FACULTAS ARTIS INFORMATICÆ • UNIVERSITAS MASARYKIANA •

FI MU

Faculty of Informatics
Masaryk University Brno

# Probabilistic Bisimulation: Naturally on Distributions

by

Holger Hermanns
Jan Krčál
Jan Křetínský

FI MU Report Series FIMU-RS-2014-3

April 2014

Publications in the FI MU Report Series are in general accessible via WWW:

http://www.fi.muni.cz/reports/

Further information can be obtained by contacting:

Faculty of Informatics
Masaryk University
Botanická 68a
602 00 Brno
Czech Republic

# Probabilistic Bisimulation: Naturally on Distributions

Holger Hermanns
Saarland University – Computer Science, Saarbrücken, Germany
hermanns@cs.uni-saarland.de

Jan Krčál
Saarland University – Computer Science, Saarbrücken, Germany
krcal@cs.uni-saarland.de

Jan Křetínský
IST Austria
jan.kretinsky@ist.ac.at

April 20, 2014

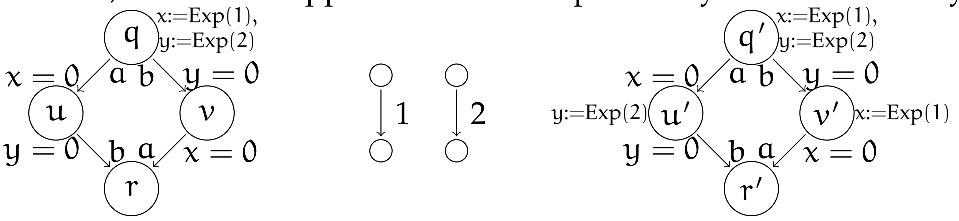
**Abstract**

In contrast to the usual understanding of probabilistic systems as stochastic processes, recently these systems have also been regarded as transformers of probabilities. In this paper, we give a *natural* definition of strong bisimulation for probabilistic systems corresponding to this view that treats probability *distributions* as first-class citizens. Our definition applies in the same way to discrete systems as well as to systems with uncountable state and action spaces. Several examples demonstrate that our definition refines the understanding of behavioural equivalences of probabilistic systems. In particular, it solves a longstanding open problem concerning the representation of memoryless continuous time by memoryfull continuous time. Finally, we give algorithms for computing this bisimulation not only for finite but also for classes of uncountably infinite systems.

# 1 Introduction

Continuous time concurrency phenomena can be addressed in two principal manners: On the one hand, *timed automata* (TA) extend interleaving concurrency with real-valued clocks [2]. On the other hand, time can be represented by memoryless stochastic time, as in *continuous time Markov chains* (CTMC) and extensions, where time is represented in the form of exponentially distributed random delays [37, 35, 6, 26]. TA and CTMC variations have both been applied to very many intriguing cases, and are supported by powerful real-time, respectively stochastic time model checkers [3, 42] with growing user bases. The models are incomparable in expressiveness, but if one extends timed automata with the possibility to sample from exponential distributions [5, 12, 33], there appears to be a natural bridge from CTMC to TA. This kind of stochastic semantics of timed automata has recently gained considerable popularity by the statistical model checking approach to TA analysis [16, 15].

Still there is a disturbing difference, and this difference is the original motivation [14] of the work presented in this paper. The obvious translation of an exponentially distributed delay into a clock expiration sampled from the very same exponential probability distribution fails in the presence of concurrency. This is because the translation is not fully compatible with the natural interleaving concurrency semantics for TA respectively CTMC. This is illustrated by the following example, which in the middle displays two small CTMC, which are supposed to run independently and concurrently.



On the left and right we see two stochastic automata (a variation of timed automata formally defined in Section 3). They have clocks $x$ and $y$ which are initialized by sampling from exponential distributions, and then each run down to $0$. The first one reaching $0$ triggers a transition and the other clock keeps on running unless resampled, which happens on the right, but not on the left. The left model is obtained by first translating the respective CTMC, and then applying the natural TA interleaving semantics, while the right model is obtained by first applying the equally natural CTMC interleaving semantics prior to translation.

The two models have subtly different semantics in terms of their underlying dense probabilistic timed transition systems. This can superficially be linked to the memoryless property of exponential distributions, yet there is no formal basis for proving equivalence. This paper closes this gap, which has been open for at least 15 years, by introducing a natural *continuous-space distribution-based* bisimulation. This result is embedded in several further intriguing application contexts and algorithmic achievements for this novel bisimulation.

The theory of bisimulations is a well-established and elegant framework to describe equivalence between processes based on their behaviour. In the standard semantics of probabilistic systems [43, 51], when a probabilistic step from a state to a distribution is taken, the random choice is resolved and we instead continue from one of the successor states. Recently, there has been considerable interest in instead regarding probabilistic systems as deterministic transformers of probability *distributions* [41, 1, 23], where the choice is not resolved and we continue from the distribution over successors. Thus, instead of the current state the transition changes the current distribution over the states. Although the distribution semantics is very natural in many contexts [34], it has been only partially reflected in the study of bisimulations [34, 22, 27, 26].

Our definition arises as an unusual, but very simple instantiation of the standard coalgebraic framework for bisimulations [48]. (No knowledge of coalgebra is required from the reader though.) Despite its simplicity, the resulting notion is surprisingly fruitful, not only because it indeed solves the longstanding correspondence problem between CTMC and TA with stochastic semantics.

Firstly, it is more adequate than other equivalences when applied to systems with distribution semantics, including large-population models where different parts of the population act differently [44]. Indeed, as argued in [30], some equivalent states are not identified in the standard probabilistic bisimulations and too many are identified in the recent distribution based bisimulations [22, 27]. Our approach allows for a bisimulation identifying precisely the desired states [30].

Secondly, our bisimulation over distributions induces an equivalence *on states*, and this relation equates behaviourally indistinguishable states which in many settings are unnecessarily distinguished by standard bisimulations. We shall discuss this phenomenon in the context of several applications. Nevertheless, the key idea to work with distributions instead of single states also bears disadvantages. The main difficulty is that even for finite systems the space of distributions is uncountable, thus bisimulation is difficult to compute. However, we show that it admits a concise representation using methods of linear algebra and we provide an algorithm for computing it. Further, in order to cover e.g. continuous-time systems, we need to handle both uncountably many states (that store the sampled time) and labels (real time durations). Fortunately, there is an elegant way to do so using the standard coalgebra framework. Moreover, it can easily be further generalized, e.g. adding rewards to the generic definition is a trivial task.

**Our contribution** is the following:

- We give a natural definition of bisimulation from the distribution perspective for systems with generally uncountable spaces of states and labels.
- We argue by means of several applications that the definition can be considered more useful than the classical notions of probabilistic bisimulation.
- We provide an algorithm to compute this distributional bisimulation on finite non-deterministic probabilistic systems, and present a decision algorithm for uncountable continuous-time systems induced by the stochastic automata mentioned above.

A full version of this paper is available Appendix.

## 2 Probabilistic bisimulation on distributions

A (potentially uncountable) set $S$ is a *measurable space* if it is equipped with a $\sigma$-algebra, which we denote by $\Sigma(X)$. The elements of $\Sigma(X)$ are called *measurable sets*. For a measurable space $S$, let $\mathcal{D}(S)$ denote the set of *probability measures* (or *probability distributions*) over $S$. The following definition is similar to the treatment of [58].

**Definition 2.1.** *A* non-deterministic labelled Markov process *(NLMP) is a tuple* $\mathbf{P} = (S, L, \{\tau_a \mid a \in L\})$ *where S is a measurable space of* states*, L is a measurable space of* labels*, and* $\tau_a : S \to \Sigma(\mathcal{D}(S))$ *assigns to each state* $s$ *a measurable set of probability measures* $\tau_a(s)$ *available in* $s$ *under* $a$.[(1)]

When in a state $s \in S$, NLMP reads a label $a \in L$ and *non-deterministically* chooses a successor distribution $\mu \in \mathcal{D}(S)$ that is in the set of convex combinations[(2)] over $\tau_a(s)$,

[(1)] We further require that for each $s \in S$ we have $\{(a, \mu) | \mu \in \tau_a(s)\} \in \Sigma(L) \otimes \Sigma(\mathcal{D}(S))$ and for each $A \in \Sigma(L)$ and $Y \in \Sigma(\mathcal{D}(S))$ we have $\{s \in S \mid \exists a \in A. \tau_a(s) \cap Y \neq \emptyset\} \in \Sigma(S)$. Here $\Sigma(\mathcal{D}(S))$ is the Giry $\sigma$-algebra [32] over $\mathcal{D}(X)$.

[(2)] A distribution $\mu \in \mathcal{D}(S)$ is a *convex combination* of a set $M \in \Sigma(\mathcal{D}(S))$ of distributions if there is a measure $\nu$ on $\mathcal{D}(S)$ such that $\nu(M) = 1$ and $\mu = \int_{\mu' \in \mathcal{D}(S)} \mu' \nu(d\mu')$.

denoted by $s \xrightarrow{a} \mu$. If there is no such distribution, the process halts. Otherwise, it moves into a successor state according to $\mu$. Considering convex combinations is necessary as it gives more power than pure resolution of non-determinism [49].

**Example 2.2.** *If all sets are finite, we obtain* probabilistic automata (PA) *defined [49] as a triple* $(S, L, \longrightarrow)$ *where* $\longrightarrow \subseteq S \times L \times \mathcal{D}(S)$ *is a probabilistic transition relation with* $(s, a, \mu) \in \longrightarrow$ *if* $\mu \in \tau_a(s)$.

**Example 2.3.** *In the continuous setting, consider a random number generator that also remembers the previous number. We set* $L = [0, 1]$, $S = [0, 1] \times [0, 1]$ *and* $\tau_x(\langle \mathit{new}, \mathit{last} \rangle) = \{\mu_x\}$ *for* $x = \mathit{new}$ *and* $\emptyset$ *otherwise, where* $\mu_x$ *is the uniform distribution on* $[0, 1] \times \{x\}$. *If we start with a uniform distribution over* $S$, *the measure of successors under any* $x \in L$ *is* $0$. *Thus in order to get any information of the system we have to consider successors under sets of labels, e.g. intervals.*

For a measurable set $A \subseteq L$ of labels, we write $s \xrightarrow{A} \mu$ if $s \xrightarrow{a} \mu$ for some $a \in A$, and denote by $S_A := \{s \mid \exists \mu : s \xrightarrow{A} \mu\}$ the set of states having some outgoing label from $A$. Further, we can lift this to probability distributions by setting $\mu \xrightarrow{A} \nu$ if $\nu = \frac{1}{\mu(S_A)} \int_{s \in S_A} \nu_s \, \mu(\mathrm{d}\, s)$ for some measurable function assigning to each state $s \in S_A$ a measure $\nu_s$ such that $s \xrightarrow{A} \nu_s$. Intuitively, in $\mu$ we restrict to states that do not halt under $A$ and consider all possible combinations of their transitions; we scale up by $\frac{1}{\mu(S_A)}$ to obtain a distribution again.

**Example 2.4.** *In the previous example, let* $\upsilon$ *be the uniform distribution. Due to the independence of the random generator on previous values, we get* $\upsilon \xrightarrow{[0,1]} \upsilon$. *Similarly,* $\upsilon \xrightarrow{[0.1,0.2]} \upsilon_{[0.1,0.2]}$ *where* $\upsilon_{[0.1,0.2]}$ *is uniform on* $[0, 1]$ *in the first component and uniform on* $[0.1, 0.2]$ *in the second component, with no correlation.*

Using this notation, a non-deterministic and probabilistic system such as NLMP can be regarded as a non-probabilistic, thus solely non-deterministic, labelled transition system over the uncountable space of probability distributions. The natural bisimulation from this distribution perspective is as follows.

