Comparing Labelled Markov Decision Processes

Stefan Kiefer
Department of Computer Science, University of Oxford, UK
stekie@cs.ox.ac.uk

Qiyi Tang
Department of Computer Science, University of Oxford, UK
qiyi.tang@cs.ox.ac.uk

Abstract

A labelled Markov decision process is a labelled Markov chain with nondeterminism, i.e., together with a strategy a labelled MDP induces a labelled Markov chain. The model is related to interval Markov chains. Motivated by applications of equivalence checking for the verification of anonymity, we study the algorithmic comparison of two labelled MDPs, in particular, whether there exist strategies such that the MDPs become equivalent/inequivalent, both in terms of trace equivalence and in terms of probabilistic bisimilarity. We provide the first polynomial-time algorithms for computing memoryless strategies to make the two labelled MDPs inequivalent if such strategies exist. We also study the computational complexity of qualitative problems about making the total variation distance and the probabilistic bisimilarity distance less than one or equal to one.

2012 ACM Subject Classification Theory of computation → Program verification; Theory of computation → Models of computation; Mathematics of computing → Probability and statistics

Keywords and phrases Markov decision processes, Markov chains, Behavioural metrics

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2020.49

Related Version A full version of the paper is [24], available at `https://arxiv.org/abs/2009.11643`.

Funding *Stefan Kiefer*: Supported by a Royal Society University Fellowship.

Acknowledgements We thank the anonymous reviewers of this paper for their constructive feedback.

1 Introduction

Given a model of computation (e.g., finite automata), and two instances of it, are they semantically equivalent (i.e., do they accept the same language)? Such *equivalence* problems can be viewed as a fundamental question for almost any model of computation. As such, they permeate computer science, in particular, theoretical computer science.

In *labelled Markov chains (LMCs)*, which are Markov chains whose states (or, equivalently, transitions) are labelled with an observable letter, there are two natural and very well-studied versions of equivalence, namely *trace (or language) equivalence* and *probabilistic bisimilarity*.

The *trace equivalence* problem has a long history, going back to Schützenberger [33] and Paz [29] who studied *weighted* and *probabilistic* automata, respectively. Those models generalize LMCs, but the respective equivalence problems are essentially the same. It can be extracted from [33] that equivalence is decidable in polynomial time, using a technique based on linear algebra. Variants of this technique were developed in [38, 16]. More recently, the efficient decidability of the equivalence problem was exploited, both theoretically and practically, for the verification of probabilistic systems, see, e.g., [22, 23, 30, 28, 27]. In those works, equivalence naturally expresses properties such as obliviousness and anonymity, which are difficult to formalize in temporal logic. In a similar vein, inequivalence can mean detectibility and the lack of anonymity.

Probabilistic bisimilarity is an equivalence that was introduced by Larsen and Skou [26]. It is finer than trace equivalence, i.e., probabilistic bisimilarity implies trace equivalence. A similar notion for Markov chains, called *lumpability*, can be traced back at least to the classical text by Kemeny and Snell [21]. Probabilistic bisimilarity can also be computed in polynomial time [2, 13, 39]. Indeed, in practice, computing the bisimilarity quotient is fast and has become a backbone for highly efficient tools for probabilistic verification such as Prism [25] and Storm [19].

In this paper, we study equivalence problems for *(labelled) Markov decision processes (MDPs)*, which are LMCs plus nondeterminism, i.e., each state may have several *actions* (or "moves") one of which is chosen by a controller, potentially randomly. An MDP and a controller *strategy* together induce an LMC (potentially with infinite state space, depending on the complexity of the strategy). The nondeterminism in MDPs gives rise to a spectrum of equivalence queries: one may ask about the existence of strategies for two given MDPs such that the induced LMCs become trace/bisimulation equivalent, or such that they become trace/bisimulation *in*equivalent. Another potential dimension of this spectrum is whether to consider general strategies or more restricted ones, such as memoryless or even memoryless deterministic (MD) ones.

In this paper, we focus on *memoryless* strategies, for several reasons. First, these questions for unrestricted strategies quickly lead to undecidability. For example, in [17, Theorem 3.1] it was shown that whether there exists a general strategy such that a given MDP becomes trace equivalent with a given LMC is undecidable. Second, memoryless strategies are sufficient for a wide range of objectives in MDPs, and their simplicity means that even if it was known that a general strategy exists to accomplish (in)equivalence one might still wonder if there *also* exists a memoryless strategy. Third, probabilistic bisimilarity is a less natural notion for LMCs induced by general strategies: such LMCs will in general have an infinite state space, even when the MDP is finite. Fourth, applying a memoryless strategy in an MDP is related to choosing an instance of an *interval Markov chain (IMC)*. IMCs are like Markov chains, but the transitions are labelled not with probabilities but with probability intervals. IMCs were introduced by Jonsson and Larsen [20] and have been well studied in verification-related domains [34, 7, 12, 3, 6], but also in areas such as systems biology, security or communication protocols, see, e.g., [11]. Selecting a memoryless strategy in an MDP corresponds to selecting a probability from each interval (one out of generally uncountably many). *Parametric Markov chains* and *parametric MDPs* are further related models, see, e.g., [18, 41] and the references therein.

LMCs can also be compared in terms of their *distance*. We consider two natural distance functions between two LMCs: the *total variation* distance (between the two trace distributions) and the *probabilistic bisimilarity* distance [15]. Both distances can be at most 1. The total variation (resp. probabilistic bisimilarity) distance is 0 if and only if the LMCs are trace equivalent (resp. probabilistic bisimilar). Further, the probabilistic bisimilarity distance is an upper bound on the total variation distance [8]. It was shown in [9] (resp. [37]) that whether the total variation (resp. probabilistic bisimilarity) distance of two LMCs equals 1 can be decided in polynomial time. This raises the question whether these results can be extended to MDPs, i.e., what is the complexity of deciding whether there exists a memoryless strategy to make the distance less than 1 or equal to 1, respectively. It turns out that some of these problems are closely related to the corresponding (in)equivalence problem.

Instead of comparing two MDPs with initial distributions/states, one may equivalently compare two initial distributions/states in a single MDP (by taking a disjoint union of the states). In this paper we study the computational complexity of the following problems:

- $\mathrm{TV} = 0$ ($\mathrm{TV} > 0$), which asks whether there is a memoryless strategy such that the two initial distributions are (not) trace equivalent in the induced labelled Markov chain;
- $\mathrm{TV} = 1$ ($\mathrm{TV} < 1$), which asks whether there is a memoryless strategy such that the two initial distributions (do not) have total variation distance one;
- $\mathrm{PB} = 0$ ($\mathrm{PB} > 0$), which asks whether there is a memoryless strategy such that the two initial states are (not) probabilistic bisimilar;
- $\mathrm{PB} = 1$ ($\mathrm{PB} < 1$), which asks whether there is a memoryless strategy such that the two initial states (do not) have probabilistic bisimilarity distance one.

In Sections 3 and 4 we provide the first polynomial-time algorithms for $\mathrm{TV} > 0$ and $\mathrm{PB} > 0$, respectively. We also show how to compute memoryless strategies that witness trace and probabilistic bisimulation inequivalence, respectively. In Section 5 we discuss $\mathrm{TV} = 1$ and $\mathrm{PB} = 1$, and in Section 6 we establish the complexity of the remaining four problems, which are about making the distance small ($= 0$ or <1). We conclude in Section 7. Table 1 summarises the results in the paper. Missing proofs can be found in the full version of this paper [24].

Table 1 Summary of the results. These results also imply results for the problems which state "for all memoryless strategies". For example, $\mathrm{TV} > 0$ is the complement of the decision problem whether for all memoryless strategies the two initial distributions are trace equivalent in the induced labelled Markov chains.

Problem	Complexity
$\mathrm{TV} = 0$	$\exists\mathbb{R}$-complete
$\mathrm{TV} > 0$	in P
$\mathrm{TV} = 1$	NP-hard and in $\exists\mathbb{R}$
$\mathrm{TV} < 1$	$\exists\mathbb{R}$-complete
$\mathrm{PB} = 0$	NP-complete
$\mathrm{PB} > 0$	in P
$\mathrm{PB} = 1$	NP-complete
$\mathrm{PB} < 1$	NP-complete

2 Preliminaries

We write $\mathbb{R}$ for the set of real numbers and $\mathbb{N}$ the set of nonnegative integers. Let S be a finite set. We denote by $\mathrm{Distr}(S)$ the set of probability distributions on S. By default we view vectors, i.e., elements of $\mathbb{R}^S$, as row vectors. For a vector $\mu \in [0,1]^S$ we write $|\mu| := \sum_{s\in S} \mu(s)$ for its L_1-norm. A vector $\mu \in [0,1]^S$ is a distribution (resp. subdistribution) over S if $|\mu| = 1$ (resp. $0 < |\mu| \leq 1$). We denote column vectors by boldface letters; in particular, $\mathbf{1} \in \{1\}^S$ and $\mathbf{0} \in \{0\}^S$ are column vectors all whose entries are 1 and 0, respectively. For $s \in S$ we write δ_s for the (Dirac) distribution over S with $\delta_s(s) = 1$ and $\delta_s(r) = 0$ for $r \in S \setminus \{s\}$. For a (sub)distribution μ we write $\mathrm{support}(\mu) = \{s \in S \mid \mu(s) > 0\}$ for its support.

