Weak Bisimulation is Sound and Complete for PCTL*

Josée Desharnais*
Département d'Informatique
Université Laval
Québec, Canada, G1K 7P4

Vineet Gupta
Google Inc.
2400 Bayshore Parkway
Mountain View CA 94043 USA

Radha Jagadeesan†
School of CTI
De Paul University
Chicago, Illinois 60604-2287 USA

Prakash Panangaden‡
School of Computer Science
McGill University
Montréal, Canada, H3A 2A7

November 22, 2009

Abstract

We investigate weak bisimulation of probabilistic systems in the presence of nondeterminism, i.e. labelled concurrent Markov chains (LCMC) with silent transitions. We develop an approach based on allowing convex combinations of computations, similar to Segala and Lynch's use of randomized schedulers.

The definition of weak bisimulation destroys the additivity property of the probability distributions, yielding instead *capacities*. The mathematics behind capacities naturally captures the intuition that when we deal with nondeterminism we must work with bounds on the possible probabilities rather than with their exact values.

Our analysis leads to three new developments:

- We identify a characterization of "image finiteness" for countable-state systems and present a new definition of weak bisimulation for these LCMCs. We prove that our definition coincides with that of Philippou, Lee and Sokolsky for finite state systems.

*Research supported by NSERC.

†Research supported by NSF.

‡Research supported in part by NSERC and MITACS.

- We show that bisimilar states have matching computations. The notion of matching involves *convex combinations* of transitions.
- We study a minor variant of the probabilistic logic pCTL* — the variation arises from an extra path formula to address action labels. We show that bisimulation is sound and complete for this variant of pCTL*.

This is an extended complete version of a paper that was presented at CONCUR 2002.

1 Introduction

The main object of this paper is to study systems that combine probability, concurrency and nondeterminism. We focus in particular on weak bisimulation. The importance of weak bisimulation comes from the need for abstraction. In order to construct larger programs from smaller programs one works with the composition mechanisms of the language. When doing so it is necessary to hide internal actions and work with weak rather than strong bisimulation.

In the purely probabilistic context, the study of strong bisimulation was initiated by Larsen and Skou [LS91], and an equivalence notion was developed, similar to the queuing theory notion of "lumpability" [KS60]. This theory has been extended to continuous state spaces and continuous distributions [BDEP97, DEP98, DGJP02a] and, in the discrete setting, to weak bisimulation [BH97].

The study of weak bisimulation for systems with probability and nondeterminism is sensitive to the underlying model. The two principal models are the *alternating* model [Han94] - where there are two disjoint classes of states, probabilistic states and nondeterministic states - and the nonalternating model [SL94]. Weak bisimulation for finite-state systems in the alternating model with distinct nondeterministic and probabilistic states was defined by Philippou, Lee and Sokolsky [PLS00] whereas weak bisimulation for the nonalternating model was studied by Segala and Lynch [SL94]. Our study is set in the context of the alternating model and follows [PLS00].

We explore the subtle consequences of the benign looking definitions of [PLS00]. The most significant change from ordinary probability theory is that the "probabilities" no longer satisfy additivity[1]. In the presence of nondeterminism, we are describing a *set* of probability distributions $\{Q_i\}$ for a given state s and a given weak transition label a. The "probabilities"

[1] Additivity: P is additive if for disjoint sets A, B, $P(A \cup B) = P(A) + P(B)$.

ascribed by [PLS00] arise by majorizing over this set, i.e. $P(s, a, E)$, the probability of reaching a set of states E from state s on weak transition labelled a, is given by $\max_i Q_i(E)$ for any subset of states E.

The second important change is that the notion of matching has changed radically. The essence of any bisimulation notion is that transitions of one process can be matched with transitions in the bisimilar process. In order to match computation paths on given weak labels one needs to take convex-linear combinations of computations. The "computations" (to be defined precisely later) now have a convex space structure. This means that the space is closed under the formation of arbitrary *convex* combinations: if $\{c_i | i \in I\}$ is a set of computations then $\sum_i \lambda_i c_i$ is also a computation where $0 \leq \lambda_i \leq 1$ and $\sum_i \lambda_i = 1$. In example 2.9 we discuss this point in detail. Essentially randomized schedulers allow one to take just such combinations. This convex structure is important and allows us to use some standard ideas of convexity: for example, the fact that the convex closure of a compact set is compact.

The three main points that we make can be summarized as follows.

- First, we generalize the definitions of [PLS00] to a large class of infinite-state systems satisfying a compactness property. Informally, compactness is a topological formalization of finite branching. In this context, compactness enables us to capture a robust notion of "image finiteness" for weak transitions that hide internal actions. The compact systems that we consider include all finite state systems including those with cycles.

- Second, we adapt the ideas on randomized schedulers from Segala's work on probabilistic IO automata [SL95]. On the one hand, randomized schedulers do not change the semantics: the sups that one computes are the same. From the point of view of linear algebra, one can visualize this geometrically by seeing that convex combinations do not introduce new extremal points, thus the suprema of probabilities used in [PLS00] are preserved. On the other hand, these schedulers enable us to perform a fine-grained analysis of the structure of computations in bisimilar systems. This analysis permits us to establish that bisimilar states s, t satisfy a familiar property: "for every distribution of states induced by a resolution of non-deterministic choices from s, there exists a resolution of non-deterministic choices from t that results in a matching distribution on states." We show simple examples that demonstrate that this matching property *requires* the presence of convex combinations.

- Third, we analyze the structure that arises by majorizing over a set of probability distributions. This structure is called a capacity — for our purposes, capacities are monotone functions from sets (with inclusion order) to the reals (with the usual order) that preserve sups (resp. infs) of increasing (resp. decreasing) sequences of sets. Capacities are not necessarily additive. Indeed, the capacities induced by the definitions of [PLS00] only satisfy: $P(s, a, A) + P(s, a, B) \geq P(s, a, A \cup B)$ for disjoint sets of states A, B.

  This loss of additivity has already been recognized in various situations in mathematics [Cho53, Del72, Mey66] and in economics [Sch84]. Economic studies distinguish risk, the relative probabilities of the events are known, from uncertainty, there is no unique assignment of probabilities to events, this is what computer scientists call nondeterminism. Risk is modelled using probability. The modelling of uncertainty is via a *set* of probability measures that are consistent with the known information. The structure obtained by majorizing this set of probability measures does not satisfy additivity and is a capacity. A rich theory of capacities was already available for our use. This theory meshes very well with the idea that uncertainty in probability distributions should be captured by giving upper and lower bounds on probabilities and expectation values. We show that the key equations that are demanded by this theory are met by the capacities that arise in the context of weak bisimulation.

Soundness and Completeness of weak bisimulation for probabilistic logics. A fundamental application of these ideas, and the original impetus for these investigations, is the analysis of soundness and completeness of bisimulation for probabilistic logics. We study a minor variant of the probabilistic logic pCTL* [dA97] – the variation arises from an extra path formula to address action labels – and is inspired by the variants of probabilistic logics that deal with action labels [SL94, Han94] . We show that bisimulation is sound and complete for this variant of pCTL*. Our soundness and completeness proofs rely crucially on all three developments identified above.

Organization of this paper. The rest of this paper is organized as follows. First, in section 2, we review the basic definitions of the model (the "alternating model") and weak probabilistic bisimulation and associated results to make the paper self-contained. Section 3 identifies the class of countable systems to which our study applies. In section 4 we show that our definition is equivalent to that of Philippou, Lee and Sokolsky [PLS00]. In section 5 we show that the capacities defined in the development of weak bisimulation satisfy the axioms required of capacities. Finally, in section 6, we use the machinery that has been developed to prove soundness and completeness results for the logic.

2 Background and Definitions

We begin with a review of the underlying framework — our definitions are adapted from [PLS00]. We work in the context of the "alternating model" for labelled concurrent Markov chains [Han94], labelled transition systems with non-determinism and probability.

Definition 2.1 *A labelled concurrent Markov chain (henceforth LCMC), is a tuple*
$\mathcal{K} = (K, \mathtt{Act}, \longrightarrow, k_0)$*, where*
(1) $K = K_p \cup K_n$*, a countable set, is partitioned into the probabilistic states,* K_p*, and the nondeterministic states* K_n*,* k_0 *is the start state.*
(2) $\mathtt{Act}$ *is a finite set of action symbols that contains a special action* τ*.*
(3) The transition relation $\longrightarrow = \longrightarrow_p \cup \longrightarrow_n$ *is partitioned into probabilistic and nondeterministic transitions.* $\longrightarrow_n \subseteq K_n \times \mathtt{Act} \times K_p$ *is* image-finite, *i.e. for each* $s \in K_n$ *and* $a \in \mathtt{Act}$*, the set* $\{s' \in K_p \mid s \xrightarrow{a} s'\}$ *is finite.* $\longrightarrow_p \subseteq K_p \times (0,1] \times K_n$ *satisfies that for each* $s \in K_p$*,* $\sum_{(s,\pi,t)\in\longrightarrow_p} \pi \leq 1$*.*

A state is either probabilistic - in which case the transitions are probabilistic and unlabelled - or nondeterministic, in which case the transitions are finite-branching and labelled, possibly by a τ transition. We allow subprobability distributions so the probabilities need not add up to 1. In particular, we allow some probabilistic states to be dead states so that the transition probabilities associated with such a state are zero. The probabilistic branching can be countable at a state. In this paper, we will work with countable state systems.