**Definition 2.5.** *Let* $(S, L, \{\tau_a \mid a \in L\})$ *be a NLMP and* $R \subseteq \mathcal{D}(S) \times \mathcal{D}(S)$ *be a symmetric relation. We say that* $R$ *is a (strong)* probabilistic bisimulation *if for each* $\mu \mathrel{R} \nu$ *and measurable* $A \subseteq L$

1. $\mu(S_A) = \nu(S_A)$, *and*

2. *for each* $\mu \xrightarrow{A} \mu'$ *there is a* $\nu \xrightarrow{A} \nu'$ *such that* $\mu' \mathrel{R} \nu'$.

*We set $\mu \sim \nu$ if there is a probabilistic bisimulation $\mathsf{R}$ such that $\mu \,\mathsf{R}\, \nu$.*

**Example 2.6.** *Considering Example 2.3, states $\{x\} \times [0,1]$ form a class of $\sim$ for each $x \in [0,1]$ as the old value does not affect the behaviour. More precisely, $\mu \sim \nu$ iff marginals of their first component are the same.*

**Naturalness.** Our definition of bisimulation is not created ad-hoc as it often appears for relational definitions, but is actually an instantiation of the standard bisimulation for a particular *coalgebra*. Although this aspect is not necessary for understanding the paper, it is another argument for naturalness of our definition. For reader's convenience, we present a short introduction to coalgebras and the formal definitions in Appendix. Here we only provide an intuitive explanation by example.

Non-deterministic labelled transition systems are essentially given by the transition function $S \to \mathcal{P}(S)^{L}$; given a state $s \in S$ and a label $a \in L$, we can obtain the set of the successors $\{s' \in S \mid s \xrightarrow{a} s'\}$. The transition function corresponds to a coalgebra, which induces a bisimulation coinciding with the classical one of Park and Milner [46]. Similarly, PA are given by the transition function $S \to \mathcal{P}(\mathcal{D}(S))^{L}$; instead of successors there are distributions over successors. Again, the corresponding coalgebraic bisimulation coincides with the classical ones of Larsen and Skou [43] and Segala and Lynch [50].

In contrast, our definition can be obtained by considering states $S'$ to be distributions in $\mathcal{D}(S)$ over the original state space and defining the transition function to be $S' \to ([0,1] \times \mathcal{P}(S'))^{\Sigma(L)}$. The difference to the standard non-probabilistic case is twofold: firstly, we consider all measurable sets of labels, i.e. all elements of $\Sigma(L)$; secondly, for each label set we consider the mass, i.e. element of $[0,1]$, of the current state distribution that does not deadlock, i.e. can perform some of the labels. These two aspects form the crux of our approach and distinguish it from other approaches.

# 3 Applications

We now argue by some concrete application domains that the distribution view on bisimulation yields a fruitful notion.

## 3.1 Memoryless vs. memoryfull continuous time.

First, we reconsider the motivating discussion from Section 1 revolving around the difference between continuous time represented by real-valued clocks, respectively memoryless stochastic time. For this we introduce a simple model of *stochastic automata* [12].

**Definition 3.1.** *A* stochastic automaton (SA) *is a tuple* $\mathcal{S} = (\mathcal{Q}, \mathcal{C}, \mathcal{A}, \rightarrow, \kappa, \mathsf{F})$ *where* $\mathcal{Q}$ *is a set of locations,* $\mathcal{C}$ *is a set of clocks,* $\mathcal{A}$ *is a set of actions,* $\rightarrow \subseteq \mathcal{Q} \times \mathcal{A} \times 2^{\mathcal{C}} \times \mathcal{Q}$ *is a set of edges,* $\kappa : \mathcal{Q} \rightarrow 2^{\mathcal{C}}$ *is a clock setting function, and* $\mathsf{F}$ *assigns to each clock its distribution over* $\mathbb{R}_{\geq 0}$*.*

Avoiding technical details, $\mathcal{S}$ has the following NLMP semantics $\mathbf{P}_{\mathcal{S}}$ with state space $S = \mathcal{Q} \times (\mathbb{R}_{\geq 0})^{\mathcal{C}}$, assuming it is initialized in some location $q_0$: When a location $q$ is entered, for each clock $c \in \kappa(q)$ a positive *value* is chosen randomly according to the distribution $\mathsf{F}(c)$ and stored in the state space. Intuitively, the automaton idles in location $q$ with all all clock values decreasing at the same speed until some edge $(q, a, X, q')$ becomes *enabled*, i.e. all clocks from $X$ have value $\leq 0$. After this *idling time* $t$, the action $a$ is taken and the automaton enters the next location $q'$. If an edge is enabled on entering a location, it is taken immediately, i.e. $t = 0$. If more than one edge become enabled simultaneously, one of them is chosen non-deterministically. Its formal definition is given in Appendix. We now are in the position to harvest Definition 2.5, to arrive at the novel bisimulation for stochastic automata.

**Definition 3.2.** *We say that locations* $q_1, q_2$ *of an SA* $\mathcal{S}$ *are* probabilistic bisimilar*, denoted* $q_1 \sim q_2$*, if* $\mu_1 \sim \mu_2$ *in* $\mathbf{P}_{\mathcal{S}}$ *where each* $\mu_i$ *corresponds to the location being* $q_i$*, any* $c \notin \kappa(q_i)$ *being* $0$*, and any* $c \in \kappa(q_i)$ *being independently set to a random value according to* $\mathsf{F}(c)$*.*

This bisimulation identifies $q$ and $q'$ from Section 1 unlike any previous bisimulation on SA [12]. In Section 4.2 we discuss how to compute this bisimulation, despite being continuous-space. Recall that the model initialized by $q$ is obtained by first translating two simple CTMC, and then applying the natural interleaving semantics, while the model, of $q'$ is obtained by first applying the equally natural CTMC interleaving semantics prior to translation. The bisimilarity of these two models generalizes to the whole universe of CTMC and SA:

**Theorem 3.3.** *Let* $\mathsf{SA}(\mathcal{C})$ *denote the stochastic automaton corresponding to a CTMC* $\mathcal{C}$*. For any CTMC* $\mathcal{C}_1, \mathcal{C}_2$*, we have*

$$\mathsf{SA}(\mathcal{C}_1) \parallel_{\mathsf{SA}} \mathsf{SA}(\mathcal{C}_1) \ \sim \ \mathsf{SA}(\mathcal{C}_1 \parallel_{\mathsf{CT}} \mathcal{C}_1).$$

Here, $\|_{\mathrm{CT}}$ and $\|_{\mathrm{SA}}$ denotes the interleaving parallel composition of SA [13] (echoing TA parallel composition) and CTMC [37, 35] (Kronecker sum of their matrix representations), respectively.

### 3.2 Bisimulation for partial-observation MDP (POMDP).

A POMDP is a quadruple $\mathcal{M} = (S, L, \delta, \mathcal{O})$ where (as in an MDP) $S$ is a set of states, $A$ is a set of actions, and $\delta : S \times \mathcal{A} \to \mathcal{D}(S)$ is a transition function. Furthermore, $\mathcal{O} \subseteq 2^S$ partitions the state space. The choice of actions is resolved by a policy yielding a Markov chain. Unlike in an MDP, such choice is not based on the knowledge of the current state, only on knowing that the current state belongs into an *observation* $o \in \mathcal{O}$. POMDPs have a wide range of applications in robotic control, automated planning, dialogue systems, medical diagnosis, and many other areas [52].

In the analysis of POMDP, the distributions over states, called *beliefs*, arise naturally. They allow for transforming the POMDP $\mathcal{M}$ into a fully observable NLMP $D_{\mathcal{M}} = (S, \mathcal{O}, \longrightarrow)$ with continuous space, by setting $(s, \xrightarrow{o}, \mu) \in \longrightarrow$ if $s \in o$ and $\delta(s, a) = \mu$ for some $a \in A$. Although probabilistic bisimulations over beliefs have been already considered [7, 38], no connection of this particular case to general probabilistic bisimulation has been studied. We can set $\mu \sim \mu'$ in $\mathcal{M}$ if $\mu \sim \mu'$ in $D_{\mathcal{M}}$. In Section 4.1, we shall provide an algorithm for computing bisimulations over beliefs in finite POMDP. Previously, there was only an algorithm [38] for computing bisimulations on distributions of Markov *chains* with partial observation.

### 3.3 Further applications.

Probabilistic automata are especially apt for compositional modelling of *distributed systems*. The only information a component in a distributed system has about the current state of another component stems from their mutual communication. Therefore, each component can be also viewed from the outside as a partial-observation system. Thus, also in this context, distribution bisimulation is a natural concept.

Furthermore we can understand a PA as a description, in the sense of [28, 44], of a representative *agent* in a large homogeneous *population*. The distribution view then naturally represents the ratios of agents being currently in the individual states and labels given to this large population of PAs correspond to global control actions [28]. For more details on applications, see Appendix.

# 4 Algorithms

In this section, we discuss computational aspects of deciding our bisimulation. Since $\sim$ is a relation over distributions over the system's state space, it is uncountably infinite even for simple finite systems, which makes it in principle intricate to decide. Fortunately, the bisimulation relation has a linear structure, and this allows us to employ methods of linear algebra to work with it effectively. Moreover, important classes of continuous-space systems can be dealt with, since their structure can be exploited. We exemplify this on a subset of deterministic stochastic automata, for which we are able to provide an algorithm to decide bisimilarity.

## 4.1 Finite systems – greatest fixpoints

Let us fix a PA $(S, L, \longrightarrow)$. We apply the standard approach by starting with $\mathcal{D}(S) \times \mathcal{D}(S)$ and pruning the relation until we reach the fixpoint $\sim$. In order to represent $\sim$ using linear algebra, we identify a distribution $\mu$ with a vector $(\mu(s_1), \dots, \mu(s_{|S|})) \in \mathbb{R}^{|S|}$.

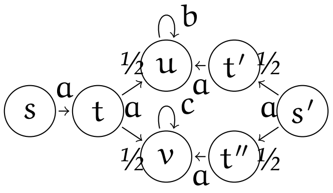
Although the space of distributions is uncountable, we construct an implicit representation of $\sim$ by a system of equations written as columns in a matrix $E$.

**Definition 4.1.** *A matrix* $E$ *with* $|S|$ *rows is a* bisimulation matrix *if for some bisimulation* $R$, *for any distributions* $\mu, \nu$

$$\mu \, R \, \nu \quad \textit{iff} \quad (\mu - \nu)E = 0.$$

For a bisimulation matrix $E$, an equivalence class of $\mu$ is then the set $(\mu + \{\rho \mid \rho E = 0\}) \cap \mathcal{D}(S)$, the set of distributions that are equal modulo $E$.

**Example 4.2.** *The bisimulation matrix* $E$ *below encodes that several conditions must hold for two distributions* $\mu, \nu$ *to be bisimilar. Among others, if we multiply* $\mu - \nu$ *with e.g. the second column, we must get* $0$. *This translates to* $(\mu(v) - \nu(v)) \cdot 1 = 0$, *i.e.* $\mu(v) = \nu(v)$. *Hence for bisimilar distributions, the measure of* $v$ *has to be the same. This proves that* $u \not\sim v$ *(here we identify states and their Dirac distributions). Similarly, we can prove that* $t \sim \frac{1}{2}t' + \frac{1}{2}t''$. *Indeed, if we multiply the corresponding difference vector* $(0, 0, 1, -\frac{1}{2}, -\frac{1}{2}, 0, 0)$ *with any column of the matrix, we obtain* $0$.



$$\begin{array}{r}s: \\ s': \\ t: \\ t': \\ t'': \\ u: \\ v:\end{array}\begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & ½ & ½ \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

Note that the unit matrix is always a bisimulation matrix, not relating anything with anything but itself. For which bisimulations do there exist bisimulation matrices? We say a relation $R$ over distributions is *linear* if $\mu R \nu$ and $\mu' R \nu'$ imply $\big(p\mu + (1-p)\mu'\big)\ R\ \big(p\nu + (1-p)\nu'\big)$ for any $p \in [0,1]$.

**Lemma 4.3.** *For every linear bisimulation there exists a corresponding bisimulation matrix.*

Since $\sim$ is linear (see Appendix), there is a bisimulation matrix corresponding to $\sim$. It is a least restrictive bisimulation matrix $E$ (note that all bisimulation matrices with the least possible dimension have identical solution space), we call it *minimal bisimulation matrix*. We show that the necessary and sufficient condition for $E$ to be a bisimulation matrix is *stability* with respect to transitions.

