Explainability is a Game for Probabilistic Bisimilarity Distances

Emily Vlasman
Department of Computer Science, Oxford University, UK

Anto Nanah Ji
Department of Electrical Engineering and Computer Science, York University, Toronto, Canada

James Worrell
Department of Computer Science, Oxford University, UK

Franck van Breugel
Department of Electrical Engineering and Computer Science, York University, Toronto, Canada

Abstract

We revisit a game from the literature that characterizes the probabilistic bisimilarity distances of a labelled Markov chain. We illustrate how an optimal policy of the game can explain these distances. Like the games that characterize bisimilarity and probabilistic bisimilarity, the game is played on pairs of states and matches transitions of those states. To obtain more convincing and interpretable explanations than those provided by generic optimal policies, we restrict to optimal policies that delay reaching observably inequivalent state pairs for as long as possible (called 1-maximal) while quickly reaching equivalent ones (called 0-minimal). We present iterative algorithms that compute 1-maximal and 0-minimal policies and prove an exponential lower bound for the number of iterations of the algorithm that computes 1-maximal policies.

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

Keywords and phrases probabilistic bisimilarity distance, labelled Markov chain, game, policy, explainability

Digital Object Identifier 10.4230/LIPIcs.CONCUR.2025.36

Funding *James Worrell*: EPSRC Fellowship EP/X033813/1.
Franck van Breugel: Natural Sciences and Engineering Research Council of Canada.

Acknowledgements The authors would like to thank the referees for their constructive feedback.

1 Introduction

Probabilistic bisimilarity, a fundamental notion introduced by Larsen and Skou [39], captures which states of a model with randomness are considered behaviourally equivalent. As shown by Katoen, Kemna, Zapreev, and Jansen [31], reducing a model by identifying states that are probabilistic bisimilar often speeds up probabilistic model checking. That is, the time it takes to reduce a model and subsequently check a property of the reduced model is often less than the time needed to check the property of the original model.

As shown by Giacalone, Jou, and Smolka [25], behavioural equivalences such as probabilistic bisimilarity are *not robust*. Even the smallest changes to the probabilities in the model may result in different states being identified as behaviourally equivalent and, hence, may lead to different reduced models. These models may even satisfy different properties.

Giacalone et al. suggested distances as a robust alternative to equivalences. Desharnais, Gupta, Jagadeesan, and Panangaden [14] proposed a quantitative generalization of probabilistic bisimilarity: *probabilistic bisimilarity distances* (or distances for short). To each pair

of states, a real number in the unit interval $[0, 1]$ is assigned that captures the behavioural similarity of the states. The smaller this number, the more alike the states behave. As shown by Desharnais et al., states are probabilistic bisimilar if and only if their distance is zero.

During the last two decades, efficient algorithms have been developed to approximate and compute probabilistic bisimilarity distances (see, for example, the work of Tang [51]). Once we have computed that the distance between two states is, say $\frac{1}{8}$, it begs for an explanation. In this paper we address the question *how to explain probabilistic bisimilarity distances*.

Logic has been extensively used to explain behavioural equivalences. Logics have been designed such that for each pair of states that is not behaviourally equivalent there exists a formula of the logic such one state satisfies the formula and the other state does not. For example, for models with nondeterminism, such as labelled transition systems, bisimilarity, due to Milner [42] and Park [44], is a key behavioural equivalence. The Hennessy-Milner logic [27] provides a logical characterization of bisimilarity. For labelled Markov chains, which model systems with randomness, probabilistic bisimilarity is logically characterized by the probabilistic modal logic of [39]. Rady and Van Breugel [47] explained the distance of a pair of states by means of an infinite sequence of formulas of a logic. The major drawback of their approach is that the explanation is in general not finitely representable. In this paper, we use games to explain distances. As we will see, these can be finitely represented. Before introducing our game, we first review some related work from the literature as our approach is different from the commonly used Ehrenfeucht-Fraïssé-like games [19, 24], yet uses some ingredients of those games.

1.1 Bisimilarity

As was shown by Stirling (see, for example, [50]), bisimilarity for labelled transition systems can be characterized by means of a two-player game. In the literature we find different names for these players including Spoiler, Adversary, and Attacker for the first one and Duplicator, Prover, and Defender for the other player. Here, we use Spoiler and Duplicator. The game starts in a state pair (s, t). Spoiler tries to show that s and t are not bisimilar, whereas Duplicator tries to prove that they are.

The game is played in rounds. If a round starts in state pair (s, t), Spoiler chooses a state $u_1 \in \{s, t\}$ and an outgoing transition of u_1 with target, say, s'. Duplicator uses the other state $u_2 \in \{s, t\} \setminus \{u_1\}$ and chooses one of the outgoing transitions of u_2 with target, say, t' (preferably with the same label as s' – otherwise Duplicator loses). The next round of the game continues in the state pair (s', t'). The objective of Spoiler is to reach a state pair with different labels whereas Duplicator tries to avoid ever reaching such a state pair. As has been shown by Stirling [50, Section 2.4], states s and t are bisimilar if and only if Duplicator can avoid ever reaching a state pair with different labels when the game is started in state pair (s, t), no matter how Spoiler plays.

As pointed out by Fijalkow, Klin, and Panangaden [21], "this classical bisimulation game is elegant because it allows one to characterize a global property of behaviours (bisimilarity) in terms of a game whose rules only depend on local considerations."

1.2 Probabilistic Bisimilarity

Several characterizations of probabilistic bisimilarity for labelled Markov chains in terms of a game can be found in the literature. We will briefly review two of those next. The games are also played in rounds by two players. As in the game for bisimilarity, also in the games for probabilistic bisimilarity, the objective of Spoiler is to reach a state pair with different labels whereas Duplicator tries to avoid ever reaching such a state pair.

Let us first consider the game introduced by Desharnais, Laviolette, and Tracol in [17]. If the game is in state pair (s,t), Spoiler chooses a state $u_1 \in \{s,t\}$ and a set of states U_1. Duplicator uses the other state $u_2 \in \{s,t\} \setminus \{u_1\}$ and chooses a set of states U_2 such that the probability of transitioning to a state in U_1 from state u_1, that is, the sum of probabilities of the transitions from u_1 to a state in U_1, is less than or equal to the probability of transitioning to a state in U_2 from u_2. Such a U_2 always exists as Duplicator can pick the set of all states. Subsequently, Spoiler chooses $i \in \{1,2\}$ and a state $s' \in U_i$. Finally, Duplicator chooses a state $t' \in U_{3-i}$ (preferably with the same label as s' – otherwise Duplicator loses). The next round of the game starts in the state pair (s',t'). As in the bisimulation game, the objective of Spoiler is to reach a state pair with different labels whereas Duplicator tries to avoid ever reaching such a state pair. Desharnais, Laviolette, and Tracol [17, Theorem 5] proved that states s and t are probabilistic bisimilar if and only if Duplicator can avoid ever reaching a state pair with different labels when the game is started in state pair (s,t) no matter how Spoiler plays. Forejt, Jancar, Kiefer, and Worrell [23] generalize this game to a setting with both randomness and nondeterminism. Ford, Beohar, König, Milius, and Schröder [22] consider an even more general setting.

Fijalkow, Klin, and Panangaden [21] propose a slightly simpler game that characterizes probabilistic bisimilarity. If the game is in state-pair (s,t) then Spoiler chooses a set of states U such that the probability of transitioning from state s to a state in U is different from the probability of transitioning from state t to a state in U. If no such choice exists then Spoiler loses the game. Subsequently, Duplicator picks a state u that is in U and a state v that is not in U (preferably with the same label as u – otherwise Duplicator loses), and the game continues in the state pair (u,v). Fijalkow, Klin, and Panangaden [21, Theorem 8] showed that states s and t are probabilistic bisimilar if and only if Duplicator can avoid ever reaching a pair of states with different labels and does not lose, or Spoiler loses when the game is started in state pair (s,t) no matter how Spoiler plays.

Fijalkow, Klin, and Panangaden [21] mentioned that "the connection between metrics and bisimulation is well understood but it is possible that via the game one might gain a more quantitative understanding of the numerical significance of the metric." This paper confirms that this is the case.

1.3 Probabilistic Bisimilarity Distances

König and Mika-Michalski [36] generalize the game of Desharnais, Laviolette, and Tracol in two dimensions. Firstly, they consider distances instead of equivalences[1]. Secondly, they present a general framework based on the category of sets and functions, an endofunctor on that category, and coalgebras of that endofunctor. Below, we describe the game resulting from an instantiation of their framework so that it is applicable to distances for labelled Markov chains. The game starts in a triple (s,t,ϵ), where s and t are states and $\epsilon \in [0,1)$. Spoiler tries to show that the distance of s and t is greater than ϵ, whereas Duplicator tries to demonstrate that the distance is at most ϵ. Like all the games we discussed before, the game is played in rounds. Each round consists of the following steps. Spoiler chooses a state $u_1 \in \{s,t\}$ and a fuzzy set of states U_1. Duplicator uses the other state $u_2 \in \{s,t\} \setminus \{u_2\}$ and chooses a fuzzy set of states U_2 such that the expectation of transitioning from u_1 to U_1 minus expectation of transitioning from u_2 to U_2 is at most ϵ. Such a U_2 always exists.

