Logical, Metric, and Algorithmic Characterisations of Probabilistic Bisimulation

Yuxin Deng[1] **Wenjie Du**[2]

March 25, 2011
CMU-CS-11-110

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

[1]Author affiliations: Carnegie Mellon University, Shanghai Jiao Tong University, and Chinese Academy of Sciences
[2]Author affiliation: Shanghai Normal University

This work was partially supported by the Qatar National Research Fund under grant NPRP 09-1107-1-168, the Natural Science Foundation of China under Grant No. 61033002 and the Opening Fund of Top Key Discipline of Computer Software and Theory in Zhejiang Provincial Colleges at Zhejiang Normal University. The statements made herein are solely the responsibility of the authors.

Keywords: Probabilistic concurrency theory, bisimulation, logic, metric, algorithm

Abstract

Many behavioural equivalences or preorders for probabilistic processes involve a lifting operation that turns a relation on states into a relation on distributions of states. We show that several existing proposals for lifting relations can be reconciled to be different presentations of essentially the same lifting operation. More interestingly, this lifting operation nicely corresponds to the Kantorovich metric, a fundamental concept used in mathematics to lift a metric on states to a metric on distributions of states, besides the fact the lifting operation is related to the maximum flow problem in optimisation theory.

The lifting operation yields a neat notion of probabilistic bisimulation, for which we provide logical, metric, and algorithmic characterisations. Specifically, we extend the Hennessy-Milner logic and the modal mu-calculus with a new modality, resulting in an adequate and an expressive logic for probabilistic bisimilarity, respectively. The correspondence of the lifting operation and the Kantorovich metric leads to a natural characterisation of bisimulations as pseudometrics which are post-fixed points of a monotone function. We also present an "on the fly" algorithm to check if two states in a finitary system are related by probabilistic bisimilarity, exploiting the close relationship between the lifting operation and the maximum flow problem.

1 Introduction

In the last three decades a wealth of behavioural equivalences have been proposed in concurrency theory. Among them, *bisimilarity* [43, 48] is probably the most studied one as it admits a suitable semantics, an elegant co-inductive proof technique, as well as efficient decision algorithms.

In recent years, probabilistic constructs have been proven useful for giving quantitative specifications of system behaviour. The first papers on probabilistic concurrency theory [25, 5, 38] proceed by *replacing* nondeterministic with probabilistic constructs. The reconciliation of nondeterministic and probabilistic constructs starts with [27] and has received a lot of attention in the literature [67, 54, 40, 53, 29, 41, 3, 32, 44, 6, 57, 42, 14, 15, 13, 12].

We shall also work in a framework that features the co-existence of probability and nondeterminism. More specifically, we deal with *probabilistic labelled transition systems (pLTSs)* [14] which are an extension of the usual labelled transition systems (LTSs) so that a step of transition is in the form $s \xrightarrow{a} \Delta$, meaning that state s can perform action a and evolve into a distribution Δ over some successor states. In this setting state s is related to state t by a relation $\mathcal{R}$, say probabilistic simulation, written $s\ \mathcal{R}\ t$, if for each transition $s \xrightarrow{a} \Delta$ from s there exists a transition $t \xrightarrow{a} \Theta$ from t such that Θ can somehow simulate the behaviour of Δ according to $\mathcal{R}$. To formalise the mimicking of Δ by Θ, we have to *lift* $\mathcal{R}$ to be a relation $\mathcal{R}^\dagger$ between distributions over states and require $\Delta\ \mathcal{R}^\dagger\ \Theta$.

Various approaches of lifting relations have appeared in the literature; see e.g. [37, 54, 14, 8, 12]. We will show that although those approaches appear different, they can be reconciled. Essentially, there is only one lifting operation, which has been presented in different forms. Moreover, we argue that the lifting operation is interesting itself. This is justified by its intrinsic connection with some fundamental concepts in mathematics, notably *the Kantorovich metric* [34]. For example, it turns out that our lifting of binary relations from states to distributions nicely corresponds to the lifting of metrics from states to distributions by using the Kantorovich metric. In addition, the lifting operation is closely related to *the maximum flow problem* in optimisation theory, as observed by Baier et al. [2].

A good scientific concept is often elegant, even seen from many different perspectives. Bisimulation is one of such concepts in the traditional concurrency theory, as it can be characterised in a great many ways such as fixed point theory, modal logics, game theory, coalgebras etc. We believe that probabilistic bisimulation is also one of such concepts in probabilistic concurrency theory. As an evidence, we will provide in this paper three characterisations, from the perspectives of modal logics, metrics, and decision algorithms.

1. Our logical characterisation of probabilistic bisimulation consists of two aspects: *adequacy* and *expressivity* [50]. A logic $\mathcal{L}$ is adequate when two states are bisimilar if and only if they satisfy exactly the same set of formulae in $\mathcal{L}$. The logic is expressive when each state s has a characteristic formula φ_s in $\mathcal{L}$ such that t is bisimilar to s if and only if t satisfies φ_s. We will introduce a probabilistic choice modality to capture the behaviour of distributions. Intuitively, distribution Δ satisfies the formula $\bigoplus_{i\in I} p_i \cdot \varphi_i$ if there is a decomposition of Δ into a convex combination some distributions, $\Delta = \sum_{i\in I} p_i \cdot \Delta_i$, and each Δ_i confirms to the property specified by φ_i. When the new modality is added to the Hennessy-Milner logic [28] we obtain an adequate logic for probabilistic bisimilarity; when it is added to the modal mu-calculus [36] we obtain an expressive logic.

2. By metric characterisation of probabilistic bisimulation, we mean to give a pseudometric

such that two states are bisimilar if and only if their distance is 0 when measured by the pseudometric. More specifically, we show that bisimulations correspond to pseudometrics which are post-fixed points of a monotone function, and in particular bisimilarity corresponds to a pseudometric which is the greatest fixed point of the monotone function.

3. As to the algorithmic characterisation, we propose an "on the fly" algorithm that checks if two states are related by probabilistic bisimilarity. The schema of the algorithm is to approximate probabilistic bisimilarity by iteratively accumulating information about state pairs (s, t) where s and t are not bisimilar. In each iteration we dynamically constructs a relation $\mathcal{R}$ as an approximant. Then we verify if every transition from one state can be matched up by a transition from the other state, and their resulting distributions are related by the lifted relation $\mathcal{R}^\dagger$, which involves solving the maximum flow problem of an appropriately constructed network, by taking advantage of the close relation between our lifting operation and the above mentioned maximum flow problem.

Related work Probabilistic bisimulation was first introduced by Larsen and Skou [37]. Later on, it was investigated in a great many probabilistic models. An adequate logic for probabilistic bisimulation in a setting similar to our pLTSs has been studied in [33, 49]. It is also based on an probabilistic extension of the Hennessy-Milner logic. The main difference from our logic in Section 5.1 is the introduction of the operator $[\cdot]_p$. Intuitively, a distribution Δ satisfies the formula $[\varphi]_p$ when the set of states satisfying φ is measured by Δ with probability at least p. So the formula $[\varphi]_p$ can be expressed by our logic in terms of the probabilistic choice $\bigoplus_{i\in I} p_i{\cdot}\varphi_i$ by setting $I = \{1, 2\}$, $p_1 = p$, $p_2 = 1 - p$, $\varphi_1 = \varphi$, and $\varphi_2 = true$. When restricted to deterministic pLTSs (i.e., for each state and for each action, there exists at most one outgoing transition from the state), probabilistic bisimulations can be characterised by simpler forms of logics, as observed in [37, 16, 49].

An expressive logic for nonprobabilistic bisimulation has been proposed in [55]. In this paper we partially extend the results of [55] to a probabilistic setting that admits both probabilistic and nondeterministic choice. We present a probabilistic extension of the modal mu-calculus [36], where a formula is interpreted as the set of states satisfying it. This is in contrast to the probabilistic semantics of the mu-calculus as studied in [29, 41, 42] where formulae denote lower bounds of probabilistic evidence of properties, and the semantics of the generalised probabilistic logic of [6] where a mu-calculus formula is interpreted as a set of deterministic trees that satisfy it.

The Kantorovich metric has been used by van Breugel *et al.* for defining behavioural pseudometrics on fully probabilistic systems [61, 64, 60] and reactive probabilistic systems [62, 63, 58, 59]; and by Desharnais *et al.* for labelled Markov chains [17, 19] and labelled concurrent Markov chains [18]; and later on by Ferns *et al.* for Markov decision processes [23, 24]; and by Deng *et al.* for action-labelled quantitative transition systems [7]. One exception is [20], which proposes a pseudometric for labelled Markov chains without using the Kantorovich metric. Instead, it is based on a notition of ϵ-bisimulation, which relaxes the definition of probabilistic bisimulation by allowing small perturbation of probabilities. In this paper we are mainly interested in the correspondence of our lifting operation to the Kantorovich metric. The metric characterisation of probabilistic bisimulation in Section 6 is merely a direct consequence of this correspondence.

Decision algorithms for probabilistic bisimilarity and similarity have been considered by Baier et al. in [2] and Zhang et al. in [68]. Their algorithms are global in the sense that a whole state space has to be fully generated in advance. In contrast, "on the fly" algorithms are local in the sense that the state space is dynamically generated which is often more efficient to determine that one state fails to be related to another. Our algorithm in Section 7 is inspired by [2] because we also reduce the problem of checking if two distributions are related by a lifted relation to the maximum flow problem of a suitable network. We generalise the local algorithm of checking nonprobabilistic bisimilarity [22, 39] to the probabilistic setting.

This paper provides a relatively comprehensive account of probabilistic bisimulation. Some of the results or their variants were mentioned previously in [7, 9, 10, 11]. Here they are presented in a uniform way and equipped with detailed proofs.

Outline of the paper The paper proceeds by recalling a way of lifting binary relations from states to distributions, and showing its coincidence with a few other ways in Section 2. The lifting operation is justified in Section 3 in terms of its correspondence to the Kantorovich metric and the maximum flow problem. In Section 4 we define probabilistic bisimulation and show its infinite approximation. In Section 5 we introduce a probabilistic choice modality, then extend the Hennessy-Milner logic and the modal mu-calculus so to obtain two logics that are adequate and expressive, respectively. In Section 6 we characterise probabilistic bisimulations as pseudometrics. In Section 7 we exploit the correspondence of our lifting operation to the maximum flow problem, and present a polynomial time decision algorithm. Finally, Section 8 concludes the paper.

2 Lifting relations

In the probabilistic setting, formal systems are usually modelled as distributions over states. To compare two systems involves the comparison of two distributions. So we need a way of lifting relations on states to relations on distributions. This is used, for example, to define probabilistic bisimulation as we shall see in Section 4. A few approaches of lifting relations have appeared in the literature. We will take the one from [12], and show its coincidence with two other approaches.

We first fix some notation. A (discrete) probability distribution over a set S is a mapping $\Delta : S \to [0,1]$ with $\sum_{s\in S} \Delta(s) = 1$. The *support* of Δ is given by $\lceil \Delta \rceil := \{\, s \in S \mid \Delta(s) > 0 \,\}$. In this paper we only consider finite state systems, so it suffices to use distributions with finite support; let $\mathcal{D}(S)$, ranged over by Δ, Θ, denote the collection of all such distributions over S. We use $\overline{s}$ to denote the point distribution, satisfying $\overline{s}(t) = 1$ if $t = s$, and 0 otherwise. If $p_i \geq 0$ and Δ_i is a distribution for each i in some finite index set I, then $\sum_{i\in I} p_i \cdot \Delta_i$ is given by

$$(\sum_{i\in I} p_i \cdot \Delta_i)(s) \quad = \quad \sum_{i\in I} p_i \cdot \Delta_i(s)$$

If $\sum_{i\in I} p_i = 1$ then this is easily seen to be a distribution in $\mathcal{D}(S)$. Finally, the *product* of two probability distributions Δ, Θ over S, T is the distribution $\Delta \times \Theta$ over $S \times T$ defined by $(\Delta \times \Theta)(s,t) := \Delta(s) \cdot \Theta(t)$.

