow)$ be an NPLTS and $s \in S$. An acyclic NPLTS $\mathcal{Z} = (Z, A, \longrightarrow_{\mathcal{Z}})$ is a* structure-preserving resolution *of $s$, written $\mathcal{Z} \in Res_{\rm sp}(s)$, iff there exists a correspondence function $corr_{\mathcal{Z}} : Z \to S$ such that $s = corr_{\mathcal{Z}}(z_s)$, for some $z_s \in Z$, and for all $z \in Z$ it holds that:*

- *If $z \xrightarrow{a}_{\mathcal{Z}} \Delta$ then $corr_{\mathcal{Z}}(z) \xrightarrow{a} \Gamma$, with $corr_{\mathcal{Z}}$ being injective over $supp(\Delta)$ and satisfying $\Delta(z') = \Gamma(corr_{\mathcal{Z}}(z'))$ for all $z' \in supp(\Delta)$.*
- *At most one transition departs from $z$.* ■

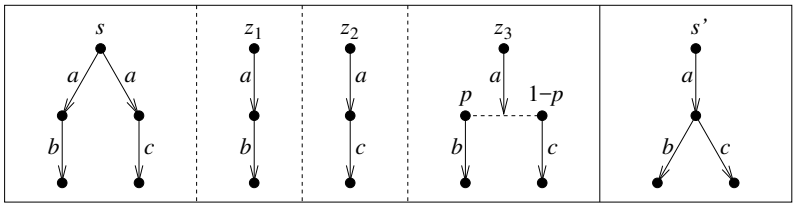


**Fig. 2.** An example of structure modification induced by a randomized scheduler

### 3.2 Structure-Modifying Resolutions via Randomization

If the current state has $n \in \mathbb{N}_{\geq 1}$ outgoing transitions, a *randomized scheduler* generates $p_i \in \mathbb{R}_{[0,1]}$ for $i = 1, \ldots, n$ such that $\sum_{i=1}^{n} p_i \leq 1$ and then selects transition $i$ with probability $p_i$ or stops with probability $1 - \sum_{i=1}^{n} p_i$. A deterministic scheduler is a special case in which $p_i = 1$ for some $i$ or $p_i = 0$ for each $i$.

Randomized schedulers, proposed in [22] and applied to the definition of probabilistic trace [23] and testing [24] semantics, probabilistically combine transitions of the original model. Therefore, the resulting resolutions are not necessarily isomorphic to submodels of the original model (or of its unfolding) because a *modification* of the structure of the original model may have taken place. An example of this phenomenon is shown in Fig. 2, where the NPLTS in the leftmost part admits under randomized schedulers the three maximal resolutions depicted next to it in the figure. The resolution starting with $z_3$ is obtained by combining the two $a$-transitions departing from $s$ with probabilities $p$ and $1 - p$.

The formalization via a correspondence function of a resolution stemming from a randomized scheduler is not an easy task. The reason is that, according to [22], a combined transition may derive from several *differently labeled* transitions, as shown in the central part of the forthcoming Fig. 3. In other words, a resolution of a simple probabilistic automaton [22], in which every transition has a single label, may have a transition with *several* labels, thereby deviating from a simple probabilistic automaton and hence from an NPLTS.

Similar to [3], below we formalize a resolution induced by a variant of randomized scheduler consistent with the definition of probabilistic bisimilarity given in [25] for simple probabilistic automata. At the current state, the scheduler decides to stop or to perform a certain action among the available ones; in the latter case, it takes a convex combination (i.e., the sum of the values $p_i$ is 1) of the outgoing transitions *identically labeled* with that action. To compensate for the impossibility of combining differently labeled transitions, we admit self-combinations; e.g., in Fig. 3 a combination of the $a$-transition departing from $s$ with itself $n$ times is able to reproduce the situation in the rightmost part of the same figure, which is equivalent to the one in the central part.

**Definition 4.** *Let $\mathcal{L} = (S, A, \longrightarrow)$ be an NPLTS and $s \in S$. An acyclic NPLTS $\mathcal{Z} = (Z, A, \longrightarrow_{\mathcal{Z}})$ is a* structure-modifying resolution via randomization *of $s$, written $\mathcal{Z} \in Res_{\mathrm{sm,r}}(s)$, iff there exists a correspondence function $corr_{\mathcal{Z}} : Z \to S$ such that $s = corr_{\mathcal{Z}}(z_s)$, for some $z_s \in Z$, and for all $z \in Z$ it holds that:*

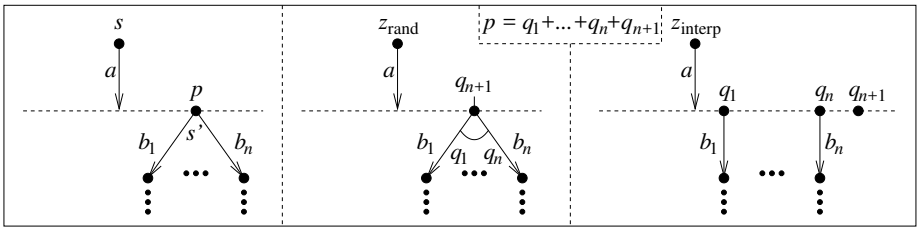


**Fig. 3.** Equivalent resolutions induced by randomized and interpolating schedulers

- *If* $z \xrightarrow{a}_{\mathcal{Z}} \Delta$ *then there exist* $n \in \mathbb{N}_{\geq 1}$, $p_i \in \mathbb{R}_{]0,1]}$ *for* $1 \leq i \leq n$ *summing up to* 1, *and* $corr_{\mathcal{Z}}(z) \xrightarrow{a} \Gamma_i$ *for* $1 \leq i \leq n$, *with* $corr_{\mathcal{Z}}$ *being injective when considered from* $supp(\Delta)$ *to the disjoint union of the sets* $supp(\Gamma_i)$ *and satisfying* $\Delta(z') = \sum_{i=1}^{n} p_i \cdot \Gamma_i(corr_{\mathcal{Z}}(z'))$ *for all* $z' \in supp(\Delta)$.
- *At most one transition departs from* $z$. ■

Injectivity cannot be directly imposed as in Def. 3, otherwise in Fig. 2 the NPLTS model starting with $z_3$ would not be a legal resolution induced by the self-combination of the $a$-transition departing from $s'$ in the rightmost part, and hence $s'$ would not be considered trace equivalent to $s$ in the leftmost part.

### 3.3 Structure-Modifying Resolutions via Interpolation

For every state in the support of the target distribution of the current transition, an *interpolating scheduler* splits it into $n \in \mathbb{N}_{\geq 1}$ copies, each having a single outgoing transition or no transitions at all, to which probabilities are assigned whose sum is the overall probability of the original state, and then selects one of the copies based on its probability. A deterministic scheduler is a special case in which $n = 1$.