**Definition 4.4.** *For a $|S| \times |S|$ matrix $P$, we say that a matrix $E$ with $|S|$ rows is* $P$-stable *if for every $\rho \in \mathbb{R}^{|S|}$,*

$$\rho E = 0 \implies \rho P E = 0 \tag{1}$$

We first briefly explain the stability in a simpler setting.

### 4.1.1 Action-deterministic systems.

Let us consider PA where in each state, there is at most one transition. For each $a \in L$, we let $P_a = (p_{ij})$ denote the transition matrix such that for all $i, j$, if there is (unique) transition $s_i \xrightarrow{a} \mu$ we set $p_{ij}$ to $\mu(s_j)$, otherwise to $0$. Then $\mu$ evolves under $a$ into $\mu P_a$. Denote $\mathbf{1} = (1, \ldots, 1)^\top$.

**Proposition 4.5.** *In an action-deterministic PA, $E$ containing $\mathbf{1}$ is a bisimulation matrix iff it is $P_a$-stable for all $a \in L$.*

To get a minimal bisimulation matrix $E$, we start with a single vector $\mathbf{1}$ which stands for an equation saying that the overall probability mass in bisimilar distributions is the same. Then we repetitively multiply all vectors we have by all the matrices $P_a$ and add each resulting vector to the collection if it is linearly independent of the current collection, until there are no changes. In Example 4.2, the second column of $E$ is obtained as $P_c\mathbf{1}$, the fourth one as $P_a(P_c\mathbf{1})$ and so on.

The set of all columns of $E$ is thus given by the described iteration

$$\{P_a \mid a \in L\}^*\mathbf{1}$$

modulo linear dependency. Since $P_a$ have $|S|$ rows, the fixpoint is reached within $|S|$ iterations yielding $1 \le d \le |S|$ equations. Each class then forms an $(|S|-d)$-dimensional affine subspace intersected with the set of probability distributions $\mathcal{D}(S)$. This is also the principle idea behind the algorithm of [57] and [22].

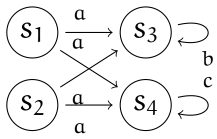
### 4.1.2 Non-deterministic systems.

In general, for transitions under $A$, we have to consider $c_i^A$ non-deterministic choices in each $s_i$ among all the outgoing transitions under some $a \in A$. We use variables $w_i^j$ denoting the probability that j-th transition, say $(s_i, a_i^j, \mu_i^j)$, is taken by the scheduler/player[(3)] in $s_i$. We sum up the choices into a "non-deterministic" transition matrix $P_A^W$ with parameters $W$ whose $i$th row equals $\sum_{j=1}^{c_i^A} w_i^j \mu_i^j$. It describes where the probability mass moves from $s_i$ under $A$ depending on the collection $W$ of the probabilities the player gives each choice. By $\mathcal{W}_A$ we denote the set of all such $W$.

A simple generalization of the approach above would be to consider $\{P_A^W \mid A \subseteq L, W \in \mathcal{W}_A\}^*\mathbf{1}$. However, firstly, the set of these matrices is uncountable whenever there are at least two transitions to choose from. Secondly, not all $P_A^W$ may be used as the following example shows.

**Example 4.6.** *In each bisimulation class in the following example, the probabilities of* $s_1+s_2$, $s_3$, *and* $s_4$ *are constant, as can also be seen from the bisimulation matrix* $E$, *similarly to Example 4.2. Further,* $E$ *can be obtained as* $(\mathbf{1}\ \ P_c\mathbf{1}\ \ P_b\mathbf{1})$. *Observe that* $E$ *is* $P_{\{a\}}^W$*-stable for* $W$ *that maximizes the probability of going into the "class"* $s_3$ *(both* $s_1$ *and* $s_2$ *go to* $s_3$, *i.e.* $w_1^1 = w_2^1 = 1$*); similarly for the "class"* $s_4$.

[(3)] We use the standard notion of Spoiler-Duplicator bisimulation game (see e.g. [48]) where in $\{\mu_0, \mu_1\}$ Spoiler chooses $i \in \{0,1\}$, $A \subseteq L$, and $\mu_i \xrightarrow{A} \mu_i'$, Duplicator has to reply with $\mu_{1-i} \xrightarrow{A} \mu_{1-i}'$ such that $\mu_i(S_A) = \mu_{i-1}(S_A)$, and the game continues in $\{\mu_0', \mu_1'\}$. Spoiler wins iff at some point Duplicator cannot reply.



$$P^W_{\{a\}} = \begin{pmatrix} 0 & 0 & w_1^1 & w_2^2 \\ 0 & 0 & w_2^1 & w_2^2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \qquad E = \begin{pmatrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}$$

*However, for* $W$ *with* $w_1^1 \neq w_2^1$*, e.g.* $s_1$ *goes to* $s_3$ *and* $s_2$ *goes with equal probability to* $s_3$ *and* $s_4$ *(*$w_1^1 = 1, w_2^1 = w_2^2 = \frac{1}{2}$*), we obtain from* $P^W_{\{a\}}E$ *a new independent vector* $(0, 0.5, 0, 0)^\top$ *enforcing a partition finer than* $\sim$*. This does not mean that Spoiler wins the game when choosing such mixed* $W$ *in some* $\mu$*, it only means that Duplicator needs to choose a* different $W'$ *in a bisimilar* $\nu$ *in order to have* $\mu P^W_A \sim \nu P^{W'}_A$ *for the successors.*

A fundamental observation is that we get the correct bisimulation when Spoiler is restricted to finitely many "extremal" choices and Duplicator is restricted for such extremal $W$ to respond only with the very same $W$.

To this end, consider $M^W_A = P^W_A E$ where $E$ is the current matrix with each of $e$ columns representing an equation. Intuitively, the $i$th row of $M^W_A$ describes how much of $s_i$ is moved to various classes when a step is taken. Denote the linear forms in $M^W_A$ over $W$ by $m_{ij}$. Since the players can randomize and mix choices which transition to take, the set of vectors $\{(m_{i1}(w_i^1, \ldots, w_i^{c_i}), \ldots, m_{ib}(w_i^1, \ldots, w_i^{c_i})) \mid w_i^1, \ldots, w_i^{c_i} \geq 0, \sum_{j=1}^{c_i} w_i^j = 1\}$ forms a convex polytope denoted by $C_i$. Each vector in $C_i$ is thus the $i$th row of the matrix $M^W_A$ where some concrete weights $w_i^j$ are "plugged in". This way $C_i$ describes all the possible choices in $s_i$ and their effect on where the probability mass is moved.

Denote vertices (extremal points) of a convex polytope $P$ by $\mathcal{E}(P)$. Then $\mathcal{E}(C_i)$ correspond to pure (non-randomizing) choices that are "extremal" w.r.t. $E$. Note that now if $s_j \sim s_k$ then $C_j = C_k$, or equivalently $\mathcal{E}(C_j) = \mathcal{E}(C_k)$. Indeed, for every choice in $s_j$ there needs to be a matching choice in $s_k$ and vice versa. However, since we consider bisimulation between generally non-Dirac distributions, we need to combine these extremal choices. We define the set $\mathcal{E}(C) \subseteq \prod_{i=1}^{|S|} \mathcal{E}(C_i)$ to contain a tuple $c = (c_1 \cdots c_{|S|})$ iff the $c_i$'s are "extremal in (some) same direction", i.e. $\sum_{i=1}^{|S|} c_i$ is a vertex (extremal choice) of the polytope generated by points $\{\sum_{i=1}^{|S|} c_i' \mid \forall i : c_i' \in C_i\}$. Each $c \in \mathcal{E}(C)$ is a tuple of vertices, and thus corresponds to particular choices, denoted by $W(c)$.

**Proposition 4.7.** *Let* $E$ *be a matrix containing* $\mathbf{1}$*. It is a bisimulation matrix iff it is* $P^{W(c)}_A$*-stable for all* $A \subseteq L$ *and* $c \in \mathcal{E}(C)$*.*

```
Input    : Probabilistic automaton $(S, L, \rightarrow)$
Output   : A minimal bisimulation matrix $E$
foreach $A \subseteq L$ do
    compute $P_A^W$                                   //non-deterministic transition matrix
$E \leftarrow (\mathbf{1})$
repeat
    foreach $A \subseteq L$ do
        $M_A^W \leftarrow P_A^W E$                    //polytope of all choices
        compute $\mathcal{E}(C)$ from $M_A^W$          //vertices, i.e. extremal choices
        foreach $c \in \mathcal{E}(C)$ do
            $M_A^{W(c)} \leftarrow M_A^W$ with values $W(c)$ plugged in
            $E_{new} \leftarrow$columns of $M_A^{W(c)}$ linearly independent of columns of $E$
            $E \leftarrow (E\ E_{new})$
until $E$ does not change
```

**Algorithm 1:** Bisimulation on probabilistic automata

**Theorem 4.8.** *Algorithm 1 computes a minimal bisimulation matrix.*

The running time is exponential. We leave the question whether linear programming or other methods [36] can yield $E$ in polynomial time open. The algorithm can easily be turned into one computing other bisimulation notions from the literature, for which there were no algorithms so far, see Section 5.

## 4.2 Continuous-time systems - least fixpoints

Turning our attention to continuous systems, we finally sketch an algorithm for deciding bisimulation $\sim$ over a subclass of stochastic automata, this constitutes the first algorithm to compute a bisimulation on the uncountably large semantical object.

We need to adopt two restrictions. First, we consider only *deterministic* SA, where the probability that two edges become enabled at the same time is zero (when initiated in any location). Second, to simplify the exposition, we restrict all distributions occurring to exponential distributions. Notably, even for this class, our bisimulation is strictly coarser than the one induced by standard bisimulations [37, 35, 6] for continuous-time Markov chains. At the end of the section we discuss possibilities for extending the class of supported distributions. Both the restrictions can be effectively checked on SA.

**Theorem 4.9.** *Let $\mathcal{S} = (\mathcal{Q}, \mathcal{C}, \mathcal{A}, \rightarrow, \kappa, F)$ be a deterministic SA over exponential distributions. There is an algorithm to decide in time polynomial in $|\mathcal{S}|$ and exponential in $|\mathcal{C}|$ whether $q_1 \sim q_2$ for any locations $q_1, q_2$.*

The rest of the section deals with the proof. We fix $\mathcal{S} = (\mathcal{Q}, \mathcal{C}, \mathcal{A}, \rightarrow, \kappa, F)$ and $q_1, q_2 \in \mathcal{Q}$. First, we straightforwardly abstract the NLMP semantics $\mathbf{P}_{\mathcal{S}}$ by a NLMP $\hat{\mathbf{P}}$ over state space $\hat{S} = \mathcal{Q} \times (\mathbb{R}_{\geq 0} \cup \{-\})^{\mathcal{C}}$ where all negative values of clocks are expressed by one element $-$. Let $\xi$ denote the obvious mapping of distributions $\mathcal{D}(S)$ onto $\mathcal{D}(\hat{S})$. Then $\xi$ preserves bisimulation since two states $s_1, s_2$ that differ only in negative values satisfy $\xi(\tau_a(s_1)) = \xi(\tau_a(s_2))$ for all $a \in L$.

**Lemma 4.10.** *For any distributions $\mu, \nu$ on $S$ we have $\mu \sim \nu$ iff $\xi(\mu) \sim \xi(\nu)$.*

Second, similarly to an embedded Markov chain of a CTMC, we further abstract the NLMP $\hat{\mathbf{P}}$ by a *finite* deterministic PA $\bar{D} = (\bar{S}, \mathcal{A}, \longrightarrow)$ such that each state of $\bar{D}$ is a distribution over the uncountable state space $\hat{S}$.