Definition 2.1 Given two sets S and T and a relation $\mathcal{R} \subseteq S \times T$. Then $\mathcal{R}^\dagger \subseteq \mathcal{D}(S) \times \mathcal{D}(T)$ is the smallest relation that satisfies:

1. $s \mathrel{\mathcal{R}} t$ implies $\overline{s} \mathrel{\mathcal{R}^\dagger} \overline{t}$
2. $\Delta_i \mathrel{\mathcal{R}^\dagger} \Theta_i$ implies $(\sum_{i\in I} p_i \cdot \Delta_i) \mathrel{\mathcal{R}^\dagger} (\sum_{i\in I} p_i \cdot \Theta_i)$, where I is a finite index set and $\sum_{i\in I} p_i = 1$.

The lifting construction satisfies the following useful property whose proof is straightforward thus omitted.

Proposition 2.2 Suppose $\mathcal{R} \subseteq S \times S$ and $\sum_{i\in I} p_i = 1$. If $(\sum_{i\in I} p_i \cdot \Delta_i)\ \mathcal{R}^\dagger\ \Theta$ then $\Theta = \sum_{i\in I} p_i \cdot \Theta_i$ for some set of distributions Θ_i such that $\Delta_i\ \mathcal{R}^\dagger\ \Theta_i$. □

We now look at alternative presentations of Definition 2.1. The proposition below is immediate.

Proposition 2.3 Let Δ and Θ be distributions over S and T, respectively, and $\mathcal{R} \subseteq S \times T$. Then $\Delta\ \mathcal{R}^\dagger\ \Theta$ if and only if Δ, Θ can be decomposed as follows:

1. $\Delta = \sum_{i\in I} p_i \cdot \overline{s_i}$, where I is a finite index set and $\sum_{i\in I} p_i = 1$
2. For each $i \in I$ there is a state t_i such that $s_i\ \mathcal{R}\ t_i$
3. $\Theta = \sum_{i\in I} p_i \cdot \overline{t_i}$. □

An important point here is that in the decomposition of Δ into $\sum_{i\in I} p_i \cdot \overline{s_i}$, the states s_i are *not necessarily distinct*: that is, the decomposition is not in general unique. Thus when establishing the relationship between Δ and Θ, a given state s in Δ may play a number of different roles.

From Definition 2.1, the next two properties follows. In fact, they are sometimes used in the literature as definitions of lifting relations instead of being properties (see e.g. [54, 37]).

Theorem 2.4

1. Let Δ and Θ be distributions over S and T, respectively. Then $\Delta\ \mathcal{R}^\dagger\ \Theta$ if and only if there exists a weight function $w : S \times T \to [0, 1]$ such that
   (a) $\forall s \in S : \sum_{t\in T} w(s, t) = \Delta(s)$
   (b) $\forall t \in T : \sum_{s\in S} w(s, t) = \Theta(t)$
   (c) $\forall (s, t) \in S \times T : w(s, t) > 0 \Rightarrow s\ \mathcal{R}\ t$.
2. Let Δ, Θ be distributions over S and $\mathcal{R}$ is an equivalence relation. Then $\Delta\ \mathcal{R}^\dagger\ \Theta$ if and only if $\Delta(C) = \Theta(C)$ for all equivalence class $C \in S/\mathcal{R}$, where $\Delta(C)$ stands for the accumulation probability $\sum_{s\in C} \Delta(s)$.

Proof:

1. ($\Rightarrow$) Suppose $\Delta\ \mathcal{R}^\dagger\ \Theta$. By Proposition 2.3, we can decompose Δ and Θ such that $\Delta = \sum_{i\in I} p_i \cdot \overline{s_i}$, $\Theta = \sum_{i\in I} p_i \cdot \overline{t_i}$, and $s_i\ \mathcal{R}\ t_i$ for all $i \in I$. We define the weight function w by letting $w(s, t) = \sum\{p_i \mid s_i = s, t_i = t, i \in I\}$ for any $s \in S, t \in T$. This weight function can be checked to meet our requirements.
   (a) For any $s \in S$, it holds that
   $$\begin{array}{rcl} \sum_{t\in T} w(s,t) & = & \sum_{t\in T} \sum\{p_i \mid s_i = s, t_i = t, i \in I\} \\ & = & \sum\{p_i \mid s_i = s, i \in I\} \\ & = & \Delta(s) \end{array}$$
   (b) Similarly, we have $\sum_{s\in S} w(s, t) = \Theta(t)$.
   (c) For any $s \in S, t \in T$, if $w(s, t) > 0$ then there is some $i \in I$ such that $p_i > 0$, $s_i = s$, and $t_i = t$. It follows from $s_i\ \mathcal{R}\ t_i$ that $s\ \mathcal{R}\ t$.

($\Leftarrow$) Suppose there is a weight function w satisfying the three conditions in the hypothesis. We construct the index set $I = \{(s,t) \mid w(s,t) > 0, s \in S, t \in T\}$ and probabilities $p_{(s,t)} = w(s,t)$ for each $(s,t) \in I$.

(a) It holds that $\Delta = \sum_{(s,t)\in I} p_{(s,t)} \cdot \overline{s}$ because, for any $s \in S$,

$$\begin{array}{rcl} (\sum_{(s,t)\in I} p_{(s,t)} \cdot \overline{s})(s) & = & \sum_{(s,t)\in I} w(s,t) \\ & = & \sum\{w(s,t) \mid w(s,t) > 0, t \in T\} \\ & = & \sum\{w(s,t) \mid t \in T\} \\ & = & \Delta(s) \end{array}$$

(b) Similarly, we have $\Theta = \sum_{(s,t)\in I} w(s,t) \cdot \overline{t}$.

(c) For each $(s,t) \in I$, we have $w(s,t) > 0$, which implies $s \mathrel{\mathcal{R}} t$.

Hence, the above decompositions of Δ and Θ meet the requirement of the lifting $\Delta \mathrel{\mathcal{R}^\dagger} \Theta$.

2. ($\Rightarrow$) Suppose $\Delta \mathrel{\mathcal{R}^\dagger} \Theta$. By Proposition 2.3, we can decompose Δ and Θ such that $\Delta = \sum_{i\in I} p_i \cdot \overline{s_i}$, $\Theta = \sum_{i\in I} p_i \cdot \overline{t_i}$, and $s_i \mathrel{\mathcal{R}} t_i$ for all $i \in I$. For any equivalence class $C \in S/\mathcal{R}$, we have that

$$\begin{array}{rcl} \Delta(C) = \sum_{s\in C} \Delta(s) & = & \sum_{s\in C} \sum\{p_i \mid i \in I, s_i = s\} \\ & = & \sum\{p_i \mid i \in I, s_i \in C\} \\ & = & \sum\{p_i \mid i \in I, t_i \in C\} \\ & = & \Theta(C) \end{array}$$

where the equality in the third line is justified by the fact that $s_i \in C$ iff $t_i \in C$ since $s_i \mathrel{\mathcal{R}} t_i$ and $C \in S/\mathcal{R}$.

($\Leftarrow$) Suppose, for each equivalence class $C \in S/\mathcal{R}$, it holds that $\Delta(C) = \Theta(C)$. We construct the index set $I = \{(s,t) \mid s \mathrel{\mathcal{R}} t \text{ and } s,t \in S\}$ and probabilities $p_{(s,t)} = \frac{\Delta(s)\Theta(t)}{\Delta([s]_\mathcal{R})}$ for each $(s,t) \in I$, where $[s]_\mathcal{R}$ stands for the equivalence class that contains s.

(a) It holds that $\Delta = \sum_{(s,t)\in I} p_{(s,t)} \cdot \overline{s}$ because, for any $s' \in S$,

$$\begin{array}{rcl} (\sum_{(s,t)\in I} p_{(s,t)} \cdot \overline{s})(s') & = & \sum_{(s',t)\in I} p_{(s',t)} \\ & = & \sum\{\frac{\Delta(s')\Theta(t)}{\Delta([s']_\mathcal{R})} \mid s' \mathrel{\mathcal{R}} t,\ t \in S\} \\ & = & \sum\{\frac{\Delta(s')\Theta(t)}{\Delta([s']_\mathcal{R})} \mid t \in [s']_\mathcal{R}\} \\ & = & \frac{\Delta(s')}{\Delta([s']_\mathcal{R})} \sum\{\Theta(t) \mid t \in [s']_\mathcal{R}\} \\ & = & \frac{\Delta(s')}{\Delta([s']_\mathcal{R})} \Theta([s']_\mathcal{R}) \\ & = & \frac{\Delta(s')}{\Delta([s']_\mathcal{R})} \Delta([s']_\mathcal{R}) \\ & = & \Delta(s') \end{array}$$

(b) Similarly, we have $\Theta = \sum_{(s,t)\in I} p_{(s,t)} \cdot \overline{t}$.

(c) For each $(s,t) \in I$, we have $s \mathrel{\mathcal{R}} t$.

Hence, the above decompositions of Δ and Θ meet the requirement of the lifting $\Delta \mathrel{\mathcal{R}^\dagger} \Theta$. $\square$

3 Justifying the lifting operation

In our opinion, the lifting operation given in Definition 2.1 is not only concise but also on the right track. This is justified by its intrinsic connection with some fundamental concepts in mathematics, notably the Kantorovich metric.

3.1 Justification by the Kantorovich metric

We begin with some historical notes. The *transportation problem* has been playing an important role in linear programming due to its general formulation and methods of solution. The original transportation problem, formulated by the French mathematician G. Monge in 1781 [45], consists of finding an optimal way of shovelling a pile of sand into a hole of the same volume. In the 1940s, the Russian mathematician and economist L.V. Kantorovich, who was awarded a Nobel prize in economics in 1975 for the theory of optimal allocation of resources, gave a relaxed formulation of the problem and proposed a variational principle for solving the problem [34]. Unfortunately, Kantorovich's work went unrecognized during a long period of time. The later known *Kantorovich metric* has appeared in the literature under different names, because it has been rediscovered historically several times from different perspectives. Many metrics known in measure theory, ergodic theory, functional analysis, statistics, etc. are special cases of the general definition of the Kantorovich metric [65]. The elegance of the formulation, the fundamental character of the optimality criterion, as well as the wealth of applications, which keep arising, place the Kantorovich metric in a prominent position among the mathematical works of the 20th century. In addition, this formulation can be computed in polynomial time [47], which is an appealing feature for its use in solving applied problems. For example, it is widely used to solve a variety of problems in business and economy such as market distribution, plant location, scheduling problems etc. In recent years the metric attracted the attention of computer scientists [9]: it has been used in various different areas in computer science such as probabilistic concurrency, image retrieval, data mining, bioinformatics, etc.

Roughly speaking, the Kantorovich metric provides a way of measuring the distance between two distributions. Of course, this requires first a notion of distance between the basic elements that are aggregated into the distributions, which is often referred to as the *ground distance*. In other words, the Kantorovich metric defines a "lifted" distance between two distributions of mass in a space that is itself endowed with a ground distance. There are a host of metrics available in the literature (see e.g. [26]) to quantify the distance between probability measures; see [52] for a comprehensive review of metrics in the space of probability measures. The Kantorovich metric has an elegant formulation and a natural interpretation in terms of the transportation problem.

We now recall the mathematical definition of the Kantorovich metric. Let (X, m) be a separable metric space. (This condition will be used by Theorem 3.4 below.)

Definition 3.1 Given any two Borel probability measures Δ and Θ on X, the *Kantorovich distance* between Δ and Θ is defined by

$$K(\Delta, \Theta) = \sup \left\{ \left| \int f d\Delta - \int f d\Theta \right| : ||f|| \leq 1 \right\}.$$

where $||\cdot||$ is the *Lipschitz semi-norm* defined by $||f|| = \sup_{x \neq y} \frac{|f(x)-f(y)|}{m(x,y)}$ for a function $f : X \to \mathbb{R}$ with $\mathbb{R}$ being the set of all real numbers.

The Kantorovich metric has an alternative characterisation. We denote by $\mathbf{P}(X)$ the set of all Borel probability measures on X such that for all $z \in X$, if $\Delta \in \mathbf{P}(X)$ then $\int_X m(x,z)\Delta(x) < \infty$. We write $M(\Delta, \Theta)$ for the set of all Borel probability measures on the product space $X \times X$ with marginal measures Δ and Θ, i.e. if $\Gamma \in M(\Delta,\Theta)$ then $\int_{y\in X} d\Gamma(x,y) = d\Delta(x)$ and $\int_{x\in X} d\Gamma(x,y) = d\Theta(y)$ hold.