Every probabilistic state s induces a probability distribution Q on K_n given by $Q(t) = \sum_{(s,\pi,t)\in\longrightarrow_p} \pi$ for every $t \in K$, and for $E \subseteq K$, $Q(E) = \sum_{t\in E} Q(t)$. We sometimes write $s \to_p Q$ to emphasize this distribution. Indeed, one can take the view that the "real" states are the nondeterministic states and the probabilistic states are really just names for certain probability distributions.

The LCMC model does not need to be strictly alternating. One can work with a model that only restricts states to be either purely nondeterministic or purely probabilistic and does not enforce strict alternation. We discuss this variant at the end of this section.

Every sequence, say σ, of transitions has as an associated probability $\mathtt{prob}(\sigma)$, obtained by multiplying the probabilities occurring on the path. We attribute 1 to a nondeterministic transition in a path, and multiply together probabilities of all the probabilistic transitions.

Similarly, every sequence σ of transitions has an associated weak sequence of labels $\mathtt{Weak}(\sigma) \in (\mathtt{Act} - \{\tau\})^*$, obtained by removing the labels of τ-transitions. Nondeterministic transitions with label τ and probabilistic transitions do not contribute to the weak label. We will say that a path of τ transitions and probabilistic transitions has weak label ε.

We define *computations* of an LCMC as transition trees obtained by unfolding the LCMC from the root, resolving the nondeterministic choices (i.e. each nondeterministic state has at most one transition coming out of it) and taking either all probabilistic choices at a probabilistic state or none. A computation can thus be viewed as a purely (sub)probabilistic labelled Markov chain. We refer to the set of all the probabilistic transitions from a probabilistic state as a *fan*.

Definition 2.2 *A* computation *of an LCMC is a (possibly infinite) subtree of the tree obtained by partially unfolding the LCMC. In a computation every nondeterministic state has at most one transition coming out of it and if a probabilistic transition is included then the entire fan of that probabilistic transition is included.*

We are interested in transitions with particular weak labels.

Definition 2.3 *Let $\mathcal{K}$ be a LCMC, $a \in \mathtt{Act}$. An a-computation from $s \in K$ is a computation such that every path from the root has weak label a or ε.*

It may seem peculiar to allow an a-computation to have paths labelled by ε. This is done to allow for a computation where the a transition has not happened yet (or may never happen). For example, any a-computation of state s of Figure 1(b) must include the infinite path $(sw)^\omega$ which has label ε. However, when we associate probability distributions with computations we will not count the paths labelled with ε, we insist that the paths that contribute to the distribution have weak label a.

Each computation induces a distribution on its leaf states in the standard way — the probability of a leaf node is the probability of the (unique) path going to it. We actually use a somewhat looser correspondence between computations and distributions. We allow many distributions to be induced by a given computation; the requirement of matching is weakened to an inequality.

Definition 2.4 *Let $\mathcal{K}$ be a LCMC, $s \in K$, and let Q be a distribution on states.*
We write $s \overset{a}{\Rightarrow} Q$, and we call it a basic transition*, if there is an a-computation such that for all $s_i \in K$, $Q(s_i) \leq \sum_\sigma \mathtt{prob}(\sigma)$ where the summation is taken over paths σ with weak label a that start in s and end in the leaf s_i.*

Note that we have not required equality here. We are saying that a weak transition $s \overset{a}{\Rightarrow} Q$ means that the a-computation starting from s produces a distribution than *dominates* Q rather than exactly matching Q. This will allow greater flexibility with manipulating weak transitions especially when we form convex combinations, for example, in the proof of Lemma 3.3 below.

We extend this notation to convex[2] combinations of distributions.

Definition 2.5 *Let $s_i \overset{a}{\Rightarrow} Q_i$ and let $\sum_i \lambda_i \leq 1$, where all $\lambda_i \geq 0$. We define the notation: $s_i \overset{a}{\Rightarrow} (\sum_i \lambda_i Q_i)$ to stand for the transition to the* convex sum of distributions.

Such a transition can be viewed as the "weighted superposition" of the transitions $s_i \overset{a}{\Rightarrow} Q_i$. Note that $s \overset{a}{\Rightarrow} [\lambda \times Q_1 + (1-\lambda) \times Q_2]$ is reminiscent of the randomized schedulers [SL95].

Transitions from states to distributions as above are one way to the definition of bisimulation. Another way is through transitions from states to sets of states, which is how strong bisimulation is defined for labelled Markov processes in [BDEP97, DGJP02a]. The "probability" from a state s to a subset of states via a path with weak label a is defined by taking the supremum over all possible a-computations.

Definition 2.6 *Let $\mathcal{K}$ be a LCMC, $s \in K, E \subseteq K$. Then, the probability of going from s to $E \subseteq K$ via a, denoted by $P(s,a,E)$, is defined as:*

$$P(s,a,E) = \sup\{Q(E) \mid s \overset{a}{\Rightarrow} Q\}.$$

The supremum in this definition is the source of the subtlety of weak bisimulation: $P(s,a,.)$ does not satisfy additivity. The two following examples show the importance of taking the supremum.

[2] Strictly speaking, these are not convex combinations since we have $\sum_i \lambda_i \leq 1$ rather than $= 1$.

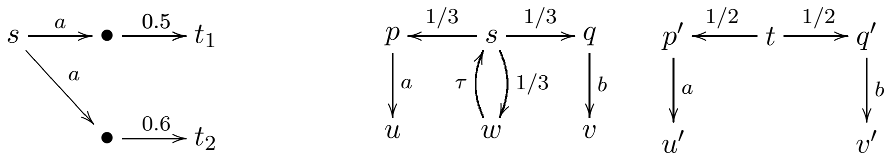


Figure 1: (a) Additivity Fails (b) Matching with linear combinations

Example 2.7 Consider the transition system in Figure 1(a), where unnecessary probabilistic transitions have been elided. We have $P(s,a,\{t_1\}) = 0.5$, $P(s,a,\{t_2\}) = 0.6$, $P(s,a,\{t_1,t_2\}) = 0.6$. Thus additivity does not hold, and we must take the sup over all computations in the definition of $P(s,a,E)$.

Example 2.8 Consider the transition system on the left of Figure 1(b). Then

$$P(s,a,\{u\}) = \sum_{i\geq 1}(\frac{1}{3})^i = 1/2.$$

The next example shows the importance of allowing linear combinations when matching computations with given weak labels.

Example 2.9 Consider the transition systems of Figure 1(b). Intuitively we would like to say that the states s and t are weakly bisimilar. We would also like to say p, p' and q, q' are weakly bisimilar.

The probability of starting from s and reaching u on a weak a label is $1/2$ and the same is true for reaching u' from t. Note that we need to sum over all possible paths that include the τ-loop if we want to get the answer $1/2$ starting from s. Thus the a-computation from t that includes u' gives a probability of $1/2$ to u' and can be matched by the infinite computation from s that loops infinitely through w and gives probability $1/2$ to u. However, we have absolutely no way of matching the distribution induced by the computation including only one step from s. Indeed, this computation induces the distribution that gives probability $1/3$ to each one of u, w and v. The only way to match it is to take a linear combination, namely the distribution δ_t induced by the trivial computation consisting only of state t, and the distribution P induced by the one-step computation. The required combination is thus $1/3 \times \delta_t + 2/3 \times P$.

We are now ready to define weak bisimulation. Given an equivalence relation R, we say a set E is R-closed if $E = R(E) := \{s \mid \exists t \in E \text{ such that } tRs\}$ and we use $[u]_R$ to stand for the equivalence class of a state u.

Definition 2.10 *An equivalence relation R on K is a weak bisimulation if for all $s, t \in K$ such that $s\ R\ t$ and all R-closed $E \subseteq K$, we have:*

$$(\forall a \in \mathtt{Act})\ [P(s, a, E) = P(t, a, E)].$$

There is a maximum weak bisimulation, denoted by $\approx$. We write $[u]$ for the bisimulation class of the state u.

A LCMC $\mathcal{K}$ is *bisimulation collapsed* if each bisimulation equivalence class is a singleton.

The equational laws supported by this definition extend the usual ones for nondeterministic labelled transition systems or purely probabilistic transition systems. Indeed, the usual relations that witness the bisimulation are carried over essentially unchanged, for example, $\tau.\mathcal{K} \approx \mathcal{K}$, and unfolding a LCMC yields a weakly bisimilar system. See [BS01] for a full axiomatization of equational laws for finite processes (without loops, so the transition system is a tree).