- The set $\bar{S}$ is the set of states reachable via the transitions relation defined below from the distributions $\mu_1, \mu_2$ corresponding to $q_1$, $q_2$ (see Definition 3.2).
- Let us fix a state $\mu \in \bar{S}$ (note that $\mu \in \mathcal{D}(\hat{S})$) and an action $a \in \mathcal{A}$ such that in the NLMP $\hat{\mathbf{P}}$ an $a$-transition occurs with positive probability, i.e. $\mu \xrightarrow{A_a} \nu$ for some $\nu$ and for $A_a = \{a\} \times \mathbb{R}_{\geq 0}$. Thanks to restricting to deterministic SA, $\hat{\mathbf{P}}$ is also deterministic and such a distribution $\nu$ is uniquely defined. We set $(\mu, a, M) \in \longrightarrow$ where $M$ is the discrete distribution that assigns probability $p_{q,f}$ to state $\nu_{q,f}$ for each $q \in \mathcal{Q}$ and $f : \mathcal{C} \rightarrow \{-, +\}$ where $p_{q,f} = \nu(\hat{S}_{q,f})$, $\nu_{q,f}$ is the conditional distribution $\nu_q(X) := \nu(X \cap \hat{S}_{q,f})/\nu(\hat{S}_{q,f})$ for any measurable $X \subseteq \hat{S}$, and $\hat{S}_{q,f} = \{(q', \nu) \in \hat{S} \mid q' = q, \nu(c) \geq 0 \text{ iff } f(c) = + \text{ for each } c \in \mathcal{C}\}$ the set of states with location $q$ and where the sign of clock values matches $f$.

For exponential distributions all the reachable states $\nu \in \bar{S}$ correspond to some location $q$ where the subset $X \subseteq \mathcal{C}$ is newly sampled, hence we obtain:

**Lemma 4.11.** *For a deterministic SA over exponential distributions, $|\bar{S}| \leq |\mathcal{Q}| 2^{|\mathcal{C}|}$.*

Instead of a greatest fixpoint computation as employed for the discrete algorithm, we take a complementary approach and prove or disprove bisimilarity by a least fixpoint procedure. We start with the initial pair of distributions (states in $\bar{D}$) which generates further requirements that we impose on the relation and try to satisfy them. We work with a *tableau*, a rooted tree where each node is either an *inner node* with a pair of discrete probability distributions over states of $\bar{D}$ as a label, a *repeated node* with a label that already appears somewhere between the node and the root, or a *failure node* denoted by $\square$, and the children of each inner node are obtained by one *rule* from $\{\textbf{Step}, \textbf{Lin}\}$. A tableau not containing $\square$ is *successful*.

**Step** For a node $\mu \sim \nu$ where $\mu$ and $\nu$ have *compatible timing*, we add for each label $a \in L$ one child node $\mu_a \sim \nu_a$ where $\mu_a$ and $\nu_a$ are the unique distributions such that $\mu \xrightarrow{a} \mu_a$ and $\nu \xrightarrow{a} \nu_a$. Otherwise, we add one failure node. We say that $\mu$ and $\nu$ have compatible timing if for all actions $a \in \mathcal{A}$ we have $\mu(S_{A_a}) = \nu(S_{A_a})$ and if for all actions $a \in \mathcal{A}$ with $\mu(S_{A_a}) > 0$ we have that $\mu$ restricted to $S_{A_a}$ is equivalent to $\nu$ restricted to $S_{A_a}$.

**Lin** For a node $\mu \sim \nu$ linearly dependent on the set of remaining nodes in the tableau, we add one child (repeat) node $\mu \sim \nu$. Here, we understand each node $\mu \sim \nu$ as a vector $\mu - \nu$ in the $|S_\mathcal{S}|$-dimensional vector space.

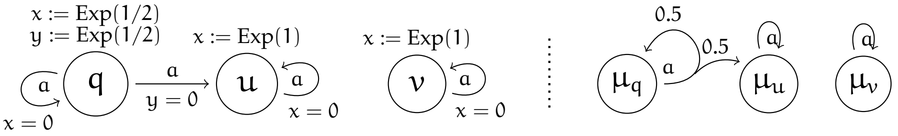
Note that compatibility of timing is easy to check. Furthermore, the set of rules is correct and complete w.r.t. bisimulation in $\hat{\mathbf{P}}$.

**Lemma 4.12.** *There is a successful tableau from $\mu \sim \nu$ iff $\mu \sim \nu$ in $\hat{\mathbf{P}}$. Moreover, the set of nodes of a successful tableau is a subset of a bisimulation.*

We get Theorem 4.9 since $q_1 \sim q_2$ iff $\xi(\mu_1) \sim \xi(\mu_2)$ in $\hat{\mathbf{P}}$ and since, thanks to **Lin**:

**Lemma 4.13.** *There is a successful tableau from $\mu \sim \nu$ iff there is a finite successful tableau from $\mu \sim \nu$ of size polynomial in $|\bar{S}|$.*

**Example 4.14.** *Let us demonstrate the rules by a simple example. Consider the following stochastic automaton $\mathcal{S}$ on the left.*



*Thanks to the exponential distributions, $\bar{D}$ on the right has also only three states where $\mu_q = q \otimes \mathsf{Exp}(1/2) \otimes \mathsf{Exp}(1/2)$ is the product of two exponential distributions with rate 1/2, $\mu_u = u \otimes \mathsf{Exp}(1)$, and $\mu_v = v \otimes \mathsf{Exp}(1)$. Note that for both clocks x and y, the probability of getting to zero first is 0.5.*

$$\dfrac{1\cdot\mu_u \sim 1\cdot\mu_v}{1\cdot\mu_u \sim 1\cdot\mu_v}\,\textit{\textbf{Step}} \qquad\qquad \begin{array}{c} 1\cdot\mu_q + 0\cdot\mu_u \sim 1\cdot\mu_v \\ \hline \frac{1}{2}\cdot\mu_q + \frac{1}{2}\cdot\mu_u \sim 1\cdot\mu_v \\ \hline \frac{1}{4}\cdot\mu_q + \frac{3}{4}\cdot\mu_u \sim 1\cdot\mu_v \\ \hline \ldots \end{array}\begin{array}{l} \textit{\textbf{Step}} \\ \textit{\textbf{Step}} \\ \textit{\textbf{Step}} \end{array}$$

*The finite tableau on the left is successful since it ends in a repeated node, thus it proves* $u \sim v$*. The infinite tableau on the right is also successful and proves* $q \sim v$*. When using only the rule* ***Step****, it is necessarily infinite as no node ever repeats. The rule* ***Lin*** *provides the means to truncate such infinite sequences. Observe that the third node in the tableau on the right above is linearly dependent on its ancestors.*

**Remark 4.15.** *Our approach can be turned into a complete proof system for bisimulation on models with* expolynomial *distributions* [4]*. Thanks to their properties, the states of the discrete transition system* $\bar{D}$ *can be expressed symbolically. In fact, we conjecture that the resulting semi-algorithm can be twisted to a decision algorithm for this expressive class of models. Being technically demanding, it is out of scope of this paper.*

# 5 Related work and discussion

For an overview of coalgebraic work on probabilistic bisimulations we refer to a survey [53]. A considerable effort has been spent to extend this work to continuous-space systems: the solution of [18] (unfortunately not applicable to $\mathbb{R}$), the construction of [24] (described by [48] as "ingenious and intricate"), sophisticated measurable selection techniques in [21], and further approaches of [20] or [58]. In contrast to this standard setting where relations between states and their successor distributions must be handled, our work uses directly relations on distributions which simplifies the setting. The coalgebraic approach has also been applied to trace semantics of uncountable systems [40]. Coalgebraic treatment of probabilistic bisimulation is still very lively [47].

Recently, distribution-based bisimulations have been studied. In [22], a bisimulation is defined in the context of language equivalence of Rabin's deterministic probabilistic automata and also an algorithm to compute the bisimulation on them. However, only finite systems with no non-determinism are considered. The most related to our notion are the very recent independently developed [27] and [55]. However, none of them is

[4] With density that is positive on an interval $[\ell, u)$ for $\ell \in \mathbb{N}_0$, $u \in \mathbb{N} \cup \{\infty\}$ given piecewise by expressions of the form $\sum_{i=0}^{I} \sum_{j=0}^{J} a_{ij} x^i e^{-\lambda_{ij} x}$ for $a_{ij}, \lambda_{ij} \in \mathbb{R} \cup \{\infty\}$. This class contains many important distributions such as exponential, or uniform, and enables efficient approximation of others.

applicable in the continuous setting and for neither of the two any algorithm has previously been given. Nevertheless, since they are close to our definition, our algorithm with only small changes can actually compute them. Although the bisimulation of [27] in a rather complex way extends [22] to the non-deterministic case reusing their notions, it can be equivalently rephrased as our Definition 2.5 only considering singleton sets $A \subseteq L$. Therefore, it is sufficient to only consider matrices $P_A^W$ for singletons $A$ in our algorithm. Apart from being a weak relation, the bisimulation of [55] differs in the definition of $\mu \xrightarrow{A} \nu$: instead of restricting to the states of the support that can perform *some* action of $A$, it considers those states that can perform *exactly* actions of $A$. Here each $i$th row of each transition matrix $P_A^W$ needs to be set to zero if the set of labels from $s_i$ is different from $A$.

There are also bisimulation relations over distributions that, however, coincide with the classical [43] on Dirac distributions and are only directly lifted to non-Dirac distributions. Thus they fail to address the motivating correspondence problem from Section 1 and are less precise for large-population models. Moreover, no algorithms were given. They were considered for finite [11, 34] and uncountable [8] state spaces.

There are other bisimulations that identify more states than the classical [43] such as [54] and [4] designed to match a specific logic. Further, weak bisimulations coarser than usual state based analogues were given in [26, 25, 19], which also inspires our work, especially their approach to internal transitions. However, they are quite different from our notion as in the case without internal transitions they basically coincide with lifting [34] of the classical bisimulation [43]. Another approach to obtain coarser equivalences on probabilistic automata is via testing scenarios [56].

# 6 Conclusion

We have introduced a general and natural notion of a distribution-based probabilistic bisimulation, have shown its applications in different settings and have provide algorithms to compute it for finite and some classes of infinite systems. As to future work, the precise complexity of the finite case is certainly of interest. Further, the tableaux decision method opens the arena for investigating wider classes of continuous-time systems where the new bisimulation is decidable.

**Acknowledgement** We would like to thank Pedro D'Argenio, Filippo Bonchi, Daniel Gebler, and Matteo Mio for valuable feedback and discussions.

# References

[1] M. Agrawal, S. Akshay, B. Genest, and P. Thiagarajan. Approximate verification of the symbolic dynamics of Markov chains. In *LICS*, 2012.

[2] R. Alur and D. Dill. A theory of timed automata. *Theor. Comput. Sci.*, 126(2):183–235, 1994.

[3] G. Behrmann, A. David, K. G. Larsen, P. Pettersson, and W. Yi. Developing uppaal over 15 years. *Softw., Pract. Exper.*, 41(2):133–142, 2011.

[4] M. Bernardo, R. D. Nicola, and M. Loreti. Revisiting bisimilarity and its modal logic for nondeterministic and probabilistic processes. Technical Report 06, IMT Lucca, 2013.

[5] M. Bravetti and P. D'Argenio. Tutte le algebre insieme: Concepts, discussions and relations of stochastic process algebras with general distributions. In *Validation of Stochastic Systems*, 2004.

[6] M. Bravetti, H. Hermanns, and J.-P. Katoen. YMCA: Why Markov Chain Algebra? *Electr. Notes Theor. Comput. Sci.*, 162:107–112, 2006.

[7] P. Castro, P. Panangaden, and D. Precup. Equivalence relations in fully and partially observable Markov decision processes. In *IJCAI*, 2009.

[8] S. Cattani. *Trace-based Process Algebras for Real-Time Probabilistic Systems*. PhD thesis, University of Birmingham, 2005.

[9] K. Chatterjee, L. Doyen, and T. Henzinger. Qualitative analysis of partially-observable Markov decision processes. In *MFCS*, 2010.