Definition 3.2 For $\Delta, \Theta \in \mathbf{P}(X)$, we define the metric L as follows:

$$L(\Delta,\Theta) = \inf\left\{ \int m(x,y)d\Gamma(x,y) : \Gamma \in M(\Delta,\Theta)\right\}.$$

Lemma 3.3 If (X, m) is a separable metric space then K and L are metrics on $\mathbf{P}(X)$. □

The famous Kantorovich-Rubinstein duality theorem gives a dual representation of K in terms of L.

Theorem 3.4 [Kantorovich-Rubinstein [35]] If (X, m) is a separable metric space then for any two distributions $\Delta, \Theta \in \mathbf{P}(X)$ we have $K(\Delta,\Theta) = L(\Delta,\Theta)$. □

In view of the above theorem, many papers in the literature directly take Definition 3.2 as the definition of the Kantorovich metric. Here we keep the original definition, but it is helpful to understand K by using L. Intuitively, a probability measure $\Gamma \in M(\Delta,\Theta)$ can be understood as a *transportation* from one unit mass distribution Δ to another unit mass distribution Θ. If the distance $m(x, y)$ represents the cost of moving one unit of mass from location x to location y then the Kantorovich distance gives the optimal total cost of transporting the mass of Δ to Θ. We refer the reader to [66] for an excellent exposition on the Kantorovich metric and the duality theorem.

Many problems in computer science only involve finite state spaces, so discrete distributions with finite supports are sometimes more interesting than continuous distributions. For two discrete distributions Δ and Θ with finite supports $\{x_1, ..., x_n\}$ and $\{y_1, ..., y_l\}$, respectively, minimizing the total cost of a discretised version of the transportation problem reduces to the following linear programming problem:

$$(1)\qquad \begin{array}{ll} \text{minimize} & \sum_{i=1}^n \sum_{j=1}^l \Gamma(x_i, y_j) m(x_i, y_j) \\ \text{subject to} & \bullet\ \forall 1 \le i \le n : \sum_{j=1}^l \Gamma(x_i, y_j) = \Delta(x_i) \\ & \bullet\ \forall 1 \le j \le l : \sum_{i=1}^n \Gamma(x_i, y_j) = \Theta(y_j) \\ & \bullet\ \forall 1 \le i \le n, 1 \le j \le l : \Gamma(x_i, y_j) \ge 0. \end{array}$$

Since (1) is a special case of the discrete mass transportation problem, some well-known polynomial time algorithm like [47] can be employed to solve it, which is an attractive feature for computer scientists.

Recall that a pseudometric is a function that yields a non-negative real number for each pair of elements and satisfies the following: $m(s, s) = 0$, $m(s, t) = m(t, s)$, and $m(s, t) \le m(s, u) + m(u, t)$, for any $s, t \in S$. We say a pseudometric m is 1-bounded if $m(s, t) \le 1$ for any s and t. Let Δ and Θ be distributions over a finite set S of states. In [61] a 1-bounded pseudometric m on S is lifted to be a 1-bounded pseudometric $\hat{m}$ on $\mathcal{D}(S)$ by setting the distance $\hat{m}(\Delta,\Theta)$ to be the value of the following linear programming problem:

$$(2)\qquad \begin{array}{ll} \text{maximize} & \sum_{s\in S}(\Delta(s) - \Theta(s))x_s \\ \text{subject to} & \bullet\ \forall s,t \in S : x_s - x_t \le m(s,t) \\ & \bullet\ \forall s \in S : 0 \le x_s \le 1. \end{array}$$

This problem can be dualised and then simplified to yield the following problem:

$$\text{(3)} \qquad \begin{array}{ll} \text{minimize} & \sum_{s,t\in S} y_{st} m(s,t) \\ \text{subject to} & \bullet\ \forall s \in S : \sum_{t\in S} y_{st} = \Delta(s) \\ & \bullet\ \forall t \in S : \sum_{s\in S} y_{st} = \Theta(t) \\ & \bullet\ \forall s,t \in S : y_{st} \geq 0. \end{array}$$

Now (3) is in exactly the same form as (1).

This way of lifting pseudometrics via the Kantorovich metric as given in (3) has an interesting connection with the lifting of binary relations given in Definition 2.1.

Theorem 3.5 Let R be a binary relation and m a pseudometric on a state space S satisfying

$$\text{(4)} \qquad s\ R\ t \quad \text{iff} \quad m(s,t) = 0$$

for any $s, t \in S$. Then it holds that

$$\Delta\ \ R^{\dagger}\ \ \Theta \quad \text{iff} \quad \hat{m}(\Delta, \Theta) = 0$$

for any distributions $\Delta, \Theta \in \mathcal{D}(S)$.

Proof: Suppose $\Delta\ \ R^{\dagger}\ \ \Theta$. From Theorem 2.4(1) we know there is a weight function w such that

1. $\forall s \in S : \sum_{t\in S} w(s,t) = \Delta(s)$
2. $\forall t \in S : \sum_{s\in S} w(s,t) = \Theta(t)$
3. $\forall s,t \in S : w(s,t) > 0 \Rightarrow s\ R\ t$.

By substituting $w(s,t)$ for $y_{s,t}$ in (3), the three constraints there can be satisfied. For any $s, t \in S$ we distinguish two cases:

1. either $w(s,t) = 0$
2. or $w(s,t) > 0$. In this case we have $s\ R\ t$, which implies $m(s,t) = 0$ by (4).

Therefore, we always have $w(s,t)m(s,t) = 0$ for any $s, t \in S$. Consequently, $\sum_{s,t\in S} w(s,t)m(s,t) = 0$ and the optimal value of the problem in (3) must be 0, i.e. $\hat{m}(\Delta, \Theta) = 0$, and the optimal solution is determined by w.

The above reasoning can be reversed to show that the optimal solution of (3) determines a weight function, thus $\hat{m}(\Delta, \Theta) = 0$ implies $\Delta\ \ R^{\dagger}\ \ \Theta$. $\square$

The above property will be used in Section 6 to give a metric characterisation of probabilistic bisimulation (cf. Theorem 6.9).

3.2 Justification by network flow

The lifting operation discussed in Section 2 is also related to the maximum flow problem in optimisation theory. This was already observed by Baier et al. in [2].

We briefly recall the basic definitions of networks. More details can be found in e.g. [21]. A *network* is a tuple $\mathcal{N} = (N, E, \perp, \top, c)$ where (N, E) is a finite directed graph (i.e. N is a set of nodes and $E \subseteq N \times N$ is a set of edges) with two special nodes $\perp$ (the *source*) and $\top$ (the *sink*) and a *capability* c, i.e. a function that assigns to each edge $(v, w) \in E$ a non-negative number $c(v, w)$. A *flow function* f for $\mathcal{N}$ is a function that assigns to edge e a real number $f(e)$ such that

- $0 \leq f(e) \leq c(e)$ for all edges e.
- Let $in(v)$ be the set of incoming edges to node v and $out(v)$ the set of outgoing edges from node v. Then, for each node $v \in N \backslash \{\perp, \top\}$,

$$\sum_{e \in in(v)} f(e) \;=\; \sum_{e \in out(v)} f(e).$$

The *flow* $F(f)$ of f is given by

$$F(f) \;=\; \sum_{e \in out(\perp)} f(e) - \sum_{e \in in(\perp)} f(e).$$

The *maximum flow* in $\mathcal{N}$ is the supremum (maximum) over the flows $F(f)$, where f is a flow function in $\mathcal{N}$.

We will see that the question whether $\Delta\ \mathcal{R}^\dagger\ \Theta$ can be reduced to a maximum flow problem in a suitably chosen network. Suppose $\mathcal{R} \subseteq S \times S$ and $\Delta, \Theta \in \mathcal{D}(S)$. Let $S' = \{s' \mid s \in S\}$ where s' are pairwise distinct new states, i.e. $s' \in S$ for all $s \in S$. We create two states $\perp$ and $\top$ not contained in $S \cup S'$ with $\perp \neq \top$. We associate with the pair (Δ, Θ) the following network $\mathcal{N}(\Delta, \Theta, \mathcal{R})$.

- The nodes are $N = S \cup S' \cup \{\perp, \top\}$.
- The edges are $E = \{(s, t') \mid (s, t) \in \mathcal{R}\} \cup \{(\perp, s) \mid s \in S\} \cup \{(s', \top) \mid s \in S\}$.
- The capability c is defined by $c(\perp, s) = \Delta(s)$, $c(t', \top) = \Theta(t)$ and $c(s, t') = 1$ for all $s, t \in S$.

The next lemma appeared as Lemma 5.1 in [2].

Lemma 3.6 Let S be a finite set, $\Delta, \Theta \in \mathcal{D}(S)$ and $\mathcal{R} \subseteq S \times S$. The following statements are equivalent.

1. There exists a weight function w for (Δ, Θ) with respect to $\mathcal{R}$.
2. The maximum flow in $\mathcal{N}(\Delta, \Theta, \mathcal{R})$ is 1. □

Since the lifting operation given in Definition 2.1 can also be stated in terms of weight functions, we obtain the following characterisation using network flow.

Theorem 3.7 Let S be a finite set, $\Delta, \Theta \in \mathcal{D}(S)$ and $\mathcal{R} \subseteq S \times S$. Then $\Delta\ \mathcal{R}^\dagger\ \Theta$ if and only if the maximum flow in $\mathcal{N}(\Delta, \Theta, \mathcal{R})$ is 1.

Proof: Combining Theorem 2.4(1) and Lemma 3.6. □

The above property will play an important role in Section 7 to give an "on the fly" algorithm for checking probabilistic bisimilarity.

4 Probabilistic bisimulation

With a solid base of the lifting operation, we can proceed to define a probabilistic version of bisimulation. We start with a probabilistic generalisation of labelled transition systems (LTSs).

Definition 4.1 A *probabilistic labelled transition system* (pLTS)[1] is a triple $\langle S, \mathsf{Act}, \rightarrow\rangle$, where

1. S is a set of states;
2. Act is a set of actions;
3. $\rightarrow \; \subseteq \; S \times \mathsf{Act} \times \mathcal{D}(S)$ is the transition relation.

As with LTSs, we usually write $s \xrightarrow{a} \Delta$ in place of $(s, a, \Delta) \in \rightarrow$. A pLTS is *finitely branching* if for each state $s \in S$ the set $\{\langle \alpha, \Delta\rangle \mid s \xrightarrow{\alpha} \Delta, \alpha \in \mathsf{Act}, \Delta \in \mathcal{D}(S)\}$ is finite; if moreover S is finite, then the pLTS is *finitary*.

In a pLTS, one step of transition leaves a single state but might end up in a set of states; each of them can be reached with certain probability. An LTS may be viewed as a degenerate pLTS, one in which only point distributions are used.

Let s and t are two states in a pLTS, we say t can simulate the behaviour of s if the latter can exhibit action a and lead to distribution Δ then the former can also perform a and lead to a distribution, say Θ, which can mimic Δ in successor states. We are interested in a relation between two states, but it is expressed by invoking a relation between two distributions. To formalise the mimicking of one distribution by the other, we make use of the lifting operation investigated in Section 2.

Definition 4.2 A relation $\mathcal{R} \subseteq S \times S$ is a *probabilistic simulation* if $s \;\mathcal{R}\; t$ implies

- if $s \xrightarrow{a} \Delta$ then there exists some Θ such that $t \xrightarrow{a} \Theta$ and $\Delta \;\mathcal{R}^\dagger\; \Theta$.

If both $\mathcal{R}$ and $\mathcal{R}^{-1}$ are probabilistic simulations, then $\mathcal{R}$ is a *probabilistic bisimulation.* The largest probabilistic bisimulation, denoted by $\sim$, is called *probabilistic bisimilarity.*

As in the nonprobabilistic setting, probabilistic bisimilarity can be approximated by a family of inductively defined relations.

Definition 4.3 Let S be the state set of a pLTS. We define:

- $\sim_0 := S \times S$
- $s \sim_{n+1} t$, for $n \geq 0$, if
  1. whenever $s \xrightarrow{a} \Delta$, there exists some Θ such that $t \xrightarrow{a} \Theta$ and $\Delta \sim_n^\dagger \Theta$;
  2. whenever $t \xrightarrow{a} \Theta$, there exists some Δ such that $s \xrightarrow{a} \Delta$ and $\Delta \sim_n^\dagger \Theta$.
- $\sim_\omega := \bigcap_{n \geq 0} \sim_n$

[1] Essentially the same model has appeared in the literature under different names such as *NP-systems* [30], *probabilistic processes* [31], *simple probabilistic automata* [53], *probabilistic transition systems* [32] etc. Furthermore, there are strong structural similarities with *Markov Decision Processes* [51, 15].