Interpolating schedulers, proposed in [10], probabilistically split states of the original model thereby inducing resolutions possibly modifying the structure of the original model. As mentioned in [10], for each resolution obtained from an interpolating (resp. randomized) scheduler, there exists a resolution obtained from a randomized (resp. interpolating) scheduler with the same trace distribution. This can be seen in Fig. 3, where in the leftmost part we have a state $s'$ reached with probability $p$ in the target distribution of an $a$-transition. The resolution in the central part, induced by a randomized scheduler that combines the transitions departing from $s'$, is equivalent to the resolution in the rightmost part, induced by an interpolating scheduler that splits state $s'$, where $\sum_{i=1}^{n+1} q_i = p$.

Resolutions arising from interpolating schedulers were natively defined in [10] through a correspondence function that maps all split states to the original state from which they derive. Unlike Defs. 3 and 4, the constraint on transitions is formulated with respect to the states in the support of the corresponding transition of the *original model* – rather than the states in the support of the transition of the resolution – and the preservation of the overall probability associated with each such state makes injectivity requirements unnecessary.

**Definition 5.** *Let $\mathcal{L} = (S, A, \longrightarrow)$ be an NPLTS and $s \in S$. An acyclic NPLTS $\mathcal{Z} = (Z, A, \longrightarrow_{\mathcal{Z}})$ is a* structure-modifying resolution via interpolation *of $s$, written $\mathcal{Z} \in Res_{\mathrm{sm,i}}(s)$, iff there exists a correspondence function $corr_{\mathcal{Z}} : Z \to S$ such that $s = corr_{\mathcal{Z}}(z_s)$, for some $z_s \in Z$, and for all $z \in Z$ it holds that:*

- *If $z \xrightarrow{a}_{\mathcal{Z}} \Delta$ then $corr_{\mathcal{Z}}(z) \xrightarrow{a} \Gamma$, with $corr_{\mathcal{Z}}$ satisfying for all $s \in supp(\Gamma)$ $\Gamma(s) = \sum_{z' \in supp(\Delta)}^{corr_{\mathcal{Z}}(z')=s} \Delta(z')$.*
- *At most one transition departs from $z$.* ■

A variant of the structure-modifying resolution above has been proposed in [6], which combines the effect of interpolating and randomized schedulers.

## 4 Consequences of the Excessive Power of Schedulers

Although deterministic schedulers are very intuitive, the rigid preservation they ensure about the structure of the original model, together with their freedom of performing choices inconsistent with each other in states with equivalent continuations, causes the resulting probabilistic trace equivalence to be overdiscriminating, thereby violating certain desirable properties. This also happens, to a much lesser extent, with randomized and interpolating schedulers. In the following, after presenting in Sect. 4.1 the definition of some probabilistic behavioral equivalences, we illustrate in Sect. 4.2 a number of anomalies.

### 4.1 Equivalences for Nondeterministic and Probabilistic Processes

The spectrum of behavioral equivalences for nondeterministic and probabilistic processes was studied in [4]. Here we focus on the two endpoints of the spectrum by recalling the definitions of bisimulation and trace semantics.

Probabilistic bisimilarity requires that two NPLTS models are able to mimic each other behavior stepwise, in terms of the probability of reaching the same class of equivalent states when executing the same action [20,25]. Its definition does not need to explicitly resort to resolutions, as these are implicitly built while selecting a single transition from each pair of states.

**Definition 6.** *Let $(S, A, \longrightarrow)$ be an NPLTS and $s_1, s_2 \in S$. We write $s_1 \sim_{\mathrm{PB}} s_2$ iff there exists a probabilistic bisimulation $\mathcal{B}$ over $S$ such that $(s_1, s_2) \in \mathcal{B}$. An equivalence relation $\mathcal{B}$ over $S$ is a* probabilistic bisimulation *iff, whenever $(s_1, s_2) \in \mathcal{B}$, then for all $a \in A$ it holds that for each $s_1 \xrightarrow{a} \Delta_1$ there exists $s_2 \xrightarrow{a} \Delta_2$ such that for all equivalence classes $C \in S/\mathcal{B}$:*

$$\Delta_1(C) \;=\; \Delta_2(C)$$

■

In contrast, trace equivalence requires that two NPLTS models possess the same trace distributions, i.e., the same family of sets of action sequences weighted with their execution probabilities, where each set is related to a specific resolution of nondeterminism [23]. Its definition, which abstracts from branching points of process behavior, explicitly relies on $Res(\_)$, with which we denote any of the sets of resolutions introduced in Defs. 3 to 5.

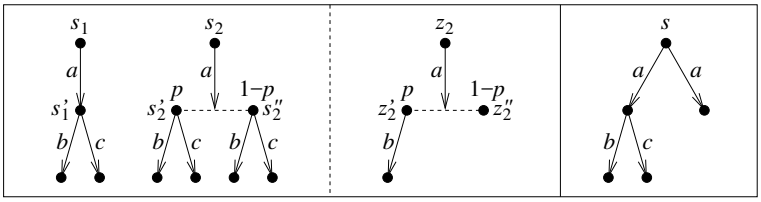


**Fig. 4.** Violation of $s_1 \sim_{\mathrm{PB}} s_2 \Longrightarrow s_1 \sim_{\mathrm{PTr}} s_2$ (maximality does not help)

**Definition 7.** *Let $(S, A, \longrightarrow)$ be an NPLTS and $s_1, s_2 \in S$. We write $s_1 \sim_{\mathrm{PTr}} s_2$ iff for each $\mathcal{Z}_1 \in Res(s_1)$ there exists $\mathcal{Z}_2 \in Res(s_2)$ such that for all traces $\alpha \in A^*$:*

$$prob(\mathcal{CC}(z_{s_1}, \alpha)) \;=\; prob(\mathcal{CC}(z_{s_2}, \alpha))$$

*and also the condition obtained by exchanging $\mathcal{Z}_1$ with $\mathcal{Z}_2$ is satisfied.* ■

### 4.2 Anomalies and Counterexamples

We now present a number of counterexamples showing that:

- $\sim_{\mathrm{PTr}}$ is not coarser than $\sim_{\mathrm{PB}}$ under deterministic schedulers.
- $\sim_{\mathrm{PTr}}$ is not a congruence w.r.t. action prefix under deterministic schedulers.
- $\sim_{\mathrm{PTr}}$ is not backward compatible with its version for fully prob. processes.

Consider the two NPLTS models in the leftmost part of Fig. 4. It holds that $s_1 \sim_{\mathrm{PB}} s_2$, but $s_1 \not\sim_{\mathrm{PTr}} s_2$ because of the resolution in the central part of Fig. 4, where trace $a\,b$ is executable with probability $p$ instead of 1. This resolution belongs to $Res_{\mathrm{sp}}(s_2) \setminus Res_{\mathrm{sp}}(s_1)$ as it does not preserve the structure of the NPLTS whose initial state is $s_1$. Notice that the same resolution belongs to $Res_{\mathrm{sm,r}}(s_1)$, if the $a$-transition of $s_1$ is combined with itself, and to $Res_{\mathrm{sm,i}}(s_1)$, if $z_2'$ and $z_2''$ are both mapped to $s_1'$.