[10] L. Cheung. *Reconciling nondeterministic and probabilistic choices*. PhD thesis, Institute for Computing and Information Sciences, Radboud University Nijmegen, 2006.

[11] S. Crafa and F. Ranzato. A spectrum of behavioral relations over ltss on probability distributions. In *CONCUR*, 2011.

[12] P. D'Argenio and J.-P. Katoen. A theory of stochastic systems part I: Stochastic automata. *Inf. Comput.*, 203(1):1–38, 2005.

[13] P. D'Argenio and J.-P. Katoen. A theory of stochastic systems. part II: Process algebra. *Inf. Comput.*, 203(1):39–74, 2005.

[14] P. R. D'Argenio and C. Baier. What is the relation between CTMC and TA?, 1999. Personal communication.

[15] A. David, K. Larsen, A. Legay, M. Mikucionis, D. Poulsen, J. van Vliet, and Z. Wang. Statistical model checking for networks of priced timed automata. In *FORMATS*, 2011.

[16] A. David, K. Larsen, A. Legay, M. Mikucionis, and Z. Wang. Time for statistical model checking of real-time systems. In *CAV*, 2011.

[17] L. de Alfaro, T. Henzinger, and R. Jhala. Compositional methods for probabilistic systems. In *CONCUR*, 2001.

[18] E. de Vink and J. Rutten. Bisimulation for probabilistic transition systems: A coalgebraic approach. In *ICALP*, 1997.

[19] Y. Deng and M. Hennessy. On the semantics of Markov automata. *Inf. Comput.*, 222:139–168, 2013.

[20] J. Desharnais, V. Gupta, R. Jagadeesan, and P. Panangaden. Approximating labeled Markov processes. In *LICS*, 2000.

[21] E.-E. Doberkat. Semi-pullbacks and bisimulations in categories of stochastic relations. In *ICALP*, 2003.

[22] L. Doyen, T. Henzinger, and J.-F. Raskin. Equivalence of labeled Markov chains. *Int. J. Found. Comput. Sci.*, 19(3):549–563, 2008.

[23] L. Doyen, T. Massart, and M. Shirmohammadi. Limit synchronization in Markov decision processes. *CoRR*, abs/1310.2935, 2013.

[24] A. Edalat. Semi-pullbacks and bisimulation in categories of Markov processes. *Mathematical Structures in Computer Science*, 9(5):523–543, 1999.

[25] C. Eisentraut, H. Hermanns, J. Krämer, A. Turrini, and L. Zhang. Deciding bisimilarities on distributions. In *QEST*, 2013.

[26] C. Eisentraut, H. Hermanns, and L. Zhang. On probabilistic automata in continuous time. In *LICS*, 2010.

[27] Y. Feng and L. Zhang. When equivalence and bisimulation join forces in probabilistic automata. *CoRR*, abs/1311.3396, 2013.

[28] N. Gast and B. Gaujal. A mean field approach for optimization in discrete time. *Discrete Event Dynamic Systems*, 21(1):63–101, 2011.

[29] N. Gast, B. Gaujal, and J.-Y. L. Boudec. Mean field for Markov decision processes: From discrete to continuous optimization. *IEEE Trans. Automat. Contr.*, 57(9):2266–2280, 2012.

[30] S. Georgievska and S. Andova. Probabilistic may/must testing: retaining probabilities by restricted schedulers. *Formal Asp. Comput.*, 24(4-6):727–748, 2012.

[31] S. Giro and P. D'Argenio. Quantitative model checking revisited: Neither decidable nor approximable. In *FORMATS*, 2007.

[32] M. Giry. A categorical approach to probability theory. In *Categorical aspects of topology and analysis*. Springer, 1982.

[33] P. G. Harrison and B. Strulo. Spades - a process algebra for discrete event simulation. *J. Log. Comput.*, 10(1):3–42, 2000.

[34] M. Hennessy. Exploring probabilistic bisimulations, part i. *Formal Asp. Comput.*, 2012.

[35] H. Hermanns, U. Herzog, and V. Mertsiotakis. Stochastic process algebras - between lotos and markov chains. *Computer Networks*, 30(9-10):901–924, 1998.

[36] H. Hermanns and A. Turrini. Deciding probabilistic automata weak bisimulation in polynomial time. In *FSTTCS*, 2012.

[37] J. Hillston. *A Compositional Approach to Performance Modelling*. Cambridge University Press, New York, NY, USA, 1996.

[38] D. Jansen, F. Nielson, and L. Zhang. Belief bisimulation for hidden Markov models - logical characterisation and decision algorithm. In *NASA Formal Methods*, 2012.

[39] B. Jovanovic and R. Rosenthal. Anonymous sequential games. *Journal of Mathematical Economics*, 17(1):77–87, 1988.

[40] H. Kerstan and B. König. Coalgebraic trace semantics for probabilistic transition systems based on measure theory. In *CONCUR*, 2012.

[41] V. Korthikanti, M. Viswanathan, G. Agha, and Y. Kwon. Reasoning about mdps as transformers of probability distributions. In *QEST*, 2010.

[42] M. Z. Kwiatkowska, G. Norman, and D. Parker. Prism 4.0: Verification of probabilistic real-time systems. In *CAV*, vol. 6806 of *Lecture Notes in Computer Science*. Springer, 2011.

[43] K. Larsen and A. Skou. Bisimulation through probabilistic testing. In *POPL*, 1989.

[44] R. May et al. Biological populations with nonoverlapping generations: stable points, stable cycles, and chaos. *Science*, 186(4164):645–647, 1974.

[45] C. McCaig, R. Norman, and C. Shankland. From individuals to populations: A mean field semantics for process algebra. *Theor. Comput. Sci.*, 412(17):1557–1580, 2011.

[46] R. Milner. *Communication and concurrency*. PHI Series in computer science. Prentice Hall, 1989.

[47] M. Mio. Upper-expectation bisimilarity and lukasiewicz $\mu$-calculus. In *FoSSaCS*, 2014.

[48] D. Sangiorgi and J. Rutten. *Advanced Topics in Bisimulation and Coinduction*. Cambridge University Press, New York, NY, USA, 1st edition, 2011.

[49] R. Segala. *Modeling and Verification of Randomized Distributed Real-time Systems*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 1995.

[50] R. Segala and N. Lynch. Probabilistic simulations for probabilistic processes. In *CONCUR*, 1994.

[51] R. Segala and N. A. Lynch. Probabilistic simulations for probabilistic processes. In *CONCUR*, vol. 836 of *Lecture Notes in Computer Science*. Springer, 1994.

[52] G. Shani, J. Pineau, and R. Kaplow. A survey of point-based pomdp solvers. *AAMAS*, 27(1):1–51, 2013.

[53] A. Sokolova. Probabilistic systems coalgebraically: A survey. *Theor. Comput. Sci.*, 412(38):5095–5110, 2011.

[54] L. Song, L. Zhang, and J. Godskesen. Bisimulations meet PCTL equivalences for probabilistic automata. In *CONCUR*, 2011.

[55] L. Song, L. Zhang, and J. C. Godskesen. Late weak bisimulation for markov automata. *CoRR*, abs/1202.4116, 2012.

[56] M. Stoelinga and F. Vaandrager. A testing scenario for probabilistic automata. In *ICALP*, 2003.

[57] W. Tzeng. A polynomial-time algorithm for the equivalence of probabilistic automata. *SIAM J. Comput.*, 21(2):216–227, 1992.

[58] N. Wolovick. *Continuous probability and nondeterminism in labeled transaction systems*. PhD thesis, Universidad Nacional de Córdoba, 2012.

# A Bisimulation coalgebraically

## A.1 Short introduction to coalgebras

Definitions of bisimulations can be given in terms of relations and we did so. However, for two reasons we also give a coalgebraic definition that induces our relational definition. Firstly, due to the general framework our definition will cover a spectrum of bisimulations depending on the interpretation of the coalgebra and is applicable to more complex systems, automatically yielding the bisimulation definitions. Secondly, any ad-hoc features of a simple coalgebraic definition are more visible and can be clearly identified, whereas it is difficult to distinguish which of two similar relational definitions is more natural. As we assume no previous knowledge of categorical notions we give a brief introduction to coalgebras in the spirit of [48].

A *functor* $F$ (on sets) assigns to each set $X$ a set $F(X)$, and to each set function $f : X \to Y$ a set function $F(f) : F(X) \to F(Y)$ such that two natural conditions are satisfied: (i) the identity function on $X$ is mapped to the identity function on $F(X)$ and (ii) a composition $f \circ g$ is mapped to a composition $F(f) \circ F(g)$.

**Example A.1.** *The powerset functor $\mathcal{P}(-)$ maps a set $X$ to the set $\mathcal{P}(X)$ of its subsets and a function $f : X \to Y$ to $\mathcal{P}(f) : \mathcal{P}(X) \to \mathcal{P}(Y)$ by $U \mapsto \{f(x) \mid x \in U\}$.*

*Similarly, for a fixed set $L$, the operator $(-)^L$ mapping $X$ to the set $X^L$ of functions $L \to X$ is a functor, where the image of $f : X \to Y$ is $F(f) : X^L \to Y^L$ given by mapping $u : L \to X$ to $f \circ u : L \to Y$.*

For a functor $F$, an $F$-*coalgebra* is a pair of the carrier set (or state space) $S$ and the operation function $\text{next} : S \to F(S)$. Intuitively, the function next describes the transition to the next step.

**Example A.2.** *A transition system $(S, \to)$ with $\to \subseteq S \times S$ can be understood as a $\mathcal{P}(-)$-coalgebra by setting $\text{next}(s) = \{s' \mid s \to s'\}$. And vice versa, every $\mathcal{P}$-coalgebra gives rise to a transition system.*

*A labelled transition system $(S, L, \to)$ with the set of labels $L$ and $\to \subseteq S \times L \times S$ can be seen as a $(\mathcal{P}(-))^L$-coalgebra with $\text{next} : S \to (\mathcal{P}(S))^L$ given by $\text{next}(s)(a) \to \{s' \mid s \xrightarrow{a} s'\}$.*

A *bisimulation* on an $F$-coalgebra $(S, \text{next})$ is a an $F$-coalgebra $(R, \overline{\text{next}})$ with $R \subseteq S \times S$ such that the two projections $\pi_1 : R \to S$ and $\pi_2 : R \to S$ make the following diagram commute:[5]

[5] I.e. $\text{next} \circ \pi_1 = F(\pi_1) \circ \overline{\text{next}}$ and $\text{next} \circ \pi_2 = F(\pi_2) \circ \overline{\text{next}}$.

$$\begin{tikzcd}
S \arrow[d, "next"'] & R \arrow[l, "\pi_1"'] \arrow[r, "\pi_2"] \arrow[d, "\overline{next}"'] & S \arrow[d, "next"] \\
F(S) & F(R) \arrow[l, "F(\pi_1)"] \arrow[r, "F(\pi_2)"'] & F(S)
\end{tikzcd}$$

**Example A.3.** *For LTS, the coalgebraic bisimulation coincides with the classical one of Park and Milner [46], where a symmetric* $R$ *is a bisimulation if for every* $sRt$ *and* $s \xrightarrow{a} s'$ *there is* $t \xrightarrow{a} t'_{s,a,s',t}$ *with* $s' R t'_{s,a,s',t'}$. *Indeed, given a classical bisimulation* $R$*, one can define* $\mathrm{next}(\langle s,t\rangle)(a)$ *to contain for every* $s \xrightarrow{a} s'$ *the matching pair* $\langle s', t'_{s,a,s',t}\rangle$ *and symmetrically for* $t$*. Since all these pairs are from* $R$*,* $(R, \overline{\mathrm{next}})$ *is indeed a coalgebra. Further, the projection* $F(\pi_1)$ *of* $\mathrm{next}(\langle s,t\rangle)$ *yields for each* $a \in L$ *all and nothing but the successors of* $s$ *under* $a$*, symmetrically for* $t$*, hence the commuting.*

*Conversely, given a coalgebraic bisimulation* $(R, \overline{\mathrm{next}})$*, the commuting of* $\pi_1$ *guarantees that* $\mathrm{next}(\langle s,t\rangle)(a)$ *yields all and nothing but the successors of* $s$ *under* $a$*. Hence, for each* $s \xrightarrow{a} s'$ *there must be* $\langle s', t'\rangle \in \mathrm{next}(\langle s,t\rangle)(a) \subseteq R$*, moreover, with* $t \xrightarrow{a} t'$ *due to* $\pi_2$ *commuting.*

As we have seen, the coalgebraic definition coincides with the relational one for non-probabilistic systems. One can use the same theory for finite probabilistic systems, too. Let $\mathcal{D}(X)$ denote the set of simple distributions, i.e. functions $f : X \to [0,1]$ such that $f$ is non-zero for only finitely many elements $x_1, \ldots, x_n$ and $\sum_{i=1}^{n} f(x_i) = 1$. Note that $\mathcal{D}(-)$ can be understood as a functor.