A *labelled Markov chain* (LMC) is a quadruple $\langle S, L, \tau, \ell\rangle$ consisting of a nonempty finite set S of states, a nonempty finite set L of labels, a transition function $\tau : S \to \mathrm{Distr}(S)$, and a labelling function $\ell : S \to L$.

We denote by $\tau(s)(t)$ the transition probability from s to t. Similarly, we denote by $\tau(s)(E) = \sum_{t\in E} \tau(s)(t)$ the transition probability from s to $E \subseteq S$. A trace in a LMC $\mathcal{M}$ is a sequence of labels $w = a_1 a_2 \cdots a_n$ where $a_i \in L$. We denote by $L^{\leq n}$ the set of traces of length at most n. Let $M : L \to [0,1]^{S\times S}$ specify the transitions, so that $\sum_{a\in L} M(a)$

is a stochastic matrix, $M(a)(s,t) = \tau(s)(t)$ if $\ell(s) = a$ and $M(a)(s,t) = 0$ otherwise. We extend M to the mapping $M : L^* \to [0,1]^{S\times S}$ with $M(w) = M(a_1)\cdots M(a_n)$ for a trace $w = a_1 \cdots a_n$. If the LMC is in state s, then with probability $M(w)(s,s')$ it emits a trace w and moves to state s' in $|w|$ steps. For a trace $w \in L^*$, we define $Run(w) := \{w\}L^\omega$; i.e., $Run(w)$ is the set of traces starting with w. To an initial distribution π on S, we associate the probability space $(L^\omega, \mathcal{F}, \Pr_{\mathcal{M},\pi})$, where $\mathcal{F}$ is the σ-field generated by all basic cylinders $Run(w)$ with $w \in L^*$ and $\Pr_{\mathcal{M},\pi} : \mathcal{F} \to [0,1]$ is the unique probability measure such that $\Pr_{\mathcal{M},\pi}(Run(w)) = |\pi M(w)|$. We generalize the definition of $\Pr_{\mathcal{M},\pi}$ to subdistributions π in the obvious way, yielding sub-probability measures. We may drop the subscript $\mathcal{M}$ if it is clear from the context.

Given two initial distributions μ and ν, the *total variation distance* between μ and ν is defined as follows:

$$d_{tv}(\mu,\nu) = \sup_{E\in\mathcal{F}} |\Pr_\mu(E) - \Pr_\nu(E)|.$$

We write $\mu \equiv \nu$ to denote that μ and ν are trace equivalent, i.e., $|\Pr_\mu(Run(w))| = |\Pr_\nu(Run(w))|$ holds for all $w \in L^*$. We have that trace equivalence and the total variation distance being zero are equivalent [9, Proposition 3(a)].

The pseudometric *probabilistic bisimilarity distance* of Desharnais et al. [14] , which we denote by d_{pb}, is a function from $S \times S$ to $[0,1]$, that is, an element of $[0,1]^{S\times S}$. It can be defined as the least fixed point of the following function:

$$\Delta(d)(s,t) = \begin{cases} 1 & \text{if } \ell(s) \neq \ell(t) \\ \min\limits_{\omega\in\Omega(\tau(s),\tau(t))} \sum\limits_{u,v\in S} \omega(u,v)\, d(u,v) & \text{otherwise} \end{cases}$$

where the set $\Omega(\mu,\nu)$ of *couplings* of $\mu,\nu \in \mathrm{Distr}(S)$ is defined as $\Omega(\mu,\nu) = \left\{ \omega \in \mathrm{Distr}(S\times S) \;\middle|\; \sum_{t\in S}\omega(s,t) = \mu(s) \wedge \sum_{s\in S}\omega(s,t) = \nu(t) \right\}$. Note that a coupling $\omega \in \Omega$ is a joint probability distribution with marginals μ and ν (see, e.g., [4, page 260-262]).

An equivalence relation $R \subseteq S \times S$ is a *probabilistic bisimulation* if for all $(s,t) \in R$, $\ell(s) = \ell(t)$ and $\tau(s)(E) = \tau(t)(E)$ for each R-equivalence class E. *Probabilistic bisimilarity*, denoted by $\sim_{\mathcal{M}}$ (or $\sim$ when $\mathcal{M}$ is clear), is the largest probabilistic bisimulation. For all $s,t \in S$, $s \sim t$ if and only if $d_{pb}(s,t) = 0$ [14, Theorem 1].

A *(labelled) Markov decision process* (MDP) is a tuple $\langle S, \mathcal{A}, L, \varphi, \ell\rangle$ consisting of a finite set S of states, a finite set $\mathcal{A}$ of actions, a finite set L of labels, a partial function $\varphi : S \times \mathcal{A} \nrightarrow \mathrm{Distr}(S)$ denoting the probabilistic transition, and a labelling function $\ell : S \to L$. The set of available actions in a state s is $\mathcal{A}(s) = \{\mathsf{m} \in \mathcal{A} \mid \varphi(s,\mathsf{m}) \text{ is defined}\}$. A *memoryless* strategy for an MDP is a function $\alpha : S \to \mathrm{Distr}(\mathcal{A})$ that given a state s, returns a probability distribution on all the available actions at that state. Such strategies are also known as positional, as they do not depend on the history of past states. A strategy α is *memoryless deterministic* (MD) if for all states s there exists an action $\mathsf{m} \in \mathcal{A}(s)$ such that $\alpha(s)(\mathsf{m}) = 1$; we thus view an MD strategy as a function $\alpha : S \to \mathcal{A}$.

For the remainder of the paper, we fix an MDP $\mathcal{D} = \langle S, \mathcal{A}, L, \varphi, \ell\rangle$. Given a memoryless strategy α for $\mathcal{D}$, an LMC $\mathcal{D}(\alpha) = \langle S, L, \tau, \ell\rangle$ is induced, where $\tau(s)(t) = \sum_{\mathsf{m}\in\mathcal{A}(s)} \alpha(s)(\mathsf{m}) \cdot \varphi(s,\mathsf{m})(t)$. The matrix M_α specifies the transitions of the LMC $\mathcal{D}(\alpha)$ as is defined previously.

We fix two initial distributions μ and ν on S (resp. two initial states s and t) for problems related to total variation distance (resp. probabilistic bisimilarity distance).

3 Trace Inequivalence

In this section we show that one can decide in polynomial time whether there exists a memoryless strategy α so that $\mu \not\equiv \nu$ in $\mathcal{D}(\alpha)$. In terms of the notation from the introduction, we show that $\mathrm{TV} > 0$ is in P. Define the following column-vector spaces.

$\mathcal{V}_1 = \langle M_{\alpha_1}(a_1)M_{\alpha_2}(a_2)\cdots M_{\alpha_m}(a_m)\mathbf{1} : \alpha_i \text{ is a memoryless strategy}; a_i \in L\rangle$ and

$\mathcal{V}_2 = \langle M_\alpha(w)\mathbf{1} : \alpha \text{ is a memoryless strategy}; w \in L^*\rangle$ and

$\mathcal{V}_3 = \langle M_\alpha(w)\mathbf{1} : \alpha \text{ is an MD strategy}; w \in L^*\rangle$.