One may be tempted to admit only *maximal* resolutions in the definition of probabilistic trace equivalences, but the problem would still be there if a $c$-transition departed from $z_2''$. Moreover, by so doing, probabilistic trace equivalences would no longer be compatible with trace equivalence. For instance, the former would not identify the two fully nondeterministic, trace equivalent NPLTS models in Fig. 4 whose initial states are $s_1$ and $s$, because the maximal resolution of $s$ with an $a$-transition only – featuring traces $\varepsilon$ and $a$ – is not matched by the two maximal resolutions of $s_1$ – resp. featuring also $a\,b$ and $a\,c$.

Let us move to examine the two NPLTS models in the leftmost part of Fig. 5. After the two $a$-transitions, two distributions are reached that are probabilistic trace equivalent, in the sense that for each class of equivalent states they both assign the same probability to that class. However, it holds that $s_3 \not\sim_{\mathrm{PTr}} s_4$ due to the resolution in the rightmost part of Fig. 5, where trace $a\,a'\,b$ is executable with probability $p$ instead of 1. This resolution belongs to $Res_{\mathrm{sp}}(s_3) \setminus Res_{\mathrm{sp}}(s_4)$ as it does not preserve the structure of the NPLTS whose initial state

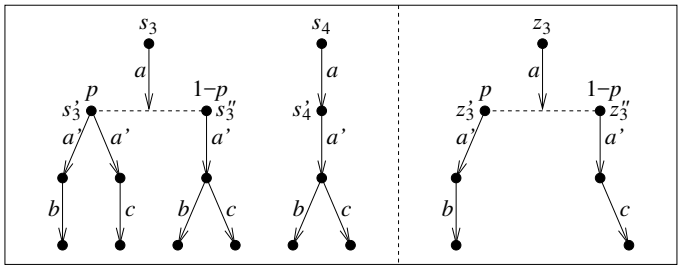


**Fig. 5.** Violation of congruence with respect to action prefix: $s_3 \not\sim_{\mathrm{PTr}} s_4$

is $s_4$. The same resolution belongs to $Res_{\mathrm{sm,r}}(s_4)$, if the $a$-transition of $s_4$ is combined with itself, and to $Res_{\mathrm{sm,i}}(s_4)$, if $z_3'$ and $z_3''$ are both mapped to $s_4'$.

This example reveals that, under deterministic schedulers, probabilistic trace equivalence is not a congruence with respect to the action prefix operator, which concatenates the execution of an action with a process. The difference with trace equivalence for fully nondeterministic processes is that in our setting the continuation after an action is *not a single process*, but a probability distribution over processes. The problem arises when several equivalent states are in the support of the same distribution, as in the target distribution of the $a$-transition of $s_3$, thereby allowing schedulers to act inconsistently.

We finally study the two NPLTS models in the leftmost part of Fig. 6. They are identified by the trace equivalence for fully probabilistic processes of [17], which does not use schedulers as in those processes there are no nondeterministic choices to be solved. However, it turns out that $s_5 \not\sim_{\mathrm{PTr}} s_6$ because $\sim_{\mathrm{PTr}}$ does make use of schedulers, in particular their capability of *stopping the execution*. This is witnessed by the resolution in the rightmost part of Fig. 6, where not only trace $a\,b\,c_1$ but also trace $a\,b$ is executable with probability $p$. This resolution belongs only to $Res_{\mathrm{sp}}(s_6)$ as it does not preserve the structure of the NPLTS whose initial state is $s_5$. It does not even belong to $Res_{\mathrm{sm,r}}(s_5) \cup Res_{\mathrm{sm,i}}(s_5)$ because in the NPLTS starting with $s_5$, after performing the $a$-transition and the $b$-transition, the $c_1$-transition can be executed with probability $p$, while the $c_1$-transition in the resolution can be executed with probability 1 and hence its source state cannot be mapped to the source state of the former $c_1$-transition.

This further example highlights that schedulers inducing structure-modifying resolutions are not exempt from shortcomings despite their greater flexibility. The considered resolution would be ruled out by imposing maximality but, as we have seen at the beginning of this section, that may generate other anomalies.

## 5 Anomaly Avoidance via Coherent Resolutions

The anomalies shown in Figs. 4 to 6 are due to the freedom of schedulers of making different decisions in equivalent states and cause probabilistic trace equivalence to be overdiscriminating. We thus propose to limit the excessive power of

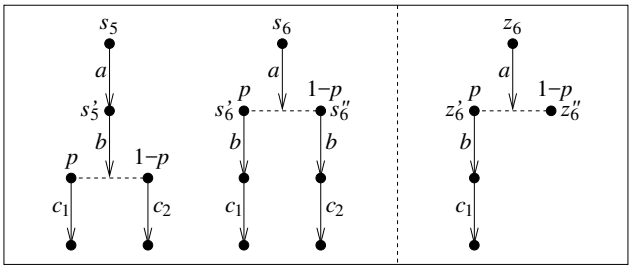


**Fig. 6.** Incompatibility w.r.t. fully prob. processes: $s_5 \not\sim_{\mathrm{PTr}} s_6$ (levelwise coherency)

schedulers by restricting them to yield *coherent resolutions*. This means that, if several states in the support of the target distribution of a transition are equivalent, then the decisions made by the scheduler in those states have to be coherent with each other, so that the states to which they correspond in any resolution are equivalent as well. The *coherency constraint* implementing this idea will be expressed by reasoning on *coherent trace distributions*, i.e., families of sets of traces weighted with their execution probabilities in a given resolution, built through the following operations.

**Definition 8.** *Let $A \neq \emptyset$ be a countable set. For $a \in A$, $p \in \mathbb{R}$, $TD \subseteq 2^{A^* \times \mathbb{R}}$, and $T \subseteq A^* \times \mathbb{R}$ we define:*

$$a \,.\, TD = \{a \,.\, T \mid T \in TD\} \qquad a \,.\, T = \{(a\,\alpha, p') \mid (\alpha, p') \in T\}$$
$$p \cdot TD = \{p \cdot T \mid T \in TD\} \qquad p \cdot T = \{(\alpha, p \cdot p') \mid (\alpha, p') \in T\}$$
$$tr(TD) = \{tr(T) \mid T \in TD\} \quad tr(T) = \{\alpha \in A^* \mid (\alpha, p') \in T \text{ for some } p' \in \mathbb{R}\}$$

*while for $TD_1, TD_2 \subseteq 2^{A^* \times \mathbb{R}}$ we define:*

$$TD_1 + TD_2 = \begin{cases} \{T_1 + T_2 \mid T_1 \in TD_1 \wedge T_2 \in TD_2 \wedge tr(T_1) = tr(T_2)\} \\ \qquad\qquad \text{if } tr(TD_1) = tr(TD_2) \\ \{T_1 + T_2 \mid T_1 \in TD_1 \wedge T_2 \in TD_2\} \\ \qquad\qquad \text{otherwise} \end{cases}$$