[1] Desharnais et al. consider equivalence relations indexed by $\epsilon \in [0,1]$. Above, we presented their approach for $\epsilon = 0$.

Spoiler chooses $i \in \{1, 2\}$ and a state s'. Duplicator chooses a state t' with $U_i(s') \leq U_{3-i}(t')$. Duplicator can always choose a U_2 in the second step so that it can pick a t' in the fourth step with $U_i(s') \leq U_{3-i}(t')$. The next round starts in $(s', t', U_{3-i}(t') - U_i(s'))$. As shown by König and Mika-Michalski [36, Theorem 35 and 38], the distance of states s and t is at most ϵ if and only if Duplicator can avoid ever reaching a state pair with different labels when the game is started in (s, t, ϵ), no matter how Spoiler plays.

König, Mika-Michalski, and Schröder [37] present generic algorithms for computing strategies of both Spoiler and Duplicator.

Komorida, Katsumata, Hu, Klin, and Hasuo [35, Table 2] generalize the game of Fijalkow, Klin, and Panangaden to a quantitative setting using fibrations and coalgebras. Instantiating their framework to our setting amounts to the following game. The game starts in (s, t, ϵ). Spoiler chooses a fuzzy subset of states U such that the expectation of transitioning from state s to U and the expectation of transitioning from state t to U differ by more than ϵ. If such a U does not exist, Spoiler loses. Duplicator chooses states s' and t' as well as $\epsilon' \in [0, 1)$ such that $U(s')$ and $U(t')$ differ by more than ϵ'. As shown in [35, Theorem 5.11], the distance of states s and t is at most ϵ if and only if Duplicator can avoid ever reaching a state pair with different labels or Spoiler loses when the game is started in (s, t, ϵ), no matter how Spoiler plays.

1.4 Our Game

State pairs that have distance zero, which we call 0-*pairs*, are probabilistic bisimilar. We can explain their behavioural equivalence, and hence their zero distance, by the games discussed in Section 1.2. State pairs that have different labels, which we call 1-*pairs* as their distance is one, are observably different. Their different labelling explains their distance. It remains to explain the distances of the other state pairs, that is, those state pairs that have the same label but are not probabilistic bisimilar, which we call ?-*pairs*. We explain the distances of the ?-pairs by means of a game.

The foundation of the game that we study in this paper is an alternative characterization of the distances given by Chen, Van Breugel, and Worrell [10, Theorem 8]. This characterization underlies the algorithm of Bacci, Bacci, Larsen, and Mardare [1] to compute the distances. Tang [51] showed that their algorithm is an instance of Howard's *policy iteration* [28]. Howard's generic algorithm works on *Markov decision processes*. A Markov decision process is a $1\frac{1}{2}$-player game: one ordinary player and randomness which accounts for the remaining half. In our setting, the ordinary player takes on the role of Duplicator and the randomness embodies the Spoiler.

Tang [51, Section 5.3] defined the specific Markov decision processes so that Howard's policy iteration algorithm computes the distances. As shown by Tang [51, Section 6.1], one needs to decide probabilistic bisimilarity before running Howard's policy iteration algorithm. Probabilistic bisimilarity can be efficiently decided (see, for example, the algorithm of Derisavi, Hermanns, and Sanders [13]). Hence, at the time we start the game we know which states pairs are 0-pairs and 1-pairs.

Like Stirling's bisimilarity game, in our game we match transitions. In our setting we match parts of transitions. For example, consider the transitions of states 0 and 1 in Figure 1. The transition from state 1 to state 2 can be matched by part (of probability $\frac{3}{16}$) of the transition from 0 to 2. Similarly, the transition from 0 to 3 can be matched by part (of probability $\frac{1}{4}$) of the transition from 1 to 3 and the self loops of 0 and 1 can be matched. The parts that remain, part of the transition from 0 and 2 and part of the transition from 1 to 3, both have probability $\frac{1}{16}$ and can be matched as well. This matching is depicted in Figure 2.

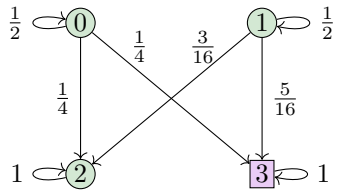


Figure 1 A labelled Markov chain.

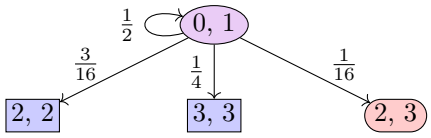


Figure 2 A coupling of the transitions of states 1 and 2 of the labelled Markov chain of Figure 1.

These matchings are also knows as *couplings*, a notion introduced by Doebling [18]. The set of couplings is known as the *transportation polytope*. There are infinitely many ways to match parts of transitions. However, as we will see, there is a finite set of couplings so that all other ones can be obtained as convex combinations of those in the finite set (the vertices of the transportation polytope). Although this set is finite, its size may be exponential in the number of states of the labelled Markov chain.

In our game the player tries match transitions, that is, chooses a coupling, in order to avoid 1-pairs. Therefore, also this game is *elegant* in that it characterizes a global property of behaviours (the distances) in terms of a game whose rules only depend on local considerations (the couplings). As we will see later, [10, Theorem 8] implies that the player can restrict to strategies, also known as *policies* in this setting, that consist of a choice of a coupling for each ?-pair. Such policies are known as deterministic and memoryless.

Given a policy, our game unfolds as follows. Let us start the game in ?-pair (s, t). The player uses their policy which provides a coupling of the transitions of s and t. Randomly, respecting the probabilities associated with the coupling, a matching of parts of transitions is chosen. This matching takes the game to a state pair (u, v). If this state pair is a 0-pair (depicted as a blue rectangle) or a 1-pair (depicted as a red rounded rectangle), the game stops. Otherwise, the game continues in ?-pair (u, v). The game reaches a 0- or 1-pair with probability one.

In the games for bisimilarity and probabilistic bisimilarity, Duplicator tried to avoid reaching 1-pairs. As our game involves randomness, the objective of the player is to minimize the probability of reaching a 1-pair. A policy is *optimal* if it minimizes the probability of reaching a 1-pair. [10, Theorem 8] shows that optimal policies exist and that for an optimal policy the probability of reaching 1-pairs from ?-pair (s, t) coincides with the distance of s and t. The coupling depicted in Figure 2 is part of an optimal policy for the labelled Markov chain of Figure 1. Note that the probability of reaching a 1-pair from ?-pair $(1, 2)$ is $\frac{1}{18}$, the distance of 1 and 2. Hence, an optimal policy can be seen as an explanation of the distances.

1.5 1-Maximal Policies

Both policies in Figure 3 are optimal and, therefore, can be seen as an explanation of the distance of states 0 and 1. In all the above described games, including ours, (one of) the player(s) tries to avoid reaching 1-pairs as only these pairs are observably different. The policy on the left matches the transitions of 0 and 1 so that none of the state pairs reachable after

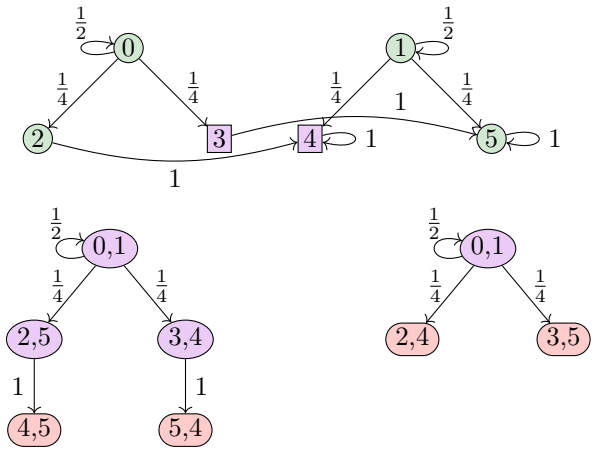


Figure 3 A labelled Markov chain and two optimal policies.

the first round of the game are observably different, whereas the one on the right matches the transitions so that two of the three reachable state pairs are 1-pairs. As a consequence, we argue that the one on the left better exemplifies the similarity of the behaviour of the states 0 and 1 and, as a result, is seen as a better explanation of the distance of 0 and 1. More generally, the larger the expected number of rounds it takes a policy to reach 1-pairs, the better it explains the distance. Therefore, policies that maximize this expected number of rounds, which we call *1-maximal*, are preferred as explanations.