In general, $\sim$ is a strictly finer relation than $\sim_\omega$. However, the two relations coincide when limited to finitely branching pLTSs.

Proposition 4.4 On finitely branching pLTSs, $\sim_\omega$ coincides with $\sim$.

Proof: It is trivial to show by induction that $s \sim t$ implies $s \sim_n t$ for all $n \geq 0$, thus $s \sim_\omega t$.

Now we show that $\sim_\omega$ is a bisimulation. Suppose $s \sim_\omega t$ and $s \xrightarrow{a} \Delta$. We have to show that there is some Θ with $t \xrightarrow{a} \Theta$ and $\Delta \sim_\omega^\dagger \Theta$. Consider the set

$$T := \{\Theta \mid t \xrightarrow{a} \Theta \wedge \Delta \not\sim_\omega^\dagger \Theta\}.$$

For each $\Theta \in T$, we have $\Delta \not\sim_\omega^\dagger \Theta$, which means that there is some $n_\Theta > 0$ with $\Delta \not\sim_{n_\Theta}^\dagger \Theta$. Since t is finitely branching, T is a finite set. Let $N = max\{n_\Theta \mid \Theta \in T\}$. It holds that $\Delta \not\sim_N^\dagger \Theta$ for all $\Theta \in T$, since by a straightforward induction on m we can show that $s \sim_n t$ implies $s \sim_m t$ for all $m, n \geq 0$ with $n > m$. By the assumption $s \sim_\omega t$ we know that $s \sim_{N+1} t$. It follows that there is some Θ with $t \xrightarrow{a} \Theta$ and $\Delta \sim_N^\dagger \Theta$, so $\Theta \notin T$ and hence $\Delta \sim_\omega^\dagger \Theta$. By symmetry we also have that if $t \xrightarrow{a} \Theta$ then there is some Δ with $s \xrightarrow{a} \Delta$ and $\Delta \sim_\omega^\dagger \Theta$. $\square$

Proposition 4.4 has appeared in [1]; here we have given a simpler proof.

5 Logical characterisation

Let $\mathcal{L}$ be a logic. We use the notation $\mathcal{L}(s)$ to stand for the set of formulae that state s satisfies. This induces an equivalence relation on states: $s \;=^{\mathcal{L}}\; t$ iff $\mathcal{L}(s) = \mathcal{L}(t)$. Thus, two states are equivalent when they satisfy exactly the same set of formulae.

In this section we consider two kinds of logical characterisations of probabilistic bisimilarity.

Definition 5.1 [Adequacy and expressivity]

1. $\mathcal{L}$ is *adequate* w.r.t. $\sim$ if for any states s and t,

$$s =^{\mathcal{L}} t \;\text{ iff }\; s \sim t.$$

2. $\mathcal{L}$ is *expressive* w.r.t. $\sim$ if for each state s there exists a *characteristic formula* $\varphi_s \in \mathcal{L}$ such that, for any states s and t,

$$t \models \varphi_s \;\text{ iff }\; s \sim t.$$

We will propose a probabilistic extension of the Hennessy-Milner logic, showing its adequacy, and then a probabilistic extension of the modal mu-calculus, showing its expressivity.

5.1 An adequate logic

We extend the Hennessy-Milner logic by adding a probabilistic choice modality to express the bebaviour of distributions.

Definition 5.2 The class $\mathcal{L}$ of modal formulae over Act, ranged over by φ, is defined by the following grammar:

$$\begin{array}{rcl}\varphi & := & \top \mid \varphi_1 \wedge \varphi_2 \mid \langle a\rangle\psi \mid \neg\varphi \\ \psi & := & \bigoplus_{i\in I} p_i \cdot \varphi_i\end{array}$$

We call φ a *state formula* and ψ a *distribution formula*. Note that a distribution formula ψ only appears as the continuation of a diamond modality $\langle a\rangle\psi$. We sometimes use the finite conjunction $\bigwedge_{i\in I}\varphi_i$ as a syntactic sugar.

The *satisfaction relation* $\models\subseteq S\times\mathcal{L}$ is defined by

- $s\models\top$ for all $s\in S$.
- $s\models\varphi_1\wedge\varphi_2$ if $s\models\varphi_i$ for $i=1,2$.
- $s\models\langle a\rangle\psi$ if for some $\Delta\in\mathcal{D}(S)$, $s\xrightarrow{a}\Delta$ and $\Delta\models\psi$.
- $s\models\neg\varphi$ if it is not the case that $s\models\varphi$.
- $\Delta\models\bigoplus_{i\in I}p_i\cdot\varphi_i$ if there are $\Delta_i\in\mathcal{D}(S)$, for all $i\in I, t\in\lceil\Delta_i\rceil$, with $t\models\varphi_i$, such that $\Delta=\sum_{i\in I}p_i\cdot\Delta_i$.

With a slight abuse of notation, we write $\Delta\models\psi$ above to mean that Δ satisfies the distribution formula ψ. The introduction of distribution formula distinguishes $\mathcal{L}$ from other probabilistic modal logics e.g. [33, 49].

It turns out that $\mathcal{L}$ is adequate w.r.t. probabilistic bisimilarity.

Theorem 5.3 [Adequacy] Let s and t be any two states in a finitely branching pLTS. Then $s\sim t$ if and only if $s=^{\mathcal{L}}t$.

Proof: ($\Rightarrow$) Suppose $s\sim t$, we show that $s\models\varphi\Leftrightarrow t\models\varphi$ by structural induction on φ.

- Let $s\models\top$, we clearly have $t\models\top$.
- Let $s\models\varphi_1\wedge\varphi_2$. Then $s\models\varphi_i$ for $i=1,2$. So by induction $t\models\varphi_i$, and we have $t\models\varphi_1\wedge\varphi_2$. By symmetry we also have $t\models\varphi_1\wedge\varphi_2$ implies $s\models\varphi_1\wedge\varphi_2$.
- Let $s\models\neg\varphi$. So $s\not\models\varphi$, and by induction we have $t\not\models\varphi$. Thus $t\models\neg\varphi$. By symmetry we also have $t\not\models\varphi$ implies $s\not\models\varphi$.
- Let $s\models\langle a\rangle\bigoplus_{i\in I}p_i\cdot\varphi_i$. Then $s\xrightarrow{a}\Delta$ and $\Delta\models\bigoplus_{i\in I}p_i\cdot\varphi_i$ for some Δ. So $\Delta=\sum_{i\in i}p_i\cdot\Delta_i$ and for all $i\in I$ and $s'\in\lceil\Delta_i\rceil$ we have $s'\models\varphi_i$. Since $s\sim t$, there is some Θ with $t\xrightarrow{a}\Theta$ and $\Delta\sim^\dagger\Theta$. By Proposition 2.2 we have that $\Theta=\sum_{i\in I}p_i\cdot\Theta_i$ and $\Delta_i\sim^\dagger\Theta_i$. It follows that for each $t'\in\lceil\Theta_i\rceil$ there is some $s'\in\lceil\Delta_i\rceil$ with $s'\sim t'$. So by induction we have $t'\models\varphi_i$ for all $t'\in\lceil\Theta_i\rceil$ with $i\in I$. Therefore, we have $\Theta\models\bigoplus_{i\in I}p_i\cdot\varphi_i$. It follows that $t\models\langle a\rangle\bigoplus_{i\in I}p_i\cdot\varphi_i$. By symmetry we also have $t\models\langle a\rangle\bigoplus_{i\in I}p_i\cdot\varphi_i\Rightarrow s\models\langle a\rangle\bigoplus_{i\in I}p_i\cdot\varphi_i$.

($\Leftarrow$) We show that the relation $=^{\mathcal{L}}$ is a probabilistic bisimulation. Suppose $s=^{\mathcal{L}}t$ and $s\xrightarrow{a}\Delta$. We have to show that there is some Θ with $t\xrightarrow{a}\Theta$ and $\Delta\;(=^{\mathcal{L}})^\dagger\;\Theta$. Consider the set

$$T:=\{\Theta\mid t\xrightarrow{a}\Theta\wedge\Theta=\sum_{s'\in\lceil\Delta\rceil}\Delta(s')\cdot\Theta_{s'}\wedge\exists s'\in\lceil\Delta\rceil,\exists t'\in\lceil\Theta_{s'}\rceil:s'\neq^{\mathcal{L}}t'\}$$

For each $\Theta \in T$, there must be some $s'_\Theta \in \lceil\Delta\rceil$ and $t'_\Theta \in \lceil\Theta_{s'_\Theta}\rceil$ such that (i) either there is a formula φ_Θ with $s'_\Theta \models \varphi_\Theta$ but $t'_\Theta \not\models \varphi_\Theta$ (ii) or there is a formula φ'_Θ with $t'_\Theta \models \varphi'_\Theta$ but $s'_\Theta \not\models \varphi'_\Theta$. In the latter case we set $\varphi_\Theta = \neg\varphi'_\Theta$ and return back to the former case. So for each $s' \in \lceil\Delta\rceil$ it holds that $s' \models \bigwedge_{\{\Theta\in T|s'_\Theta=s'\}} \varphi_\Theta$ and for each $\Theta \in T$ with $s'_\Theta = s'$ there is some $t'_\Theta \in \lceil\Theta_{s'}\rceil$ with $t'_\Theta \not\models \bigwedge_{\{\Theta\in T|s'_\Theta=s'\}} \varphi_\Theta$. Let

$$\varphi := \langle a\rangle \bigoplus_{s'\in\lceil\Delta\rceil} \Delta(s') \cdot \bigwedge_{\{\Theta\in T|s'_\Theta=s'\}} \varphi_\Theta.$$

It is clear that $s \models \varphi$, hence $t \models \varphi$ by $s =^{\mathcal{L}} t$. It follows that there must be a Θ^* with $t \xrightarrow{a} \Theta^*$, $\Theta^* = \sum_{s'\in\lceil\Delta\rceil} \Delta(s') \cdot \Theta^*_{s'}$ and for each $s' \in \lceil\Delta\rceil, t' \in \lceil\Theta^*_{s'}\rceil$ we have $t' \models \bigwedge_{\{\Theta\in T|s'_\Theta=s'\}} \varphi_\Theta$. This means that $\Theta^* \notin T$ and hence for each $s' \in \lceil\Delta\rceil, t' \in \lceil\Theta^*_{s'}\rceil$ we have $s' =^{\mathcal{L}} t'$. It follows that $\Delta \; (=^{\mathcal{L}})^\dagger \; \Theta^*$. By symmetry all transitions of t can be matched up by transitions of s. $\square$

5.2 An expressive logic

We now add the probabilistic choice modality introduced in Section 5.1 to the modal mu-calculus, and show that the resulting probabilistic mu-calculus is expressive w.r.t. probabilistic bisimilarity.

5.2.1 Probabilistic modal mu-calculus

Let *Var* be a countable set of variables. We define a set $\mathcal{L}_\mu$ of modal formulae in positive normal form given by the following grammar:

$$\begin{array}{rcl} \varphi & := & \top \mid \bot \mid \langle a\rangle\psi \mid [a]\psi \mid \varphi_1 \wedge \varphi_2 \mid \varphi_1 \vee \varphi_2 \mid X \mid \mu X.\varphi \mid \nu X.\varphi \\ \psi & := & \bigoplus_{i\in I} p_i \cdot \varphi_i \end{array}$$

where $a \in \mathsf{Act}$, I is a finite index set and $\sum_{i\in I} p_i = 1$. Here we still write φ for a state formula and ψ a distribution formula. Sometimes we also use the finite conjunction $\bigwedge_{i\in I} \varphi_i$ and disjunction $\bigvee_{i\in I} \varphi_i$. As usual, we have $\bigwedge_{i\in\emptyset} \varphi_i = \top$ and $\bigvee_{i\in\emptyset} \varphi_i = \bot$.

The two fixed point operators μX and νX bind the respective variable X. We apply the usual terminology of free and bound variables in a formula and write $fv(\varphi)$ for the set of free variables in φ.