*where for $T_1, T_2 \subseteq A^* \times \mathbb{R}$ we define:*

$$T_1 + T_2 = \{(\alpha, p_1 + p_2) \mid (\alpha, p_1) \in T_1 \wedge (\alpha, p_2) \in T_2\} \cup \{(\alpha, p) \in T_1 \cup T_2 \mid \alpha \notin tr(T_1) \cap tr(T_2)\}$$

■

Weighted trace set addition is commutative and associative. In the definition of $T_1 + T_2$, which is inspired by [3], probabilities of identical traces in the two summands are *always* added up for coherency purposes. Before Def. 3.5 of [3], the definition of $X + Y$, i.e., $T_1 + T_2$, should have included $(\alpha, q) \in X \cup Y$ in the sum anyhow, otherwise the right-to-left implication in Lemma 3.7 of [3] cannot hold as can be seen from trace $a\,b$ of the (incoherent) resolution in the central part of Fig. 4 of this paper; that definition of $X + Y$ works here instead, because of the focus on coherency.

Trace distribution addition is only commutative. Intuitively, the two summands in $TD_1 + TD_2$ represent two families of sets of weighted traces executable in the resolutions of two states in the support of a target distribution.

Every weighted trace set $T_1 \in TD_1$ is summed with every weighted trace set $T_2 \in TD_2$ – so to characterize an overall resolution – unless $TD_1$ and $TD_2$ have the same family of trace sets, in which case summation is restricted to weighted trace sets featuring the same traces for the sake of coherency. In the definition below, the double summation ensures that trace distributions $\Delta(s') \cdot TD^{\rm c}_{n-1}(s')$ exhibiting the same family $\Theta$ of trace sets will be summed up first.

**Definition 9.** *Let $\mathcal{L} = (S, A, \longrightarrow)$ be an NPLTS and $s \in S$. The* coherent trace distribution *of $s$ is the subset of $2^{A^* \times \mathbb{R}_{]0,1]}}$ defined as follows:*

$$TD^{\rm c}(s) \,=\, \bigcup_{n \in \mathbb{N}} TD^{\rm c}_n(s)$$

*where the coherent trace distribution of $s$ whose traces have length at most $n$ is defined as:*

$$TD^{\rm c}_n(s) \,=\, \begin{cases} (\varepsilon,1) \dagger \bigcup\limits_{s \xrightarrow{a} \Delta} a \,.\, \left( \sum\limits_{\Theta \in tr(\Delta, n-1)} \sum\limits_{s' \in supp(\Delta)}^{tr(TD^{\rm c}_{n-1}(s'))=\Theta} \Delta(s') \cdot TD^{\rm c}_{n-1}(s') \right) \\ \qquad\qquad\qquad\qquad\qquad\qquad \textit{if } n > 0 \textit{ and } s \textit{ has outgoing transitions} \\ \{\{(\varepsilon,1)\}\} \\ \qquad\qquad\qquad\qquad\qquad\qquad \textit{otherwise} \end{cases}$$

*for $tr(\Delta, n-1) = \{tr(TD^{\rm c}_{n-1}(s')) \mid s' \in supp(\Delta)\}$ and $(\varepsilon,1) \dagger TD = \{\{(\varepsilon,1)\} \cup T \mid T \in TD\}$.* ■

Let us reconsider the three counterexamples of Sect. 4 plus two more:

- In Fig. 4 we have $TD^{\rm c}(s'_2) = \{\{(\varepsilon,1)\}, \{(\varepsilon,1),(b,1)\}, \{(\varepsilon,1),(c,1)\}\} = TD^{\rm c}(s''_2)$ – from which $TD^{\rm c}(s_2) = \{\{(\varepsilon,1)\}, \{(\varepsilon,1),(a,1)\}, \{(\varepsilon,1),(a,1),(a\,b,1)\}, \{(\varepsilon,1),(a,1),(a\,c,1)\}\} = TD^{\rm c}(s_1)$ follows – but in the resolution $TD^{\rm c}(z'_2) = \{\{(\varepsilon,1)\}, \{(\varepsilon,1),(b,1)\}\} \neq \{\{(\varepsilon,1)\}\} = TD^{\rm c}(z''_2)$.
- In Fig. 5 we have $TD^{\rm c}(s'_3) = \{\{(\varepsilon,1)\}, \{(\varepsilon,1),(a',1)\}, \{(\varepsilon,1),(a',1),(a'\,b,1)\}, \{(\varepsilon,1),(a',1),(a'\,c,1)\}\} = TD^{\rm c}(s''_3)$ whereas in the resolution $TD^{\rm c}(z'_3) = \{\{(\varepsilon,1)\}, \{(\varepsilon,1),(a',1)\}, \{(\varepsilon,1),(a',1),(a'\,b,1)\}\} \neq \{\{(\varepsilon,1)\}, \{(\varepsilon,1),(a',1)\}, \{(\varepsilon,1),(a',1),(a'\,c,1)\}\} = TD^{\rm c}(z''_3)$.
- In Fig. 6 we have $TD^{\rm c}(s'_6) = \{\{(\varepsilon,1)\}, \{(\varepsilon,1),(b,1)\}, \{(\varepsilon,1),(b,1),(b\,c_1,1)\}\} \neq \{\{(\varepsilon,1)\}, \{(\varepsilon,1),(b,1)\}, \{(\varepsilon,1),(b,1),(b\,c_2,1)\}\} = TD^{\rm c}(s''_6)$. However, $TD^{\rm c}_1(s'_6) = \{\{(\varepsilon,1),(b,1)\}\} = TD^{\rm c}_1(s''_6)$ while in the resolution $TD^{\rm c}_1(z'_6) = \{\{(\varepsilon,1),(b,1)\}\} \neq \{\{(\varepsilon,1)\}\} = TD^{\rm c}_1(z''_6)$. This shows that we should set up separate coherency constraints relying on $TD^{\rm c}_n$ sets for every $n \in \mathbb{N}$.
- Consider the two fully probabilistic NPLTS models in the leftmost part of Fig. 7. They are identified by the trace equivalence of [17], but $s_7 \not\sim_{\rm PTr} s_8$ due to the resolution in the rightmost part of the same figure. It holds that $TD^{\rm c}_2(s'_7) = \{\{(\varepsilon,1),(b,1),(b\,c,0.3)\}\} \neq \{\{(\varepsilon,1),(b,1),(b\,c,0.2)\}\} = TD^{\rm c}_2(s''_7)$, with $TD^{\rm c}_3(s_7) = \{\{(\varepsilon,1),(a,1),(a\,b,1),(a\,b\,c,0.25)\}\} = TD^{\rm c}_3(s_8)$. However, we observe that $tr(TD^{\rm c}_2(s'_7)) = \{\{\varepsilon, b, b\,c\}\} = tr(TD^{\rm c}_2(s''_7))$ whereas $tr(TD^{\rm c}_2(z'_7)) = \{\{\varepsilon, b, b\,c\}\} \neq \{\{\varepsilon, b\}\} = tr(TD^{\rm c}_2(z''_7))$. This indicates that the coherency constraints should rely on $TD^{\rm c}_n$ sets up to the probabilities they contain, i.e., the coherency constraints should rely on $tr(TD^{\rm c}_n)$ sets.
- The violations in Figs. 6 and 7 of backward compatibility with the trace equivalence of [17] have a twofold interpretation. The former is that incoherent selections are made by the scheduler in states having the same traces of