We present a second definition of bisimulation which is similar to the one found in the non-probabilistic setting. It will be shown to be equivalent to the one above in Section 4 for *compact* LCMCs.

Proposition 2.11 *An equivalence relation R on K is a weak bisimulation iff for all $s, t \in K$ such that $s\ R\ t$ we have:*
if $s \overset{a}{\Rightarrow} Q$, there exists $t \overset{a}{\Rightarrow} Q'$ such that for all states u: $Q([u]_R) = Q'([u]_R)$.

2.1 Minor extensions to the model

The LCMC model does not need to be strictly alternating. One can work with a model that restricts states to be either purely nondeterministic or purely probabilistic but with no transitions from probabilistic states to probabilistic states. Any such transition system $\mathcal{U} = (U, \mathtt{Act}, \longrightarrow, u_0)$ has a (weak) bisimulation-preserving translation into $\mathcal{K} = (K, \mathtt{Act}, \longrightarrow, k_0)$, a strictly alternating transition system as follows. The states $K = U^p \cup U^n$ are a disjoint union of two copies of the states of U. For all $s \in U$ such that s has only nondeterministic transitions, define $s^p \xrightarrow{1} s^n$ and $s^n \xrightarrow{a} t^p$ if $s \xrightarrow{a} t$ in $\mathcal{U}$. Similarly, for all $s \in U$ such that s has only probabilistic transitions, define $s^n \xrightarrow{\tau} s^p$ and $s^p \xrightarrow{\pi} t^n$ if $s \xrightarrow{\pi} t$ in $\mathcal{U}$. There is clearly a weak bisimulation relating $\mathcal{U}$ and $\mathcal{K}$.

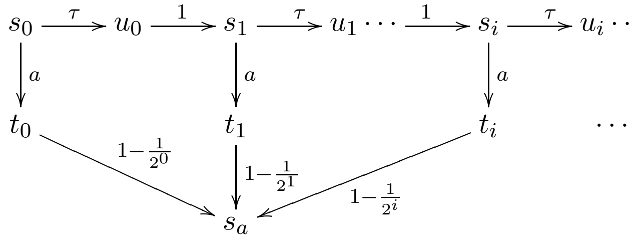
3 The compactness condition

Image finiteness plays a crucial role in the study of bisimulation in nondeterministic systems [HM85]. In the case of fully probabilistic systems and strong bisimulation it turns out – rather surprisingly – not to play a role [DEP02] despite the fact that the initial results in this subject used a strong finite branching condition [LS91].

However, here we have a combination of probability and nondeterminism and we need to control the branching. We consider countable-state LCMCs that satisfy a compactness condition. Intuitively speaking, the compactness condition can be viewed as the right generalization of "image-finiteness" for countable state LCMCs in the context of weak transitions that hide τ-labels. This compactness condition allows us to show the coincidence with the definitions of Philippou, Lee and Sokolsky [PLS00].

We first consider some preliminary motivation for considering such a condition. In general, it is not the case – even for finitely branching systems – that there is a single computation that attains the supremum of definition 2.6.

Example 3.1 *Let $\mathcal{K}$ be the LCMC described by the following diagram.*



Let $\mathcal{K}$ be an LCMC with nondeterministic states $\{s_a\} \cup \{s_i \mid i \in \mathbb{N}\}$ and probabilistic states $\{u_i, t_i \mid i \in \mathbb{N}\}$. s_0 is its start state. The state s_a has no transitions. The state s_i has two transitions, one is labelled τ to u_i, the other is labelled a to t_i. t_i has a probability $1 - \frac{1}{2^i}$ transition to s_a while u_i has a probability 1 transition to s_{i+1}. Clearly, $P(s_0, a, \{s_a\}) = 1$, but there is no single computation to witness this.

We diagnose the reason as the infinite (weak) branching at the state s_0. We now identify a large class of countable systems the class of systems that we will work with. Intuitively, this is a "compactness" condition that captures the essence of a "finite weak branching" requirement.

We begin by recalling some basic definitions from topology which can be skipped by a knowledgeable reader. In a metric space (X, d) we say that a sequence $\{x_i | i \in \mathbb{N}\}$ *converges* to x if for every $\epsilon > 0$ there is an n such that for every $k \geq n$, $d(x_k, x) < \epsilon$. We say that p is a *limit point* (or *cluster point* or *accumulation point*) of a subset S if for every $\epsilon > 0$ there is an $x \in S$ such that $x \neq p$ and $d(x, p) < \epsilon$. In other words, every open set containing p intersects S at some point other than p itself[3]. A space is said to be *sequentially compact* if every infinite sequence has a convergent subsequence. A space is said to be *limit-point compact* if every infinite set has a limit point. A topological space is *compact* if every open cover has a finite subcover. In a metric space all three notions: compact, limit-point compact and sequentially compact, coincide.

The following definition of a metric d on distributions of states is the key to getting a handle on finite-branching.

Definition 3.2 *The metric d on distributions of the states of a LCMC $\mathcal{K}$ is defined by $d(Q_1, Q_2) = \sup_{A \subseteq K} |Q_1(A) - Q_2(A)|$.*

In this metric, any computation is the limit of finite depth computations.

Lemma 3.3 *Given any weak transition $s \overset{a}{\Rightarrow} Q$, one can find a sequence of finite-depth computations with corresponding weak transitions $s \overset{a}{\Rightarrow} Q_i$ with the Q_i distributions converging to Q in the metric d.*

Proof. Let $Q = \sum_i \lambda_i \times Q_i$, where Q_i are distributions induced by basic computation trees C_i respectively. If C_i^k is a truncation of C_i, let L^k be the set of states that are leaves of both C_i and C_i^k, and define

- $Q_i^k(u) = Q_i(u)$ if u is in L^k
- For all leaves u in C_i^k that are not leaves of C_i, assign to $Q_i^k(u)$ a probability equal to the sum of the leaves of C_i that are descendants. Since C_i is a computation, this number is guaranteed to be at most the probability of the path from s to u in C_i^k (and in C_i).

Now let C_i^k be a truncation of C_i at large enough depth so that $Q(K) - \sum_{u \in L^k} Q_i^k(u) \leq \frac{1}{2^k}$. It is clear that $d(Q_i, Q_i^k) \leq \frac{1}{2^k}$. Defining $Q^k = \sum_i \lambda_i \times Q_i^k$, we get Q^k such that: $d(Q, Q^k) \leq \frac{1}{2^k}$. ∎

[3] For this definition p may or may not be in S.

Definition 3.4 *Let $\mathcal{K}$ be an LCMC and s be a state and a any label. We say s is a-compact if the set $\{Q \mid s \overset{a}{\Rightarrow} Q\}$ is compact under metric d.*

A bisimulation collapsed LCMC $\mathcal{K}$ is compact *if all states s are a-compact for all labels a (including τ). A LCMC is* compact *if its bisimulation collapse is compact.*

The following lemma simply restates the definition of compactness for LCMCs in terms of the existence of convergent subsequences; this is by far the most useful form.

Lemma 3.5 *In a compact LCMC, for any sequence of distributions Q_i such that $s \overset{a}{\Rightarrow} Q_i$, there exists $s \overset{a}{\Rightarrow} Q$ and a subsequence Q'_j of Q_j such that:*

$$(\forall \epsilon)\ (\exists j)\ (\forall i \geq j)\ (\forall E \subseteq K)|Q(E) - Q'_i(E)| < \epsilon$$

The point is that the limiting distribution is actually attained by some computation. This is typically how compactness is used.

For labelled transition systems, the compactness condition is an image-finiteness condition. Here the probability of all paths is 1 and d is the discrete metric. So, an LTS is compact iff for all states s and all labels a, the set of states reachable on a weak transition labelled a is finite.

The definition is general enough to include all finite state LCMCs, as stated in the following theorem. The proof relies crucially on weighted combinations of computations. It builds on the idea of Example 2.9 and shows that for any state s, there is a finite set of computations rooted at s such that any computation rooted at s can be built as a weighted combination of the elements of this set. This finite set is identified using the concept of *simple* computations; a similar property is used in [PLS00].

Theorem 3.6 *All finite-state systems are compact.*

In order to prove this theorem we need a standard lemma about compactness of convex closures.

Lemma 3.7 *[[Sch66], page 71] Let S be a compact subset of a locally convex topological vector space. Then, the convex closure of S is also compact*[4].

Our use of this lemma is as follows. We view a distribution on n-states as an element of the Euclidean space $\mathbb{R}^{2^n}$ of dimension 2^n, i.e. each distribution is treated as a 2^n vector that contains the probability numbers of each subset

[4] We actually need a minor variant of this lemma since our combinations are not exactly convex combinations since we allow $\sum_i \lambda_i \leq 1$.

of states. Indeed, under this embedding, the metric d is the standard l_1 norm. Since $\mathbb{R}^{2^n}$ is locally convex, this lemma applies.