We use *environments*, which binds free variables to sets of distributions, in order to give semantics to formulae. We fix a finitary pLTS and let S be its state set. Let

$$\mathsf{Env} = \{\, \rho \mid \rho : Var \to \mathcal{P}(S) \,\}$$

be the set of all environments and ranged over by ρ. For a set $V \subseteq S$ and a variable $X \in Var$, we write $\rho[X \mapsto V]$ for the environment that maps X to V and Y to $\rho(Y)$ for all $Y \neq X$.

The semantics of a formula φ can be given as the set of states satisfying it. This entails a semantic functional $[\![\ ]\!] : \mathcal{L}_\mu \to \mathsf{Env} \to \mathcal{P}(S)$ defined inductively in Figure 1, where we also apply $[\![\ ]\!]$ to distribution formulae and $[\![\psi]\!]$ is interpreted as the set of distributions that satisfy ψ. As the meaning of a closed formula φ does not depend on the environment, we write $[\![\varphi]\!]$ for $[\![\varphi]\!]_\rho$ where ρ is an arbitrary environment.

The semantics of probabilistic modal mu-calculus (pMu) is the same as that of the modal mu-calculus [36] except for the probabilistic choice modality which are satisfied by distributions. The

$$
\begin{array}{rcl}
[\![\top]\!]_\rho & = & S \\
[\![\bot]\!]_\rho & = & \emptyset \\
[\![\varphi_1 \wedge \varphi_2]\!]_\rho & = & [\![\varphi_1]\!]_\rho \cap [\![\varphi_2]\!]_\rho \\
[\![\varphi_1 \vee \varphi_2]\!]_\rho & = & [\![\varphi_1]\!]_\rho \cup [\![\varphi_2]\!]_\rho \\
[\![\langle a\rangle\psi]\!]_\rho & = & \{\, s \in S \mid \exists\Delta : s \xrightarrow{a} \Delta \ \wedge\ \Delta \in [\![\psi]\!]_\rho \,\} \\
[\![[a]\psi]\!]_\rho & = & \{\, s \in S \mid \forall\Delta : s \xrightarrow{a} \Delta \ \Rightarrow\ \Delta \in [\![\psi]\!]_\rho \,\} \\
[\![X]\!]_\rho & = & \rho(X) \\
[\![\mu X.\varphi]\!]_\rho & = & \bigcap\{\, V \subseteq S \mid [\![\varphi]\!]_{\rho[X\mapsto V]} \subseteq V \,\} \\
[\![\nu X.\varphi]\!]_\rho & = & \bigcup\{\, V \subseteq S \mid [\![\varphi]\!]_{\rho[X\mapsto V]} \supseteq V \,\} \\
[\![\bigoplus_{i\in I} p_i \cdot \varphi_i]\!]_\rho & = & \{\, \Delta \in \mathcal{D}(S) \mid \Delta = \bigoplus_{i\in I} p_i \cdot \Delta_i \ \wedge\ \forall i \in I, \forall t \in \lceil \Delta_i \rceil : t \in [\![\varphi_i]\!]_\rho \,\}
\end{array}
$$

Figure 1: Semantics of probabilistic modal mu-calculus

characterisation of *least fixed point formula* $\mu X.\varphi$ and *greatest fixed point formula* $\nu X.\varphi$ follows from the well-known Knaster-Tarski fixed point theorem [56].

We shall consider (closed) *equation systems* of formulae of the form

$$
\begin{array}{rcl}
E : X_1 & = & \varphi_1 \\
& \vdots & \\
X_n & = & \varphi_n
\end{array}
$$

where $X_1, ..., X_n$ are mutually distinct variables and $\varphi_1, ..., \varphi_n$ are formulae having at most $X_1, ..., X_n$ as free variables. Here E can be viewed as a function $E : \mathit{Var} \to \mathcal{L}_\mu$ defined by $E(X_i) = \varphi_i$ for $i = 1, ..., n$ and $E(Y) = Y$ for other variables $Y \in \mathit{Var}$.

An environment ρ is a *solution* of an equation system E if $\forall i : \rho(X_i) = [\![\varphi_i]\!]_\rho$. The existence of solutions for an equation system can be seen from the following arguments. The set Env, which includes all candidates for solutions, together with the partial order $\leq$ defined by

$$\rho \leq \rho' \text{ iff } \forall X \in \mathit{Var} : \rho(X) \subseteq \rho'(X)$$

forms a complete lattice. The *equation functional* $\mathcal{E} : \mathsf{Env} \to \mathsf{Env}$ given in the λ-calculus notation by

$$\mathcal{E} := \lambda\rho.\lambda X.[\![E(X)]\!]_\rho$$

is monotonic. Thus, the Knaster-Tarski fixed point theorem guarantees existence of solutions, and the largest solution

$$\rho_E := \bigsqcup\{\, \rho \mid \rho \leq \mathcal{E}(\rho) \,\}$$

5.2.2 Characteristic equation systems

As studied in [55], the behaviour of a process can be characterised by an equation system of modal formulae. Below we show that this idea also applies in the probabilistic setting.

Definition 5.4 Given a finitary pLTS, its *characteristic equation system* consists of one equation for each state $s_1, ..., s_n \in S$.

$$
\begin{array}{rcl}
E : X_{s_1} & = & \varphi_{s_1} \\
& \vdots & \\
X_{s_n} & = & \varphi_{s_n}
\end{array}
$$

where

$$\varphi_s := \Big(\bigwedge_{s \xrightarrow{a} \Delta} \langle a\rangle X_\Delta\Big) \wedge \Big(\bigwedge_{a\in\mathsf{Act}} [a] \bigvee_{s \xrightarrow{a} \Delta} X_\Delta\Big) \tag{5}$$

with $X_\Delta := \bigoplus_{s\in\lceil\Delta\rceil} \Delta(s)\cdot X_s$.

Theorem 5.5 Suppose E is a characteristic equation system. Then $s \sim t$ if and only if $t \in \rho_E(X_s)$.

Proof: ($\Leftarrow$) Let $\mathcal{R}= \{\, (s,t) \mid t \in \rho_E(X_s)\,\}$. We first show that

$$\Theta \in [\![X_\Delta]\!]_{\rho_E} \text{ implies } \Delta\ \mathcal{R}^\dagger\ \Theta. \tag{6}$$

Let $\Delta = \bigoplus_{i\in I} p_i\cdot\overline{s_i}$, then $X_\Delta = \bigoplus_{i\in I} p_i\cdot X_{s_i}$. Suppose $\Theta \in [\![X_\Delta]\!]_{\rho_E}$. We have that $\Theta = \bigoplus_{i\in I} p_i\cdot\Theta_i$ and, for all $i \in I$ and $t' \in \lceil\Theta_i\rceil$, that $t' \in [\![X_{s_i}]\!]_{\rho_E}$, i.e. $s_i\ \mathcal{R}\ t'$. It follows that $\overline{s_i}\ \mathcal{R}^\dagger\ \Theta_i$ and thus $\Delta\ \mathcal{R}^\dagger\ \Theta$.

Now we show that $\mathcal{R}$ is a bisimulation.

1. Suppose $s\ \mathcal{R}\ t$ and $s \xrightarrow{a} \Delta$. Then $t \in \rho_E(X_s) = [\![\varphi_s]\!]_{\rho_E}$. It follows from (5) that $t \in [\![\langle a\rangle X_\Delta]\!]_{\rho_E}$. So there exists some Θ such that $t \xrightarrow{a} \Theta$ and $\Theta \in [\![X_\Delta]\!]_{\rho_E}$. Now we apply (6).
2. Suppose $s\ \mathcal{R}\ t$ and $t \xrightarrow{a} \Theta$. Then $t \in \rho_E(X_s) = [\![\varphi_s]\!]_{\rho_E}$. It follows from (5) that $t \in [\![[a]\bigvee_{s\xrightarrow{a}\Delta} X_\Delta]\!]$. Notice that it must be the case that s can enable action a, otherwise, $t \in [\![[a]\bot]\!]_{\rho_E}$ and thus t cannot enable a either, in contradiction with the assumption $t \xrightarrow{a} \Theta$. Therefore, $\Theta \in [\![\bigvee_{s\xrightarrow{a}\Delta} X_\Delta]\!]_{\rho_E}$, which implies $\Theta \in [\![X_\Delta]\!]_{\rho_E}$ for some Δ with $s \xrightarrow{a} \Delta$. Now we apply (6).

($\Rightarrow$) We define the environment $\rho_\sim$ by

$$\rho_\sim(X_s) := \{\, t \mid s \sim t\,\}.$$

It sufficies to show that $\rho_\sim$ is a post-fixed point of $\mathcal{E}$, i.e.

$$\rho_\sim \leq \mathcal{E}(\rho_\sim) \tag{7}$$

because in that case we have $\rho_\sim \leq \rho_E$, thus $s \sim t$ implies $t \in \rho_\sim(X_s)$ which in turn implies $t \in \rho_E(X_s)$.

We first show that

$$\Delta \sim^\dagger \Theta \text{ implies } \Theta \in [\![X_\Delta]\!]_{\rho_\sim}. \tag{8}$$

Suppose $\Delta \sim^\dagger \Theta$, by Proposition 2.3 we have that (i) $\Delta = \bigoplus_{i\in I} p_i\cdot\overline{s_i}$, (ii) $\Theta = \bigoplus_{i\in I} p_i\cdot\overline{t_i}$, (iii) $s_i \sim t_i$ for all $i \in I$. We know from (iii) that $t_i \in [\![X_{s_i}]\!]_{\rho_\sim}$. Using (ii) we have that $\Theta \in [\![\bigoplus_{i\in I} p_i\cdot X_{s_i}]\!]_{\rho_\sim}$. Using (i) we obtain $\Theta \in [\![X_\Delta]\!]_{\rho_\sim}$.

Now we are in a position to show (7). Suppose $t \in \rho_\sim(X_s)$. We must prove that $t \in [\![\varphi_s]\!]_{\rho_\sim}$, i.e.

$$t \in \Big(\bigcap_{s\xrightarrow{a}\Delta} [\![\langle a\rangle X_\Delta]\!]_{\rho_\sim}\Big) \cap \Big(\bigcap_{a\in\mathsf{Act}} [\![[a] \bigvee_{s\xrightarrow{a}\Delta} X_\Delta]\!]_{\rho_\sim}\Big)$$

by (5). This can be done by showing that t belongs to each of the two parts of this intersection.

1. Rule 1: $E \to F$
2. Rule 2: $E \to G$
3. Rule 3: $E \to H$ if $X_n \notin fv(\varphi_1, ..., \varphi_n)$

$$
\begin{array}{rcl}
E: X_1 & = & \varphi_1 \\
& \vdots & \\
X_{n-1} & = & \varphi_{n-1} \\
X_n & = & \varphi_n
\end{array}
\qquad
\begin{array}{rcl}
F: X_1 & = & \varphi_1 \\
& \vdots & \\
X_{n-1} & = & \varphi_{n-1} \\
X_n & = & \nu X_n.\varphi_n
\end{array}
\qquad
\begin{array}{rcl}
G: X_1 & = & \varphi_1[\varphi_n/X_n] \\
& \vdots & \\
X_{n-1} & = & \varphi_{n-1}[\varphi_n/X_n] \\
X_n & = & \varphi_n
\end{array}
\qquad
\begin{array}{rcl}
H: X_1 & = & \varphi_1 \\
& \vdots & \\
X_{n-1} & = & \varphi_{n-1}
\end{array}
$$

Figure 2: Transformation rules

1. In the first case, we assume that $s \xrightarrow{a} \Delta$. Since $s \sim t$, there exists some Θ such that $t \xrightarrow{a} \Theta$ and $\Delta \sim^\dagger \Theta$. By (8), we get $\Theta \in [\![X_\Delta]\!]_{\rho_\sim}$. It follows that $t \in [\![\langle a \rangle X_\Delta]\!]_{\rho_\sim}$.
2. In the second case, we suppose $t \xrightarrow{a} \Theta$ for any action $a \in \mathsf{Act}$ and distribution Θ. Then by $s \sim t$ there exists some Δ such that $s \xrightarrow{a} \Delta$ and $\Delta \sim^\dagger \Theta$. By (8), we get $\Theta \in [\![X_\Delta]\!]_{\rho_\sim}$. As a consequence, $t \in [\![[a] \bigvee_{s \xrightarrow{a} \Delta} X_\Delta]\!]_{\rho_\sim}$. Since this holds for arbitrary action a, our desired result follows.