By means of the algorithm of Tang [51, Section 6.2] we can compute an optimal policy. We present an iterative algorithm that, starting from an optimal policy, computes an optimal policy that is 1-maximal. We prove an exponential lower bound for the algorithm. That is, we construct a labelled Markov chain of size $O(n)$ for which the algorithm takes $\Omega(2^n)$ iterations.

1.6 0-Minimal Policies

Both policies in Figure 4 are optimal and 1-maximal. In both the ?-pair $(0, 1)$ reaches a 1-pair with probability $\frac{1}{2}$ and is expected to do so in three rounds. For 0-pairs, the transitions can be matched so that only 0-pairs are reached[2] and, hence, 1-pairs can be avoided all together. As a result, apart from avoiding 1-pairs, a secondary objective is to reach 0-pairs. The policy on the right does that better than the one of the left because its expected number of rounds to reach 0-pairs is smaller. We present an algorithm that, starting from a 1-maximal optimal policy, computes a 1-maximal optimal policy that is 0-minimal.

Let us jump into the details. Proofs and additional particulars can be found in [43, 53].

2 Probabilistic Bisimilarity Distances

We start with formalizing the model of interest, labelled Markov chains, and the probabilistic bisimilarity distances by recalling several results from the literature. Given a finite set X, a function $\mu : X \to [0, 1]$ is a *probability distribution* on X if $\sum_{x \in X} \mu(x) = 1$. We denote the set

[2] This immediately follows from the fact that probabilistic bisimilarity is a probabilistic bisimulation (see, for example, [6, Section 4]).

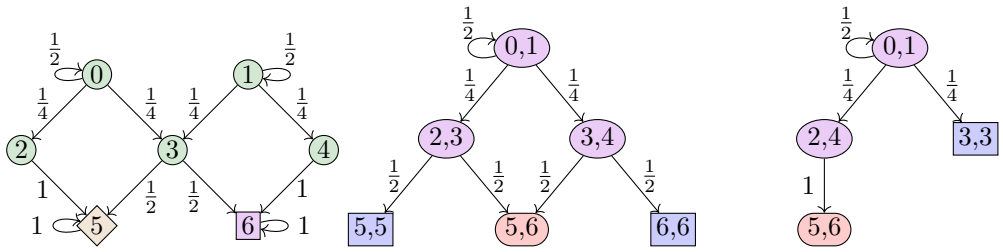


Figure 4 A labelled Markov chain and two 1-maximal optimal policies.

of probability distributions on X by $\mathcal{D}(X)$. For $\mu \in \mathcal{D}(X)$ and $A \subseteq X$, we often write $\mu(A)$ for $\sum_{x \in A} \mu(x)$. Similarly, for $\omega \in \mathcal{D}(X \times X)$, $x \in X$, and $A \subseteq X$, we usually write $\omega(x, A)$ for $\sum_{a \in A} \omega(x, a)$. For $\mu \in \mathcal{D}(X)$, we define the *support* of μ by $\mathrm{support}(\mu) = \{\, x \in X \mid \mu(x) > 0 \,\}$. To avoid clutter, for $\mu \in \mathcal{D}(X)$ and $f : X \to \mathbb{R}$ instead of $\sum_{x \in X} \mu(x)\, f(x)$ we write $\mu \cdot f$.

▶ **Definition 2.1.** *A* labelled Markov chain *is a tuple* $\langle S, L, \tau, \ell \rangle$ *consisting of*

- *a finite set* S *of* states,
- *a finite set* L *of* labels,
- *a* transition probability function $\tau : S \to \mathcal{D}(S)$, *and*
- *a* labelling function $\ell : S \to L$.

Several examples of labelled Markov chains have already been provided in the introduction. The states are represented by circles, squares, and diamonds, the labels by colours as well as shapes, and the transitions by arrows decorated with the probabilities. For the remainder of this paper, we fix a labelled Markov chain $\langle S, L, \tau, \ell \rangle$. We define probabilistic bisimilarity by means of the set $\Omega(\mu, \nu)$, which is known as the *transportation polytope* of the probability distributions μ and ν. The elements of $\Omega(\mu, \nu)$ are called *couplings*.

▶ **Definition 2.2.** *Let* $\mu, \nu \in \mathcal{D}(S)$. *The set* $\Omega(\mu, \nu)$ *is defined by*

$$\Omega(\mu, \nu) = \{\, \omega \in \mathcal{D}(S \times S) \mid \forall s \in S : \omega(s, S) = \mu(s) \wedge \omega(S, s) = \nu(s) \,\}.$$

For each $\mu, \nu \in \mathcal{D}(S)$, $\Omega(\mu, \nu)$ is a closed convex polytope. We denote the *vertices* of the transportation polytope by $V(\Omega(\mu, \nu))$.

▶ **Definition 2.3** ([30, Definition 4.3])**.** *A relation* $R \subseteq S \times S$ *is a* probabilistic bisimulation *if for all* $(s, t) \in R$, $\ell(s) = \ell(t)$ *and there exists* $\omega \in \Omega(\tau(s), \tau(t))$ *with* $\mathrm{support}(\omega) \subseteq R$. *States* s *and* t *are* probabilistic bisimilar, *denoted* $s \sim t$, *if* $(s, t) \in R$ *for some probabilistic bisimulation* R.

To define the probabilistic bisimilarity distances, it is convenient to partition the set of state pairs into 0-pairs, 1-pairs, and ?-pairs.

▶ **Definition 2.4.** *The sets* S_0^2, S_1^2, *and* $S_?^2$ *are defined by*

$$S_0^2 = \{\, (s, t) \in S \times S \mid s \sim t \,\}$$
$$S_1^2 = \{\, (s, t) \in S \times S \mid \ell(s) \neq \ell(t) \,\}$$
$$S_?^2 = (S \times S) \setminus (S_0^2 \cup S_1^2).$$

The set S_0^2 contains those state pairs that behave the same and, hence, have distance zero (see Theorem 2.6). We call these 0-pairs. The set S_1^2 contains those state pairs that have a different label and, therefore, are fundamentally differently and, hence, have distance one

(see Definition 2.5). We call these 1-pairs. The set $S^2_?$ contains the remaining state pairs. We call these ?-pairs. Note that some of these state pairs may have distance one, but cannot have distance zero (see Theorem 2.6). The probabilistic bisimilarity distances are defined in terms of the following function.

▶ **Definition 2.5.** *The function* $\Delta_1 : (S \times S \to [0,1]) \to (S \times S \to [0,1])$ *is defined by*

$$\Delta_1(d)(s,t) = \begin{cases} 0 & \text{if } (s,t) \in S^2_0 \\ 1 & \text{if } (s,t) \in S^2_1 \\ \inf\limits_{\omega \in \Omega(\tau(s),\tau(t))} \omega \cdot d & \text{otherwise.} \end{cases}$$

The functions in $S \times S \to [0,1]$ carry a natural partial order. For d, $e \in S \times S \to [0,1]$, we define $d \sqsubseteq e$ if for all s, $t \in S$, $d(s,t) \leq e(s,t)$. According to, for example, [16, Lemma 3.2], $\langle S \times S \to [0,1], \sqsubseteq \rangle$ is a complete lattice. Since the function Δ_1 is monotone (see, for example, [51, Proposition 2.1.13]), we can conclude from the Knaster-Tarski fixed point theorem [32, 52] that Δ_1 has a least fixed point, which we denote by δ_1. It maps each pair of states to a real number in the interval $[0,1]$: the *probabilistic bisimilarity distance* of the states. As we already mentioned, distance zero captures probabilistic bisimilarity.

▶ **Theorem 2.6** ([15, Theorem 4.10])**.** *For all* s, $t \in S$, $\delta_1(s,t) = 0$ *if and only if* $s \sim t$.

The 0-pairs are probabilistic bisimilar. This can be explained by means of the games mentioned in Section 1.2. The 1-pairs have different labels, which explains their difference in behaviour. Therefore, the distances of the ?-pairs remain to be explained. Hence, for the remainder of this paper, we assume that $S^2_? \neq \emptyset$ and, as a result, $S^2_1 \neq \emptyset$.

3 Policies

As we already mentioned, a policy consists of a coupling of the transitions of states s and t for each ?-pair (s,t).

▶ **Definition 3.1.** *The set* $\mathcal{P}$ *of* policies *is defined by*

$$\mathcal{P} = \{\, P \in S^2_? \to \mathcal{D}(S \times S) \mid \forall (s,t) \in S^2_? : P(s,t) \in \Omega(\tau(s), \tau(t)) \,\}.$$

Figure 3 and 4 provide examples of policies. The 0-pairs are blue rectangles, the 1-pairs are red rounded rectangles, and the ?-pairs are purple ellipses. For each ?-pair, its outgoing arrows labelled with probabilities represent a coupling. Given a policy, we can define the probability of reaching a 0- or 1-pair from any state pair as follows.