Proof.(Of the theorem) By Lemma 3.7, it suffices to show that the set of a-computations at a state s is the convex closure of a finite set $\mathcal{F}$.

Using the notation of [PLS00], we say that a basic a-computation rooted at u is *simple* if the transition chosen at a nondeterministic state depends only on the weak label of the transition from the root to the occurrence of the state. For example, suppose that we have two occurrences of a state s, say s_1, s_2, in the computation such that the weak labels of the paths from the root u to s_1 and to s_2 are the same. In this case either both have a successor or neither does, and the successors of s_1 and s_2 in the computation are the same state.

For a finite state system with finitely many labels, clearly there are only finitely many *simple* computations. We define:

$$\begin{array}{rcl} \mathcal{F}_u & = & \text{the set of simple computations from } u \\ \mathcal{F} & = & \cup_{u \in K}[\mathcal{F}_u \cup \{u \Rightarrow \mathtt{Zero}\}] \end{array}$$

where $(\forall u)$ $\mathtt{Zero}(u) = 0$. Because of the presence of the zero distributions, the set of convex combinations of $\mathcal{F}$ of computations rooted at u is given by $\{u \Rightarrow \sum_i (\lambda_i Q_i) : Q_i \in \mathcal{F}_u, \lambda_i \geq 0, \sum_i \lambda_i \leq 1\}$.

We now show that any basic computation rooted at a state s can be built as a weighted combination of computations from $\mathcal{F}$. We proceed by induction on the number of states. For each state u, we show how to eliminate "non-simple" occurrences of u. We say that a computation is "simple relative to u" if all occurrences of u in the computation make the same nondeterministic choices. We proceed in two phases.

1. We first show how to make the sub-trees rooted at an occurrence of state u "simple relative to u", i.e. all occurrences of u in this subtree make the same nondeterministic choice as the occurrence of u at the root of the subtree.

   Let this subtree be called C and let u_0 be the occurrence of the state u at the root of this subtree. Let u_i be the occurrences of u in this subtree, such that the path from u_0 to u_i has weak label ε; note that this includes u_0 itself. We will construct several basic computations C_i as follows. Let C_i be the computation rooted at u_i that (recursively) replaces the subtree at all u-descendants reachable by a ε-transition sequence by the subtree at u_i. One can visualize this by considering the result of redirecting into u_i, the incoming transition into the u_k

that are descendants of u_i with no occurrence of u on the path between u_i, u_k and then unfolding the resulting graph.

Now we construct the required computation with all the non simple nodes eliminated by taking a linear combination of the C_i. We have to combine them in such a way that the relevant probability distributions are preserved by the tree surgery just defined. Let p_i be the probability of reaching the leaves in the subtree rooted at u_i in the original computation. Then, $C = \sum \lambda_i C_i$, where $\lambda_i = p_i - \sum_k \{p_k \mid k \neq i\}$; where the k in the sum satisfies the following conditions: u_k is a descendant of u_i and there is no occurrence of u on the path between u_i and u_k. The coefficients λ_i have been adjusted to account for the leaves that have been removed in the surgery that produced the C_i from C.

2. In phase 1 we dealt with weak τ transitions. After finishing phase 1, we are left with a weighted combination of basic computations rooted at s such that all sub-trees rooted at an occurrence of state u are "simple relative to u". Consider one such basic computation C. We now show how to convert C to a weighted combination of basic computations rooted at s that are "simple relative to u".

   We perform the following for each possible non-ε weak label. Let u_i be the occurrences of u with the same weak label on the path from s to u_i and let p_i be the probability of the leaves in the subtree rooted at u_i. Let C_i be the result of replacing all the subtrees rooted at u_j by the subtree at u_i in C. Then, $C = \sum \lambda_i C_i$, where $\lambda_i = \frac{p_i}{\sum_i p_i}$.

■

For compact countable-state systems, there is a single computation yielding the maximum probability, thus resolving the issue raised by Example 3.1.

Proposition 3.8 *In a compact LCMC, for any state s and action a we have $P(s,a,E) = \sum_{s \in E} Q(s)$ for some $s \overset{a}{\Rightarrow} Q$.*

More explicitly, if $P(s,a,E) = p$ then, there exists a computation C such that: $P^C(s,a,E) = p$.

Proof. Let $\{Q_i\}$ be such that $s \overset{a}{\Rightarrow} Q_i$ and $P(s,a,E) - Q_i(E) < \frac{1}{2^i}$. In a compact metric space every sequence has a convergent subsequence. Thus there is a Q such that $s \overset{a}{\Rightarrow} Q$ and a subsequence Q'_i of Q_i such that $(\forall \epsilon)$ $(\exists i)$ $(\forall j \geq i)$ $(\forall E' \subseteq K)[|Q(E') - Q'_j(E')| < \epsilon$. This Q satisfies $Q(E) = P(s,a,E)$. ■

Example 3.9 We can modify the system of Example 3.1 by changing the transition probability from t_0 to s_a to be 1 instead of 0. This system is now compact. Note that it is not a finite-state system nor indeed weakly bisimilar to a finite-state system. Not only that, there are infinitely many different transition probabilities appearing, so it does not satisfy the minimum-deviation assumption of Larsen and Skou [LS91].

4 Coincidence with the definition of Philippou, Lee and Sokolsky

Our definition of bisimulation (Definition 2.10) is different from the definition in [PLS00]. However, the two definitions are equivalent.

We begin by presenting their definition below – we have recast it in terms of computations rather than schedulers. For any C, that is, an a-computation from s, we write P^C for the induced distribution on the leaves. Recall that $[u]_R$ stands for the equivalence class of a state u for an equivalence relation R.

Definition 4.1 *An equivalence relation R on K is a PLS-weak bisimulation iff whenever sRt:*

- *if $s \in K_n, \alpha \in \mathtt{Act}$ and $(s, \alpha, s') \in \longrightarrow$, then there exists a computation C such that $P^C(t, \alpha, [s']_R) = 1$.*
- *if $s \in K_p$, there exists a computation C such that for all $M \in K/R - [s]_R$, $Q_R(s, M) = P^C(t, \varepsilon, M)$.*

Q_R is the probability distribution from $s \in K_p$ "normalized" by weighting by the probability of exiting $[s]_R$. Let $s \rightarrow_p Q$. Then:

$$Q_R(s, M) \quad = \quad \begin{cases} Q(M), & \textit{if } Q([s]_R) = 1 \\ \frac{Q(M)}{1 - Q([s]_R)}, & \textit{otherwise.} \end{cases}$$

There is a maximum weak bisimulation, denoted by $\approx_{PLS}$.

For compact LCMCs (and hence all finite state LCMCs), $\approx$ and $\approx_{PLS}$ coincide. This theorem requires weighted combinations of computations, as illustrated by Example 2.9 and the following example.

Example 4.2 Let s be a state with no transition. Let t be a nondeterministic state with a single τ-transition to a probabilistic state t_p that has a probability 1 transition back to t. Clearly, $s \approx t$. Consider the computation from t that is the infinite chain of alternating t_p, t. This computation is matched by the weighted combination $0C$ where C is the computation from s that consists only of s.

The role of these weighted linear combinations is seen in the case (2) ⇒ (3) in the following proof.

Theorem 4.3 *The following are equivalent for compact LCMCs.*

1. $s \approx t$.
2. $s \approx_{PLS} t$.
3. *Let* $s \overset{a}{\Rightarrow} Q$. *Then, there exists* $t \overset{a}{\Rightarrow} Q'$ *such that for all states* u*:* $Q([u]_{PLS}) = Q'([u]_{PLS})$. *The reverse also holds with* t *and* s *interchanged.*

Proof. We sketch the main ideas below, the complete proof appears in Appendix A.

- (1) ⇒ (2): The key structural properties exploited in the proof are:
  - If t is a nondeterministic state, and s is a probabilistic state, such that t is weakly bisimilar to s, then there is a τ-transition from t to some t' such that t' is weakly bisimilar to s.
  - A linear programming argument is used to show that $\approx$-bisimilar probabilistic states have identical (up to $\approx$) probabilistic fans.
- (2) ⇒ (3): Using Lemma 3.3, it suffices to prove the result for finite-depth computations Q. In this case, the proof proceeds by induction on the number of transitions of computations.
  - Let C extend $s \overset{a}{\Rightarrow} Q$ by a nondeterministic transition $u \xrightarrow{b} u'$ at a leaf u. In this case, consider $t \overset{a}{\Rightarrow} Q'$, the extension of Q by matching transitions $v \overset{b}{\Rightarrow} Q_i$ from all the $v \approx_{PLS} u$ that are leaves.
  - The case when C extends $s \overset{a}{\Rightarrow} Q$ by adding a one-step probabilistic transition $u \rightarrow Q$ at a leaf u uses the ideas from example 2.9. There are two cases depending on whether $Q([u]) = 0$ or not.