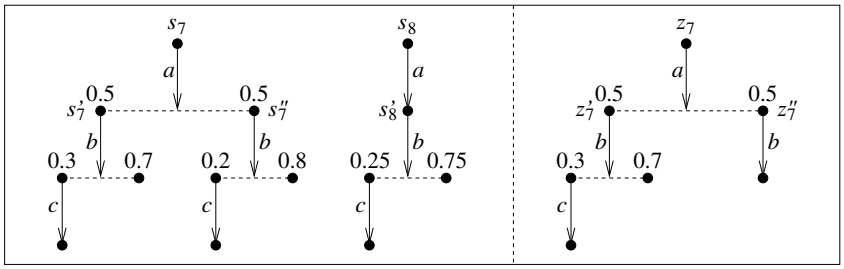


**Fig. 7.** Incompatibility w.r.t. fully prob. processes: $s_7 \not\sim_{\mathrm{PTr}} s_8$ (probability abstraction)

a certain length. The latter ascribes the lack of coherency to the fact that, in both resolutions depicted in those figures, the scheduler proceeds by selecting a transition along one direction while it stops the execution along the other direction. This is even more evident with the two fully probabilistic NPLTS models in the leftmost part of Fig. 8, which are identified by [17] but told apart by the resolution on the right, where $a\,b\,c_1$ is executable with probability 0.25, as $tr(TD_2^{\mathrm{c}}(s'_{10}))$, $tr(TD_2^{\mathrm{c}}(s''_{10}))$, and $tr(TD_2^{\mathrm{c}}(s'''_{10}))$ are pairwise different. In every coherent resolution of $s_9$, trace $a\,b\,c_1$ can be executed only with probability 0.5. This calls for a complete presence of computations of the same length in each resolution – including shorter maximal computations if any – which is different from requiring resolution maximality.

**Definition 10.** *Let $\mathcal{L} = (S, A, \longrightarrow)$ be an NPLTS, $s \in S$, and $\mathcal{Z} = (Z, A, \longrightarrow_{\mathcal{Z}}) \in Res(s)$ with correspondence function $corr_{\mathcal{Z}} : Z \to S$. We say that $\mathcal{Z}$ is a* coherent resolution *of $s$, written $\mathcal{Z} \in Res^{\mathrm{c}}(s)$, iff for all $z \in Z$, whenever $z \xrightarrow{a}_{\mathcal{Z}} \Delta$, then for all $n \in \mathbb{N}$:*

1. *$tr(TD_n^{\mathrm{c}}(corr_{\mathcal{Z}}(z'))) = tr(TD_n^{\mathrm{c}}(corr_{\mathcal{Z}}(z''))) \implies tr(TD_n^{\mathrm{c}}(z')) = tr(TD_n^{\mathrm{c}}(z''))$ for all $z', z'' \in supp(\Delta)$.*
2. *If there exists $z' \in supp(\Delta)$ such that $tr(TD_n^{\mathrm{c}}(z'))$ contains traces of length $n$, then for all $z'' \in supp(\Delta)$ either $tr(TD_n^{\mathrm{c}}(z''))$ contains traces of length $n$ too, or any $\alpha \in A^*$ occurring in $tr(TD_n^{\mathrm{c}}(z''))$ has length less than $n$ but there exists a maximal trace in $tr(TD_n^{\mathrm{c}}(corr_{\mathcal{Z}}(z'')))$ corresponding to $\alpha$.* ■

In the definition above, $Res(\_)$ denotes any of the sets of resolutions introduced in Defs. 3 to 5. From now on, we focus on $Res_{\mathrm{sp}}^{\mathrm{c}}(\_)$. Notice that the resolutions in Figs. 4 to 8 do *not* respectively belong to $Res_{\mathrm{sp}}^{\mathrm{c}}(s_2)$, $Res_{\mathrm{sp}}^{\mathrm{c}}(s_3)$, $Res_{\mathrm{sp}}^{\mathrm{c}}(s_6)$, $Res_{\mathrm{sp}}^{\mathrm{c}}(s_7)$, and $Res_{\mathrm{sp}}^{\mathrm{c}}(s_{10})$.

We conclude by proving that probabilistic trace equivalence no longer suffers from the anomalies illustrated in Sect. 4 *when using coherent resolutions* induced by deterministic schedulers. In the following, we lift a probabilistic behavioral equivalence $\sim$ from states to distributions over states by letting $\Delta_1 \sim \Delta_2$ iff $\Delta_1(C) = \Delta_2(C)$ for all equivalence classes $C$ of $\sim$. Moreover, the action prefix construction $a\,.\,\Delta$ stands for an $a$-transition whose target distribution

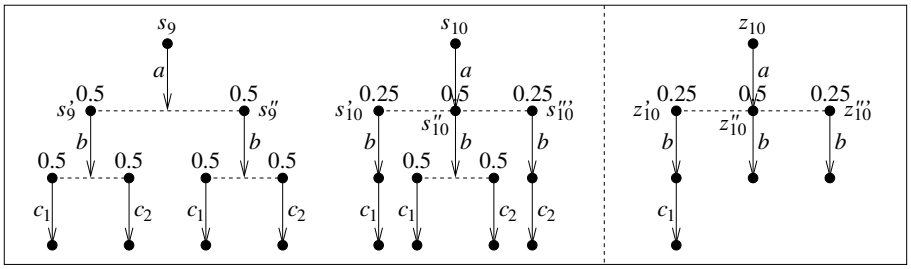


**Fig. 8.** Incompat. w.r.t. fully prob. processes: $s_9 \not\sim_{\mathrm{PTr}} s_{10}$ (levelwise completeness)

is $\Delta$, whereas $\sim^{\mathrm{fp}}_{\mathrm{PTr}}$ denotes the probabilistic trace equivalence for fully probabilistic processes defined in [17] by letting $s_1 \sim^{\mathrm{fp}}_{\mathrm{PTr}} s_2$ iff $prob(\mathcal{CC}(s_1, \alpha)) = prob(\mathcal{CC}(s_2, \alpha))$ for all $\alpha \in A^*$.

We point out that coherency was unfortunately neglected in [3,4]. In particular, property 1 below is the rectified version of a chain of results in [4] consisting of Thms. 6.5(2), 5.9(3), 4.5(2) and property 3 below is the rectified version of Thm. 3.4(2) of [3,4]; deterministic schedulers were considered in all those theorems. Property 3 now holds also in the case of randomized/interpolating schedulers by just imposing condition 2 of Def. 10.