**Example A.4.** *We can interpret* $(\mathcal{D}(-) \cup \{\bullet\})^L$*-coalgebras as finite Markov decision processes* $(S, L, Pr)$ *with* $Pr : S \times L \to \mathcal{D}(S) \cup \{\bullet\}$ *that under each action either proceed to a distribution on successors (as opposed to a non-deterministic choice in LTS) or not have the action available (the special element* $\bullet$*). The corresponding coalgebraic bisimulation can be shown to coincide with the classical one of Larsen and Skou [43], where an equivalence relation* $R$ *is a bisimulation if* $\sum_{u \in U} Pr(t,a)(u) = \sum_{u \in U} Pr(t',a)(u)$ *for every* $a \in L$*, classes* $T, U$ *of* $R$ *and* $t, t' \in T$.

In contrast, uncountable probabilistic systems are more intricate. The set of *probability measures* over $X$ (equipped with a $\sigma$-algebra clear from context) is denoted by $\mathcal{D}(X)$. Again, defining $\mathcal{D}(f)(\mu) = \mu \circ f^{-1}$ makes $\mathcal{D}(-)$ into a functor.

**Example A.5.** *We can interpret* $\mathcal{D}(-)$*-coalgebras as Markov chains with general (possibly uncountable) state space. However, it is intricate to prove that the corresponding bisimulation is defined so that it coincides with the relational definition as already mentioned in Section 1.*

**Example A.6.** *PA correspond to* $(\mathcal{P}(\mathcal{D}(-)))^L$*-coalgebras.*

## A.2 Bisimulation on distributions coalgebraically

The bisimulation we proposed is induced by a different view on the probabilistic systems. Namely, we consider distributions $\mathcal{D}(S)$ or measures $\mathcal{D}(S)$ over its state space $S$ to form the carrier of the coalgebra. A transition then changes this distribution. For instance, a Markov chain can be seen this way as a coalgebra of the identity functor.

Therefore, in order to capture the distributional semantics of NLMP and other continuous systems, we define a functor[(6)]

$$\boxed{([0,1] \times \mathcal{P}(-))^{\mathcal{P}(L)}} \tag{$\spadesuit$}$$

The vital part is not only $[0,1]$, but also the use of measurable sets of labels instead of individual labels. We can view a NLMP $\mathbf{P} = (S, L, \{\tau_a \mid a \in L\})$ as a $\spadesuit$-coalgebra with a carrier set $\mathcal{D}(S)$. The coalgebra assigns to $\mu \in \mathcal{D}(S)$ and to a set of labels $A \in \Sigma(L)$ the pair $(p, M)$ such that

- $p = \mu(S_A)$ is the measure of states that can read some $a \in A$ where $S_A = \{s \in S \mid \exists a \in A.\tau_a(s) \neq \emptyset\}$;
- $M = \emptyset$ if $\mu(S_A) = 0$, and $M$ is the set of convex combinations[(7)] over $\{\mu_\rho \mid$ measurable $\rho : S_A \to \bigcup_{a \in A} \tau_a\}$, otherwise, where

$$\mu_\rho(X) = \frac{1}{\mu(S_A)} \cdot \int_{s \in S} \rho(s)(X)\ \mu(ds) \quad \forall X \in \Sigma(S).$$

In other words, $M$ is obtained by restricting $\mu$ to the states that can read $A$ and weighting all possible combinations of their transitions.

**Lemma A.7.** *The union of $\spadesuit$-bisimulations and $\sim$ coincide.*

*Proof.* First, we prove that whenever there is $\spadesuit$-bisimulation $(R, \overline{\mathrm{next}})$ with $(\mu, \nu) \in R$ then $\mu \sim \nu$ by proving that $R \cup R^{-1}$ is a bisimulation relation. Let $a \subseteq L$ and $\mu R \nu$ or $\nu R \mu$, wlog the former (the latter follows symmetrically).

[(6)] On function, we define the functor by $\spadesuit(f)(n)(A) = (\mathrm{id} \times \mathcal{P}(f))(n(A))$. Here $\mathcal{P}(L)$ denotes only the measurable sets of labels.

[(7)] The set of convex combinations is lifted to a measurable set $Z$ of measures over $S$ as the set $\{X \mapsto \int_{\mu \in Z} \mu(X)\nu(d\,\mu) \mid \nu$ is a measure over $Z\}$.

1. The first condition of the relational bisimulation follows by

$$\begin{aligned}\mu(S_a) &= \pi_1(\mathrm{next}(\mu)(a))\\ &= \pi_1(\mathrm{next} \circ \pi_1\langle\mu,\nu\rangle(a))\\ &= \pi_1(\spadesuit\pi_1 \circ \overline{\mathrm{next}}\langle\mu,\nu\rangle(a))\\ &= \pi_1((\mathrm{id} \times \mathcal{P}\pi_1)(\overline{\mathrm{next}}\langle\mu,\nu\rangle(a)))\\ &= \mathrm{id}(\pi_1(\overline{\mathrm{next}}\langle\mu,\nu\rangle(a)))\\ &= \pi_1((\mathrm{id} \times \mathcal{P}\pi_2)(\overline{\mathrm{next}}\langle\mu,\nu\rangle(a)))\\ &= \pi_1(\spadesuit\pi_2 \circ \overline{\mathrm{next}}\langle\mu,\nu\rangle(a))\\ &= \pi_1(\mathrm{next} \circ \pi_2\langle\mu,\nu\rangle(a))\\ &= \pi_1(\mathrm{next}(\nu)(a))\\ &= \nu(S_a)\end{aligned}$$

2. For the second condition of the relational bisimulation, let $\mu \xrightarrow{a} \mu'$. Since

$$\begin{aligned}\mu' \in{} & \pi_2(\mathrm{next}(\mu))(a)\\ &= \pi_2(\mathrm{next} \circ \pi_1\langle\mu,\nu\rangle(a))\\ &= \pi_2(\spadesuit\pi_1 \circ \overline{\mathrm{next}}\langle\mu,\nu\rangle(a))\\ &= \pi_2((\mathrm{id} \times \mathcal{P}\pi_1)\Big(\overline{\mathrm{next}}\langle\mu,\nu\rangle(a)\Big))\\ &= \mathcal{P}\pi_1(\pi_2\Big(\overline{\mathrm{next}}(\langle\mu,\nu\rangle)(a)\Big))\end{aligned}$$

   there is $\nu'$ with

$$\langle\mu',\nu'\rangle \in \pi_2\Big(\overline{\mathrm{next}}(\langle\mu,\nu\rangle)(a)\Big)$$

   Since R is a coalgebra, we have $\langle\mu',\nu'\rangle \in R$, i.e. $\mu' R \nu'$.

Second, given $R = \sim$, we define $\overline{\mathrm{next}}$ making it into a coalgebra such that the bisimulation diagram commutes. Let $\mathrm{succ}_a(\mu) = \{\mu' \mid \mu \xrightarrow{a} \mu'\}$ denote the set of all $a$-successors of $\mu$. For $\mu R \nu$, we set

$$\overline{\mathrm{next}}(\langle\mu,\nu\rangle)(a) = (\mu(S_a), \{\langle\mu',\nu'\rangle \in R \cap \mathrm{succ}_a(\mu) \times \mathrm{succ}_a(\nu)\})$$

Since we imposed $\langle\mu',\nu'\rangle \in R$, $(R, \overline{\mathrm{next}})$ is a $\spadesuit$-coalgebra. Further, we prove the bisimulation diagram commutes. Firstly,

$$\begin{aligned}\mathrm{next} \circ \pi_1\langle\mu,\nu\rangle &= (\mu(S_a), \mathrm{succ}_a(\mu))\\ \mathrm{next} \circ \pi_2\langle\mu,\nu\rangle &= (\nu(S_a), \mathrm{succ}_a(\nu))\end{aligned}$$

Therefore,

$$\pi_1(\mathrm{next} \circ \pi_1\langle\mu,\nu\rangle) = \mu(S_a) = \pi_1(\spadesuit\pi_1(\overline{\mathrm{next}}\langle\mu,\nu\rangle)(a))$$

and

$$\pi_1(\mathrm{next} \circ \pi_2\langle\mu,\nu\rangle) = \nu(S_a) = \mu(S_a) = \pi_1(\spadesuit\pi_2(\overline{\mathrm{next}}\langle\mu,\nu\rangle)(a))$$

since $\mu(S_a) = \nu(S_a)$ due to $\mu \sim \nu$ and the first relational bisimulation condition. Secondly,

$$\pi_2(\mathrm{next} \circ \pi_1\langle\mu,\nu\rangle(a)) = \mathrm{succ}_a(\mu) \overset{(1)}{=} \pi_2(\spadesuit\pi_1(\overline{\mathrm{next}}\langle\mu,\nu\rangle)(a))$$
$$\pi_2(\mathrm{next} \circ \pi_2\langle\mu,\nu\rangle(a)) = \mathrm{succ}_a(\nu) \overset{(2)}{=} \pi_2(\spadesuit\pi_2(\overline{\mathrm{next}}\langle\mu,\nu\rangle)(a))$$

After we show (1) and (2), we know both components of $\spadesuit\pi_1(\overline{\mathrm{next}}\langle\mu,\nu\rangle)(a)$ are the same as of $\mathrm{next}(\pi_1\langle\mu,\nu\rangle)(a)$, and similarly for $\spadesuit\pi_2$, hence the commuting. As to (1), $\supseteq$ follows directly by $\overline{\mathrm{next}}$ defined above. For $\subseteq$, for every $\mu' \in \mathrm{succ}_a(\mu)$ there is $\nu' \in \mathrm{succ}_a(\nu)$ with $\mu' R \nu'$ due to the second realtional bisimulation condition. Thus also $\langle\mu',\nu'\rangle \in \spadesuit\pi_1(\overline{\mathrm{next}}\langle\mu,\nu\rangle)(a)$. (2) follows from symmetric argument and R being symmetric. □

## A.3 Related bisimulations

For *discrete* systmes, one could define a functor for finite probabilistic systems with nondeterminism by

$$([0,1] \times \mathcal{P}(-))^L \tag{$\heartsuit$}$$

Now a PA $(S, L, \longrightarrow)$ is a $\heartsuit$-coalgebra with the carrier set $\mathcal{D}(S)$. Indeed, the coalgebra assigns to a distribution $\mu$ and a label $a$ the pair $(p, M)$ where

- $p = \mu(S_a)$ is the probability of states that can read $a$;
- $M = \emptyset$ if $\mu(S_a) = 0$, and $M$ is the set of convex combinations over $\{\frac{1}{\mu(S_a)} \sum_{s \in S_a} \nu_s \cdot \mu(s) \mid \forall s \in S_a . s \xrightarrow{a} \nu_s\}$, otherwise. We write $\mu \xrightarrow{a} \mu'$ for every $\mu' \in M$.

**Remark A.8.** *The union of $\heartsuit$-bisimulations and bisimulation of [27], denoted by $\sim_\heartsuit$, coincide.*

Although we can use $\heartsuit$ to capture the distribution semantics of PA as above, we could as well use it differently: if we defined that a label that cannot be read in the current state is *ignored* instead of halting, the successor distribution would be defined by making a step from states that can read the label and staying elsewhere. (This approach is discussed in the next section.)