    If $Q([u]) = 0$, $u \rightarrow Q$ can be matched by computations from all the $v \approx_{PLS} u$. If $Q([u]) = r > 0$, consider the transition from u to

    Q' where: $Q'[v] = \frac{Q[v]}{1-r}$, if $u \notin [v]$ and $Q'([u]) = 0$. For any $v \approx_{PLS} u$, this computation reaches its leaves with label ε and assign probabilities in accordance with Q'. The required transition to Q from v is given by a linear combination (with coefficient $1 - r$) of this computation with the computation consisting only of v (with coefficient r).

    Consider $t \overset{a}{\Rightarrow} Q'$, the extension of Q by matching transitions $v \overset{b}{\Rightarrow} Q_i$ from all the $v \approx_{PLS} u$ that are leaves.

  In either case, the required transition from t is obtained by a linear combination $t \Rightarrow [\lambda \times Q' + (1 - \lambda) \times Q]$, where $\lambda = p/Q([u])$.

- (3) ⇒ (1): This is immediate.

■

5 Capacities from sets of measures

In this section we first review the basic theory of capacities [Cho53]. The original context that Choquet was interested in led him to impose several conditions that need not concern us here. We will present a simplified treatment and omit proofs of any results available in the literature.

We begin by recalling that the basic example 1(a) shows that we lose the additivity property crucial to the definition of a measure.

Definition 5.1 *Let S be a set and let Σ be a σ-algebra of subsets of S. A **capacity** on Σ is a non-negative real-valued set function $\nu : \Sigma \to \mathbb{R}$ such that*

- $\nu(\emptyset) = 0$
- *if $A \subseteq B$ in Σ then $\nu(A) \leq \nu(B)$,*
- *if $E_1 \subseteq E_2 \subseteq \ldots \subseteq E_n \subseteq \ldots$ with $\cup_i E_i = E$ then $\lim_{i\to\infty} \nu(E_i) = \nu(E)$,*
- *if $E_1 \supseteq E_2 \supseteq \ldots \supseteq E_n \supseteq \ldots$ with $\cap_i E_i = E$ then $\lim_{i\to\infty} \nu(E_i) = \nu(E)$.*

*If, in addition, it satisfies $\nu(A \cup B) \leq \nu(A) + \nu(B)$, it is said to be **subadditive**.*

For measures the two continuity properties are consequences of countable additivity. If we have a family of measures μ_i defined on Σ we can get subadditive capacities as follows[5].

$$\overline{\nu}(A) := \sup_i \mu_i(A).$$

It is easy to see that ν defined this way is indeed a capacity. Later we will use these properties of capacities to prove the logical characterization theorem.

We establish the key properties of the functions $\nu(E) = P(s, a, E)$ showing that they are capacities.

Lemma 5.2 *Let $s \in K$, $a \in$ `Act`. Then the function ν on the $\approx$-closed subsets of K defined as above is a subadditive capacity as per definition 5.1.*

Proof. Recall that for any a-computation C from s, we write P^C for the induced distribution on the leaves. The first property $\nu(\emptyset) = 0$ is immediate. We have from the definitions that:

- $E_1 \subseteq E_2 \Rightarrow P^C(s, a, E_1) \subseteq P^C(s, a, E_2)$. Since ν is the sup over all C of the P^C we get that $\nu(E_1) = P(s, a, E_1) = \sup_C P^C(E_1) \leq \sup_C P^C(E_2) = \nu(E_2)$.

- Let $\{E_i\}$ be an increasing sequence of $\approx$-closed sets of states. Then $P^C(s, a, \cup_i E_i) = \sup_i P^C(s, a, E_i)$ because the P^Cs are distributions (measures). Now taking sups over C we get the result $\nu(E) = \sup_C P^C(E) = \sup_C \sup_i P^C(E_i) = \sup_i \sup_C P^C(E_i) = \sup_i \nu(E_i)$.

- If $E_1 \cap E_2 = \emptyset$, $P^C(s, a, E_1 \cup E_2) = P^C(s, a, E_1) + P^C(s, a, E_2)$. Taking sups over C we get that

$$\nu(E_1 \cup E_2) = \sup_C P^C(E_1 \cup E_2) \leq \sup_C P^C(E_1) + \sup_C P^C(E_2) = \nu(E_1) + \nu(E_2).$$

Thus, the first three properties and sub-additivity follow from basic properties of sup.

The proof of the fourth property crucially uses compactness. Let $E_1 \supseteq E_2 \ldots$ be a decreasing sequence of $\approx$-closed sets of states. Let $E = \cap_k E_k$. Since $\forall k, E \subseteq E_k$ we have $\nu(E) \leq \inf_k \nu(E_k)$ using monotonicity and the definition of inf.

[5] There are examples showing that not all capacities arise in this way.

We prove $\nu(E) \geq \inf_k \nu(E_k)$ as follows. It suffices to show that

$$(\forall \epsilon > 0)\ (\exists s \overset{a}{\Rightarrow} Q)\ (\exists i)\ [Q(E) \geq \nu(E_i) - \epsilon].$$

Let $\epsilon > 0$. For each E_k there exists $s \overset{a}{\Rightarrow} P_k$ such that $P_k(E_k) \geq \nu(E_k) - \frac{\epsilon}{3}$. By compactness every sequence has a limit point. Thus there exists $s \overset{a}{\Rightarrow} Q$ and a subsequence $Q_{k'}$ of P_k such that:

$$(\forall \delta)\ (\exists j)\ (\forall i \geq j)\ (\forall A \subseteq K)[|Q(A) - Q_i(A)| < \delta].$$

Since Q is a distribution, there exists j_1 such that $(\forall i > j_1)\ Q(E) \geq [Q(E_i) - \frac{\epsilon}{3}]$. Using $\delta = \frac{\epsilon}{3}$ in the above equation relating Q and the subsequence $Q_{k'}$, we get j_2 such that :

$$(\forall i \geq j_2)\ (\forall A \subseteq K)[|Q(A) - Q_i(A)| < \delta].$$

In particular, $Q(E_i) \geq Q_i(E_i) - \frac{\epsilon}{3}$. Choosing $i = \max(j_1, j_2)$, we have:

$$Q(E) \geq Q(E_i) - \frac{\epsilon}{3} \geq Q_i(E_i) - \frac{\epsilon}{3} - \frac{\epsilon}{3} \geq \nu(E_i) - \frac{\epsilon}{3} - \frac{\epsilon}{3} - \frac{\epsilon}{3}.$$

■

6 pCTL*

We now examine the relation between our processes and a minor variant of pCTL* [ASB+95, dA97], a standard modal logic used for expressing properties of probabilistic systems. We will largely elide formal definitions, instead focusing on explaining the key differences from the treatment of de Alfaro [dA97] for Markov decision processes (that lack τ and associate *unique* probability distributions with each label at a state). It will turn out that a very small fragment of this logic suffices to characterize weak bisimulation so the completeness is achieved with a very parsimonious logic. However, it is useful to know that the truth of all pCTL* formulas is invariant under bisimulation since this logic is actually used for specification. So for soundness we want as rich a logic as possible whereas for completeness we would like as simple a logic as possible in order to make clear what is really essential to characterize bisimulation.

The logic. There are two kinds of formulas: state formulas, denoted $\phi, \phi', \ldots$, and path formulas, denoted $\psi, \psi', \ldots$. These are generated by the following grammar:

$$\begin{array}{lcl} \phi & ::= & \bot \mid \neg\phi \mid \phi \wedge \phi' \mid P_{\bowtie q}\psi \\ \psi & ::= & a \mid \phi \mid \neg\psi \mid \psi \wedge \psi' \mid \bigcirc\psi \mid \Diamond\psi \mid \psi\mathcal{U}\psi' \end{array}$$

In the above, $\bowtie$ is drawn from $\{=, \leq, \geq, <, >\}$, q is a rational in $[0, 1]$, and $a \in \mathtt{Act}$.

We ignore *atomic formulas* which are first-order logic formulas over some fixed sets of variables, functions and predicate symbols. One can assume that bisimilar states satisfy the same atomic formulas.

Policies A policy on an LCMC disambiguates all nondeterminism. The operational scheduling idea underlying a policy is that for a given history, at each nondeterministic node, the scheduler chooses exactly one labeled transition. Weak transitions are accommodated by two extensions: (a) "stuttering" is facilitated by permitting a τ transition back to the same state even if such a transition is not available explicitly; and (b) skipping of intermediate states in a ε transition is permitted; this is sometimes called "mumbling."

The use of stuttering and mumbling is quite common in treatments of fully abstract semantics for concurrent programming languages [Bro96, HdBR94]. The key point is that the computations really talk about the fringe rather than about the paths that were taken to reach the fringe. Of course, the paths do arise but the whole point of the closure conditions is to abstract away from the inessential details of the paths.