▶ **Definition 3.2.** *Let* $i \in \{0,1\}$ *and* $P \in \mathcal{P}$. *The function* $\Delta_{iP} : (S \times S \to [0,1]) \to (S \times S \to [0,1])$ *is defined by*

$$\Delta_{iP}(d)(s,t) = \begin{cases} 1 & \text{if } (s,t) \in S^2_i \\ 0 & \text{if } (s,t) \in S^2_{1-i} \\ P(s,t) \cdot d & \text{if } (s,t) \in S^2_?. \end{cases}$$

Since Δ_{iP} can be shown to be a monotone function from a complete lattice to itself, we can conclude from the Knaster-Tarski fixed point theorem that Δ_{iP} has a least fixed point. We denote the least fixed point of Δ_{iP} by δ_{iP}. For states s and t, $\delta_{iP}(s,t)$ is the probability of reaching an i-pair with respect to the policy P.

The probabilistic bisimilarity distances can be characterized in terms of a policy that minimizes the probability of reaching a 1-pair.

▶ **Theorem 3.3** ([10, Theorem 8]). $\delta_1 = \min_{P \in \mathcal{P}} \delta_{1P}$.

A policy that captures the distances, that is, the probability of reaching a 1-pair from a state pair (s,t) when using policy P equals the distance of s and t, we call optimal.

▶ **Definition 3.4.** *A policy $P \in \mathcal{P}$ is* optimal *if $\delta_{1P} = \delta_1$.*

For states s and t, the transportation polytope $\Omega(\tau(s), \tau(t))$ is generally infinite. As a result, our game, if formulated in terms of all couplings, is infinite in general. However, as we will see below, we can restrict our attention to the vertices of the transportation polytope $\Omega(\tau(s), \tau(t))$, making our game as well as the set of policies finite.

▶ **Definition 3.5.** *The set $\mathcal{V}$ of* vertex policies *is defined by*

$$\mathcal{V} = \{\, P \in \mathcal{P} \mid \forall (s,t) \in S_?^2 : P(s,t) \in V(\Omega(\tau(s), \tau(t))) \,\}.$$

Even if we restrict ourselves to vertex policies, we can still characterize the distances in terms of reachability probabilities.

▶ **Theorem 3.6** ([51, Theorem 6.1.7]). $\delta_1 = \min_{P \in \mathcal{V}} \delta_{1P}$.

According to Theorem 3.3 and 3.6, optimal and optimal vertex policies exist. We denote the set of optimal policies by $\mathcal{P}_{\text{opt}}$ and the set of optimal vertex policies by $\mathcal{V}_{\text{opt}}$. The policies depicted in Figure 3 and 4 are optimal vertex policies. The following alternative characterization of optimal policies turns out to be very useful in several of our proofs.

▶ **Proposition 3.7.** *For all $P \in \mathcal{P}$, P is optimal if and only if $\delta_1(s,t) = P(s,t) \cdot \delta_1$ for all $(s,t) \in S_?^2$.*

For every policy that is not a vertex policy, there exists a vertex policy the support graph of which is a strict subgraph. Hence, the transitions can be matched in such a way that fewer state pairs need to be considered, resulting in a simpler explanation.

▶ **Proposition 3.8.** *For all $P \in \mathcal{P}_{\text{opt}} \setminus \mathcal{V}_{\text{opt}}$ there exists $V \in \mathcal{V}_{\text{opt}}$ such that for all $(s,t) \in S_?^2$, $\text{support}(V(s,t)) \subseteq \text{support}(P(s,t))$ and $\text{support}(V(u,v)) \subset \text{support}(P(u,v))$ for some $(u,v) \in S_?^2$.*

Proof sketch. Let P be an optimal policy. For $(s,t) \in S_?^2$, we call a coupling $\omega \in \Omega(\tau(s), \tau(t))$ *optimal* for (s,t) if $\delta_1(s,t) = \omega \cdot \delta_1$. Since we can show that the set of optimal couplings for (s,t) is a closed convex polytope and the vertices of this polytope are the vertices of $\Omega(\tau(s), \tau(t))$ that are optimal, we can conclude that $P(s,t)$ is the convex combination of a set V_{st} of optimal couplings for (s,t) in $V(\Omega(\tau(s), \tau(t)))$. We can construct an optimal vertex policy V by mapping $V(s,t)$ to an optimal coupling in V_{st}. By construction, $\text{support}(V(s,t)) \subseteq \text{support}(P(s,t))$. To prove that $\text{support}(V(u,v)) \subset \text{support}(P(u,v))$ for some $(u,v) \in S_?^2$ we use the fact that a coupling is a vertex of the transportation polytope if and only if its support graph is acyclic. ◀

Tang [51, Section 6.2] presents an algorithm that computes an optimal vertex policy for a given labelled Markov chain. As we have already discussed in the introduction, although such a policy can be viewed as an explanation of the distances, we argued that 0-minimal 1-maximal optimal policies are desirable. Both 0-minimal and 1-maximal are defined in terms of expected number of rounds of the game, that is, the expected lengths of paths to 0-pairs and 1-pairs. We introduce these expected lengths next.

4 Expected Length

Let $i \in \{0,1\}$. Given an optimal policy P and a pair of states (s,t), we are interested in the expected length of paths from (s,t) to i-pairs when using P. We restrict to only those paths that reach an i-pair, that is, it is a conditional expectation. For example, consider the policy in the middle of Figure 4. To reach the 1-pair (5, 6) starting from the state pair (0, 1), there are 2 paths, each with a probability of $\frac{1}{8}$, of length 2, 2 paths, each of probability $\frac{1}{16}$, of length 3, etc. The probability of reaching a 1-pair from state pair (0, 1) is $\frac{1}{2}$. Hence, the conditional expectation is $\frac{\frac{1}{8}2+\frac{1}{8}2+\frac{1}{16}3+\frac{1}{16}3+\cdots}{\frac{1}{2}} = 3$. As we have already seen, $\delta_{1P}(s,t)$ captures the probability of all paths that reach an 1-pair from (s,t). However, when we restrict ourselves to optimal policies $\delta_{1P}(s,t) = \delta_1(s,t)$. This represents the denominator of the conditional expectation. Since P is optimal, the denominator is independent of P. Therefore, we omit it. Hence, in the following we only consider the numerator.

We define the function $\delta_0 : S \times S \to [0,1]$ by $\delta_0(s,t) = 1 - \delta_1(s,t)$. Since $\delta_{iP}(s,t)$ captures the probability of reaching an i-pair from (s,t), we can conclude that (s,t) can reach an i-pair if and only if $\delta_{iP}(s,t) \neq 0$. Because P is optimal, this is equivalent to $\delta_i(s,t) \neq 0$.

▶ **Definition 4.1.** *For $i \in \{0,1\}$, the set D_i is defined by*

$$D_i = \{\, (s,t) \in S \times S \mid \delta_i(s,t) \neq 0 \,\}.$$

Note that $S_i^2 \subseteq D_i \subseteq S_i^2 \cup S_?^2$. The expected lengths are defined in terms of the following function.

▶ **Definition 4.2.** *Let $i \in \{0,1\}$ and $P \in \mathcal{P}_{\text{opt}}$. The function $\Lambda_{iP} : (D_i \to [0,\infty)) \to (D_i \to [0,\infty))$ is defined by*

$$\Lambda_{iP}(l)(s,t) = \begin{cases} 0 & \text{if } (s,t) \in S_i^2 \\ P(s,t) \cdot (\delta_i + l) & \text{otherwise.} \end{cases}$$

Since the set S is finite and, therefore, the set D_i is finite, the set of functions $D_i \to [0,\infty)$ endowed with the sup metric[3] forms a nonempty complete metric space (see, for example, [49, Theorem 3.11]). Hence, Λ_{iP} is a function from a nonempty complete metric space to itself. To prove that Λ_{iP} has a unique fixed point, which we denote by λ_{iP}, we use the following generalization of Banach's fixed point theorem. This result dates back at least as far as the sixties. Part (a) and (b) are Banach's original fixed point theorem [4]. Part (c) can already be found in [34] and [12] contains part (d).

▶ **Theorem 4.3.** *Let (X,d) be a nonempty complete metric space. Let $f : X \to X$. If f is a power-contraction, that is, f^n is contractive for some $n \in \mathbb{N}$, then*

(a) f^n has a unique fixed point x and *(b) for all $y \in X, x = \lim_{m\in\mathbb{N}} f^{mn}(y)$.*

If f is also nonexpansive then

(c) x is the unique fixed point of f and *(d) for all $y \in X, x = \lim_{m\in\mathbb{N}} f^{m}(y)$.*

We denote P restricted to $S_?^2$ by $P_?$, that is, $P_? \in S_?^2 \to (S_?^2 \to [0,1])$.