Here and later we use the notation $\langle\cdot\rangle$ to denote the span of (i.e., the vector space spanned by) a set of vectors. By the definitions, we have that $\mu \equiv \nu$ in all LMCs induced by all memoryless strategies α if and only if $\mu M_\alpha(w)\mathbf{1} = \nu M_\alpha(w)\mathbf{1}$ holds for all memoryless strategies α and all $w \in L^*$. It follows:

▶ **Proposition 1.** *For all distributions μ, ν over S we have:*

$$\exists \text{ a memoryless strategy } \alpha \text{ such that } \mu \not\equiv \nu \text{ in } \mathcal{D}(\alpha) \iff \mu\mathbf{v} \neq \nu\mathbf{v} \text{ for some } \mathbf{v} \in \mathcal{V}_2.$$

To decide $\mathrm{TV} > 0$ and to compute the "witness" memoryless strategy such that $\mu \not\equiv \nu$ in the induced LMC, it suffices to compute a basis for $\mathcal{V}_2$; more precisely, a set of α and w such that the vectors $M_\alpha(w)\mathbf{1}$ span $\mathcal{V}_2$. As the set of memoryless strategies is uncountable, this is not straightforward. From the definitions, we know $\mathcal{V}_3 \subseteq \mathcal{V}_2 \subseteq \mathcal{V}_1$. We will show $\mathcal{V}_1 \subseteq \mathcal{V}_3$ and thus establish the equality of these three vector spaces. It follows from [17, Theorem 5.12] that computing a basis for $\mathcal{V}_1$ is in P. It follows that our problem $\mathrm{TV} > 0$ is also in P, but this does not explicitly give the witnessing memoryless strategy. Since $\mathcal{V}_2 = \mathcal{V}_3$, there must exist an MD strategy that witnesses $\mu \not\equiv \nu$. To find this MD strategy, one can go through all MD strategies (potentially exponentially many). In the following, by considering the vector spaces while restricting the word length, we show that a witness MD strategy can also be computed in polynomial time.

We define the following column-vector spaces. For each $j \in \mathbb{N}$,

$\mathcal{V}_1^j = \langle M_{\alpha_1}(a_1)M_{\alpha_2}(a_2)\cdots M_{\alpha_k}(a_k)\mathbf{1} : \alpha_i \text{ is a memoryless strategy}; a_i \in L; k \le j\rangle$ and

$\mathcal{V}_2^j = \langle M_\alpha(w)\mathbf{1} : \alpha \text{ is a memoryless strategy}; w \in L^{\le j}\rangle$ and

$\mathcal{V}_3^j = \langle M_\alpha(w)\mathbf{1} : \alpha \text{ is an MD strategy}; w \in L^{\le j}\rangle$.

Let α be an MD strategy and m be an action available at state i. Recall that an MD strategy can be viewed as a function $\alpha : S \to \mathcal{A}$. We define $\alpha^{i\to\mathsf{m}}$ to be the MD strategy such that $\alpha^{i\to\mathsf{m}}(i) = \mathsf{m}$ and $\alpha^{i\to\mathsf{m}}(s) = \alpha(s)$ for all $s \in S \setminus \{i\}$. Let $\mathbf{c}_i \in \{0,1\}^S$ be the column bit vector whose only non-zero entry is the ith one. For a set $B \subseteq \mathbb{R}^S$, we define $\langle B\rangle$ to be the vector space spanned by B.

We call a column vector an *MD vector* if it is of the form $M_\alpha(w)\mathbf{1}$ for an MD strategy α and $w \in L^*$. Let P be a set of MD strategy and word pairs, i.e., $P = \{(\alpha_1, w_1), (\alpha_2, w_2), \cdots, (\alpha_m, w_m)\}$ where α_i is an MD strategy and $w_i \in L^*$. We define a function $\mathcal{B}$ transforming such a set P to the set of corresponding MD vectors, i.e., $\mathcal{B}(P) = \{M_{\alpha_1}(w_1)\mathbf{1}, M_{\alpha_2}(w_2)\mathbf{1}, \cdots, M_{\alpha_n}(w_n)\mathbf{1}\}$.

▶ **Lemma 2.** *Let $j \in \mathbb{N}$. For all MD strategies α_1 and α_2, $a \in L$ and $w \in L^{\le j}$, we have $M_{\alpha_1}(a)M_{\alpha_2}(w)\mathbf{1} \in \langle \mathcal{V}_1^j \cup \mathcal{B}(\{(\alpha, aw)\})\rangle$ where α is the MD strategy defined by*

$$\alpha(i) = \begin{cases} \alpha_1(i) & \text{if } \mathbf{c}_i \notin \mathcal{V}_1^j \\ \alpha_2(i) & \text{otherwise} \end{cases}$$

The next lemma shows that a basis for $\mathcal{V}_1^j$ for some $j < |S|$ consisting only of MD vectors can be computed in polynomial time.

▶ **Lemma 3.** *Let $j \in \mathbb{N}$ with $j < |S|$. One can compute in polynomial time a set $P_j = \{(\alpha_0, w_0), \cdots, (\alpha_k, w_k)\}$ in which all α_i are MD strategies and all w_i are in $L^{\leq j}$ such that $\mathcal{B}(P_j)$ is a basis of $\mathcal{V}_1^j$.*

Proof sketch. We prove this lemma by induction on j. The base case where $j = 0$ is vacuously true with $P_0 = \{(\alpha_0, w_0)\}$ where α_0 is an arbitrary MD strategy, $w_0 = \varepsilon$ and $\mathcal{B}(P_0) = \{\mathbf{1}\}$. For the induction step, assume that we can compute in polynomial time a set $P_j = \{(\alpha_0, w_0), \cdots, (\alpha_k, w_k)\}$ where all the strategies are MD strategies and all the words are in $L^{\leq j}$ such that $\mathcal{B}(P_j)$ is a basis for $\mathcal{V}_1^j$. We show that the statement holds for $j + 1$. Define

$$\Sigma = \{\alpha_0\} \cup \{\alpha_0^{s \to \mathsf{m}} : s \in S,\ \mathsf{m} \in \mathcal{A}(s)\} \qquad \text{and} \qquad \mathbb{M} = \{M_\alpha(a) \in \mathbb{R}^{S \times S} : \alpha \in \Sigma,\ a \in L\}.$$

Next, we present Algorithm 1 which computes a set P_{j+1} in polynomial time such that

$$\text{for all } M \in \mathbb{M} \text{ and all } \mathbf{b} \in \mathcal{B}(P_j) : M \cdot \mathbf{b} \in \langle \mathcal{B}(P_{j+1}) \rangle \tag{1}$$

Algorithm 1 Polynomial-time algorithm computing P_{j+1}.

```
1 P_{j+1} := P_j
2 foreach α_1 ∈ Σ, a ∈ L and (α_2, w) ∈ P_j do
3     if M_{α_1}(a)M_{α_2}(w)1 ∉ ⟨B(P_{j+1})⟩ then
4         add (α, aw) to P_{j+1} where α is the MD strategy defined as
                α(i) = { α_1(i)  if c_i ∉ V_1^j
                       { α_2(i)  otherwise.
5     end
6 end
```

All the vectors in $\mathcal{B}(P_{j+1})$ are linearly independent, as we only add a pair if the corresponding vector is linearly independent to the existing vectors in $\mathcal{B}(P_{j+1})$ (lines 3-4). Since $\mathcal{B}(P_j)$ is a basis for $\mathcal{V}_1^j$, we can decide whether $\mathbf{c}_i \in \mathcal{V}_1^j$ for $i \in S$ in polynomial time, and thus compute a pair (α, aw) on line 4 in polynomial time. Since $|\Sigma|$ and $|L|$ are polynomial in the size of the MDP, $|P_j| < |S|$, the number of iterations is polynomial in the size of the MDP. The construction of P_{j+1} is then in polynomial time. It remains to show that after adding (α, aw) to P_{j+1} (line 4), we have $M \cdot \mathbf{b} = M_{\alpha_1}(a)M_{\alpha_2}(w)\mathbf{1} \in \langle \mathcal{B}(P_{j+1}) \rangle$. Since the pair (α_2, w) is in P_j, we have $w \in L^{\leq j}$. Then,

$$\begin{aligned}
& M \cdot \mathbf{b} \\
= \ & M_{\alpha_1}(a)M_{\alpha_2}(w)\mathbf{1} \\
\in \ & \langle \mathcal{V}_1^j \cup \mathcal{B}(\{(\alpha, aw)\}) \rangle \qquad \text{[Lemma 2]} \\
= \ & \langle \mathcal{B}(P_j) \cup \mathcal{B}(\{(\alpha, aw)\}) \rangle \qquad [\mathcal{B}(P_j) \text{ is a basis for } \mathcal{V}_1^j \text{ by induction hypothesis}] \\
= \ & \langle \mathcal{B}\big(P_j \cup \{(\alpha, aw)\}\big) \rangle
\end{aligned}$$

Since $P_j \subseteq P_{j+1}$ (line 1), we have $\mathcal{B}(P_j) \subseteq \mathcal{B}(P_{j+1})$. By adding the pair (α, aw) to P_{j+1}, we have $\langle \mathcal{B}\big(P_j \cup \{(\alpha, aw)\}\big) \rangle \subseteq \langle \mathcal{B}(P_{j+1}) \rangle$, and thus $M \cdot \mathbf{b} \in \langle \mathcal{B}(P_{j+1}) \rangle$.