Rather than formalize these operational intuitions directly, using definition 2.2, we formalize a local snapshot of the purely probabilistic chain that results from the use of such a scheduler.

Definition 6.1 *Let $\mathcal{K}$ be a LCMC.*

- *A* behaviour *is a finite sequence of states and labels $s = s_0, l_0, s_1, l_1, s_2, \ldots$ obtained from the transitions of the LCMC with possible stuttering and mumbling as described above. In other words: either s_i makes an l_i labelled transition to s_{i+1} according to the transition relation of the LCMC and possibly sequences of τ transitions before and after l_i (mumbling), or s_i and s_{i+1} are the same and the label is a ε (stuttering).*
- *A* basic policy *is a map from behaviours to $S \subseteq K \times \mathtt{Act} \times (0, 1]$ such that $\sum_S \pi_3(S) = 1$; here the notation π_3 means that we project out the*

third component of the triples (the actual probabilities), thus the sum is over the probabilities of the triples in S.

A basic policy η is valid *for an LCMC $\mathcal{K}$ if for all behaviors $s_0, l_0, s_1, l_1, s_2, \ldots, s_i$ there is a computation (as per definition 2.2) rooted at s_i validating*

$$\eta(s_0, l_0, s_1, l_1, s_2, \ldots, s_i),$$

i.e.

$$\forall (s, a, r) \in \eta(s_0, l_0, s_1, l_1, s_2, \ldots, s_i),$$

r equals the sum of the probabilities of the paths from root s_i with weak label a that end in leaf s in the computation[6]*. A behavior $s_0, l_0, s_1, l_1, s_2, \ldots s_j$ satisfies the basic policy η if for all $0 \leq i < j$, there exists an $r > 0$ such that $(s_{i+1}, l_i, r) \in \eta(s_0, l_0, \ldots s_i)$.*

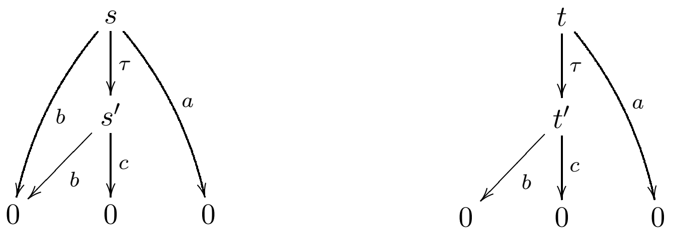
General policies are constructed as linear combinations of basic policies: $\sum_i \lambda_i \eta_i$ (where $\lambda_i > 0, \sum_i \lambda_i = 1$. A linear combination of basic policies valid for $\mathcal{K}$ is defined to be valid for $\mathcal{K}$.

When we interpret formulas over an LCMC $\mathcal{K}$, we will only consider policies valid for $\mathcal{K}$. Any policy η valid for $\mathcal{K}$ defines a measure μ_η on the paths of the resulting computation C_η in a standard way[7].

Example 6.2 Consider the policy η such that $\eta(s_0, l_0, s_1, l_1, s_2, \ldots, s_i) = (s_i, \tau, 1)$. It is valid at s_i via the computation that only has a root s_i. The behavior $s_0, l_0, s_1, l_1, s_2, \ldots, s_i, s_i$ is the sole behavior that immediately extends $s_0, l_0, s_1, l_1, s_2, \ldots, s_i$ as per this policy. So, this policy models one-step of stuttering at state s_i after behavior $s_0, l_0, s_1, l_1, s_2, \ldots, s_i$.

Here is an example showing the effect of mumbling.

Example 6.3



[6] Thus, in contrast to the definition of a-computations in earlier sections, the validating computation can have different weak labels on different paths from the root to the leaves.

[7] We elide well-known measure-theoretic details in this paper.

Consider the systems shown above. According to our definitions the states s and t should be weakly bisimilar. If we were to look for a direct match of the path $s \xrightarrow{b} 0$ from t we would only find $t \xrightarrow{\tau} t' \xrightarrow{b} 0$ and there is no state in the former path that is weakly bisimilar to t'. With mumbling, however, we are allowed to drop intermediate steps and we have the "mumbled" version of the path, namely $t \xrightarrow{b} 0$.

6.1 The formal semantics of the logic

The basic semantic relation is of the form $s \models \phi$ for state formulas and $\alpha \models \psi$ for path formulas where α is a behavior and s is a state, ϕ is a state formula and ψ is a path formula. The semantics of the path formulas is exactly as in standard linear temporal logic, indeed this *is* standard linear temporal logic. For $a \in \texttt{Act}$, the path formula a is true of a behaviors $s_0, a, s_1, \ldots$ whose first weak label is a. Formally,

$$\begin{array}{ll} s_0\ a\ s_1 \ldots \models a & \\ s_0\ a \ldots \models \phi & \text{if } s_0 \models \phi \\ s_0\ a_0\ s_1\ a_1 \ldots \models \bigcirc\psi & \text{if } s_1\ a_1 \ldots \models \psi. \end{array}$$

We omit the other cases as they follow the standard pattern.

Policies and the probabilistic quantifier For state formulas the only interesting point is the probabilistic quantifier. Let $\mathcal{K}$ be an LCMC. Fix a (possibly general) policy η. A set of behaviors is measurable if the set of the corresponding paths in η is measurable. By a routine structural induction, we can show that the sets of behaviors that satisfy path formulas are measurable.

The state formula $P_{\bowtie q}\psi$ is true at a state s if for all policies η, the measure of the set of behaviors that satisfy ψ is in the $\bowtie$ relation to q. More precisely, let $\mu_{\eta,s}$ be the measure induced on the set of paths starting from s with the policy η, then

$$s \models P_{\bowtie q}\psi \text{ if } \mu_{\eta,s}(\{\alpha | \alpha \models \psi\}) \bowtie q.$$

Example 6.4 We illustrate how the logic captures minimum and maximum probabilities, where the minima (resp. maxima) are taken over the set of policies.

Let ψ be a path formula. The formula $P_{\leq q}\neg\psi$ is true at a state s if for all policies η that are valid for $\mathcal{K}$, the measure of the set of behaviors that satisfy $\neg\psi$ is at most q.

The formula $P_{\geq q}\neg\psi$ is true at a state s if for all policies η, the measure of the set of behaviors that satisfy $\neg\psi$ is at least q.

One needs to be aware of the expressive power of the logic especially with regards to divergence. With the closure of the policies under stuttering we are effectively adding self loops everywhere.

Example 6.5 Consider the two processes shown in Figure 2. For the process on the left there is a divergent τ-loop (recall that probabilistic transitions are viewed as being unlabelled) coming out of the state s_1; apart from this the states s_1 and s_2 are identical. It is clear that the pairs of states (t_1, t_2) and (p_1, p_2) are bisimilar. By our definitions (s_1, s_2) are also bisimilar. It may appear that this contradicts the claim that the logic is sound for weak bisimulation because it appears that the formula $P_{=1}(\Diamond a)$ is satisfied by s_2 *for any possible scheduler* whereas this formula is not satisfied by s_1 under the (unfair) scheduler which always chooses the τ action in s_1. However, the effect of closing under stuttering is precisely to add the same τ loop to s_2, and thus the two states indeed satisfy the same formulas; in this case neither will satisfy the formula just mentioned. Thus, in this example neither process will satisfy any nontrivial formula of the form $P_{\geq q}\phi$.

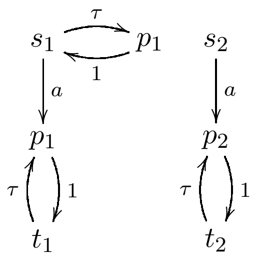


Figure 2: The logic does not detect divergence

How would we distinguish s_1 or s_2 from a dead state `nil` without any transition? One can use negation to, in effect, "get an existential quantification over policies." In fact positive formulas with universal quantification over policies are not very useful. One really wants to have existential quantification over policies and these one gets with negation.

Example 6.6 Consider the formula $\neg P_{<q}\phi$. A state s satisfies this formula if $s \not\models P_{<q}\phi$. According to our semantics, this means that *there is a policy*, say η, such that in the Markov chain starting from s following policy η, the measure of the paths satisfying ϕ is at least q.

The states s_1 (and s_2) in the above example satisfy $\neg P_{<1}[a]$ which a dead state does not.

6.2 Soundness of bisimulation

The key to the proof, as might be expected, is to show that the paths and computations out of bisimilar states "match" sufficiently.

First, we consider behaviors. The following proposition is a standard use of the co-inductive definition of bisimulation. We do not need it but it is worth noting. The proof is omitted because it can be done on similar lines to Lemma 6.8 which follows. That lemma is needed in the proof of the soundness theorem.