[3] The distance of $k, l \in D_i \to [0,\infty)$ is defined as $\max_{(s,t)\in D_i} |k(s,t) - l(s,t)|$.

▶ **Proposition 4.4.** *For all $i \in \{0,1\}$, $P \in \mathcal{P}_{\mathrm{opt}}$, k, $l \in D_i \to [0,\infty)$, $(s,t) \in D_i$, and $n \in \mathbb{N}$,*
(a) *if $(s,t) \in S_i^2$ then $\Lambda_{iP}^{n+1}(k)(s,t) = \Lambda_{iP}^{n+1}(l)(s,t)$, and*
(b) *if $(s,t) \in D_i \setminus S_i^2$ then $|\Lambda_{iP}^{n+1}(k)(s,t) - \Lambda_{iP}^{n+1}(l)(s,t)| \leq P_?^n(s,t)(S_?^2)\,\|k-l\|$.*
From the above proposition we can conclude that Λ_{iP} is a power-contraction and nonexpansive.

▶ **Theorem 4.5.** *For all $i \in \{0,1\}$ and $P \in \mathcal{P}_{\mathrm{opt}}$, Λ_{iP} has unique fixed point.*

Proof sketch. Since Δ_{iP} is monotone and nonexpansive, we can conclude from [5, Theorem 1] that

$$\delta_{iP} = \lim_{n\in\mathbb{N}} \Delta_{iP}^n(\bot), \tag{1}$$

where $\bot : S \times S \to [0,1]$ is defined by $\bot(s,t) = 0$. Let $(s,t) \in D_i$. Hence, $\delta_i(s,t) \neq 0$. Since P is optimal, for $(s,t) \in D_i$ we have that $\delta_{iP}(s,t) \neq 0$. Therefore, from (1) we can conclude that exists $n_{st} \in \mathbb{N}$ such that for all $n \geq n_{st}$ we have that $\Delta_{iP}^{n+1}(\bot)(s,t) > 0$. As a consequence, we can prove that $P_?^n(s,t)(S_?^2) < 1$.

Let $n = (\max_{(s,t)\in D_i} n_{st}) + 1$ and $c = \max_{(s,t)\in D_i} P_?^n(s,t)(S_?^2)$. Note that $c < 1$. From Proposition 4.4 we can deduce that Λ_{iP}^n is c-contractive. By means of Theorem 4.3(c) we can complete the proof. ◀

The above result is a key ingredient of the proof of Proposition 5.3, which is in turn crucial in our proof of correctness and termination of Algorithm 1.

5 1-Maximal Policies

As discussed in the introduction and illustrated in Figure 3, an optimal policy that maximizes the expected length of paths to 1-pairs is desirable. This maximal expected length is defined as follows.

▶ **Definition 5.1.** *The function $\lambda_1 : D_1 \to [0,\infty]$ is defined by*

$$\lambda_1(s,t) = \sup_{P\in\mathcal{P}_{\mathrm{opt}}} \lambda_{1P}(s,t).$$

An optimal policy that realizes that maximal expected length is called 1-maximal.

▶ **Definition 5.2.** *A policy $P \in \mathcal{P}_{\mathrm{opt}}$ is* 1-maximal *if $\lambda_{1P} = \lambda_1$.*

We denote the set of optimal 1-maximal policies by $\mathcal{P}_{\mathrm{opt}}^{\max}$ and the set of optimal 1-maximal vertex policies by $\mathcal{V}_{\mathrm{opt}}^{\max}$. Similar to Proposition 3.7, we provide an alternative characterization of 1-maximal that we use in several of our proofs.

▶ **Proposition 5.3.** *For all $P \in \mathcal{P}_{\mathrm{opt}}$, P is 1-maximal if and only if $\lambda_1(s,t) = P(s,t)\cdot(\delta_1 + \lambda_1)$ for all $(s,t) \in D_1 \setminus S_1^2$.*

Below we present a policy iteration algorithm that computes a 1-maximal optimal vertex policy from an optimal vertex policy. In our algorithm we use the following function.

▶ **Definition 5.4.** *The function $\Lambda_1 : (D_1 \to [0,\infty)) \to (D_1 \to [0,\infty))$ is defined by*

$$\Lambda_1(l)(s,t) = \begin{cases} 0 & \text{if } (s,t) \in S_1^2 \\ \max\limits_{\omega\in V(\Omega(\tau(s),\tau(t)))\wedge\delta_1(s,t)=\omega\cdot\delta_1} \omega\cdot(\delta_1 + l) & \text{otherwise.} \end{cases}$$

The following theorem is crucial for proving that the vertex policy our algorithm computes is 1-maximal.

▶ **Theorem 5.5.** *λ_1 is the unique fixed point of Λ_1.*

Our algorithm starts from an optimal vertex policy P, which can be obtained by the policy iteration algorithm of [51, Section 6.2]. It computes λ_{1P} for that policy P (line 1), which can be done by means of standard algorithms (see, for example, [3, Section 10.1.1]). As long as there is a state pair in $D_1 \setminus S_1^2$ that is not locally 1-maximal with respect to the current policy (line 2), the policy at (s,t) is improved to a locally 1-maximal choice (line 3). This boils down to solving a linear programming problem. After this change to the policy P, we recompute λ_{1P} (line 4). The loop maintains the invariant that P is an optimal vertex policy as ω in line 3 is a vertex and satisfies $\delta_1(s,t) = \omega \cdot \delta_1$ (see Proposition 3.7). At termination, we have that λ_{1P} is a fixed point of Λ_1 and, by Theorem 5.5, equals λ_1 and, therefore, is 1-maximal.

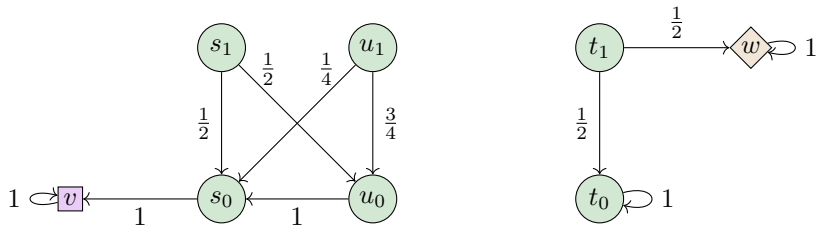
Algorithm 1 1-maximal optimal vertex policy.

Input: optimal vertex policy P
Output: 1-maximal optimal vertex policy

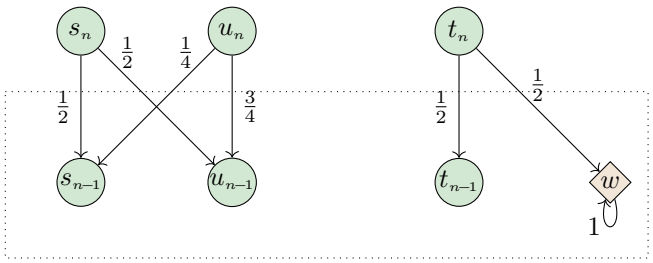
1: compute λ_{1P}
2: **while** $\exists (s,t) \in D_1 \setminus S_1^2 : \Lambda_1(\lambda_{1P})(s,t) > \lambda_{1P}(s,t)$ **do**
3: $\quad P(s,t) \leftarrow \underset{\omega \in V(\Omega(\tau(s),\tau(t))) \wedge \delta_1(s,t) = \omega \cdot \delta_1}{\arg\max} \omega \cdot (\delta_1 + \lambda_{1P})$
4: $\quad$ compute λ_{1P}
5: **end while**
6: **return** P

Next, we prove an exponential lower bound for the above algorithm.

▶ **Definition 5.6.** *The labelled Markov chain C_0 is defined as*



If $n > 0$ then the labelled Markov chain C_n is defined as



The dashed square represents the labelled Markov chain C_{n-1}.

Note that C_n has $3n+8$ states and $6n+11$ transitions and, hence, is of size $\mathcal{O}(n)$. Applying the above algorithm to the labelled Markov chain C_n results in an exponential number of iterations.

▶ **Theorem 5.7.** *For each $n \in \mathbb{N}$, the labelled Markov chain C_n of size $\mathcal{O}(n)$ is such that Algorithm 1 takes $\Omega(2^n)$ iterations.*

Proof sketch. We define a sequence of 2^n optimal vertex policies and show that there exists an execution of Algorithm 1 that cycles through all those policies. The policies only differ for the (s_i, t_i) pairs. The two transitions of s_i are matched with the two transitions of t_i in the two obvious ways. Which of those two matchings is chosen for each (s_i, t_i) pair is captured by means of the (reversed) Gray code. Hence, the exponential lower bound proof relies on the order in which the non-locally optimal (s_i, t_i) pairs are chosen (for which the current matching is replaced with the other matching) rather than the fact that the transportation polytopes may have exponentially many vertices (in our example that is not the case). ◀

6 0-Minimal Policies

As we discussed in the introduction and Figure 4 illustrates, a 1-maximal optimal policy that minimizes the expected length of paths to 0-pairs is a preferable explanation. As in the previous section, we first capture the minimal expected length.