Finally, we show that the set P_{j+1} satisfies $\mathcal{V}_1^{j+1} = \langle \mathcal{B}(P_{j+1}) \rangle$. We have

$$\begin{aligned} \langle \mathcal{B}(P_{j+1}) \rangle &\subseteq \mathcal{V}_3^{j+1} && \text{for all } (\alpha, w) \in P_{j+1} : \alpha \text{ is an MD strategy and } w \in L^{\leq j+1} \\ &\subseteq \mathcal{V}_1^{j+1} && \text{from the definitions} \end{aligned}$$

We prove the other direction $\mathcal{V}_1^{j+1} \subseteq \langle \mathcal{B}(P_{j+1}) \rangle$ in [24]. ◀

Combining classical linear algebra arguments about equivalence checking (see, e.g., [38]) with Lemma 3, we obtain:

▶ **Lemma 4.**
1. *For all $j < |S|$ we have $\mathcal{V}_1^j = \mathcal{V}_2^j = \mathcal{V}_3^j$.*
2. *We have $\mathcal{V}_1 = \mathcal{V}_2 = \mathcal{V}_3 = \mathcal{V}_1^{|S|-1} = \mathcal{V}_2^{|S|-1} = \mathcal{V}_3^{|S|-1}$.*

Thus we obtain:

▶ **Proposition 5.** *One can compute in polynomial time a set $P = \{(\alpha_0, w_0), \cdots, (\alpha_k, w_k)\}$ of MD strategy and word pairs such that $\mathcal{B}(P)$ is a basis of $\mathcal{V}_2$.*

Proof. By Lemma 4 it suffices to invoke Lemma 3 for $j = |S| - 1$. ◀

Now we can prove the main theorem of this section.

▶ **Theorem 6.** *The problem* $\mathrm{TV} > 0$ *is in* P*. Further, for any positive instance of the problem* $\mathrm{TV} > 0$*, we can compute in polynomial time an MD strategy α and a word w that witness $\mu \not\equiv \nu$, i.e., $\Pr_{\mu,\mathcal{D}(\alpha)}(Run(w)) \neq \Pr_{\nu,\mathcal{D}(\alpha)}(Run(w))$.*

Proof. A polynomial algorithm follows naturally from Proposition 5 and Proposition 1. We first compute a set P of MD strategy and word pairs such that $\mathcal{B}(P)$ is a basis for $\mathcal{V}_2$. For each $\mathbf{b} \in \mathcal{B}(P)$, we check whether $\mu\mathbf{b} \neq \nu\mathbf{b}$ and output "yes" indicating a positive instance if the inequality holds. Otherwise, we have $\mu\mathbf{b} = \nu\mathbf{b}$ for all $\mathbf{b} \in \mathcal{B}(P)$, and the algorithm outputs "no" indicating that $\mu \equiv \nu$ holds for all memoryless strategies.

If the instance is positive, there exists a vector $\mathbf{b} \in \mathcal{B}(P)$ such that $\mu\mathbf{b} \neq \nu\mathbf{b}$. Since $\mathbf{b}$ is an MD vector which corresponds to a pair $(\alpha, w) \in P$, we have $\mu M_\alpha(w)\mathbf{1} \neq \nu M_\alpha(w)\mathbf{1}$, equivalently $\Pr_{\mu,\mathcal{D}(\alpha)}(Run(w)) \neq \Pr_{\nu,\mathcal{D}(\alpha)}(Run(w))$. ◀

4 Probabilistic Bisimulation Inequivalence

In this section we show that one can decide in polynomial time whether there exists a memoryless strategy α so that $s \not\sim t$ in $\mathcal{D}(\alpha)$, i.e., we show that $\mathrm{PB} > 0$ is in P.

For some MDPs, there might be memoryless strategies such that $s \not\sim t$ in the induced LMC but no such strategy is MD. The MDP in Figure 1 is such an example. Similar to the *or*-gate construction of [8, Theorem 2], we have $s \sim t$ if and only if $q_1 \sim q_2$ or $q_2 \sim q_3$. We have $q_2 \sim q_1$ if the MD strategy maps q_2 to the action that goes to state u, otherwise $q_2 \sim q_3$ if the MD strategy maps q_2 to the action that goes to state v. This rules out the algorithm that goes through all the MD strategies.

We define an equivalence relation and run the classical polynomial-time partition refinement as shown in Algorithm 2, with an equivalence relation $\equiv_X$ defined below. At the beginning, all states are in the same equivalence class. In a refinement step, a pair of states is split if there *could* exist a memoryless strategy that makes them not probabilistic bisimilar. Two states s, t remain in the same equivalence class until the end if and only if they are probabilistic bisimilar under all memoryless strategies.

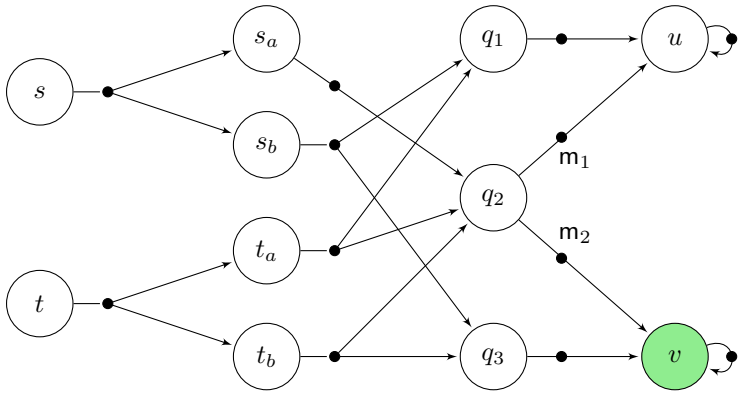

Figure 1 In this MDP no MD strategy witnesses $s \not\sim t$. All states have the same label except state v. By default the transition probabilities out of each action are uniformly distributed.

Algorithm 2 Partition Refinement.

$i = 0; X_0 := \{S\}$
repeat
  $i := i + 1$
  $X_i := S/{\equiv_{X_{i-1}}}$
until $X_i = X_{i-1}$

The correctness of this approach is not obvious, as some splits that occurred in different iterations of the algorithm may have been due to different, potentially contradictory, memoryless strategies. Furthermore, the algorithm does not compute a memoryless strategy that witnesses $s \not\sim t$. The key to solving both problems will be Lemma 11.

A partition of the states S is a set X consisting of pairwise disjoint subsets E of S with $\bigcup_{E \in X} = S$. Recall that $\varphi(s, \mathsf{m})(s')$ is the transition probability from s to s' when choosing action m. Similarly, $\varphi(s, \mathsf{m})(E)$ is the transition probability from s to $E \subseteq S$ when choosing action m. We write $\varphi(s, \mathsf{m})(X)$ to denote the vector (probability distribution) $(\varphi(s, \mathsf{m})(E))_{E \in X}$. We define $\varphi(s)(X) = \{\varphi(s, \mathsf{m})(X) : \mathsf{m} \in \mathcal{A}(s)\}$, which is a set of probabilistic distributions over the partition X when choosing all available actions of s. Each partition is associated with an equivalence relation $\equiv_X$ on S: $s \equiv_X s'$ if and only if

- $\ell(s) = \ell(s')$;
- $s \neq s' \implies |\varphi(s)(X)| = 1$ and $\varphi(s)(X) = \varphi(s')(X)$.

Let $S/{\equiv_X}$ denote the set of equivalence classes with respect to $\equiv_X$, which forms a partition of S. We present in Table 2 the partitions of running the algorithm on the MDP in Figure 1. Notice that states s and t are no longer in the same equivalence class at the end.

Table 2 Example of running Algorithm 2 on the MDP in Figure 1.

$X_0 = \{S\}$
$X_1 = \big\{\{v\}, S \setminus \{v\}\big\}$
$X_2 = \big\{\{v\}, \{q_2\}, \{q_3\}, S \setminus \{v, q_2, q_3\}\big\}$
$X_3 = \big\{\{v\}, \{q_2\}, \{q_3\}, \{s_a\}, \{s_b\}, \{t_a\}, \{t_b\}, \{s, t, q_1, u\}\big\}$
$X_4 = \big\{\{v\}, \{q_2\}, \{q_3\}, \{s_a\}, \{s_b\}, \{t_a\}, \{t_b\}, \{s\}, \{t\}, \{q_1, u\}\ \big\}$

The following lemma is standard, and claims that the partition gets finer.