Proposition 6.7 *Let $s \approx t$. Then, for any behavior $s, l_0, s_1, l_1, s_2, \ldots$ from s, there is a behavior with equal trace, $t, l_0, t_1, l_1, t_2, \ldots$, from t such that: $(\forall i)$ $[s_i \approx t_i]$.*

Next, we move to policies and induced computations. For this, we follow the proof of Theorem 4.3 (in particular the implication (2) $\Rightarrow$ (3)). This proof has already shown that given a computation C from a state s, and given t bisimilar to s, there is a computation C' from t that assigns the same probabilities to the leaves of C. We will now generalize this to all paths — given an a-computation C_η induced by a policy η from a state s, we show that for any bisimilar state t, there is a policy η' that assigns at least the probabilities assigned by η to all the paths in C_η. We use the equivalence of our definitions with those of Philippou, Lee and Sokolsky [PLS00]. The first case of their definition permits the simulation of non-deterministic edges. The second case of their definition permits the simulation of probabilistic branches.

Lemma 6.8 *Let s, t be bisimilar states. Let η be a policy and let C_η be the induced η-computation from s. Then, there is a policy η' such that every path in C_η is a behavior in the η' computation from t with the same probability.*

Proof. It suffices to prove this for the case where η is a basic policy.

The proof is a routine induction. C_η has countably many transitions. Consider any ordering o of these transitions such that a transition occurs after all the transitions leading up to it. We construct $C_{\eta'}$ by mimicking transitions in the order prescribed by o. Our induction hypothesis is that at the i'th stage: every path in the subtree induced by the first i transitions (as per o) is a behavior in $C^i_{\eta'}$ computation from t with at least the same probability.

Let the $i+1$'st transition be a transition at u. Let p be the probability of the path from s to u in C_η. Let V be the set of leaves v in $C^i_{\eta'}$ such that:

- $v \approx u$
- The path from s to u in C_η is a behavior corresponding to the path from t to v in $C^i_{\eta'}$.

The measure of V in $C^i_{\eta'}$, say q, is at least p by the induction hypothesis.

There are two cases based on the kind of the $(i+1)$st transition.

1. The $(i+1)$st transition is a nondeterministic transition $u \xrightarrow{b} u'$. This transition can be matched by computations from all elements of V: by definition these computations reach $[u']$ with probability 1 on weak label b.

2. The $(i+1)$st transition is a probabilistic transition $u \to Q$. There are two cases depending on whether $Q([u]) = 0$ or not.

   If $Q([u]) = 0$, this transition can be matched by computations from all elements of V: by theorem 4.3 these computations reach the leaves with label ε and assign probabilities in accordance with Q.

   If $Q([u]) = r > 0$, consider the transition from u to Q' where: $Q'[v] = \frac{Q[v]}{1-r}$, if $u \notin [v]$ and $Q'([u]) = 0$. Pick any element $v \in V$. Since $v \approx u$, by theorem 4.3, this computation reaches the leaves with label ε and assign probabilities in accordance with Q'. The required transition to Q from v is given by a linear combination (with coefficient $1-r$) of this computation with the computation consisting only of v (with coefficient r).

In either case, let $C^{i'}_{\eta'}$ be the extension of $C^i_{\eta'}$ by these matching transitions. $C^{i+1}_{\eta'}$ is got by a linear combination $t \Rightarrow [\lambda \times C^{i'}_{\eta'} + (1-\lambda) \times C^i_{\eta'}]$, where $\lambda = p/q$. ∎

Lemmas 6.7 and 6.8 yield the desired theorem by a standard induction on the structure of formulas.

Theorem 6.9 *If $s \approx\ t$, then for all pCTL* state formulas ϕ, $s \models \phi$ iff $t \models \phi$.*

Proof. We sketch the case of $P_{\geq q}\psi$. Let s satisfy $P_{\geq q}\psi$. Every policy induces a set of computations from s. For every computation from s, using lemma 6.8, there is a computation from t that attributes a larger measure to the set of behaviors from t that satisfy ψ. Hence, t satisfies $P_{\geq q}\psi$. ∎

6.3 Completeness

We proceed now to completeness. Here the fact that we have a capacity plays a key role, as we use the downward continuity property of capacities.

We identify $\mathcal{L}$, a sub-fragment of the state formulas of the pCTL* variant above, that suffices for completeness. These are generated by the following grammar:

$$\phi \quad ::= \quad \top \mid \ \phi_1 \wedge \phi_2 \mid \phi_1 \vee \phi_2 \mid \neg\phi \mid \ \langle a\rangle_{\geq q}\phi$$

where $a \in \texttt{Act}$ (including τ), q is a rational and $\langle a\rangle_{\geq q}\phi$ is shorthand for $\neg P_{\leq q}[a \wedge \bigcirc\phi]$; recall the discussion of example 6.6. Thus, a state s satisfies $\langle a\rangle_{\geq q}\phi$ iff there is a policy η such that the computation induced by η assigns probability greater than q to the states satisfying ϕ reachable on a weak a transition. More succinctly, s satisfies $\langle a\rangle_{\geq q}\phi$ if $P(s, a, \{t \mid t \text{ satisfies } \phi\}) \geq q$.

Theorem 6.10 *If two states satisfy the same formulas of $\mathcal{L}$, then they are bisimilar.*

Proof. Let R be the equivalence relation defined by the formulas of $\mathcal{L}$. Let s and t be two R-related states. We need to prove that for every R-closed set X, $P(s, a, X) = P(t, a, X)$, where $a \neq \tau$ and also an analogous proof for an empty weak label. By using formulas of the form $\langle a\rangle_{\geq q}\phi$, we obtain the required equality for sets of states X that are denotations of formulas, i.e. $X = \{s' \mid s' \text{ satisfies } \phi\}$, $\phi \in \mathcal{L}$.

Since the state space is countable every R-closed set is a countable union of equivalence classes. Every equivalence class is described by countably many formulas and - since we have negation - can be described as the intersection of countably many sets of the form $\{s|s \text{ satisfies } \phi\}$. Thus every R-closed set, say Y, is of the form

$$Y = \cup_{i=1}^{\infty} \cap_{j=1}^{\infty} X_{ij}$$

where the X_{ij} are the denotations of formulas.

We define

$$Y_i := \cap_{j=1}^{\infty}[\cup_{k=1}^{i} X_{kj}].$$

Note that Y_i forms an increasing family in the subset ordering. Furthermore $\cup_{i=1}^{\infty} Y_i = Y$ by distributivity. Now, for each i, the sets $Z_i^{(l)} := \cap_{j=1}^{l} \cup_{k=1}^{i} X_{kj}$ are a decreasing family as l increases and they are the denotations of formulas, since there is conjunction and disjunction in the logic. Thus the two capacities will agree on each $Z_i^{(l)}$ and - since $\lim_{l\to\infty} Z_i^{(l)} = Y_i$ - by up continuity they will agree on Y_i and thus - by down continuity - they agree on Y.

The proof for $P(s, \varepsilon, X) = P(t, \varepsilon, X)$ is similar except for the use of the formulas $\langle\tau\rangle\phi$ and is omitted.

∎

Note how we used the up and down continuity properties of capacities to get from having the transition probabilities[8] agree on sets definable by formulas was as good as knowing that they agree everywhere. In other situations [DGJP02a], we have used Dynkin's $\lambda - \pi$ theorem to short circuit most of the above proof. Here, of course, that theorem does not apply (because we do not have σ-additivity); instead we used the continuity properties of capacities to complete the argument.

7 Conclusions

The main thrust of the present paper has been to elucidate the interaction between probability and nondeterminism in the alternating model. The definition of weak bisimulation that we have used generalizes the elegant treatment of Philippou, Lee and Sokolsky from finite state to countable systems. We have emphasized two features of their definition that were left implicit by them, namely the loss of additivity and the need for considering convex-linear structure when matching weak transitions. The main new result of our analysis is that weak bisimulation is sound and complete for (a minor variant of) pCTL*.

It is worth taking a retrospective view of some of the mathematical ideas in the proofs. The basic problem, with which we have had to struggle, is the loss of σ-additivity. The heart of any completeness proof of this type is an argument to establish that equality of the transition probabilities to sets of states *defined by the logic* forces equality of all the transition probabilities.

[8] Strictly speaking, we should say "transition capacities", but this is too ugly.

Such an argument rests on theorems that guarantee equality of measures given equality on a suitable generating set for the σ-field. These uniqueness theorems heavily rely on σ-additivity. Thus, we were led to consider what structure we do have given that we do not have a probability measure. The fact that we have capacities and, in particular, that capacities satisfy strong continuity properties (both upward and downward) turns out to be strong enough to establish the results that we need. To conclude we need to argue that we really have capacities. Here the compactness property turns out to be crucial.

The other major mathematical innovation (of course implicit in the works of previous authors) is the use of linear combinations in matching. This is really taking "out of the closet" ideas that are understood as randomized schedulers or other such devices. However, having done so it becomes clear that linear algebra and linear programming plays a key role in matching. In particular Claire Jones' remarkably prescient splitting lemma [Jon90] is clearly part of a general pattern [vBW01] where linear programming ideas, and duality in particular play a key role.