▶ **Definition 6.1.** *The function $\lambda_0 : D_0 \to [0, \infty)$ is defined by*

$$\lambda_0(s,t) = \inf_{P \in \mathcal{P}^{\max}_{\text{opt}}} \lambda_{0P}(s,t).$$

A 1-maximal optimal policy that matches that minimal expected length is called 0-minimal.

▶ **Definition 6.2.** *A policy $P \in \mathcal{P}^{\max}_{\text{opt}}$ is* 0-minimal *if $\lambda_{0P} = \lambda_0$.*

Below we present a policy iteration algorithm that computes a 0-minimal 1-maximal optimal vertex policy from a 1-maximal optimal vertex policy. In our algorithm we use the following function.

▶ **Definition 6.3.** *The function $\Lambda_0 : (D_0 \to [0, \infty)) \to (D_0 \to [0, \infty))$ is defined by*

$$\Lambda_0(l)(s,t) = \begin{cases} 0 & \text{if } (s,t) \in S_0^2 \\ \min\limits_{\omega \in V(\Omega(\tau(s),\tau(t))) \wedge \delta_1(s,t) = \omega \cdot \delta_1 \wedge \lambda_1(s,t) = \omega \cdot (\delta_1 + \lambda_1)} \omega \cdot (\delta_0 + l) & \text{otherwise.} \end{cases}$$

The key ingredient of the correctness proof of our algorithm is the following result.

▶ **Theorem 6.4.** *λ_0 is the unique fixed point of Λ_0.*

Our algorithm that computes a 0-minimal 1-maximal vertex policy from a 1-maximal vertex policy (Algorithm 2) is very similar in structure to Algorithm 1. Instead of focussing on 1-pairs, we concentrate on 0-pairs. Furthermore, we maintain as a loop invariant that P is not only an optimal vertex policy but also that it is 1-maximal.

Algorithm 2 0-minimal 1-maximal optimal vertex policy.

Input: 1-maximal optimal vertex policy P
Output: 0-minimal 1-maximal optimal vertex policy
1: compute λ_{0P}
2: **while** $\exists (s,t) \in D_0 \setminus S_0^2 : \Lambda_0(\lambda_{0P})(s,t) < \lambda_{0P}(s,t)$ **do**
3: $\quad P(s,t) \leftarrow \underset{\omega \in V(\Omega(\tau(s),\tau(t))) \wedge \delta_1(s,t)=\omega\cdot\delta_1 \wedge \lambda_1(s,t)=\omega\cdot(\delta_1+\lambda_1)}{\arg\min} \omega \cdot (\delta_0 + \lambda_{0P})$
4: $\quad$ compute λ_{0P}
5: **end while**
6: **return** P

Algorithm 1 and 2 both optimize locally in line 3 to obtain a global optimum. This may remind the reader of the "elegance" quote of Fijalkow et al. in the introduction.

7 Symmetric Policies

Since probabilistic bisimilarity distances are symmetric, that is, $\delta_1(s,t) = \delta_1(t,s)$ for all states s and t, one may wonder whether $\delta_1(s,t)$ can be explained similarly to $\delta_1(t,s)$. We call a policy P symmetric if $P(s,t)$ and $P(t,s)$ are mirror images.

▶ **Definition 7.1.** *Let $P \in \mathcal{P}$. P is* symmetric *if for all $(s,t),(u,v) \in S_?^2$,*

$$P(s,t)(u,v) = P(t,s)(v,u).$$

We fix a total order $\prec$ on the set of states S. This allows us to turn a policy into a symmetric one as follows.

▶ **Definition 7.2.** *Let $P \in \mathcal{P}$. We define $P_\prec$ by*

$$P_\prec(s,t)(u,v) = \begin{cases} P(s,t)(u,v) & \text{if } s \prec t \\ P(t,s)(v,u) & \text{if } s \succ t. \end{cases}$$

This construction preserves all the properties of policies in which we are interested.

▶ **Theorem 7.3.** *For all $P \in \mathcal{P}$,*
(a) *$P_\prec$ is a symmetric policy,*
(b) *if P is a vertex policy then $P_\prec$ is a vertex policy,*
(c) *if P is optimal then $P_\prec$ is optimal,*
(d) *if P is 1-maximal then $P_\prec$ is 1-maximal, and*
(e) *if P is 0-minimal then $P_\prec$ is 0-minimal.*

As a result, once we have computed a 0-minimal 1-maximal optimal vertex policy P, we can turn it into a symmetric 0-minimal 1-maximal optimal vertex policy $P_\prec$. Generally, such a symmetric policy is simpler as $\delta_1(s,t)$ is explained similarly to $\delta_1(t,s)$. In the graphical representation of a symmetric policy, we only need to represent the state pairs (s,t) with $s \preceq t$, where an arrow from (s,t) to (u,v) with $u \succ v$ is depicted as a twisted arrow.

8 Experimental Results

We have implemented Algorithm 1 and 2 in Java[4]. To compute an optimal vertex policy, we used a Java implementation[5] of the algorithm presented in [51, Section 6.2]. In our experimental evaluation we used the two examples of [51, Section 9.3], namely an example of

[4] The code is available at github.com/antoNanahJi/Explainability
[5] The code is available at bitbucket.org/discoveri/probabilistic-bisimilarity-distances.

Table 1 Average (and standard deviation) in milliseconds of time to compute optimal, 1-maximal, and 0-minimal policies.

	optimal	1-maximal	0-minimal
Crowds	913 (28)	1002 (24)	1099 (27)
Leader	494 (14)	593 (19)	576 (21)
Miller-Rabin	32,149 (149)	479 (2)	1,069 (5)
Quicksort	218,232 (8,971)	38,528 (1,139)	46,416 (1,165)
Two dies	3,142 (1)	2,271 (1)	1,569 (9)

two dies due to Knuth and Yao [33] and randomized quick sort, as well as the Miller-Rabin primality test [41, 46], the crowds protocol by Reiter and Rubin in [48], and the leader election protocol by Itai and Rodeh [29].

The experiments were run on an Intel machine with an i7-8700T CPU and 15 GB of RAM. For each of the five examples, we first computed an optimal vertex policy, then a 1-maximal optimal vertex policy, and finally a 0-minimal 1-maximal optimal vertex policy. We timed 55 trails for each of the three stages of the computation and report the average (and standard deviation) in milliseconds in Table 1. Since a Java virtual machine needs to perform just-in-time compilation and optimization, we discarded the first eight trails. Garbage collection was triggered in between trials to minimize its impact on our measurements.

As can be seen, the algorithms tend to perform well in practice despite the exponential lower bound. In most cases, computing the 1-maximal and 0-minimal optimal policies is significantly faster than computing optimal policies. Recall that if policy P is optimal then λ_{1P} captures the distances. Hence, once we have computed the distances by means of policy iteration, we can construct a symmetric 0-minimal 1-maximal optimal vertex policy that explains the distances, often taking less time than computing the distances.

Our implementation also provides a graphical representation (DOT format) of the policy.

9 Conclusion

We have shown that explainability is a game in the context of probabilistic bisimilarity distances of labelled Markov chains. More precisely, symmetric 0-minimal 1-maximal optimal vertex policies for the $1\frac{1}{2}$-player games presented in this paper explain the distances.

Apart from the area of probabilistic model checking, distances similar to the ones studied in this paper also play a role in other fields including control theory [26], fault-tolerance [9], privacy [11], quantum computing [20], reinforcement learning [40], and systems-biology [38]. As a consequence, we anticipate that our results are widely applicable.

The main contributions of this paper are the following.

- We provide a unified overview of games related to probabilistic bisimilarity and probabilistic bisimilarity distances in the introduction of the paper.
- We argue that symmetric 0-minimal 1-maximal optimal vertex policies explain these distances.
- We develop algorithms to compute symmetric 0-minimal 1-maximal optimal vertex policies and prove them correct.
- We prove an exponential lower bound for the algorithm that computes a 1-maximal optimal vertex policy from an optimal vertex policy.
- We have implemented all algorithms in Java. The code is open-source.

Several questions remain open. For example, we conjecture that the algorithm that computes a 0-minimal 1-maximal optimal policy from a 1-maximal optimal policy also has an exponential lower bound. Determining upper bounds for both our algorithms is left for future work.