▶ **Lemma 7.** *For all $i \in \mathbb{N}$, we have $\equiv_{X_{i+1}} \subseteq \equiv_{X_i}$.*

If the loop in Algorithm 2 is performed $|S|-1$ times then $X_{|S|-1}$ consists of $|S|$ one-element sets. Hence at most after $|S|-1$ refinement steps the partition X_i cannot be refined. We aim at proving that $s \equiv_{X_{|S|-1}} t$ if and only if $s \sim_{\mathcal{D}(\alpha)} t$ for all memoryless strategies α. In the following lemma we show the forward direction:

▶ **Lemma 8.** *Let X be a partition and $X = S/\equiv_X$. We have $\equiv_X \subseteq \sim_{\mathcal{D}(\alpha)}$ for all memoryless strategies α.*

For the converse, to guarantee $\equiv_{X_{|S|-1}}$ is not too fine, it suffices to show that there exists a memoryless strategy α' such that $\sim_{\mathcal{D}(\alpha')} \subseteq \equiv_X$ where $X = S/\equiv_X$. To do that, we define the equivalence relations $\sim^i_{\mathcal{D}(\alpha)}$ with $0 \le i \le |S|$ for all memoryless strategies α.

Let α be a memoryless strategy. Let τ be the transition function for the LMC $\mathcal{D}(\alpha)$. Define the equivalence relation $\sim^i_{\mathcal{D}(\alpha)}$ with $0 \le i \le |S|$ on S: $s \sim^i_{\mathcal{D}(\alpha)} s'$ if and only if

- $\ell(s) = \ell(s')$;
- $i > 0 \implies \tau(s)(E) = \tau(s')(E)$ for all $E \in S/\sim^{i-1}_{\mathcal{D}(\alpha)}$.

Note that for the LMC $\mathcal{D}(\alpha)$, we have $\sim^{i+1}_{\mathcal{D}(\alpha)} \subseteq \sim^i_{\mathcal{D}(\alpha)}$ for all $i \in \mathbb{N}$ and $\sim^{|S|-1}_{\mathcal{D}(\alpha)}$ is the probabilistic bisimilarity for the LMC $\mathcal{D}(\alpha)$ (see, e.g., [2]).

Since the witness strategy might not be MD, we compute a set of prime numbers that can be used to form the weights of the actions. The prime numbers are used to rule out certain "accidental" bisimulations. We denote by size($\mathcal{D}$) the size of the representation of an object $\mathcal{D}$. We represent rational numbers as quotients of integers written in binary.

For $u \in S$, $\mathsf{m} \in \mathcal{A}(u)$ and $E \subseteq S$, we express $\varphi(u,\mathsf{m})(E)$ as an irreducible fraction $\frac{a_{u,\mathsf{m},E}}{b_{u,\mathsf{m},E}}$ where $a_{u,\mathsf{m},E}$ and $b_{u,\mathsf{m},E}$ are coprime integers. Similarly, for $u \in S$, $\mathsf{m}_1, \mathsf{m}_2 \in \mathcal{A}(u)$ and $E \subseteq S$, $\varphi(u,\mathsf{m}_1)(E) - \varphi(u,\mathsf{m}_2)(E)$ is expressed as an irreducible fraction $\frac{c_{u,\mathsf{m}_1,\mathsf{m}_2,E}}{d_{u,\mathsf{m}_1,\mathsf{m}_2,E}}$ that $c_{u,\mathsf{m}_1,\mathsf{m}_2,E}$ and $d_{u,\mathsf{m}_1,\mathsf{m}_2,E}$ are coprime integers. Let $N \subseteq \mathbb{N}$ be the following set:

$$\begin{aligned} N \quad = \quad & \{b_{u,\mathsf{m},E} : u \in S,\ \mathsf{m} \in \mathcal{A}(u) \text{ and } E \in \textstyle\bigcup_i X_i\} \cup \\ & \{c_{u,\mathsf{m}_1,\mathsf{m}_2,E} : u \in S,\ \mathsf{m}_1, \mathsf{m}_2 \in \mathcal{A}(u),\ E \in \textstyle\bigcup_i X_i \text{ and } c_{u,\mathsf{m}_1,\mathsf{m}_2,E} > 0\}. \end{aligned}$$

We denote by $\theta(x)$ the number of different prime factors of a positive integer x, and by $\theta(N)$ the number of different prime factors in N where N is a set of positive integers.

▶ **Lemma 9.** *$\theta(N)$ is polynomial in size($\mathcal{D}$).*

Using the prime number theorem, we obtain the following lemma which guarantees that one can find $|S|$ extra different prime numbers other than the prime factors in N in time polynomial in size($\mathcal{D}$).

▶ **Lemma 10.** *One can find $|S|$ different prime numbers in time polynomial in size($\mathcal{D}$) such that any of them is coprime to all numbers in the set N.*

To each $u \in S$, we assign a different prime number p_u that is coprime with all $b \in N$. This can be done in polynomial time by Lemma 10. We have

$$p_u \nmid b \text{ for all } b \in N \quad \text{and} \quad u \neq v \implies p_u \neq p_v \text{ for all } u, v \in S \tag{2}$$

We define a partial memoryless strategy for $\mathcal{D}$ to be a partial function $\alpha' : S \nrightarrow \mathrm{Distr}(\mathcal{A})$ that, given a state $s \in S$, returns $\alpha'(s) \in \mathrm{Distr}(\mathcal{A}(s))$ if $\alpha'(s)$ is defined. A memoryless strategy α is compatible with a partial memoryless strategy α', written as $\alpha \sqsupseteq \alpha'$, if and only if $\alpha(s) = \alpha'(s)$ for all s such that $\alpha'(s)$ is defined. We construct the partial memoryless strategy iteratively.

▶ **Lemma 11.** *Let $i \in \mathbb{N}$ with $i \leq |S|$. One can compute in polynomial time a partial strategy α'_i such that $\sim^i_{\mathcal{D}(\alpha)} \subseteq \equiv_{X_i}$ for all $\alpha \sqsupseteq \alpha'_i$.*

Proof sketch. We prove the statement by induction on i. Let $s, t \in S$. The base case is $i = 0$. By definition, we have if $s \not\equiv_{X_0} t$ then $\ell(s) \neq \ell(t)$. We also have if $\ell(s) \neq \ell(t)$, then $s \not\sim^0_{\mathcal{D}(\alpha)} t$ in $\mathcal{D}(\alpha)$ for all memoryless strategy α. We simply let α'_0 be the empty partial function such that $\alpha \sqsupseteq \alpha'_0$ holds for any memoryless strategy α.

For the induction step, assume that we can compute in polynomial time a partial strategy α'_i such that $\sim^i_{\mathcal{D}(\alpha)} \subseteq \equiv_{X_i}$ for all $\alpha \sqsupseteq \alpha'_i$, i.e., if $s \not\equiv_{X_i} t$ then $s \not\sim^i_{\mathcal{D}(\alpha)} t$ in $\mathcal{D}(\alpha)$. We show the statement holds for $i + 1$.

Algorithm 3 Polynomial-time algorithm constructing α'_{i+1}.

```
1 α'_{i+1} := α'_i
2 foreach u ∈ S such that |φ(u)(X_i)| = 1 and |φ(u)(X_{i+1})| ≠ 1 do
3 |   pick m_1, m_2 ∈ A(u) such that for a set E ∈ X_{i+1} : φ(u, m_1)(E) > φ(u, m_2)(E)
4 |   α'_{i+1}(u)(m_1) := 1/p_u
5 |   α'_{i+1}(u)(m_2) := 1 − 1/p_u
6 end
```

Algorithm 3 computes the partial memoryless strategy α'_{i+1} in polynomial time. We show that α'_j does not overwrite α'_k for all $k < j$. It follows that for any $\alpha \sqsupseteq \alpha'_{i+1}$, it satisfies $\alpha \sqsupseteq \alpha'_i$. Let $\alpha \sqsupseteq \alpha'_{i+1}$. Assume $s \not\equiv_{X_{i+1}} t$. We distinguish the two cases: $s \not\sim^i_{\mathcal{D}(\alpha)} t$ and $s \sim^i_{\mathcal{D}(\alpha)} t$. For both cases we can derive $s \not\sim^{i+1}_{\mathcal{D}(\alpha)} t$, i.e., $\sim^{i+1}_{\mathcal{D}(\alpha)} \subseteq \equiv_{X_{i+1}}$ as desired. The details can be found in [24]. ◀

For example, let p_{q_2}, the prime number assigned to state q_2 in Figure 1, be 3 which is coprime with numbers in $N = \{1, 2\}$.[1] We show how the partial strategy α'_1 is constructed. On line 1 of Algorithm 3, α'_1 is equal to α'_0, the empty partial function. Since $|\varphi(q_2)(X_0)| = 1$ and $|\varphi(q_2)(X_1)| = 2$, we enter the for loop. We can pick $\mathsf{m}_1, \mathsf{m}_2 \in \mathcal{A}(q_2)$ and $E = S \setminus \{v\} \in X_1$ on line 3, since $\varphi(q_2, \mathsf{m}_1)(E) = 1 > 0 = \varphi(q_2, \mathsf{m}_2)(E)$. We then define the strategy for q_2 (line 4 and 5): $\alpha'_1(q_2)(\mathsf{m}_1) = \frac{1}{3}$ and $\alpha'_1(q_2)(\mathsf{m}_2) = \frac{2}{3}$. We have completed the construction of α'_1 as $|\varphi(u)(X_0)| = |\varphi(u)(X_1)| = 1$ for all other state u.