□

5.2.3 Characteristic formulae

So far we know how to construct the characteristic equation system for a finitary pLTS. As introduced in [46], the three transformation rules in Figure 2 can be used to obtain from an equation system E a formula whose interpretation coincides with the interpretation of X_1 in the greatest solution of E. The formula thus obtained from a characteristic equation system is called a *characteristic formula*.

Theorem 5.6 Given a characteristic equation system E, there is a characteristic formula φ_s such that $\rho_E(X_s) = [\![\varphi_s]\!]$ for any state s. □

The above theorem, together with the results in Section 5.2.2, gives rise to the following corollary.

Corollary 5.7 For each state s in a finitary pLTS, there is a characteristic formula φ_s such that $s \sim t$ iff $t \in [\![\varphi_s]\!]$. □

6 Metric characterisation

In the definition of probabilistic bisimulation probabilities are treated as labels since they are matched only when they are identical. One may argue that this does not provide a robust relation: Processes that differ for a very small probability, for instance, would be considered just as different as processes that perform completely different actions. This is particularly relevant to many applications where specifications can be given as perfect, but impractical processes and other, practical processes are considered acceptable if they only differ from the specification with a negligible probability.

To find a more flexible way to differentiate processes, researchers in this area have borrowed from mathematics the notion of metric[2]. A metric is defined as a function that associates a distance with a pair of elements. Whereas topologists use metrics as a tool to study continuity and convergence, we will use them to provide a measure of the difference between two processes that are not quite bisimilar.

Since different processes may behave the same, they will be given distance zero in our metric semantics. So we are more interested in pseudometrics than metrics.

In the rest of this section, we fix a finite state pLTS $(S, \mathsf{Act}, \longrightarrow)$ and provide the set of pseudometrics on S with the following partial order.

Definition 6.1 The relation $\preceq$ for the set $\mathcal{M}$ of 1-bounded pseudometrics on S is defined by

$$m_1 \preceq m_2 \text{ if } \forall s, t : m_1(s,t) \geq m_2(s,t).$$

Here we reverse the ordering with the purpose of characterizing bisimilarity as the *greatest* fixed point (cf: Corollary 6.10).

Lemma 6.2 $(\mathcal{M}, \preceq)$ is a complete lattice.

Proof: The top element is given by $\forall s, t : \top(s,t) = 0$; the bottom element is given by $\bot(s,t) = 1$ if $s \neq t$, 0 otherwise. Greatest lower bounds are given by $(\bigsqcap X)(s,t) = \sup\{m(s,t) \mid m \in X\}$ for any $X \subseteq \mathcal{M}$. Finally, least upper bounds are given by $\bigsqcup X = \bigsqcap \{m \in \mathcal{M} \mid \forall m' \in X : m' \preceq m\}$. $\square$

Definition 6.3 $m \in \mathcal{M}$ is a *state-metric* if, for all $\epsilon \in [0,1)$, $m(s,t) \leq \epsilon$ implies:

- if $s \xrightarrow{a} \Delta$ then there exists some Δ' such that $t \xrightarrow{a} \Delta'$ and $\hat{m}(\Delta, \Delta') \leq \epsilon$

where the lifted metric $\hat{m}$ was defined in (2) via the Kantorovich metric. Note that if m is a state-metric then it is also a metric. By $m(s,t) \leq \epsilon$ we have $m(t,s) \leq \epsilon$, which implies

- if $t \xrightarrow{a} \Delta'$ then there exists some Δ such that $s \xrightarrow{a} \Delta$ and $\hat{m}(\Delta', \Delta) \leq \epsilon$.

In the above definition, we prohibit ϵ to be 1 because we use 1 to represent the distance between any two incomparable states including the case where one state may perform a transition and the other may not.

The greatest state-metric is defined as

$$m_{max} = \bigsqcup \{m \in \mathcal{M} \mid m \text{ is a state-metric}\}.$$

It turns out that state-metrics correspond to bisimulations and the greatest state-metric corresponds to bisimilarity. To make the analogy closer, in what follows we will characterize m_{max} as a fixed point of a suitable monotone function on $\mathcal{M}$. First we recall the definition of Hausdorff distance.

[2]For simplicity, in this section we use the term metric to denote both metric and pseudometric. All the results are based on pseudometrics.

Definition 6.4 Given a 1-bounded metric d on Z, the *Hausdorff distance* between two subsets X, Y of Z is defined as follows:

$$H_d(X,Y) = \max\{\sup_{x\in X}\inf_{y\in Y} d(x,y), \sup_{y\in Y}\inf_{x\in X} d(y,x)\}$$

where $\inf\ \emptyset = 1$ and $\sup\ \emptyset = 0$.

Next we define a function F on $\mathcal{M}$ by using the Hausdorff distance.

Definition 6.5 Let $der(s,a) = \{\Delta \mid s \xrightarrow{a} \Delta\}$. $F(m)$ is a pseudometric given by:

$$F(m)(s,t) = \sup_{a\in\mathsf{Act}}\{H_{\hat{m}}(der(s,a), der(t,a))\}.$$

Thus we have the following property.

Lemma 6.6 For all $\epsilon \in [0,1)$, $F(m)(s,t) \leq \epsilon$ if and only if:

- if $s \xrightarrow{a} \Delta$ then there exists some Δ' such that $t \xrightarrow{a} \Delta'$ and $\hat{m}(\Delta,\Delta') \leq \epsilon$;
- if $t \xrightarrow{a} \Delta'$ then there exists some Δ such that $s \xrightarrow{a} \Delta$ and $\hat{m}(\Delta',\Delta) \leq \epsilon$. □

The above lemma can be proved by directly checking the definition of F, as can the next lemma.

Lemma 6.7 m is a state-metric if and only if $m \preceq F(m)$. □

Consequently we have the following characterisation:

$$m_{max} = \bigsqcup\{m \in \mathcal{M} \mid m \preceq F(m)\}.$$

Lemma 6.8 F is monotone on $\mathcal{M}$. □

Because of Lemma 6.2 and 6.8, we can apply Knaster-Tarski fixed point theorem, which tells us that m_{max} is the greatest fixed point of F. Furthermore, by Lemma 6.7 we know that m_{max} is indeed a state-metric, and it is the greatest state-metric.

We now show the correspondence between state-metrics and bisimulations.

Theorem 6.9 Given a binary relation $\mathcal{R}$ and a pseudometric $m \in \mathcal{M}$ on a finite state pLTS such that

$$m(s,t) = \begin{cases} 0 & \text{if } s\,\mathcal{R}\,t \\ 1 & \text{otherwise.} \end{cases} \tag{9}$$

Then $\mathcal{R}$ is a probabilistic bisimulation if and only if m is a state-metric.

Proof: The result can be proved by using Theorem 3.5, which in turn relies on Theorem 2.4 (1). Below we give an alternative proof that uses Theorem 2.4 (2) instead.

Given two distributions Δ, Δ' over S, let us consider how to compute $\hat{m}(\Delta,\Delta')$ if $\mathcal{R}$ is an equivalence relation. Since S is finite, we may assume that $V_1, ..., V_n \in S/\mathcal{R}$ are all the equivalence classes of S under $\mathcal{R}$. If $s,t \in V_i$ for some $i \in 1..n$, then $m(s,t) = 0$, which implies $x_s = x_t$ by the first constraint of (2). So for each $i \in 1..n$ there exists some x_i such that $x_i = x_s$ for all $s \in V_i$. Thus, some summands of (2) can be grouped together and we have the following linear program:

$$\sum_{i \in 1..n} (\Delta(V_i) - \Delta'(V_i))x_i \tag{10}$$

with the constraint $x_i - x_j \leq 1$ for any $i, j \in 1..n$ with $i \neq j$. Briefly speaking, if $\mathcal{R}$ is an equivalence relation then $\hat{m}(\Delta, \Delta')$ is obtained by maximizing the linear program (10).

($\Rightarrow$) Suppose $\mathcal{R}$ is a bisimulation and $m(s,t) = 0$. From the assumption in (9) we know that $\mathcal{R}$ is an equivalence relation. By the definition of m we have $s\ \mathcal{R}\ t$. If $s \xrightarrow{a} \Delta$ then $t \xrightarrow{a} \Delta'$ for some Δ' such that $\Delta\ \mathcal{R}^\dagger\ \Delta'$. To show that m is a state-metric it suffices to prove $m(\Delta, \Delta') = 0$. We know from $\Delta\ \mathcal{R}^\dagger\ \Delta'$ and Theorem 2.4 (2) that $\Delta(V_i) = \Delta'(V_i)$, for each $i \in 1..n$. It follows that (10) is maximized to be 0, thus $\hat{m}(\Delta, \Delta') = 0$.

($\Leftarrow$) Suppose m is a state-metric and has the relation in (9). Notice that $\mathcal{R}$ is an equivalence relation. We show that it is a bisimulation. Suppose $s\ \mathcal{R}\ t$, which means $m(s,t) = 0$. If $s \xrightarrow{a} \Delta$ then $t \xrightarrow{a} \Delta'$ for some Δ' such that $\hat{m}(\Delta, \Delta') = 0$. To ensure that $\hat{m}(\Delta, \Delta') = 0$, in (10) the following two conditions must be satisfied.

1. No coefficient is positive. Otherwise, if $\Delta(V_i) - \Delta'(V_i) > 0$ then (10) would be maximized to a value not less than $(\Delta(V_i) - \Delta'(V_i))$, which is greater than 0.
2. It is not the case that at least one coefficient is negative and the other coefficients are either negative or 0. Otherwise, by summing up all the coefficients, we would get

$$\Delta(S) - \Delta'(S) < 0$$

which contradicts the assumption that Δ and Δ' are distributions over S.

Therefore the only possibility is that all coefficients in (10) are 0, i.e., $\Delta(V_i) = \Delta'(V_i)$ for any equivalence class $V_i \in S/\ \mathcal{R}$. It follows from Theorem 2.4 (2) that $\Delta\ \mathcal{R}^\dagger\ \Delta'$. So we have shown that $\mathcal{R}$ is indeed a bisimulation. □

Corollary 6.10 Let s and t be two states in a finite state pLTS. Then $s \sim t$ if and only if $m_{max}(s,t) = 0$.

Proof: ($\Rightarrow$) Since $\sim$ is a bisimulation, by Theorem 6.9 there exists some state-metric m such that $s \sim t$ iff $m(s,t) = 0$. By the definition of m_{max} we have $m \preceq m_{max}$. Therefore $m_{max}(s,t) \leq m(s,t) = 0$.

($\Leftarrow$) From m_{max} we construct a pseudometric m as follows.

$$m(s,t) = \begin{cases} 0 & \text{if } m_{max}(s,t) = 0 \\ 1 & \text{otherwise.} \end{cases}$$

Since m_{max} is a state-metric, it is easy to see that m is also a state-metric. Now we construct a binary relation $\mathcal{R}$ such that $\forall s, s' : s\ \mathcal{R}\ s'$ iff $m(s, s') = 0$. If follows from Theorem 6.9 that $\mathcal{R}$ is a bisimulation. If $m_{max}(s,t) = 0$, then $m(s,t) = 0$ and thus $s\ \mathcal{R}\ t$. Therefore we have the required result $s \sim t$ because $\sim$ is the largest bisimulation. □

7 Algorithmic characterisation

In this section we propose an "on the fly" algorithm for checking if two states in a finitary pLTS are bisimilar.

An important ingredient of the algorithm is to check if two distributions are related by a lifted relation. Fortunately, Theorem 3.7 already provides us a method for deciding whether $\Delta\ \mathcal{R}^\dagger\ \Theta$, for two given distributions Δ, Θ and a relation $\mathcal{R}$. We construct the network $\mathcal{N}(\Delta, \Theta, \mathcal{R})$ and compute the maximum flow with well-known methods, as sketched in Algorithm 1.

Algorithm 1 Check$(\Delta, \Theta, \mathcal{R})$

Input: A nonempty finite set S, distributions
    $\Delta, \Theta \in \mathcal{D}(S)$ and $\mathcal{R} \subseteq S \times S$
Output: If $\Delta\ \mathcal{R}^\dagger\ \Theta$ then "yes" else "no"
Method:
    Construct the network $\mathcal{N}(\Delta, \Theta, \mathcal{R})$
    Compute the maximum flow F in $\mathcal{N}(\Delta, \Theta, \mathcal{R})$
    If $F < 1$ then return "no" else "yes".