In [53], Vlasman introduced two properties of policies, label conflict and probabilistic bisimilar conflict, argued that policies free of such conflicts are desirable for explaining probabilistic bisimilarity distances, and presented algorithms to remove such conflicts from optimal policies. We have already shown that 1-maximal optimal policies are free of label conflicts (see [43]). It is an open problem whether a 0-minimal 1-maximal optimal policy is free of probabilistic bisimilar conflicts. We conjecture that "transitivity" of policies, which seems closely related to the fact that probabilistic bisimilarity distances satisfy the triangle inequality, may play a role in tackling that problem. Apart from symmetry and "transitivity," it could be worthwhile exploring other forms of structure of policies to further reduce the complexity of the explanation.

We conjecture that the policy iteration algorithm of [51, Section 6.2], as well as Algorithm 1 and 2 can be modified so that they maintain that the policy is symmetric as a loop invariant. This may allows us to restrict our attention to only half of the state pairs currently being considered in the algorithm.

As we mentioned in the introduction, our game can be viewed as a Markov decision process presented by Tang in [51, Section 5.3]. Several definitions and results can be expressed in the parlance of Markov decision processes. For example, the function Δ_1 presented in Definition 2.5 can be seen as a Bellman equation (see, for example, [45, Section 4.3]) – note that a slight generalization of Δ_1 was already defined in, for example, [7, Definition 15], which predates Tang's presentation of the Markov decision process by more than a decade. Similarly, for a policy P and a state pair (s,t), $P(s,t)$ can be seen a decision rule and Proposition 3.7 then characterizes a conserving decision rule (see, for example, [45, Section 6.2.4]).

Our results are related to the work of Busatto-Gaston et al. [8]. They consider bi-objective problems on Markov decision processes. In particular, they minimize the (conditional) expected number of steps to a target while guaranteeing the maximal probability of reaching it. In this paper, we consider three objectives (optimal, 1-maximal, and 0-minimal). They propose a simple two-step pruning algorithm. Our algorithms show similarities to their algorithm. For example, Definition 5.4 corresponds to pruning the Markov decision process by restricting to couplings ω that satisfy $\delta_1(s,t) = \omega \cdot \delta_1$. Since the size of the Markov decision process representing our game may be exponential in the number of states of the labelled Markov chain, we do not explicitly construct the Markov decision process, nor do we explicitly prune it.

The aim of our work is to provide a human-interpretable explanation of probabilistic bisimilarity distances. According to Theorem 3.3, for a policy P we have that δ_{1P} provides an upper bound for δ_1. As a result, a policy can also be viewed as a certificate (see, for example, [2] for work on certificates for Markov decision processes) for an upper bound of the probabilistic bisimilarity distances. In this case, we would aim for a small policy P so that δ_{1P} can be computed quickly. We consider this an interesting line of future work.

References

1 Giorgio Bacci, Giovanni Bacci, Kim Larsen, and Radu Mardare. On-the-fly exact computation of bisimilarity distances. In Nir Piterman and Scott Smolka, editors, *Proceedings of the 19th International Conference on Tools and Algorithms for the Construction and Analysis of*

Systems, volume 7795 of *Lecture Notes in Computer Science*, pages 1–15, Rome, Italy, March 2013. Springer-Verlag. doi:10.1007/978-3-642-36742-7_1.

2 Christel Baier, Calvin Chau, and Sascha Klüppelholz. Certificates and witnesses for multi-objective queries in Markov decision processes. In Jane Hillston, Sadegh Soudjani, and Masaki Waga, editors, *Proceedings of the 1st International Joint Conference on Quantitative Evaluation of Systems and Formal Modeling and Analysis of Timed Systems*, volume 14996 of *Lecture Notes in Computer Science*, pages 1–18, Calgary, Canada, September 2024. Springer-Verlag. doi:10.1007/978-3-031-68416-6_1.

3 Christel Baier and Joost-Pieter Katoen. *Principles of Model Checking*. The MIT Press, Cambridge, MA, USA, 2008.

4 Stefan Banach. Sur les opérations dans les ensembles abstraits et leurs applications aux equations intégrales. *Fundamenta Mathematicae*, 3:133–181, 1922. doi:10.4064/fm-3-1-133-181.

5 Franck van Breugel. On behavioural pseudometrics and closure ordinals. *Information Processing Letters*, 112(19):715–718, October 2012. doi:10.1016/J.IPL.2012.06.019.

6 Franck van Breugel. Probabilistic bisimilarity distances. *SIGLOG News*, 4(4):33–51, October 2017. doi:10.1145/3157831.3157837.

7 Franck van Breugel, Claudio Hermida, Michael Makkai, and James Worrell. An accessible approach to behavioural pseudometrics. In Luís Caires, Giuseppe Italiano, Luís Monteiro, Catuscia Palamidessi, and Moti Yung, editors, *Proceedings of the 32nd International Colloquium on Automata, Languages and Programming*, volume 3580 of *Lecture Notes in Computer Science*, pages 1018–1030, Lisbon, Portugal, July 2005. Springer-Verlag. doi:10.1007/11523468_82.

8 Damien Busatto-Gaston, Debraj Chakraborty, Anirban Majumdar, Sayan Mukherjee, Guillermo Pérez, and Jean-François Raskin. Bi-objective lexicographic optimization in Markov decision processes with related objectives. In Étienne André and Jun Sun, editors, *Proceedings of the 21st International Symposium on Automated Technology for Verification and Analysis*, volume 14215 of *Lecture Notes in Computer Science*, pages 203–223, Signapore, October 2023. Springer-Verlag. doi:10.1007/978-3-031-45329-8_10.

9 Pablo Castro, Pedro D'Argenio, Ramiro Demasi, and Luciano Putruele. Measuring masking fault-tolerance. In *Proceedings of the 25th International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, volume 11428 of *Lecture Notes in Computer Science*, pages 375–392, Prague, Czech Republic, April 2019. Springer-Verlag. doi:10.1007/978-3-030-17465-1_21.

10 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. doi:10.1007/978-3-642-28729-9_29.

11 Dmitry Chistikov, Andrzej Murawski, and David Purser. Bisimilarity distances for approximate differential privacy. In Shuvendu Lahiri and Chao Wang, editors, *Proceedings of the 16th International Symposium on Automated Technology for Verification and Analysis*, volume 11138 of *Lecture Notes in Computer Science*, pages 194–210, Los Angeles, CA, USA, October 2018. Springer-Verlag. doi:10.1007/978-3-030-01090-4_12.

12 Eric Denardo. Contraction mappings in the theory underlying dynamic programming. *SIAM Review*, 9(2):165–177, April 1967. doi:10.1137/1009030.

13 Salem Derisavi, Holger Hermanns, and William Sanders. Optimal state-space lumping in Markov chains. *Information Processing Letters*, 87(6):309–315, September 2003. doi:10.1016/S0020-0190(03)00343-0.

14 Josée Desharnais, Vineet Gupta, Radha Jagadeesan, and Prakash Panangaden. Metrics for labeled Markov systems. In Jos Baeten and Sjouke Mauw, editors, *Proceedings of 10th International Conference on Concurrency Theory*, volume 1664 of *Lecture Notes in Computer Science*, pages 258–273, Eindhoven, the Netherlands, August 1999. Springer-Verlag. doi:10.1007/3-540-48320-9_19.

15 Josée Desharnais, Vineet Gupta, Radha Jagadeesan, and Prakash Panangaden. Metrics for labelled Markov processes. *Theoretical Computer Science*, 318(3):323–354, June 2004. doi:10.1016/J.TCS.2003.09.013.

16 Josée Desharnais, Radha Jagadeesan, Vineet Gupta, and Prakash Panangaden. The metric analogue of weak bisimulation for probabilistic processes. In *Proceedings of the 17th Annual IEEE Symposium on Logic in Computer Science*, pages 413–422, Copenhagen, Denmark, July 2002. IEEE. doi:10.1109/LICS.2002.1029849.

17 Josée Desharnais, François Laviolette, and Mathieu Tracol. Approximate analysis of probabilistic processes: logic, simulation and games. In *Proceedings of the 5th International Conference on the Quantitative Evaluation of Systems*, pages 264–273, Saint-Malo, France, September 2008. IEEE. doi:10.1109/QEST.2008.42.

18 Wolfgang Doeblin. Exposé de la théorie des chaînes simples constantes de Markov à un nombre fini d'états. *Revue mathématique de l'union interbalkanique*, 2:77–105, 1938.

19 Andrzej Ehrenfeucht. An application of games to the completeness problem for formalized theories. *Fundamenta Mathematicae*, 49:129–141, 1961.

20 Yuan Feng, Runyao Duan, and Mingsheng Ying. Bisimulation for quantum processes. In *Proceedings of the 38th Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, pages 523–534, Austin, TX, USA, January 2011. ACM. doi:10.1145/1926385.1926446.

21 Nathanaël Fijalkow, Bartek Klin, and Prakash Panangaden. Expressiveness of probabilistic modal logics, revisited. In Ioannis Chatzigiannakis, Piotr Indyk, Fabian Kuhn, and Anca Muscholl, editors, *Proceedings of 44th International Colloquium on Automata, Languages and Programming*, volume 80 of *Leibniz International Proceedings in Informatics*, pages 105:1–105:12, Warsaw, Poland, July 2017. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPICS.ICALP.2017.105.