**Theorem 1.** *Let $\mathcal{L} = (S, A, \longrightarrow)$ be an NPLTS, $s_1, s_2 \in S$, $\Delta_1, \Delta_2 \in Distr(S)$. Under coherent resolutions induced by deterministic schedulers it holds that:*

1. $s_1 \sim_{\mathrm{PB}} s_2 \implies s_1 \sim_{\mathrm{PTr}} s_2$.
2. $\Delta_1 \sim_{\mathrm{PTr}} \Delta_2 \implies a \,.\, \Delta_1 \sim_{\mathrm{PTr}} a \,.\, \Delta_2$ *for all* $a \in A$.
3. *If $\mathcal{L}$ is fully probabilistic, then* $s_1 \sim_{\mathrm{PTr}} s_2 \iff s_1 \sim^{\mathrm{fp}}_{\mathrm{PTr}} s_2$. ■

We finally observe that looser coherency constraints, based on weighted trace sets rather than trace distributions as in Def. 10, would not work. Similar to $TD^{\mathrm{c}}(s)$ in Def. 9, one may define $T^{\mathrm{c}}(s)$ by considering all weighted traces executable from $s$ at once – i.e., without keeping track of the resolutions in which they are feasible – and use it for coherency purposes, but then probabilistic trace equivalent NPLTS models like the ones in Fig. 9 would be told apart. Indeed, we would have $tr(T^{\mathrm{c}}(s'_1)) = \{\epsilon, b, b\,c_1, b\,c_2, b\,c\} = tr(T^{\mathrm{c}}(s'_2))$ – whereas $tr(TD^{\mathrm{c}}(s'_1)) \neq tr(TD^{\mathrm{c}}(s'_2))$ – hence in any coherent resolution of $s'$ traces $a\,b\,c_1$, $a\,b\,c_2$, $a\,b\,c$ could only be executed with probability 0.5 if present, while $s''$ admits coherent resolutions in which those traces have execution probability 0.25.

## 6 Conclusions

To guarantee a number of desirable properties for probabilistic trace equivalence over probabilistic automata, we have proposed a set of coherency constraints as a solution to the problem – addressed also in [12] for a different probabilistic model and equivalence – of limiting the excessive power of schedulers.

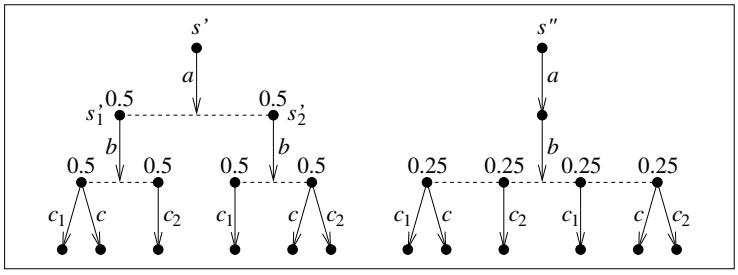


**Fig. 9.** Using weighted trace sets for coherency breaks probabilistic trace equivalence

The highlighted anomalies mostly have to do with structure-preserving resolutions generated by deterministic schedulers, so one may wonder why not to avoid those schedulers altogether. The first reason is that, as shown in [4], the use of a specific family of schedulers has an impact on the discriminating power of behavioral equivalences, so there might be situations in which considering deterministic schedulers is more appropriate. The second reason is that, as witnessed by Fig. 6, some of the examined anomalies affect also equivalences defined on structure-modifying resolutions generated by randomized/interpolating schedulers. The third reason is that in more general frameworks, like the ULTraS metamodel [2] of which probabilistic automata are an instance, the applicability of deterministic schedulers is always possible, while this might not be the case for other families of schedulers.

**Acknowledgement.** We would like to thank Valeria Vignudelli for pointing out the property violation illustrated in Fig. 4 and Rob van Glabbeek for the valuable discussions on interpolating and randomized schedulers.

## References

1. C. Baier, J.-P. Katoen, H. Hermanns, and V. Wolf. Comparative branching-time semantics for Markov chains. *Information and Computation*, 200:149–214, 2005.
2. M. Bernardo. Genesis and evolution of ULTraS: Metamodel, metaequivalences, metaresults. In *Models, Languages, and Tools for Concurrent and Distributed Programming*, volume 11665 of *LNCS*, pages 92–111. Springer, 2019.
3. M. Bernardo, R. De Nicola, and M. Loreti. Revisiting trace and testing equivalences for nondeterministic and probabilistic processes. *Logical Methods in Computer Science*, 10(1:16):1–42, 2014.
4. M. Bernardo, R. De Nicola, and M. Loreti. Relating strong behavioral equivalences for processes with nondeterminism and probabilities. *Theoretical Computer Science*, 546:63–92, 2014.
5. M. Bernardo, D. Sangiorgi, and V. Vignudelli. On the discriminating power of testing equivalences for reactive probabilistic systems: Results and open problems. In *Proc. of the 11th Int. Conf. on the Quantitative Evaluation of Systems (QEST 2014)*, volume 8657 of *LNCS*, pages 281–296. Springer, 2014.