As shown in [4], computing the maximum flow in a network can be done in time $O(n^3/\log n)$ and space $O(n^2)$, where n is the number of nodes in the network. So we immediately have the following result.

Lemma 7.1 The test whether $\Delta\ \mathcal{R}^\dagger\ \Theta$ can be done in time $O(n^3/\log n)$ and space $O(n^2)$. □

We now present a bisimilarity-checking algorithm by adapting the algorithm proposed in [39] for value-passing processes, which in turn was inspired by [22].

The main procedure in the algorithm is **Bisim**(s, t). It starts with the initial state pair (s, t), trying to find the smallest bisimulation relation containing the pair by matching transitions from each pair of states it reaches. It uses three auxiliary data structures:

- $NotBisim$ collects all state pairs that have already been detected as not bisimilar.
- $Visited$ collects all state pairs that have already been visited.
- *Assumed* collects all state pairs that have already been visited and assumed to be bisimilar.

The core procedure, **Match**, is called from function **Bis** inside the main procedure **Bisim**. Whenever a new pair of states is encountered it is inserted into $Visited$. If two states fail to match each other's transitions then they are not bisimilar and the pair is added to $NotBisim$. If the current state pair has been visited before, we check whether it is in $NotBisim$. If this is the case, we return $false$. Otherwise, a loop has been detected and we make assumption that the two states are bisimilar, by inserting the pair into *Assumed*, and return *true*. Later on, if we find that the two states are not bisimilar after finishing searching the loop, then the assumption is wrong, so we first add the pair into $NotBisim$ and then raise the exception $WrongAssumption$, which forces the function **Bis** to run again, with the new information that the two states in this pair are not bisimilar. In this case, the size of $NotBisim$ has been increased by at least one. Hence, **Bis** can only be called for finitely many times. Therefore, the procedure **Bisim**(s, t) will terminate. If it

Algorithm 2 Bisim(s,t)

Bisim$(s,t) = \{$
$NotBisim := \{\}$
fun Bis(s,t)=$\{$
  $Visited := \{\}$
  $Assumed := \{\}$
  Match$(s,t)\}$
$\}$ **handle** $WrongAssumption \Rightarrow$ **Bis**(s,t)
return Bis(s,t)

Match$(s,t) =$
$Visited := Visisted \cup \{(s,t)\}$
$b = \bigwedge_{a \in A}$ **MatchAction**(s,t,a)
if $b = false$ **then**
  $NotBisim := NotBisim \cup \{(s,t)\}$
  if $(s,t) \in Assumed$ **then**
    raise $WrongAssumption$
  end if
end if
return b

MatchAction$(s,t,a) =$
for all $s \xrightarrow{a} \Delta_i$ **do**
  for all $t \xrightarrow{a} \Theta_j$ **do**
    $b_{ij} =$ **MatchDistribution**(Δ_i, Θ_j)
  end for
end for
return $(\bigwedge_i(\bigvee_j b_{ij})) \underline{\wedge (\bigwedge_j(\bigvee_i b_{ij}))}$

MatchDistribution$(\Delta, \Theta) =$
Assume $\lceil \Delta \rceil = \{s_1, ..., s_n\}$ and $\lceil \Theta \rceil = \{t_1, ..., t_m\}$
$\mathcal{R} := \{(s_i, t_j) \mid$ **Close**$(s_i, t_j) = true\}$
return Check$(\Delta, \Theta, \mathcal{R})$

Close$(s,t) =$
if $(s,t) \in NotBisim$ **then**
  return $false$
else if $(s,t) \in Visited$ **then**
  $Assumed := Assumed \cup \{(s,t)\}$
  return $true$
else
  return Match(s,t)
end if

returns $true$, then the set $(Visited - NotBisim)$ constitutes a bisimulation relation containing the pair (s, t).

The main difference from the algorithm of checking non-probabilistic bisimilarity in [39] is the introduction of the procedure **MatchDistribution**(Δ, Θ), where we approximate $\sim$ by a binary relation $\mathcal{R}$ which is coarser than $\sim$ in general, and we check the validity of $\Delta\ \mathcal{R}^\dagger\ \Theta$. If $\Delta\ \mathcal{R}^\dagger\ \Theta$ does not hold, then $\Delta \sim^\dagger \Theta$ is invalid either and **MatchDistribution**(Δ, Θ) returns *false* correctly. Otherwise, the two distributions Δ and Θ are considered equivalent with respect to $\mathcal{R}$ and we move on to match other pairs of distributions. The correctness of the algorithm is stated in the following theorem.

Theorem 7.2 Given two states s_0 and t_0 in a finitary pLTS, the function **Bisim**(s_0, t_0) terminates, and it returns *true* if and only if $s_0 \sim t_0$.

Proof: Let **Bis**$_i$ stand for the i-th execution of the function **Bis**. Let $Assumed_i$ and $NotBisim_i$ be the set $Assumed$ and $NotBisim$ at the end of **Bis**$_i$. When **Bis**$_i$ is finished, either a $WrongAssumption$ is raised or no $WrongAssumption$ is raised. In the former case, $Assumed_i \cap NotBisim_i \neq \emptyset$; in the latter case, the execution of the function **Bisim** is completed. From function **Close** we know that $Assumed_i \cap NotBisim_{i-1} = \emptyset$. Now it follows from the simple fact $NotBisim_{i-1} \subseteq NotBisim_i$ that $NotBisim_{i-1} \subset NotBisim_i$. Since we are considering finitary pLTSs, there is some j such that $NotBisim_{j-1} = NotBisim_j$, when all the non-bisimilar state pairs reachable from s_0 and t_0 have been found and **Bisim** must terminate.

For the correctness of the algorithm, we consider the relation $\mathcal{R}_i = Visited_i - NotBisim_i$, where $Visited_i$ is the set $Visited$ at the end of **Bis**$_i$. Let **Bis**$_k$ be the last execution of **Bis**. For each $i \leq k$, the relation $\mathcal{R}_i$ can be regarded as an approximation of $\sim$, as far as the states appeared in $\mathcal{R}_i$ are concerned. Moreover, $\mathcal{R}_i$ is a coarser approximation because if two states s, t are revisited but their relation is unknown, they are assumed to be bisimilar. Therefore, if **Bis**$_k(s_0, t_0)$ returns $false$, then $s_0 \not\sim t_0$. On the other hand, if **Bis**$_k(s_0, t_0)$ returns $true$, then $\mathcal{R}_k$ constitutes a bisimulation relation containing the pair (s_0, t_0). This follows because **Match**$(s_0, t_0) = true$ which basically means that whenever $s\ \mathcal{R}_k\ t$ and $s \xrightarrow{a} \Delta$ there exists some transition $t \xrightarrow{a} \Theta$ such that **Check**$(\Delta, \Theta, \mathcal{R}_k) = true$, i.e. $\Delta\ \mathcal{R}_k^\dagger\ \Theta$. Indeed, this rules out the possibility that $s_0 \not\sim t_0$ as otherwise we would have $s_0 \not\sim_\omega t_0$ by Proposition 4.4, that is $s_0 \not\sim_n t_0$ for some $n > 0$. The latter means that some transition $s_0 \xrightarrow{a} \Delta$ exists such that for all $t_0 \xrightarrow{a} \Theta$ we have $\Delta \not\sim_{n-1}^\dagger \Theta$, or symmetrically with the roles of s_0 and t_0 exchanged, i.e. Δ and Θ can be distinguished at level n, so a contradiction arises. $\square$

Below we consider the time and space complexities of the algorithm.

Theorem 7.3 Let s and t be two states in a pLTS with n states in total. The function **Bisim**(s, t) terminates in time $O(n^7/\log n)$ and space $O(n^2)$.

Proof: The number of state pairs is bounded by n^2. In the worst case, each execution of the function **Bis**(s, t) only yields one new pair of states that are not bisimilar. The number of state pairs examined in the first execution of **Bis**(s, t) is at most $O(n^2)$, in the second execution is at most $O(n^2 - 1)$, $\cdots$. Therefore, the total number of state pairs examined is at most $O(n^2 + (n^2 - 1) + \cdots + 1) = O(n^4)$. When a state pair (s, t) is examined, each transition of s is compared with all transitions of t labelled with the same action. Since the pLTS is finitely branching, we could assume that each state has at most c outgoing transitions. Therefore, for each state pair, the number of comparisons of transitions is bound by c^2. As a comparison of two transitions calls the function **Check** once, which requires time $O(n^3/\log n)$ by Lemma 7.1. As a result, examining each state pair takes time $O(c^2n^3/\log n)$. Finally, the worst case time complexity of executing $\mathbf{Bisim}(s,t)$ is $O(n^7/\log n)$.

The space requirement of the algorithm is easily seen to be $O(n^2)$, in view of Lemma 7.1. □

Remark 7.4 With mild modification, the above algorithm can be adapted to check probabilistic similarity. We simply remove the underlined part in the function **MatchAction**; the rest of the algorithm remains unchanged. Similar to the analysis in Theorems 7.2 and 7.3, the new algorithm can be shown to correctly check probabilistic similarity over finitary pLTSs; its worst case time and space complexities are still $O(n^7/\log n)$ and $O(n^2)$, respectively.

8 Conclusion

To define behavioural equivalences or preorders for probabilistic processes often involves a lifting operation that turns a binary relation $\mathcal{R}$ on states into a relation $\mathcal{R}^\dagger$ on distributions over states. We have shown that several different proposals for lifting relations can be reconciled. They are nothing more than different forms of essentially the same lifting operation. More interestingly, we have discovered that this lifting operation corresponds well to the Kantorovich metric, a fundamental concept used in mathematics to lift a metric on states to a metric on distributions over states, besides the fact the lifting operation is related to the maximum flow problem in optimisation theory.

The lifting operation leads to a neat notion of probabilistic bisimulation, for which we have provided logical, metric, and algorithmic characterisations.

1. We have introduced a probabilistic choice modality to specify the behaviour of distributions of states. Adding the new modality to the Hennessy-Milner logic and the modal mu-calculus results in an adequate and an expressive logic w.r.t. probabilistic bisimilarity, respectively.

2. Due to the correspondence of the lifting operation and the Kantorovich metric, bisimulations can be naturally characterised as pseudometrics which are post-fixed points of a monotone function, and bisimilarity as the greatest post-fixed point of the funciton.

3. We have presented an "on the fly" algorithm to check if two states in a finitary pLTS are bisimilar. The algorithm is based on the close relationship between the lifting operation and the maximum flow problem.

In the belief that a good scientific concept is often elegant, even seen from different perspectives, we consider the lifting operation and probabilistic bisimulation as two concepts in probabilistic concurrency theory that are formulated in the right way.

References

[1] C. Baier. On the algorithmic verification of probabilistic systems, 1998. Habilitation Thesis, Universitt Mannheim.

[2] C. Baier, B. Engelen, and M. E. Majster-Cederbaum. Deciding bisimilarity and similarity for probabilistic processes. *Journal of Computer and System Sciences*, 60(1):187–231, 2000.

[3] E. Bandini and R. Segala. Axiomatizations for probabilistic bisimulation. In *Proceedings of the 28th International Colloquium on Automata, Languages and Programming*, volume 2076 of *Lecture Notes in Computer Science*, pages 370–381. Springer, 2001.

[4] J. Cheriyan, T. Hagerup, and K. Mehlhorn. Can a maximum flow be computed on O(nm) time? In *Proceedings of the 17th International Colloquium on Automata, Languages and Programming*, volume 443 of *Lecture Notes in Computer Science*, pages 235–248. Springer, 1990.

[5] I. Christoff. Testing equivalences and fully abstract models for probabilistic processes. In *Proceedings the 1st International Conference on Concurrency Theory*, volume 458 of *Lecture Notes in Computer Science*, pages 126–140. Springer, 1990.

[6] R. Cleaveland, S. P. Iyer, and M. Narasimha. Probabilistic temporal logics via the modal mu-calculus. *Theoretical Computer Science*, 342(2-3):316–350, 2005.

[7] Y. Deng, T. Chothia, C. Palamidessi, and J. Pang. Metrics for action-labelled quantitative transition systems. *Electronic Notes in Theoretical Computer Science*, 153(2):79–96, 2006.