▶ **Theorem 12.** *One can compute in polynomial time a memoryless strategy β such that $\sim_{\mathcal{D}(\beta)} \subseteq \sim_{\mathcal{D}(\alpha)}$ for all memoryless strategies α.*

Proof. By invoking Lemma 11 for $i = |S| - 1$, a partial strategy $\alpha'_{|S|-1}$ can be computed in polynomial time such that $\sim^{|S|-1}_{\mathcal{D}(\alpha)} \subseteq \equiv_{X_{|S|-1}}$ for all $\alpha \sqsupseteq \alpha'_{|S|-1}$. Since $\sim^{|S|-1}_{\mathcal{D}(\alpha)} = \sim_{\mathcal{D}(\alpha)}$, we have $\sim_{\mathcal{D}(\alpha)} \subseteq \equiv_{X_{|S|-1}}$ for all $\alpha \sqsupseteq \alpha'_{|S|-1}$. Let β be a memoryless strategy defined by

$$\beta(u) = \begin{cases} \alpha'_{|S|-1}(u) & \text{if } \alpha'_{|S|-1}(u) \text{ is defined} \\ \delta_{\mathsf{m}_u} \text{ where } \mathsf{m}_u \in \mathcal{A}(u) & \text{otherwise} \end{cases}$$

By definition the memoryless strategy β is compatible with $\alpha'_{|S|-1}$. We have:

$$\begin{aligned} \sim_{\mathcal{D}(\beta)} &\subseteq \equiv_{X_{|S|-1}} && \beta \sqsupseteq \alpha'_{|S|-1} \\ &\subseteq \sim_{\mathcal{D}(\alpha)} \text{ for all strategy } \alpha && X_{|S|-1} = S/{\equiv_{X_{|S|-1}}} \text{ and Lemma 8} \end{aligned}$$ ◀

[1] We have $2 \in N$ since $\varphi(s, \mathsf{m}_s)(\{s_a\}) = \frac{1}{2}$ where m_s is the only available action at state s.

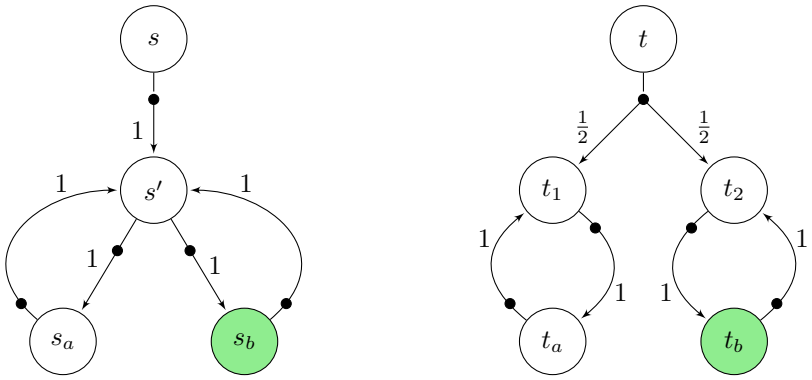

Figure 2 In this MDP, no MD strategy witnesses $d_{tv}(\delta_s, \delta_t) = 1$ (nor $d_{pb}(s,t) = 1$). States s_b and t_b have label b while all other states have label a.

▶ **Corollary 13.** *The problem* PB > 0 *is in* P*. Further, for any positive instance of the problem* PB > 0*, we can compute in polynomial time a memoryless strategy that witnesses* $s \not\sim t$*.*

5 The Distance One Problems

In this section, we summarise the results for the two distance one problems, namely TV $= 1$ and PB $= 1$. The *existential theory of the reals, ETR,* is the set of valid formulas of the form

$$\exists x_1 \ldots \exists x_n \; R(x_1, \ldots, x_n),$$

where R is a boolean combination of comparisons of the form $p(x_1, \ldots, x_n) \sim 0$, in which $p(x_1, \ldots, x_n)$ is a multivariate polynomial (with rational coefficients) and $\sim \in \{<, >, \leq, \geq, =, \neq\}$. The complexity class $\exists\mathbb{R}$ [32] consists of those problems that are many-one reducible to ETR in polynomial time. Since ETR is NP-hard and in PSPACE [5, 31], we have NP $\subseteq \exists\mathbb{R} \subseteq$ PSPACE.

For some MDPs there exist memoryless strategies that make $d_{tv}(\delta_s, \delta_t) = 1$ but no such strategy is MD. For example, consider the MDP in Figure 2 which has two MD strategies. We have $d_{tv}(\delta_s, \delta_t) = \frac{1}{2}$ which is less than 1 in the LMC induced by any of the two MD strategies, and $d_{tv}(\delta_s, \delta_t) = 1$ in the LMC induced by any other strategy. Thus, we cannot simply guess an MD strategy. We show that the problem TV $= 1$ is in $\exists\mathbb{R}$, using the characterization from [9, Theorem 21] of total variation distance 1 in LMCs and some reasoning on convex polyhedra:

▶ **Theorem 14.** *The problem* TV $= 1$ *is in* $\exists\mathbb{R}$*.*

The problem TV $= 1$ is NP-hard, and PB $= 1$ is NP-complete. The hardness results for both problems are by reductions from the Set Splitting problem. Given a finite set S and a collection $\mathcal{C}$ of subsets of S, Set Splitting asks whether there is a partition of S into disjoint sets S_1 and S_2 such that no set in $\mathcal{C}$ is a subset of S_1 or S_2.

Let $\langle S, \mathcal{C} \rangle$ be an instance of Set Splitting where $S = \{e_1, \cdots, e_n\}$ and $\mathcal{C} = \{C_1, \cdots, C_m\}$ is a collection of subsets of S. We construct an MDP $\mathcal{D}$ consisting of the following states: two states s and t, a state e_i for each element in S, twin states C_j and C'_j for each element in $\mathcal{C}$, two sink states u and v. State v has label b while all other states have label a. State s (t) has a single action which goes with uniform probability $\frac{1}{m}$ to states C_i (C'_i) for $1 \leq i \leq m$. For each

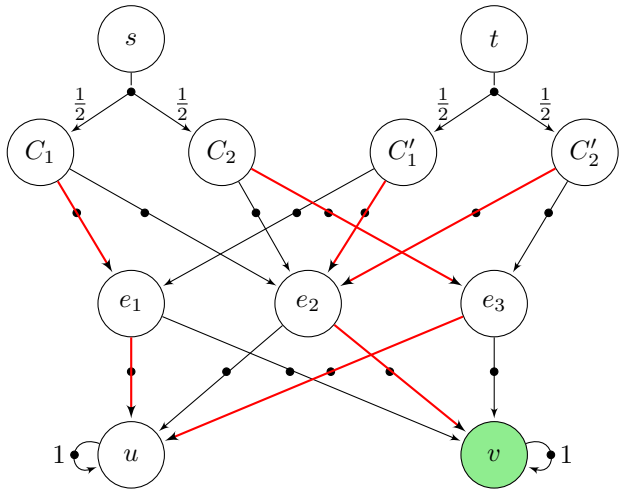

Figure 3 The MDP in the reduction from Set Splitting for NP-hardness of TV = 1 (or PB = 1).

$e_i \in C_j$, there is an action from state C_j and C'_j leading to state e_i with probability one. Each state e_i has two actions going to the sink states u and v with probability one, respectively. We have: $\langle S, \mathcal{C} \rangle \in$ Set Splitting $\iff$ $\exists$ memoryless strategy α such that $d_{tv}(\delta_s, \delta_t) = 1$ in $\mathcal{D}(\alpha)$.