Moreover, we could easily extend the functor to systems with real rewards (as in [7]) simply by adding $\mathbb{R}$ to get $\mathbb{R} \times ([0,1] \times \mathcal{P}(-))^L$ for rewards on states or $([0,1] \times \mathcal{P}(\mathbb{R} \times -))^L$ on transitions etc. Similarly, for systems without the inner non-determinism like Rabin automata, we could simplify the functor to $([0,1] \times -)^L$. The only important and novel part of the functor is $[0,1]$ stating the overall probability mass that performs the step. (This is also the only difference to non-probabilistic coalgebraic functors.) In all the cases, the generic ♡-bisimulation keeps the same shape. What changes is the induced relational bisimulation.

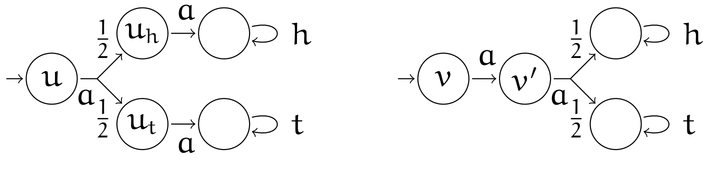
# B Applications

In the following subsections, we justify the proposed bisimulation yielded by ♠ by reviewing its application areas and comparing it to other bisimulations in these areas.

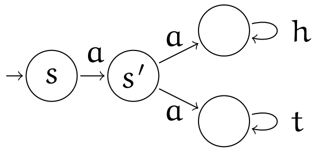
## B.1 Bisimulation in compositional modelling of distributed systems

Probabilistic automata are apt for compositional modelling of communicating parallel systems. This way, the whole system is built bottom-up connecting smaller components into larger by the parallel composition operator. To tackle the state space explosion, minimisation algorithms can be applied throughout the process after each composition. Computing the quotient according to a bisimulation serves well as a minimisation algorithm if the bisimulation is a congruence w.r.t. parallel composition. This condition is satisfied by the (also distribution-based) strong bisimulation recently defined by Hennessy [34], denoted by $\sim_{Hen}$. This is not the case with $\sim$ as shown in the following example.

**Example B.1.** *According to our definition, $u \sim v$ because $\frac{1}{2}u_h + \frac{1}{2}u_t \sim v'$. In contrast, $u \not\sim_{Hen} v$. Therefore, $\sim_{Hen}$ is strictly finer than $\sim$. Actually, $\sim_{Hen}$ coincides (on Dirac distributions) with the standard probabilistic bisimulation of Larsen and Skou [43] which distinguishes $u$ and $v$ as well.*



*Let $\|_A$ denotes the* CSP*-style full synchronization on labels from* $A$ *and interleaving on* $L \setminus A$. *Then* $\sim$ *is not a congruence w.r.t.* $\|_A$ *as* $u \|_L s \not\sim v \|_L s$ *for* $s$ *depicted below.*



This is actually a classical example, due to [49], modelling a process $u$ (or $v$) generating a secret by tossing a coin and the process $s$ guessing the secret. If $s$ guesses correctly, they synchronize forever on $h$ or $t$; otherwise, they halt. In $u \|_L s$, the non-determinism can be resolved by a *scheduler* in such a way that the guesser makes a correct guess with probability 1 which is not possible in $v \|_L s$ because the secret is generated later. This is overly pessimistic in the context of *distributed systems* where the guesser observes only the communication with the tosser and not its state. Namely, the systems $u \|_L s$ and $v \|_L s$ exhibit the same behaviour (correct guess with probability at most 1/2) if the non-determinism is resolved by *distributed* schedulers [17, 10, 31]. This means that the non-determinism in each component of the composition is resolved independently of the state of the other component.

## B.2 Bisimulation for partially observable MDPs

In the distributed setting it is natural to assume that the state space of each component is *fully* unobservable from outside. This is a special case of *partially* observable systems, such as partially observable Markov decision processes (POMDP). POMDPs have a wide range of applications in robotic control, automated planning, dialogue systems, medical diagnosis, and many other areas [52].

In the analysis of POMDP, the distributions over states, called *beliefs*, arise naturally and yield a continuous-space (fully observable) belief MDP. Therefore, probabilistic bisimulations over beliefs have been already studied [7, 38]. However, no connection of this particular case to general probabilistic bisimulation has been studied.

There are various (equivalent) definitions of POMDP, we use one close to computational game theory [9].

**Definition B.2.** *A* partially observable Markov decision process (POMDP) *is a tuple* $\mathcal{M} = (S, \delta, \mathcal{O})$ *where* $S$ *is a set of states,* $\delta \subseteq S \times \mathcal{D}(S)$ *is a transition relation, and* $\mathcal{O} \subseteq 2^S$ *is a set of observations that partition the state space.*

This formalism is also known as *labelled Markov decision processes* [22] where state labels correspond to observations. Such a state-labelled system $\mathcal{M} = (S, \delta, \mathcal{O})$ can be easily translated to an action-labelled PA $D_{\mathcal{M}} = (S, \mathcal{O}, \longrightarrow)$ where $s \xrightarrow{o} \mu$ if $s \in o$ and $(s, \mu) \in \delta$. This way, we can define $\mu \sim \mu'$ in $\mathcal{M}$ if $\mu \sim \mu'$ in $D_{\mathcal{M}}$.

Hence, in Section 4.1, we give the first algorithm for computing bisimulations over beliefs in finite POMDP. Previously, there was only an algorithm [38] for computing bisimulations on distributions of Markov chains with partial observation.

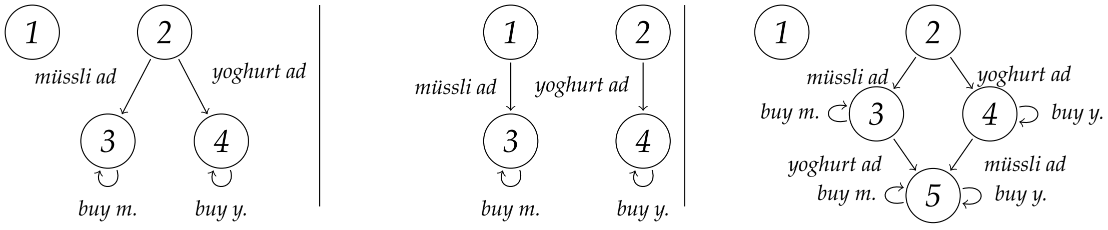
## B.3 Bisimulation for large-population models

In the sense of [28, 45, 44, 39], we can understand PA as a description of one *agent* in a large homogeneous population. For example a *chemical compounds*, a *node of a computer grid*, or a *customer of a chain store*.

The distribution perspective is a natural one – the distribution specifies the ratios of agents being currently in the individual states. For a Markov chain, this gives a deterministic process over the continuous space of distributions.

The non-determinism of PA has also a natural interpretation. Labels given to this large population of PAs correspond to global control actions [29, 28] such as *manipulation with the chemical solution*, a *broadcast within the grid*, or a *marketing campaign of the chain store*. Agents react to this control action if currently in a state with transition under this label, otherwise they ignore it. Multiple transitions under this label correspond to multiple ways how the agent may react.

**Example B.3.** *Let us illustrate the idea by an example of three models of customers of a chain store with half of the population in state* 1 *and half of the population in state* 3.



*It is natural to assume that these three models can be distinguished. Indeed none of the populations are bisimilar according to our definition. Note however, that the related distribution-based bisimulation of [27] that allows only singletons* A *in Definition 2.5 does not distinguish the first and the second population. Their definition actually extends the bisimulation of [22] defined on input-enabled models; they naturally transform general probabilistic automata to input-enabled* *ones by directing the missing transitions into a newly added sink state. Observe that the similarly natural alternative approach of adding self-loops does not distinguish the second and the third population.*

# C Formal semantics of stochastic automata

Formally, a state $(q, \xi)$ denotes being in location $q$ where each clock $c$ has value $\xi(c)$. In $(q, \xi)$, a label of the form $(a, t)$ is available if $E_a \neq \emptyset$ where $E_a$ is the set of edges that have action $a$ and become available after the idling time $t$. We set $\tau_{(a,t)}((q, \xi)) = \{\mu_e \mid e \in E_a\}$ where $\mu_e$ for an edge $e = (q, a, C, q')$ is the probability measure over states with

1. the marginal in the first component being Dirac on $q'$;
2. the marginal for any $c \notin \kappa(q')$ being Dirac on $\xi(c) - t$;
3. the marginals for each $c \in \kappa(q')$ having CDF $F(c)$, and their product being equal to the joint distribution of $\kappa(q')$.

Intuitively, it 1) moves to $q'$, 2) decreases values of clocks by $t$, and 3) sets clocks of $\kappa(q')$ to independent random values.

# D Proofs from Section 3

**Theorem 3.3.** Let $SA(\mathcal{C})$ denote the stochastic automaton corresponding to a CTMC $\mathcal{C}$. For any CTMC $\mathcal{C}_1, \mathcal{C}_2$, we have

$$SA(\mathcal{C}_1) \parallel_{SA} SA(\mathcal{C}_1) \quad \sim \quad SA(\mathcal{C}_1 \parallel_{CT} \mathcal{C}_1).$$

*Proof.* It is easy to see that the two SA have the same locations and edges, they only differ in which locations which clocks are reset. However, resetting of exponential clocks is irrelevant for bisimulation $\sim$ in SA with exponential distribution. This fact is clear from the construction of the discrete PA $\bar{D}$ in Section 4 and from correctness of this algorithm. □

# E Proofs from Section 4

## E.1 Discrete systems

We use the notation $\mu \oplus_p \nu$ to denote $(1-p)\mu + p\nu$. Further, for a (not necessarily probabilistic) measure $\mu = (\mu(s_1), \dots, \mu(s_{|S|}))$ we denote $|\mu| = \sum_{i=1}^{|S|} \mu(s_i)$. For any probability distribution $\mu$ thus $|\mu| = 1$.

**Lemma 4.3.** For every linear bisimulation there exists a corresponding bisimulation matrix.

*Proof.* Let $R$ be a linear bisimulation and $\Gamma$ an arbitrary equivalence class of $R$. Due to linearity, $\Gamma$ is closed under convex combinations. Consider $\bar{\Gamma}$ the affine closeru of $\Gamma$, i.e. the smallest set that is closed under affine combinations. Then (i) $\bar{\Gamma}$ is an affine subspace, and (ii) $\bar{\Gamma} \cap \mathcal{D}(S) = \Gamma$. This holds for every class of $R$. Hence $\{\bar{\Gamma} \mid \Gamma \text{ is an equivalence class of } R\}$ decomposes $\mathbb{R}^{|S|}$ and all $\bar{\Gamma}$ have the same difference space. By (i) we get that $\bar{\Delta} := \{\mu - \nu \mid \mu, \nu \in \bar{\Gamma}\}$ (independent of choice of $\Gamma$) is a linear space. Therefore, there is a matrix $E$ such that $\rho \in \bar{\Delta}$ iff $\rho E = 0$.