6. F. Bonchi, A. Sokolova, and V. Vignudelli. The theory of traces for systems with nondeterminism and probability. In *Proc. of the 34th ACM/IEEE Symp. on Logic in Computer Science (LICS 2019)*, pages (19:62)1–14. IEEE-CS Press, 2019.
7. S.D. Brookes, C.A.R. Hoare, and A.W. Roscoe. A theory of communicating sequential processes. *Journal of the ACM*, 31:560–599, 1984.
8. R. De Nicola. Extensional equivalences for transition systems. *Acta Informatica*, 24:211–237, 1987.
9. R. De Nicola and M. Hennessy. Testing equivalences for processes. *Theoretical Computer Science*, 34:83–133, 1984.
10. Y. Deng, R.J. van Glabbeek, C. Morgan, and C. Zhang. Scalar outcomes suffice for finitary probabilistic testing. In *Proc. of the 16th European Symp. on Programming (ESOP 2007)*, volume 4421 of *LNCS*, pages 363–378. Springer, 2007.
11. C. Derman. *Finite State Markovian Decision Processes*. Academic Press, 1970.
12. S. Georgievska and S. Andova. Probabilistic may/must testing: Retaining probabilities by restricted schedulers. *Formal Aspects of Computing*, 24:727–748, 2012.
13. R.J. van Glabbeek. The linear time – branching time spectrum I. In *Handbook of Process Algebra*, pages 3–99. Elsevier, 2001.
14. D.T. Huynh and L. Tian. On some equivalence relations for probabilistic processes. *Fundamenta Informaticae*, 17:211–234, 1992.
15. B. Jonsson, C. Ho-Stuart, and Wang Yi. Testing and refinement for nondeterministic and probabilistic processes. In *Proc. of the 3rd Int. Symp. on Formal Techniques in Real Time and Fault Tolerant Systems (FTRTFT 1994)*, volume 863 of *LNCS*, pages 418–430. Springer, 1994.
16. B. Jonsson and Wang Yi. Compositional testing preorders for probabilistic processes. In *Proc. of the 10th IEEE Symp. on Logic in Computer Science (LICS 1995)*, pages 431–441. IEEE-CS Press, 1995.
17. C.-C. Jou and S.A. Smolka. Equivalences, congruences, and complete axiomatizations for probabilistic processes. In *Proc. of the 1st Int. Conf. on Concurrency Theory (CONCUR 1990)*, volume 458 of *LNCS*, pages 367–383. Springer, 1990.
18. R.M. Keller. Formal verification of parallel programs. *Communications of the ACM*, 19:371–384, 1976.
19. J.G. Kemeny and J.L. Snell. *Finite Markov Chains*. Van Nostrand, 1960.
20. K.G. Larsen and A. Skou. Bisimulation through probabilistic testing. *Information and Computation*, 94:1–28, 1991.
21. R. Milner. *Communication and Concurrency*. Prentice Hall, 1989.
22. R. Segala. *Modeling and Verification of Randomized Distributed Real-Time Systems*. PhD Thesis, 1995.
23. R. Segala. A compositional trace-based semantics for probabilistic automata. In *Proc. of the 6th Int. Conf. on Concurrency Theory (CONCUR 1995)*, volume 962 of *LNCS*, pages 234–248. Springer, 1995.
24. R. Segala. Testing probabilistic automata. In *Proc. of the 7th Int. Conf. on Concurrency Theory (CONCUR 1996)*, volume 1119 of *LNCS*, pages 299–314. Springer, 1996.
25. R. Segala and N.A. Lynch. Probabilistic simulations for probabilistic processes. In *Proc. of the 5th Int. Conf. on Concurrency Theory (CONCUR 1994)*, volume 836 of *LNCS*, pages 481–496. Springer, 1994.
26. Wang Yi and K.G. Larsen. Testing probabilistic and nondeterministic processes. In *Proc. of the 12th Int. Symp. on Protocol Specification, Testing and Verification (PSTV 1992)*, pages 47–61. North-Holland, 1992.

# A Proofs of Results

**Proof of Thm. 1.**
Given an NPLTS $\mathcal{L} = (S, A, \longrightarrow)$, $s_1, s_2 \in S$, and $\Delta_1, \Delta_2 \in \mathit{Distr}(S)$, we proceed as follows:

1. We show that, from $(s_1, s_2) \in \mathcal{B}$ for some probabilistic bisimulation $\mathcal{B}$, it follows that $(\star)$ for each $\mathcal{Z}_1 = (Z_1, A, \longrightarrow_{\mathcal{Z}_1}) \in \mathit{Res}^{\mathrm{c}}_{\mathrm{sp}}(s_1)$ – resp. $\mathcal{Z}_2 = (Z_2, A, \longrightarrow_{\mathcal{Z}_2}) \in \mathit{Res}^{\mathrm{c}}_{\mathrm{sp}}(s_2)$ – there exists $\mathcal{Z}_2 = (Z_2, A, \longrightarrow_{\mathcal{Z}_2}) \in \mathit{Res}^{\mathrm{c}}_{\mathrm{sp}}(s_2)$ – resp. $\mathcal{Z}_1 = (Z_1, A, \longrightarrow_{\mathcal{Z}_1}) \in \mathit{Res}^{\mathrm{c}}_{\mathrm{sp}}(s_1)$ – such that for all $\alpha \in A^*$ it holds that:
$$\mathit{prob}(\mathcal{CC}(z_{s_1}, \alpha)) \;=\; \mathit{prob}(\mathcal{CC}(z_{s_2}, \alpha))$$
Starting from $s_1$, we focus on an arbitrary $\mathcal{Z}_1 = (Z_1, A, \longrightarrow_{\mathcal{Z}_1}) \in \mathit{Res}^{\mathrm{c}}_{\mathrm{sp}}(s_1)$, which we assume not to consist of a single state without transitions so to avoid trivial cases. Let $z_{s_1} \xrightarrow{a} \Delta_1$ be the initial transition of $\mathcal{Z}_1$, which we assume to derive from $s_1 \xrightarrow{a} \Gamma_1$. Since $(s_1, s_2) \in \mathcal{B}$ and $\mathcal{B}$ is a probabilistic bisimulation, there must exist $\mathcal{Z}_2 = (Z_2, A, \longrightarrow_{\mathcal{Z}_2}) \in \mathit{Res}^{\mathrm{c}}_{\mathrm{sp}}(s_2)$ with initial transition $z_{s_2} \xrightarrow{a} \Delta_2$, which we assume to derive from $s_2 \xrightarrow{a} \Gamma_2$, such that, in particular, for each $C \subseteq Z_1 \cup Z_2$, whose image via $\mathit{corr}_{\mathcal{Z}_1} \cup \mathit{corr}_{\mathcal{Z}_2}$ is an equivalence class in $S/\mathcal{B}$, it holds that:
$$\Delta_1(C) \;=\; \Gamma_1(\mathit{corr}_{\mathcal{Z}_1}(C \cap Z_1)) \;=\; \Gamma_2(\mathit{corr}_{\mathcal{Z}_2}(C \cap Z_2)) \;=\; \Delta_2(C)$$
Among all the resolutions in $\mathit{Res}^{\mathrm{c}}_{\mathrm{sp}}(s_2)$ satisfying the equality above, we choose as $\mathcal{Z}_2$ one that can execute all the traces of $\mathcal{Z}_1$ (which must exist otherwise $s_1$ could execute a trace not executable by $s_2$ and hence $s_1 \sim_{\mathrm{PB}} s_2$ would be contradicted) and only those traces (longer traces can be ruled out via pruning). Given an arbitrary $\alpha \in A^*$, we prove property $(\star)$ by proceeding by induction on $|\alpha| \in \mathbb{N}$:
   - If $|\alpha| = 0$, i.e., $\alpha = \varepsilon$, then it trivially holds that:
   $$\mathit{prob}(\mathcal{CC}(z_{s_1}, \alpha)) \;=\; 1 \;=\; \mathit{prob}(\mathcal{CC}(z_{s_2}, \alpha))$$
   - Let $|\alpha| = n+1$ for some $n \in \mathbb{N}$, with $\alpha = a'\,\alpha'$ and $|\alpha'| = n$, and suppose that property $(\star)$ holds for each trace of length $n$ when starting from two probabilistic bisimilar states. There are two cases:
     - If $a' \neq a$, since both $\mathcal{Z}_1$ and $\mathcal{Z}_2$ start with an $a$-transition, it trivially holds that:
     $$\mathit{prob}(\mathcal{CC}(z_{s_1}, \alpha)) \;=\; 0 \;=\; \mathit{prob}(\mathcal{CC}(z_{s_2}, \alpha))$$
     - If $a' = a$, we observe that an arbitrary $C \subseteq Z_1 \cup Z_2$, whose image via $\mathit{corr}_{\mathcal{Z}_1} \cup \mathit{corr}_{\mathcal{Z}_2}$ is an equivalence class in $S/\mathcal{B}$, is either reachable via both $a$-transitions, or via neither; moreover, thanks to the coherency of $\mathcal{Z}_1$ and $\mathcal{Z}_2$, either $\alpha'$ is executable in all the states of $C$, or in none of them. Let $\mathcal{G}$ be the set of subsets of $Z_1 \cup Z_2$, whose images via $\mathit{corr}_{\mathcal{Z}_1} \cup \mathit{corr}_{\mathcal{Z}_2}$ are equivalence classes in $S/\mathcal{B}$, that are reachable via both $a$-transitions and in which $\alpha'$ is executable; note that the other subsets do not contribute to $\mathit{prob}(\mathcal{CC}(z_{s_1}, \alpha))$ and $\mathit{prob}(\mathcal{CC}(z_{s_2}, \alpha))$. For each $C \in \mathcal{G}$, given an arbitrary $z_{C,1} \in C \cap \mathit{supp}(\Delta_1)$ and an arbitrary $z_{C,2} \in C \cap \mathit{supp}(\Delta_2)$ whose corresponding states in $S$ are $s_{C,1}$ and $s_{C,2}$, since $s_{C,1} \sim_{\mathrm{PB}} s_{C,2}$ and $|\alpha'| = n$ by the induction