For example, let $S = \{e_1, e_2, e_3\}$ and $\mathcal{C} = \{C_1, C_2\}$ with $C_1 = \{e_1, e_2\}$ and $C_2 = \{e_2, e_3\}$. Figure 3 shows the corresponding MDP. The MD strategy highlighted, corresponding to the partition of $S_1 = \{e_1, e_3\}$ and $S_2 = \{e_2\}$, witnesses $d_{tv}(\delta_s, \delta_t) = 1$.

▶ **Theorem 15.** *The Set Splitting problem is polynomial-time many-one reducible to* TV = 1*, hence* TV = 1 *is* NP*-hard.*

The problem PB = 1 is NP-complete. The MDP in Figure 2 is also an example of no MD strategy witnessing $d_{pb}(s, t) = 1$, which rules out the algorithm of simply guessing an MD strategy. By [36], deciding whether $d_{pb}(s, t) = 1$ in an LMC can be formulated as a reachability problem on a directed graph induced by the LMC. One can nondeterministically guess the graph induced by the LMC and use Algorithm 3 to construct a memoryless strategy that witnesses $d_{pb}(s, t) = 1$.

▶ **Theorem 16.** *The problem* PB = 1 *is* NP*-complete.*

6 Making Distances Small

In this section, we summarise the results for the remaining problems, which are all about making the distance small (equal to 0 or less than 1).

We show that TV = 0 and TV < 1 are $\exists\mathbb{R}$-complete. The proof for the membership of TV = 0 in $\exists\mathbb{R}$ is similar to [17, Theorem 4.3]. For both hardness results we provide reductions from the *Nonnegative Matrix Factorization (NMF)* problem, which asks, given a nonnegative matrix $J \in \mathbb{Q}^{n \times m}$ and a number $r \in \mathbb{N}$, whether there exists a factorization $J = A \cdot W$ with nonnegative matrices $A \in \mathbb{R}^{n \times r}$ and $W \in \mathbb{R}^{r \times m}$. The NMF problem is $\exists\mathbb{R}$-complete by [35, Theorem 2], see also [10, 40, 1] for more details on the NMF problem. The reduction is similar to [17, Theorem 4.5].

▶ **Theorem 17.** *The problem* TV = 0 *is* $\exists\mathbb{R}$*-complete.*

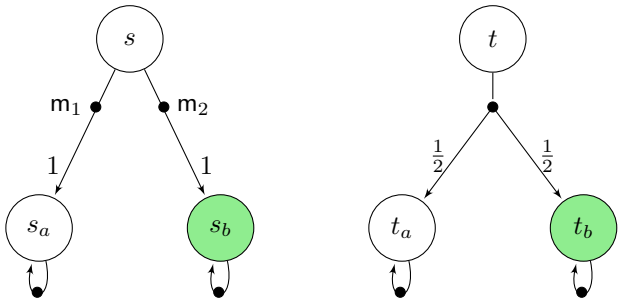

Figure 4 In this MDP, no MD strategy witnesses $d_{pb}(s,t) = 0$. States s_b and t_b have label b while all other states have label a.

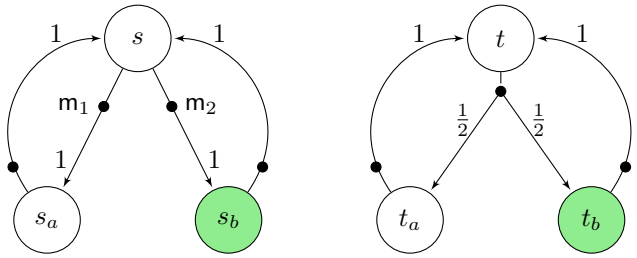

Figure 5 In this MDP, no MD strategy witnesses $d_{pb}(s,t) < 1$. States s_b and t_b have label b while all other states have label a.

▶ **Theorem 18.** *The problem* TV < 1 *is* $\exists\mathbb{R}$*-complete.*

Finally, we show that PB $= 0$ and PB < 1 are NP-complete. For some MDPs there exist memoryless strategies that make $d_{pb}(s,t) = 0$ (resp. $d_{pb}(s,t) < 1$) but no such strategy is MD. Indeed, for the MDP in Figure 4 (resp. Figure 5), it is easy to check that the only strategy α which makes $d_{pb}(s,t) = 0$ (resp. $d_{pb}(s,t) < 1$), requires randomness, that is, $\alpha(s)(\mathsf{m}_1) = \alpha(s)(\mathsf{m}_2) = \frac{1}{2}$, where m_1 and m_2 are the two available actions of state s. Thus, to show the NP upper bound, we cannot simply guess an MD strategy. Instead, one can nondeterministically guess a partition of the states and check in polynomial time if the partition is a probabilistic bisimulation.

The hardness results for both problems are by reductions from the Subset Sum problem. The reduction is similar to [17, Theorem 4.1].

▶ **Theorem 19.** *The problem* PB $= 0$ *is* NP*-complete.*

By [36], deciding whether $d_{pb}(s,t) < 1$ in an LMC can be formulated as a reachability problem on a directed graph induced by the LMC. In addition to a partition, our NP algorithm also guesses the graph induced by the LMC.

▶ **Theorem 20.** *The problem* PB < 1 *is* NP*-complete.*

7 Conclusions

We have studied the computational complexity of qualitative comparison problems in labelled MDPs. Motivated by the connection between obliviousness/anonymity and equivalence, we have devised polynomial-time algorithms to decide the existence of strategies for trace and bisimulation *in*equivalence. In case of trace inequivalence, there always exists an MD witness

strategy, and our algorithm computes it. The trace inequivalence algorithm is based on linear-algebra arguments that are considerably more subtle than in the LMC case. For bisimulation inequivalence, MD strategies may not exist, but we have devised a polynomial-time algorithm to compute a memoryless strategy witnessing inequivalence; here the randomization is based on prime numbers to rule out certain "accidental" bisimulations. The other 6 problems do not have polynomial complexity (unless $\mathsf{P} = \mathsf{NP}$), and we have established completeness results for all of them except $\mathrm{TV} = 1$, where a complexity gap between NP and $\exists\mathbb{R}$ remains.

Concerning the relationship to interval Markov chains and parametric Markov chains mentioned in the introduction, the lower complexity bounds that we have derived in this paper carry over to corresponding problems in these models. Transferring the upper bounds requires additional work, as, e.g., even the consistency problem for IMCs (i.e., whether there *exists* a Markov chain conforming to an IMC) is not obvious to solve. Nevertheless, the algorithmic insights of this paper will be needed.

References

1 Sanjeev Arora, Rong Ge, Ravi Kannan, and Ankur Moitra. Computing a nonnegative matrix factorization - provably. In *STOC*, pages 145–162. ACM, 2012.

2 Christel Baier. Polynomial time algorithms for testing probabilistic bisimulation and simulation. In Rajeev Alur and Thomas A. Henzinger, editors, *Computer Aided Verification*, pages 50–61, Berlin, Heidelberg, 1996. Springer Berlin Heidelberg.

3 Michael Benedikt, Rastislav Lenhardt, and James Worrell. LTL model checking of interval Markov chains. In Nir Piterman and Scott A. Smolka, editors, *Tools and Algorithms for the Construction and Analysis of Systems - 19th International Conference, TACAS 2013, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2013, Rome, Italy, March 16-24, 2013. Proceedings*, volume 7795 of *Lecture Notes in Computer Science*, pages 32–46. Springer, 2013.

4 Patrick Billingsley. *Probability and measure*. Wiley Series in Probability and Statistics. Wiley, New York, NY, USA, 3rd edition, 1995.

5 John Canny. Some algebraic and geometric computations in PSPACE. In *STOC*, pages 460–467, 1988.

6 Souymodip Chakraborty and Joost-Pieter Katoen. Model checking of open interval Markov chains. In Marco Gribaudo, Daniele Manini, and Anne Remke, editors, *Analytical and Stochastic Modelling Techniques and Applications*, pages 30–42. Springer International Publishing, 2015.

7 Krishnendu Chatterjee, Koushik Sen, and Thomas A. Henzinger. Model-checking omega-regular properties of interval Markov chains. In Roberto M. Amadio, editor, *Foundations of Software Science and Computational Structures, 11th International Conference, FOSSACS 2008, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2008, Budapest, Hungary, March 29 - April 6, 2008. Proceedings*, volume 4962 of *Lecture Notes in Computer Science*, pages 302–317. Springer, 2008.