For every $\mu R \nu$ we thus have $(\mu - \nu)E = 0$. In the other direction, let $\mu \in \Gamma$ and $\nu$ be arbitrary distribution such that $(\mu - \nu)E = 0$. We thus have $\mu - \nu \in \bar{\Delta}$. Since $\mu \in \bar{\Gamma}$ we thus get $\nu \in \bar{\Gamma}$. Since $\nu \in \mathcal{D}(S)$, we finally obtain $\nu \in \Gamma$ and thus $\mu R \nu$. □

**Lemma E.2.** *$\sim$ is linear.*

*Proof.* We prove that $\mu_1 \sim \nu_1$ and $\mu_2 \sim \nu_2$ imply $\mu_1 \oplus_p \mu_2 \sim \nu_1 \oplus_p \nu_2$ for any $p \in [0,1]$. This follows easily from the Spoiler-Duplicator game. Indeed, let Duplicator have a winning response to every Spoiler's strategy both in $\mu_1 \sim \nu_1$ and $\mu_2 \sim \nu_2$. Let now $p \in [0,1]$. Any Spoiler's strategy on $\mu_1 \oplus_p \mu_2 \sim \nu_1 \oplus_p \nu_2$ (w.l.o.g. attacking on the left under $a$) can be decomposed to a part acting on $(1-p)\mu_1$ resulting into $(1-p)\mu_1(S_a), (1-p)\mu_1'$ and a part acting on $p\nu$ resulting into $p\mu_2(S_a), p\mu_2'$. Duplicator has a winning response $\nu_1'$ to the former (when applied to the whole $\mu_1$) and also $\nu_2'$ to the latter (when applied to the whole $\mu_2$). Duplicator can now mix his responses resulting into $\nu_1' \oplus_p \nu_2'$, which is clearly a choice conforming both to the rules, since $(\nu_1 \oplus_p \nu_2)(S_a) = (1-p)\nu_1(S_a) + p\nu_2(S_a) = (1-p)\mu_1(S_a) + p\mu_2(S_a) = (\mu_1 \oplus_p \mu_2)(S_a)$ and also winning as the resultinig pair is again a convex combination of individual resulting pairs. □

Thus minimal bisimulation matrices always exist.

**Corollary E.3.** *There is a minimal bisimulation matrix, i.e. a matrix* $\mathsf{E}$ *such that for any* $\mu, \nu \in \mathcal{D}(S)$*, we have* $\mu \sim \nu$ *iff* $(\mu - \nu)\mathsf{E} = 0$.

We are searching for the least restrictive system $\mathsf{E}$ satisfying stability. Therefore, we can compute $\sim$, i.e. the greatest fixpoint of the bisimulation requirement of stability, as the least fixpont of the partitioning procedure of adding equations. Indeed, recall that all bisimulation matrices with the least possible dimension have the same solution space.

**Proposition 4.5.** In an action deterministic PA, $\mathsf{E}$ containing $\mathbf{1}$ is a bisimulation matrix iff it is $\mathsf{P}_a$-stable for all $a \in L$.

*Proof.* Firstly, we prove that for any $a \in L$, any bisimulation matrix $\mathsf{E}$ is $\mathsf{P}_a$-stable. Let $\rho$ be such that $\rho\mathsf{E} = 0$. Let us write $\rho = \mu - \nu$ where entries in $\mu$ and $\nu$ are non-negative. Since $\mathsf{E}$ contains $\mathbf{1}$, we have $|\mu| = |\nu|$, moreover, for the moment let us assume equal 1. Then $\rho$ is a difference of two measures $\mu - \nu$. Since $\mathsf{E}$ is a bisimulation matrix, we have $\mu \sim \nu$. Therefore, if Spoiler attacks under $a$, we have $\mu\mathsf{P}_a \sim \nu\mathsf{P}_a$. Therefore, $(\mu\mathsf{P}_a - \nu\mathsf{P}_a)\mathsf{E} = 0$, equivalently $\rho\mathsf{P}_a\mathsf{E} = 0$. In the general case, where $|\mu| = |\nu|$ is not equal 1, we can write regard them as a scalar multiples of measures, normalize them, and use the same reasoning (with the exception when they are $\mathbf{0}$, in which case the claim for $\rho = \mathbf{0}$ holds trivially).

Secondly, let $\mathsf{E}$ contain $\mathbf{1}$ and be $\mathsf{P}_a$-stable for all $a \in L$. We show that $\mathsf{R}$ defined by $\mu\mathsf{R}\nu$ iff $(\mu - \nu)\mathsf{E} = 0$ is a bisimulation relation. Consider now $A \subseteq \mathsf{L}$ singletons. The first bisimulation condition for $a \in \mathsf{L}$ follows from $(\mu - \nu)\mathsf{P}_a\mathbf{1} = 0$. The second one then from $(\mu - \nu)\mathsf{P}_a\mathsf{E} = 0$ implying $(\mu\mathsf{P}_a - \nu\mathsf{P}_a)\mathsf{E} = 0$ by stability. For general $A \subseteq \mathsf{L}$, the bisimulation condition does not generate any new requirements due to the action determinism. Since $\mathsf{S}_\mathsf{A}$ is a disjoint union of $\mathsf{S}_a$ for $a \in \mathsf{A}$, the properties follow from the properties of singeltons. □

We recall that for elements of $\mathcal{E}(\mathsf{C})$, we use the name *extreme choices* as these points are mapped to pure strategies and achieve Pareto extremal values when applied to any distributions.

**Proposition 4.7.** $\mathsf{E}$ containing $\mathbf{1}$ is a bisimulation matrix iff the matrix is $\mathsf{P}_\mathsf{A}^{W(\mathsf{c})}$-stable for all $A \subseteq L$ and $\mathsf{c} \in \mathcal{E}(\mathsf{C})$.

*Proof.* We first prove the two following facts:

(1) the extremal choices, i.e. $\mathcal{E}(\mathsf{C})$, are sufficient for Spoiler,

(2) for an extremal choice $W \in \mathcal{E}(C)$ of Spoiler, $W$ is an optimal reply of Duplicator for any distributions $\mu$ and $\nu$.

As to (1), for given $\mu \not\sim \nu$ an optimal choice of Spoiler is a $W$ such that $\mu M_a^W$ is Pareto extremal, which is always achieved also in a corner. Indeed, if a non-extremal choice $W$ is winning then in some direction from $W$ one of the extremal choices is also winning by convexity of the polytopes.

As to (2), let $\mu \sim \nu$ and $W_S$ be an extremal choice of Spoiler on $\mu$, $W_D$ an optimal (winning) response of Duplicator on $\nu$ supposed, for a contradiction, different from $W_S$. Since $W_S$ is more extreme in the direction of $W_S$ than $W_D$, $W_S$ achieves Pareto (in direction $W_S$) strictly better value than $W_D$ on $\nu$, hence also strictly better (in direction $W_S$) than $W_S$ on $\mu$. If Spoiler moved from $\nu$ by $W_D$ a matching response would be $W_S$. On the other hand, if Spoiler moved from $\nu$ by $W_S$, this choice strictly dominates $W_S$ on $\mu$ (in direction $W_S$) and thus all choices on $\mu$ (in direction $W_S$) as $W_S$ is extremal (in direction $W_S$). Hence there is no matching response for the Duplicator, a contradiction.

As a result, the bisimulation matrix requirement, written in the game fashion as follows, for all $a \in L$

$$(\mu - \nu)E = 0 \implies \forall W_S \in \mathcal{W} : \exists W_D \in \mathcal{W} : \\ \mu P_a^{W_S}\mathbf{1} = \nu P_a^{W_D}\mathbf{1} \wedge (\mu P_a^{W_S} - \nu P_a^{W_D})E = 0$$

can be transformed into: for all $a \in L$

$$(\mu - \nu)E = 0 \implies \forall W \in \mathcal{E}(C) : \\ (\mu P_a^W - \nu P_a^W)\mathbf{1} = 0 \wedge (\mu P_a^W - \nu P_a^W)E = 0$$

and since $\mathbf{1}$ is a column of $E$ also as: for all $a \in L$

$$(\mu - \nu)E = 0 \implies \forall W \in \mathcal{E}(C) : (\mu - \nu)P_a^W E = 0$$

which is nothing but $P_a^{W(c)}$-stability for all $a \in L$ and $c \in \mathcal{E}(C)$. (We deal with $\rho$ not being a difference of any two distributions by scaling as in Proposition 4.5). □

**Corollary E.6.** *Any $P_a^{W(c)}$-stable (for all $a \in L$ and $c \in \mathcal{E}(C)$) matrix containing $\mathbf{1}$ with minimal rank is a minimal bisimulation matrix.*

**Theorem 4.8.** Algorithm 1 computes a minimal bisimulation matrix in exponential time.

*Proof.* The proof follows from the previous corollary and the fact that the algorithm only adds columns required by stability on the current partitioning.

Concerning the complexity, each step is polynomial except for computing and iterating over all exponentially many extremal choices. □

## E.2 Continuous-time systems

Let us repeat the main theorem of the subsection.

**Theorem 4.9.** Let $\mathcal{S} = (\mathcal{Q}, \mathcal{C}, \mathcal{A}, \rightarrow, \kappa, F)$ be a deterministic SA over exponential distributions. There is an algorithm to decide in time polynomial in $|\mathcal{S}|$ and exponential in $|\mathcal{C}|$ whether $q_1 \sim q_2$ for any locations $q_1, q_2$.

The proof follows easily from the following lemmata.

**Lemma 4.10.** For any distributions $\mu, \nu$ on $S$ we have $\mu \sim \nu$ iff $\xi(\mu) \sim \xi(\nu)$.

*Proof.* $\Rightarrow$: Let us take the maximal bisimulation in $\mathbf{P}_{\mathcal{S}}$. We map it by $\xi$; it is easy to see that it is still a bisimulation since the operations $\xi$ and $\xrightarrow{A}$ commute for any $A \subseteq L$.

$\Leftarrow$: Let us take $\mu, \nu$ such that $\mu \not\sim \nu$. Then there is a finite sequence of set of labels $A_1, \ldots, A_n$, such that after applying this sequence, one of the conditions in Definition 2.5 is unstatisfied. Again, as the operations $\xi$ and $\xrightarrow{A}$ commute for any $A \subseteq L$, we get that also $\xi(\mu) \not\sim \xi(\nu)$. □

**Lemma 4.11.** For a deterministic SA over exponential distributions, $|\bar{S}| \leq |\mathcal{Q}| 2^{|\mathcal{C}|}$.

*Proof.* It is easy to check that for all states of the form $q \otimes \bigotimes_{c \in X \subseteq \mathcal{C}} \mathrm{Exp}(\lambda_c)$, any successor in $\bar{D}$ has the same form. □

**Lemma 4.12.** There is a successful tableau from $\mu \sim \nu$ iff $\mu \sim \nu$ in $\hat{\mathbf{P}}$. Moreover, the set of nodes of a successful tableau is a subset of a bisimulation.

*Proof.* $\Leftarrow$: We can build an infinite successful tableau only using the rule **Step**. Note that the rule exactly follows the transition relation of $\hat{\mathbf{P}}$ (only regards the distribution as a discrete convex combination of one of finitely many distributions – states of $\bar{D}$). Hence, by applying the rule **Step** from bisimilar distributions, we can obtain only tableau nodes corresponding to bisimilar distributions never reaching a failure node.

$\Rightarrow$: First, observe that if there is a successful tableau $T$ from node $\mu \sim \nu$, there also is a successful (possibly infinite) tableau $T'$ using only the rule **Step**. This is easy to observe since whenever there is an application of the **Lin** rule, you can iteratively apply the **Step** rule infinitely many times (since you can express the current node as a linear combination of nodes from which you can apply the **Step** rule; and the same inductively holds for each such successor node).

Note that by this construction, the set of nodes of $T$ is a subset of the set of nodes of $T'$. We show that for any node $\mu_1 \sim \mu_2$ in $T'$ we have $\mu_1 \sim \mu_2$ in $\hat{\mathbf{P}}$. Let us fix such a node $\mu_1 \sim \mu_2$ and let $R$ be a relation such that $\mu_1' R \mu_2'$ if $\mu_1' \sim \mu_2'$ is an ancestor of the node $\mu_1 \sim \mu_2$. Since the rule **Step** closely follows the definition of bisimulation, it is easy to see that $R$ is a bisimulation. As $R$ contains also $(\mu_1, \mu_2)$, we have $\mu_1 \sim \mu_2$. □

**Lemma 4.13.** There is a successful tableau from $\mu \sim \nu$ iff there is a finite successful tableau from $\mu \sim \nu$ of size polynomial in $|\bar{S}|$.

*Proof.* The implication $\Leftarrow$ is trivial. As regards $\Rightarrow$, let us assume that there is a successful tableau from $\mu \sim \nu$. As each node in the tableau corresponds to a vector of dimension $|\bar{S}|$, the maximal size of a set of linearly independent nodes is $|\bar{S}|$. By applying the rule **Lin** when possible we can prune the tableau into linear size. □