22 Chase Ford, Harsh Beohar, Barbara König, Stefan Milius, and Lutz Schröder. Graded monads and behavioural equivalence games. In Christel Baier and Dana Fisman, editors, *Proceedings of the 37th Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 61:1–61:13, Haifa, Israel, August 2022. ACM. doi:10.1145/3531130.3533374.

23 Vojtech Forejt, Petr Jancar, Stefan Kiefer, and James Worrell. Game characterization of probabilistic bisimilarity and applications to pushown automata. *Logical Methods in Computer Science*, 14(4:13), November 2018. doi:10.23638/LMCS-14(4:13)2018.

24 Roland Fraïssé. Sur une nouvelle classification des systèmes de relations. *Comptes Rendus*, 230:1022–1024, 1950.

25 Alessandro Giacalone, Chi-Chang Jou, and Scott Smolka. Algebraic reasoning for probabilistic concurrent systems. In *Proceedings of the IFIP WG 2.2/2.3 Working Conference on Programming Concepts and Methods*, pages 443–458, Sea of Gallilee, April 1990. North-Holland.

26 Antoine Girard and George Pappas. Approximate bisimulation: A bridge between computer science and control theory. *European Journal of Control*, 17(5):568–578, September 2011. doi:10.3166/EJC.17.568-578.

27 Matthew Hennessy and Robin Milner. On observing nondeterminism and concurrency. In Jaco de Bakker and Jan van Leeuwen, editors, *Proceedings of 7th International Colloquium on Automata, Languages and Programming*, volume 85 of *Lecture Notes in Computer Science*, pages 299–309, Noordwijkerhout, the Netherlands, July 1980. Springer-Verlag. doi:10.1007/3-540-10003-2_79.

28 Ronald Howard. *Dynamic programming and Markov processes*. The MIT Press, Cambridge, MA, USA, 1960.

29 Alon Itai and Michael Rodeh. Symmetry breaking in distributed networks. In *Proceedings of the 2nd Annual Symposium on Foundations of Computer Science*, pages 150–158, Nashville, TN, USA, October 1981. IEEE. doi:10.1109/SFCS.1981.41.

30 Bengt Jonsson and Kim Larsen. Specification and refinement of probabilistic processes. In *Proceedings of the 6th Annual Symposium on Logic in Computer Science*, pages 266–277, Amsterdam, the Netherlands, July 1991. IEEE. doi:10.1109/LICS.1991.151651.

31 Joost-Pieter Katoen, Tim Kemna, Ivan Zapreev, and David Jansen. Bisimulation minimisation mostly speeds up probabilistic model checking. In Orna Grumberg and Michael Huth, editors, *Proceedings of the 13th International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, volume 4424 of *Lecture Notes in Computer Science*, pages 87–101, Braga, Portugal, March/April 2007. Springer-Verlag. doi:10.1007/978-3-540-71209-1_9.

32 Bronisław Knaster. Un théorème sur les fonctions d'ensembles. *Annales Polonici Mathematici*, 6:133–134, 1928.

33 Donald Knuth and Andrew Yao. The complexity of nonuniform random number generation. In Joseph Traub, editor, *Proceedings of a Symposium on New Directions and Recent Results in Algorithms and Complexity*, pages 375–428, Pittsburgh, PA, USA, April 1976. Academic Press.

34 Ignace Kolodner. Fixed points. *The American Mathematical Monthly*, 71(9):906, October 1964.

35 Yuichi Komorida, Shin-ya Katsumata, Nick Hu, Bartek Klin, and Ichiro Hasuo. Codensity games for bisimilarity. In *Proceedings of the 34th Annual ACM/IEEE Symposium on Logic in Computer Science*, pages 1–13, Vancouver, Canada, June 2019. IEEE. doi:10.1109/LICS.2019.8785691.

36 Barbara König and Christina Mika-Michalski. (Metric) bisimulation games and real-valued modal logics for coalgebras. In Sven Schewe and Lijun Zhang, editors, *Proceedings of the 29th International Conference on Concurrency Theory*, volume 118 of *Leibniz International Proceedings in Informatics*, pages 37:1–37:17, Beijing, China, September 2018. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. doi:10.4230/LIPICS.CONCUR.2018.37.

37 Barbara König, Christina Mika-Michalski, and Lutz Schröder. Explaining non-bisimilarity in a coalgebraic approach: Games and distinguishing formulas. In Daniela Petrisan and Jurriaan Rot, editors, *Proceedings of the 5th IFIP WG 1.3 International Workshop on Coalgebraic Methods in Computer Science*, volume 12094 of *Lecture Notes in Computer Science*, pages 133–154, Dublin, Ireland, April 2020. Springer-Verlag. doi:10.1007/978-3-030-57201-3_8.

38 Ilya Korsunsky, Kathleen McGovern, Tom LaGatta, Loes Olde Loohuis, Terri Grosso-Applewhite, Nancy Griffeth, and Bud Mishra. Systems biology of cancer: a challenging expedition for clinical and quantitative biologists. *Frontiers in Bioengineering and Biotechnology*, 2, August 2014. doi:10.3389/fbioe.2014.00027.

39 Kim Larsen and Arne Skou. Bisimulation through probabilistic testing. In *Proceedings of the 16th Annual ACM Symposium on Principles of Programming Languages*, pages 344–352, Austin, TX, USA, January 1989. ACM. doi:10.1145/75277.75307.

40 Qiyuan Liu, Qi Zhou, Rui Yang, and Jie Wang. Robust representation learning by clustering with bisimulation metrics for visual reinforcement learning with distractions. In Brian Williams, Yiling Chen, and Jennifer Neville, editors, *Proceedings of the AAAI Conference on Artificial Intelligence*, Washington, CD, USA, February 2023. doi:10.1609/aaai.v37i7.26063.

41 Gary Miller. Riemann's hypothesis and tests for primality. In William Rounds, Nancy Martin, Jack Carlyle, and Michael Harrison, editors, *Proceedings of the 7th Annual ACM Symposium on Theory of Computing*, pages 234–239, Albuquerque, NM, USA, May 1975. ACM. doi:10.1145/800116.803773.

42 Robin Milner. *A Calculus of Communicating Systems*, volume 92 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin, Germany, 1980. doi:10.1007/3-540-10235-3.

43 Anto Nanah Ji. Explainabilty is a game for probabilistic bisimilarity distances. Master's thesis, York University, Toronto, Canada, May 2025.

44 David Park. Concurrency and automata on infinite sequences. In Peter Deussen, editor, *Proceedings of 5th GI-Conference on Theoretical Computer Science*, volume 104 of *Lecture Notes in Computer Science*, pages 167–183, Karlsruhe, Germany, March 1981. Springer-Verlag. doi:10.1007/BFB0017309.

45 Martin Puterman. *Markov Decision Processes*. Wiley, Hoboken, NJ, USA, 1994.

46 Michael Rabin. Probabilistic algorithm for testing primality. *Journal of Number Theory*, 12(1):128–138, February 1980. doi:10.1016/0022-314X(80)90084-0.

47 Amgad Rady and Franck van Breugel. Explainability of probabilistic bisimilarity distances for labelled Markov chains. In Orna Kupferman and Pawel Sobocinski, editors, *Proceedings of the 26th International Conference on Foundations of Software Science and Computational Structures*, volume 13992 of *Lecture Notes in Computer Science*, pages 285–307, Paris, France, April 2023. Springer-Verlag. doi:10.1007/978-3-031-30829-1_14.

48 Michael Reiter and Aviel Rubin. Crowds: anonymity for web transactions. *ACM Transactions on Information and System Security*, 1(1):66–92, November 1998. doi:10.1145/290163.290168.

49 Walter Rudin. *Principles of Mathematical Analysis*. McGraw Hill, New York, NY, USA, 1976.

50 Colin Stirling. Modal and temporal logics for processes. In Faron Moller and Graham Birtwistle:, editors, *Proceedings of 8th Banff Higher Order Workshop*, volume 1043 of *Lecture Notes in Computer Science*, pages 149–237, Banff, Canada, August/September 1995. Springer-Verlag. doi:10.1007/3-540-60915-6_5.

51 Qiyi Tang. *Computing probabilistic bisimilarity distances*. PhD thesis, York University, Toronto, Canada, August 2018.

52 Alfred Tarski. A lattice-theoretic fixed point theorem and its applications. *Pacific Journal of Mathematics*, 5(2):285–309, June 1955. doi:10.2140/pjm.1955.5.285.

53 Emily Vlasman. On logical and game characterizations of probabilistic bisimilarity distances for labelled Markov chains. Master's thesis, University of Oxford, Oxford, UK, September 2023.