8 Di Chen, Franck van Breugel, and James Worrell. On the complexity of computing probabilistic bisimilarity. In Lars Birkedal, editor, *Proceedings of the 15th International Conference on Foundations of Software Science and Computational Structures*, volume 7213 of *Lecture Notes in Computer Science*, pages 437–451, Tallinn, Estonia, March/April 2012. Springer-Verlag.

9 Taolue Chen and Stefan Kiefer. On the total variation distance of labelled Markov chains. In *Proceedings of the Joint Meeting of the Twenty-Third EACSL Annual Conference on Computer Science Logic (CSL) and the Twenty-Ninth Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)*, CSL-LICS '14, New York, NY, USA, 2014. ACM.

10 Joel E. Cohen and Uriel G. Rothblum. Nonnegative ranks, decompositions, and factorizations of nonnegative matrices. *Linear Algebra and its Applications*, 190:149–168, 1993.

11 Benoît Delahaye. Consistency for parametric interval markov chains. In Étienne André and Goran Frehse, editors, *2nd International Workshop on Synthesis of Complex Parameters,*

SynCoP 2015, April 11, 2015, London, United Kingdom, volume 44 of *OASICS*, pages 17–32. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2015.

12 Benoît Delahaye, Kim G. Larsen, Axel Legay, Mikkel L. Pedersen, and Andrzej Wasowski. Decision problems for interval Markov chains. In Adrian-Horia Dediu, Shunsuke Inenaga, and Carlos Martín-Vide, editors, *Language and Automata Theory and Applications - 5th International Conference, LATA 2011, Tarragona, Spain, May 26-31, 2011. Proceedings*, volume 6638 of *Lecture Notes in Computer Science*, pages 274–285. Springer, 2011.

13 Salem Derisavi, Holger Hermanns, and William H. Sanders. Optimal state-space lumping in Markov chains. *Inf. Process. Lett.*, 87(6):309–315, 2003.

14 Josée Desharnais, Vineet Gupta, Radha Jagadeesan, and Prakash Panangaden. Metrics for labeled Markov systems. In Jos Baeten and Sjouke Mauw, editors, *Proceedings of the 10th International Conference on Concurrency Theory*, volume 1664 of *Lecture Notes in Computer Science*, pages 258–273, Eindhoven, The Netherlands, August 1999. Springer-Verlag.

15 Josee Desharnais, Vineet Gupta, Radha Jagadeesan, and Prakash Panangaden. Metrics for labelled Markov processes. *Theor. Comput. Sci.*, 318(3):323–354, 2004.

16 L. Doyen, T.A. Henzinger, and J.-F. Raskin. Equivalence of labeled Markov chains. *International Journal on Foundations of Computer Science*, 19(3):549–563, 2008.

17 Nathanaël Fijalkow, Stefan Kiefer, and Mahsa Shirmohammadi. Trace refinement in labelled Markov decision processes. *Logical Methods in Computer Science*, 16(2), 2020.

18 Ernst Moritz Hahn, Holger Hermanns, and Lijun Zhang. Probabilistic reachability for parametric Markov models. *Int. J. Softw. Tools Technol. Transf.*, 13(1):3–19, 2011.

19 Christian Hensel, Sebastian Junges, Joost-Pieter Katoen, Tim Quatmann, and Matthias Volk. The probabilistic model checker Storm, 2020. `arXiv:arXiv:2002.07080`.

20 Bengt Jonsson and Kim Guldstrand Larsen. Specification and refinement of probabilistic processes. In *Proceedings of the Sixth Annual Symposium on Logic in Computer Science (LICS '91), Amsterdam, The Netherlands, July 15-18, 1991*, pages 266–277. IEEE Computer Society, 1991.

21 John G. Kemeny and J. Laurie Snell. *Finite Markov Chains*. Springer, 1976.

22 S. Kiefer, A.S. Murawski, J. Ouaknine, B. Wachter, and J. Worrell. Language equivalence for probabilistic automata. In *CAV*, volume 6806 of *LNCS*, pages 526–540. Springer, 2011.

23 S. Kiefer, A.S. Murawski, J. Ouaknine, B. Wachter, and J. Worrell. APEX: An analyzer for open probabilistic programs. In *CAV*, volume 7358 of *LNCS*, pages 693–698. Springer, 2012.

24 Stefan Kiefer and Qiyi Tang. Comparing labelled markov decision processes, 2020. `arXiv:2009.11643`.

25 M. Kwiatkowska, G. Norman, and D. Parker. PRISM 4.0: Verification of probabilistic real-time systems. In G. Gopalakrishnan and S. Qadeer, editors, *Proc. 23rd International Conference on Computer Aided Verification (CAV'11)*, volume 6806 of *LNCS*, pages 585–591. Springer, 2011.

26 Kim Guldstrand Larsen and Arne Skou. Bisimulation through probabilistic testing. *Inf. Comput.*, 94(1):1–28, 1991.

27 L. Li and Y. Feng. Quantum Markov chains: Description of hybrid systems, decidability of equivalence, and model checking linear-time properties. *Information and Computation*, 244:229–244, 2015.

28 T.M. Ngo, M. Stoelinga, and M. Huisman. Confidentiality for probabilistic multi-threaded programs and its verification. In *Engineering Secure Software and Systems*, volume 7781 of *LNCS*, pages 107–122. Springer, 2013.

29 A. Paz. *Introduction to Probabilistic Automata*. Academic Press, 1971.

30 S. Peyronnet, M. de Rougemont, and Y. Strozecki. Approximate verification and enumeration problems. In *ICTAC*, volume 7521 of *LNCS*, pages 228–242. Springer, 2012.

31 James Renegar. On the computational complexity and geometry of the first-order theory of the reals. Parts I–III. *Journal of Symbolic Computation*, 13(3):255–352, 1992.

32 Marcus Schaefer and Daniel Stefankovic. Fixed points, Nash equilibria, and the existential theory of the reals. *Theory Comput. Syst.*, 60(2):172–193, 2017. doi:10.1007/s00224-015-9662-0.

33 M.-P. Schützenberger. On the definition of a family of automata. *Information and Control*, 4:245–270, 1961.

34 Koushik Sen, Mahesh Viswanathan, and Gul Agha. Model-checking markov chains in the presence of uncertainties. In Holger Hermanns and Jens Palsberg, editors, *Tools and Algorithms for the Construction and Analysis of Systems, 12th International Conference, TACAS 2006 Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2006, Vienna, Austria, March 25 - April 2, 2006, Proceedings*, volume 3920 of *Lecture Notes in Computer Science*, pages 394–410. Springer, 2006.

35 Yaroslav Shitov. A universality theorem for nonnegative matrix factorizations, 2016. arXiv:1606.09068.

36 Qiyi Tang and Franck van Breugel. Deciding probabilistic bisimilarity distance one for labelled Markov chains. In Hana Chockler and Georg Weissenbacher, editors, *Proceedings of the 30th International Conference on Computer Aided Verification*, volume 10981 of *Lecture Notes in Computer Science*, pages 681–699, Oxford, UK, July 2018. Springer-Verlag. doi:10.1007/978-3-319-96145-3_39.

37 Qiyi Tang and Franck van Breugel. Deciding probabilistic bisimilarity distance one for probabilistic automata. *Journal of Computer and System Sciences*, 111:57–84, 2020.

38 Wen-Guey Tzeng. A polynomial-time algorithm for the equivalence of probabilistic automata. *SIAM Journal on Computing*, 21(2):216–227, 1992.

39 Antti Valmari and Giuliana Franceschinis. Simple $O(m \log n)$ time Markov chain lumping. In Javier Esparza and Rupak Majumdar, editors, *Tools and Algorithms for the Construction and Analysis of Systems, 16th International Conference, TACAS 2010, Held as Part of the Joint European Conferences on Theory and Practice of Software, ETAPS 2010, Paphos, Cyprus, March 20-28, 2010. Proceedings*, volume 6015 of *Lecture Notes in Computer Science*, pages 38–52. Springer, 2010.

40 Stephen A. Vavasis. On the complexity of nonnegative matrix factorization. *SIAM Journal on Optimization*, 20(3):1364–1377, 2009.

41 Tobias Winkler, Sebastian Junges, Guillermo A. Pérez, and Joost-Pieter Katoen. On the complexity of reachability in parametric markov decision processes. In Wan J. Fokkink and Rob van Glabbeek, editors, *30th International Conference on Concurrency Theory, CONCUR 2019, August 27-30, 2019, Amsterdam, the Netherlands*, volume 140 of *LIPIcs*, pages 14:1–14:17. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019. doi:10.4230/LIPIcs.CONCUR.2019.14.