[8] Y. Deng and W. Du. Probabilistic barbed congruence. *Electronic Notes in Theoretical Computer Science*, 190(3):185–203, 2007.

[9] Y. Deng and W. Du. Kantorovich metric in computer science: A brief survey. *Electronic Notes in Theoretical Computer Science*, 353(3):73–82, 2009.

[10] Y. Deng and W. Du. A local algorithm for checking probabilistic bisimilarity. In *Proceedings of the 4th International Conference on Frontier of Computer Science and Technology*, pages 401–407. IEEE Computer Society, 2009.

[11] Y. Deng and R. van Glabbeek. Characterising probabilistic processes logically. In *Proceedings of the 17th International Conference on Logic for Programming, Artificial Intelligence and Reasoning*, volume 6397 of *Lecture Notes in Computer Science*, pages 278–293. Springer, 2010.

[12] Y. Deng, R. van Glabbeek, M. Hennessy, and C. Morgan. Testing finitary probabilistic processes (extended abstract). In *Proceedings of the 20th International Conference on Concurrency Theory*, volume 5710 of *Lecture Notes in Computer Science*, pages 274–288. Springer, 2009.

[13] Y. Deng, R. van Glabbeek, M. Hennessy, and C. C. Morgan. Characterising testing preorders for finite probabilistic processes. *Logical Methods in Computer Science*, 4(4):1–33, 2008.

[14] Y. Deng, R. van Glabbeek, M. Hennessy, C. C. Morgan, and C. Zhang. Remarks on testing probabilistic processes. *Electronic Notes in Theoretical Computer Science*, 172:359–397, 2007.

[15] Y. Deng, R. van Glabbeek, C. C. Morgan, and C. Zhang. Scalar outcomes suffice for finitary probabilistic testing. In *Proceedings of the 16th European Symposium on Programming*, volume 4421 of *Lecture Notes in Computer Science*, pages 363–378. Springer, 2007.

[16] J. Desharnais, A. Edalat, and P. Panangaden. A logical characterization of bisimulation for labelled Markov processes. In *Proceedings of the 13th Annual IEEE Symposium on Logic in Computer Science*, pages 478–489. IEEE Computer Society Press, 1998.

[17] J. Desharnais, R. Jagadeesan, V. Gupta, and P. Panangaden. Metrics for labeled Markov systems. In *Proceedings of the 10th International Conference on Concurrency Theory*, volume 1664 of *Lecture Notes in Computer Science*, pages 258–273. Springer-Verlag, 1999.

[18] J. Desharnais, R. Jagadeesan, V. Gupta, and P. 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. IEEE Computer Society, 2002.

[19] J. Desharnais, R. Jagadeesan, V. Gupta, and P. Panangaden. Metrics for labelled markov processes. *Theoretical Computer Science*, 318(3):323–354, 2004.

[20] J. Desharnais, F. Laviolette, and M. Tracol. Approximate analysis of probabilistic processes: Logic, simulation and games. In *Proceedings of the 5th International Conference on the Quantitative Evaluaiton of Systems*, pages 264–273. IEEE Computer Society, 2008.

[21] S. Even. *Graph Algorithms.* Computer Science Press, 1979.

[22] J.-C. Fernandez and L. Mounier. Verifying bisimulations "on the fly". In *Proceedings of the 3rd International Conference on Formal Description Techniques for Distributed Systems and Communication Protocols*, pages 95–110. North-Holland, 1990.

[23] N. Ferns, P. Panangaden, and D. Precup. Metrics for finite Markov decision processes. In *Proceedings of the 20th Conference in Uncertainty in Artificial Intelligence*, pages 162–169. AUAI Press, 2004.

[24] N. Ferns, P. Panangaden, and D. Precup. Metrics for Markov decision processes with infinite state spaces. In *Proceedings of the 21st Conference in Uncertainty in Artificial Intelligence*, pages 201–208. AUAI Press, 2005.

[25] A. Giacalone, C.-C. Jou, and S. A. Smolka. Algebraic reasoning for probabilistic concurrent systems. In *Proceedings of IFIP TC2 Working Conference on Programming Concepts and Methods*, 1990.

[26] A. L. Gibbs and F. E. Su. On choosing and bounding probability metrics. *International Statistical Review*, 70(3):419–435, 2002.

[27] H. Hansson and B. Jonsson. A calculus for communicating systems with time and probabilities. In *Proceedings of IEEE Real-Time Systems Symposium*, pages 278–287. IEEE Computer Society Press, 1990.

[28] M. Hennessy and R. Milner. Algebraic laws for nondeterminism and concurrency. *Journal of the ACM*, 32(1):137–161, 1985.

[29] M. Huth and M. Kwiatkowska. Quantitative analysis and model checking. In *Proceedings of the 12th Annual IEEE Symposium on Logic in Computer Science*, pages 111–122. IEEE Computer Society, 1997.

[30] B. Jonsson, C. Ho-Stuart, and W. Yi. Testing and refinement for nondeterministic and probabilistic processes. In *Proceedings of the 3rd International Symposium on Formal Techniques in Real-Time and Fault-Tolerant Systems*, volume 863 of *Lecture Notes in Computer Science*, pages 418–430. Springer, 1994.

[31] B. Jonsson and W. Yi. Compositional testing preorders for probabilistic processes. In *Proceedings of the 10th Annual IEEE Symposium on Logic in Computer Science*, pages 431–441. Computer Society Press, 1995.

[32] B. Jonsson and W. Yi. Testing preorders for probabilistic processes can be characterized by simulations. *Theoretical Computer Science*, 282(1):33–51, 2002.

[33] B. Jonsson, W. Yi, and K. G. Larsen. Probabilistic extensions of process algebras. In *Handbook of Process Algebra*, chapter 11, pages 685–710. Elsevier, 2001.

[34] L. Kantorovich. On the transfer of masses (in Russian). *Doklady Akademii Nauk*, 37(2):227–229, 1942.

[35] L. V. Kantorovich and G. S. Rubinshtein. On a space of totally additive functions. *Vestn Lening. Univ.*, 13(7):52–59, 1958.

[36] D. Kozen. Results on the propositional mu-calculus. *Theoretical Computer Science*, 27:333–354, 1983.

[37] K. G. Larsen and A. Skou. Bisimulation through probabilistic testing. *Information and Computation*, 94(1):1–28, 1991.

[38] K. G. Larsen and A. Skou. Compositional verification of probabilistic processes. In *Proceedings of the 3rd International Conference on Concurrency Theory*, volume 630 of *Lecture Notes in Computer Science*, pages 456–471. Springer, 1992.

[39] H. Lin. "On-the-fly" instantiation of value-passing processes. In *Proceedings of FORTE'98*, volume 135 of *IFIP Conference Proceedings*, pages 215–230. Kluwer, 1998.

[40] G. Lowe. Probabilistic and prioritized models of timed CSP. *Theoretical Computer Science*, 138:315–352, 1995.

[41] A. McIver and C. Morgan. An expectation-based model for probabilistic temporal logic. Technical Report PRG-TR-13-97, Oxford University Computing Laboratory, 1997.

[42] A. McIver and C. Morgan. Results on the quantitative mu-calculus. *ACM Transactions on Computational Logic*, 8(1), 2007.

[43] R. Milner. *Communication and Concurrency.* Prentice Hall, 1989.

[44] M. M. Mislove, J. Ouaknine, and J. Worrell. Axioms for probability and nondeterminism. *Electronic Notes in Theoretical Computer Science*, 96:7–28, 2004.

[45] G. Monge. Mémoire sur la théorie des déblais et des remblais. *Histoire de l'Academie des Science de Paris*, page 666, 1781.

[46] M. Müller-Olm. Derivation of characteristic formulae. *Electronic Notes in Theoretical Computer Science*, 18:159–170, 1998.

[47] J. B. Orlin. A faster strongly polynomial minimum cost flow algorithm. In *Proceedings of the 20th ACM Symposium on the Theory of Computing*, pages 377–387. ACM, 1988.

[48] D. Park. Concurrency and automata on infinite sequences. In *Proceedings of the 5th GI Conference*, volume 104 of *Lecture Notes in Computer Science*, pages 167–183. Springer, 1981.

[49] A. Parma and R. Segala. Logical characterizations of bisimulations for discrete probabilistic systems. In *Proceedings of the 10th International Conference on Foundations of Software Science and Computational Structures*, volume 4423 of *Lecture Notes in Computer Science*, pages 287–301. Springer, 2007.

[50] A. Pnueli. Linear and branching structures in the semantics and logics of reactive systems. In *Proceedings of the 12th International Colloquium on Automata, Languages and Programming*, volume 194 of *Lecture Notes in Computer Science*, pages 15–32. Springer, 1985.

[51] M. L. Puterman. *Markov Decision Processes.* Wiley, 1994.

[52] S. Rachev. *Probability Metrics and the Stability of Stochastic Models.* Wiley New York, 1991.

[53] R. Segala. Modeling and verification of randomized distributed real-time systems. Technical Report MIT/LCS/TR-676, PhD thesis, MIT, Dept. of EECS, 1995.

[54] R. Segala and N. Lynch. Probabilistic simulations for probabilistic processes. In *Proceedings of the 5th International Conference on Concurrency Theory*, volume 836 of *Lecture Notes in Computer Science*, pages 481–496. Springer, 1994.

[55] B. Steffen and A. Ingólfsdóttir. Characteristic formulae for processes with divergence. *Information and Computation*, 110:149–163, 1994.

[56] A. Tarski. A lattice-theoretical fixpoint theorem and its application. *Pacific Journal of Mathematics*, 5:285–309, 1955.

[57] R. Tix, K. Keimel, and G. Plotkin. Semantic domains for combining probability and non-determinism. *Electronic Notes in Theoretical Computer Science*, 129:1–104, 2005.

[58] F. van Breugel, C. Hermida, M. Makkai, and J. Worrell. An accessible approach to behavioural pseudometrics. In *Proceedings of the 32nd International Colloquium on Automata, Languages and Programming*, volume 3580 of *Lecture Notes in Computer Science*, pages 1018–1030. Springer, 2005.

[59] F. van Breugel, C. Hermida, M. Makkai, and J. Worrell. Recursively defined metric spaces without contraction. *Theoretical Computer Science*, 380(1-2):143–163, 2007.

[60] F. van Breugel, B. Sharma, and J. Worrell. Approximating a behavioural pseudometric without discount for probabilistic systems. In *Proceedings of the 10th International Conference on Foundations of Software Science and Computational Structures*, volume 4423 of *Lecture Notes in Computer Science*, pages 123–137. Springer, 2007.

[61] F. van Breugel and J. Worrell. An algorithm for quantitative verification of probabilistic transition systems. In *Proceedings of the 12th International Conference on Concurrency Theory*, volume 2154 of *Lecture Notes in Computer Science*, pages 336–350. Springer, 2001.

[62] F. van Breugel and J. Worrell. Towards quantitative verification of probabilistic transition systems. In *Proceedings of the 28th International Colloquium on Automata, Languages and Programming*, volume 2076 of *Lecture Notes in Computer Science*, pages 421–432. Springer, 2001.

[63] F. van Breugel and J. Worrell. A behavioural pseudometric for probabilistic transition systems. *Theoretical Computer Science*, 331(1):115–142, 2005.

[64] F. van Breugel and J. Worrell. Approximating and computing behavioural distances in probabilistic transition systems. *Theoretical Computer Science*, 360(1-3):373–385, 2006.

[65] A. Vershik. Kantorovich metric: Initial history and little-known applications. *Journal of Mathematical Sciences*, 133(4):1410–1417, 2006.

[66] C. Villani. *Topics in Optimal Transportation*, volume 58 of *Graduate Studies in Mathematics*. American Mathematical Society, 2003.

[67] W. Yi and K. G. Larsen. Testing probabilistic and nondeterministic processes. In *Proceedings of the IFIP TC6/WG6.1 12th International Symposium on Protocol Specification, Testing and Verification*, volume C-8 of *IFIP Transactions*, pages 47–61. North-Holland, 1992.

[68] L. Zhang, H. Hermanns, F. Eisenbrand, and D. N. Jansen. Flow faster: Efficient decision algorithms for probabilistic simulations. *Logical Methods in Computer Science*, 4(4:6), 2008.