hypothesis and the coherency of $\mathcal{Z}_1$ and $\mathcal{Z}_2$ we have that:

$$prob(\mathcal{CC}(z_{C,1}, \alpha')) \;=\; prob(\mathcal{CC}(z_{C,2}, \alpha'))$$

As a consequence, by the distributivity of multiplication over addition and the compositionality of equality with respect to both operations, we have that:

$$\begin{aligned} prob(\mathcal{CC}(z_{s_1}, \alpha)) &= \sum_{C \in \mathcal{G}} \Delta_1(C) \cdot prob(\mathcal{CC}(z_{C,1}, \alpha')) = \\ &= \sum_{C \in \mathcal{G}} \Delta_2(C) \cdot prob(\mathcal{CC}(z_{C,2}, \alpha')) = prob(\mathcal{CC}(z_{s_2}, \alpha)) \end{aligned}$$

2. From $\Delta_1 \sim_{\mathrm{PTr}} \Delta_2$ it follows that $\Delta_1(C) = \Delta_2(C)$ for all $C \in S/\sim_{\mathrm{PTr}}$, hence in particular it holds that for each $s_1 \in supp(\Delta_1)$ there must exist $s_2 \in supp(\Delta_2)$ such that $s_1 \sim_{\mathrm{PTr}} s_2$, and vice versa.
   The only interesting case is the one in which we consider a trace of the form $a\,\alpha'$ and for $k \in \{1, 2\}$ a resolution $\mathcal{Z}_k = (Z_k, A, \longrightarrow_{\mathcal{Z}_k}) \in Res^{\mathrm{c}}_{\mathrm{sp}}(a\,.\,\Delta_k)$ that starts with an $a$-transition. This $a$-transition reaches with probability $p_C = \Delta_k(C)$ the set of states in $\mathcal{Z}_k$ whose corresponding original states in $\mathcal{L}$ via $corr_{\mathcal{Z}_k}$ are in the same equivalence class $C \in S/\sim_{\mathrm{PTr}}$. The reason is that, thanks to the coherency of $\mathcal{Z}_k$, two states in the support of the target distribution of the considered $a$-transition of $\mathcal{Z}_k$ possess the same traces if so do their corresponding states in $supp(\Delta_k)$, as is the case with the states of $C$.
   Given $s_{k,C} \in C \cap supp(\Delta_k)$ for $C \in S/\sim_{\mathrm{PTr}}$ and $\mathcal{Z}_{k,C} = (Z_{k,C}, A, \longrightarrow_{\mathcal{Z}_{k,C}}) \in Res^{\mathrm{c}}_{\mathrm{sp}}(s_{k,C})$ part of $\mathcal{Z}_k$, for any other $s'_{k,C} \in C \cap supp(\Delta_k)$ we observe that $\mathcal{Z}'_{k,C} = (Z'_{k,C}, A, \longrightarrow_{\mathcal{Z}'_{k,C}}) \in Res^{\mathrm{c}}_{\mathrm{sp}}(s'_{k,C})$ part of $\mathcal{Z}_k$ must match $\mathcal{Z}_{k,C}$ as $s_{k,C} \sim_{\mathrm{PTr}} s'_{k,C}$ implies $z_{s_{k,C}} \sim_{\mathrm{PTr}} z_{s'_{k,C}}$ due to the coherency of $\mathcal{Z}_k$.
   Starting from $a\,.\,\Delta_1$, we have that for all $\alpha = a\,\alpha' \in A^*$:

$$\begin{aligned} prob(\mathcal{CC}(z_{a\,.\,\Delta_1}, \alpha)) &= \sum_{C \cap supp(\Delta_1) \neq \emptyset} p_C \cdot prob(\mathcal{CC}(z_{s_{1,C}}, \alpha')) = \\ &= \sum_{C \cap supp(\Delta_2) \neq \emptyset} p_C \cdot prob(\mathcal{CC}(z_{s_{2,C}}, \alpha')) = prob(\mathcal{CC}(z_{a\,.\,\Delta_2}, \alpha)) \end{aligned}$$

   where the existence of $\mathcal{Z}_{2,C} = (Z_{2,C}, A, \longrightarrow_{\mathcal{Z}_{2,C}}) \in Res^{\mathrm{c}}_{\mathrm{sp}}(s_{2,C})$ matching $\mathcal{Z}_{1,C}$ with respect to all traces is a consequence of the existence – mentioned at the beginning of the proof – of $s_{2,C} \in supp(\Delta_2)$ such that $s_{1,C} \sim_{\mathrm{PTr}} s_{2,C}$. Therefore $\mathcal{Z}_2 = (Z_2, A, \longrightarrow_{\mathcal{Z}_2}) \in Res^{\mathrm{c}}_{\mathrm{sp}}(s_2)$, which starts with an $a$-transition and continues as $\mathcal{Z}_{2,C}$ for $s_{2,C} \in C \cap supp(\Delta_2)$, matches $\mathcal{Z}_1$ with respect to all traces.
3. If $\mathcal{L}$ is fully probabilistic, then it has a single maximal resolution, which (coincides with $\mathcal{L}$ itself if $\mathcal{L}$ is acyclic and) is the one on which the probabilities of all the traces are computed when verifying $\sim^{\mathrm{fp}}_{\mathrm{PTr}}$. Any of the other resolutions, which is considered only when verifying $\sim_{\mathrm{PTr}}$, is obtained by stopping earlier the execution of $\mathcal{L}$, in a way that is coherent along all branches of the resolution not only in terms of transition selection thanks to condition 1 of Def. 10, but also in terms of complete presence of identically labeled computations by virtue of condition 2 of Def. 10. ■