In closely related work [DGJP02a] we have shown that one can develop a metric for weak bisimulation analogous to our previous treatment of metrics for strong bisimulation [DGJP99]. In that work we heavily use linear programming and duality.

The present treatment is for discrete systems. We have preliminary results on continuous time, namely we have shown completeness for continuous stochastic logic [DP03]. To deal with continuous state spaces one has to use analogues of the linear programming theory for infinite dimensional spaces. Fortunately such theories are available and it appears that our results will go through at least under suitable compactness assumptions. There are analogues of such results for continuous-state Markov Decision Processes with rewards [FPP05], but so far only for strong bisimulation.

After the presentation of the conference version of this paper [DGJP02b], Jean Goubault-Larecq has begun a systematic investigation of the use of capacities in the context of modelling probability and non-determinism. He has explicitly worked with continuous state spaces and has tied the work to games [GL07].

An important related development has been the work carried out by Roberto Segala and his collaborators, two important papers are by Parma and Segala [PS07] and by Segala and Turrini [ST05]. Another relevant paper by Segala appears in CONCUR '06 [Seg06]. They also prove soundness and completeness of an appropriate logic for weak bisimulation using different ideas. Their work uses the framework of probabilistic automata which does not have the alternating character of our LCMC model. However, they give a bisimulation preserving translation from LCMC to probabilistic automata and from this soundness for LCMC also follows. As already remarked by Philippou et al. [PLS00], the existence of a converse bisimulation preserving translation is unlikely. Such a translation is needed for inferring our completeness results from theirs. The existence of such a translation is suggested in [ST05] but it was not clear to us. In particular the convexity properties that one needs for the probabilistic automaton case are not readily available in LCMC. There is no doubt that these papers have significantly advanced our understanding of the concepts and they have served to provide a unified conceptual framework for the different models in the extant literature.

Acknowledgements

We have benefited from discussions with Christel Baier and Roberto Segala. We thank the anonymous referees for their insightful comments. This research was supported by grants from the Natural Sciences and Engineering Council of Canada, from MITACS and from the National Science Foundation.

References

[ASB$^+$95] A. Aziz, V. Singhal, F. Balarin, R. K. Brayton, and A. L. Sangiovanni-Vincentelli. It usually works: the temporal logic of stochastic systems. In *Proceedings of the Conference on Computer-Aided Verification*, 1995.

[BDEP97] R. Blute, J. Desharnais, A. Edalat, and P. Panangaden. Bisimulation for labelled Markov processes. In *Proceedings of the Twelfth IEEE Symposium On Logic In Computer Science, Warsaw, Poland.*, 1997.

[BH97] C. Baier and H. Hermanns. Weak bisimulation for fully probabilistic processes. In *Proceedings of the 1997 International Conference on Computer Aided Verification*, number 1254 in Lecture Notes In Computer Science. Springer-Verlag, 1997.

[Bro96] Stephen Brookes. Full abstraction for a shared variable parallel language. *Information and Computation*, 127(2):145–163, 1996.

[BS01] E. Bandini and R. Segala. Axiomatizations for probabilistic bisimulation. In *Proceedings of the 28th International Colloquium on Automata, Languages and Programming*, number 2076 in Lecture Notes In Computer Science, pages 370–381. Springer-Verlag, July 2001.

[Cho53] G. Choquet. Theory of capacities. *Ann. Inst. Fourier (Grenoble)*, 5:131–295, 1953.

[dA97] L. de Alfaro. *Formal Verification of Probabilistic Systems*. PhD thesis, Stanford University, 1997. Technical Report STAN-CS-TR-98-1601.

[Del72] C. Dellacherie. *Capacités et Processus Stochastiques*. Springer-Verlag, 1972.

[DEP98] J. Desharnais, A. Edalat, and P. Panangaden. A logical characterization of bisimulation for labelled Markov processes. In *proceedings of the 13th IEEE Symposium On Logic In Computer Science, Indianapolis*, pages 478–489. IEEE Press, June 1998.

[DEP02] J. Desharnais, A. Edalat, and P. Panangaden. Bisimulation for labeled Markov processes. *Information and Computation*, 179(2):163–193, Dec 2002.

[DGJP99] J. Desharnais, V. Gupta, R. Jagadeesan, and P. Panangaden. Metrics for labeled Markov systems. In *Proceedings of CONCUR99*, number 1664 in Lecture Notes in Computer Science. Springer-Verlag, 1999.

[DGJP02a] J. Desharnais, V. Gupta, R. Jagadeesan, and P. Panangaden. The metric analogue of weak bisimulation for labelled Markov processes. In *Proceedings of the Seventeenth Annual IEEE Symposium On Logic In Computer Science*, pages 413–422, July 2002.

[DGJP02b] J. Desharnais, V. Gupta, R. Jagadeesan, and P. Panangaden. Weak bisimulation is sound and complete for $pCTL^*$. In L. Brim, P. Jancar, M. Kretinsky, and A. Kucera, editors, *Proceedings of 13th International Conference on Concurrency Theory, CONCUR02*, number 2421 in Lecture Notes In Computer Science, pages 355–370. Springer-Verlag, 2002.

[DP03] Josée Desharnais and Prakash Panangaden. Continuous stochastic logic characterizes bisimulation for continuous-time Markov processes. *Journal of Logic and Algebraic Progamming*, 56:99–115, 2003. Special issue on Probabilistic Techniques for the Design and Analysis of Systems.

[FPP05] Norm Ferns, Prakash Panangaden, and Doina Precup. Metrics for markov decision processes with infinite state spaces. In *Proceedings of the 21st Conference on Uncertainty in Artificial Intelligence*, pages 201–208, July 2005.

[GL07] Jean Goubault-Larrecq. Continuous capacities on continuous state spaces. In *Proceedings of the 34th International Colloquium on Automata, Languages and Programming (ICALP'07)*, volume 4596 of *Lecture Notes In Computer Science*, pages 764–776. Springer-Verlag, 2007.

[Han94] Hans A. Hansson. *Time and Probability in Formal Design of Distributed Systems*, volume 1 of *Real-time Safety-critical Systems*. Elseiver, 1994.

[HdBR94] E. W. Horita, J. W. de Bakker, and J. J. M. M. Rutten. Fully abstract denotational models for nonuniform concurrent languages. *Information and Computation*, 115(1):125–178, Nov 1994.

[HM85] M. Hennessy and R. Milner. Algebraic laws for nondeterminism and concurrency. *Journal of the ACM*, 32(1):137–162, 1985.

[Jon90] C. Jones. *Probabilistic Non-determinism*. PhD thesis, University of Edinburgh, 1990. CST-63-90.

[KS60] J. G. Kemeny and J. L. Snell. *Finite Markov Chains*. Van Nostrand, 1960.

[LS91] K. G. Larsen and A. Skou. Bisimulation through probablistic testing. *Information and Computation*, 94:1–28, 1991.

[Mey66] P. A. Meyer. *Probability and Potentials*. Blaisdell, 1966.

[PLS00] A. Philippou, I. Lee, and O. Sokolsky. Weak bisimulation for probabilistic processes. In C. Palamidessi, editor, *Proceedings of CONCUR 2000*, number 1877 in Lecture Notes In Computer Science, pages 334–349. Springer-Verlag, 2000.

[PS07] 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 (FOSSACS)*, number 4423 in Lecture Notes In Computer Science, pages 287–301, April 2007.

[Sch66] A. Schaefer, editor. *Topological Vector Spaces*. Springer-Verlag, 1966.

[Sch84] D. Schmeidler. Subjective probability without additivity. Technical report, Foerder Institute of Economic Research, 1984.

[Seg06] R. Segala. Probability and nondeterminism in operational models of concurrency. In *Proceedings of the 17th International Conference on Concurrency Theory CONCUR '06*, number 4137 in Lecture Notes In Computer Science, pages 64–78, 2006.

[SL94] R. Segala and N. Lynch. Probabilistic simulations for probabilistic processes. In B. Jonsson and J. Parrow, editors, *Proceedings of CONCUR94*, number 836 in Lecture Notes In Computer Science, pages 481–496. Springer-Verlag, 1994.

[SL95] R. Segala and N. Lynch. Probabilistic simulations for probabilistic processes. *Nordic Journal of Computing*, 2(2):250–273, 1995.

[ST05] R. Segala and A. Turrini. Comparative analysis of bisimulation relations on alternating and non-alternating probabilistic models. In *Proceedings of the Second International Conference on the Quantitative Evaluation of Systems (QEST)*, pages 44–53. IEEE Press, September 2005.

[vBW01] Franck van Breugel and James Worrell. An algorithm for quantitative verification of probabilistic systems. In K. G. Larsen and M. Nielsen, editors, *Proceedings of the Twelfth International Conference on Concurrency Theory - CONCUR'01*, number 2154 in Lecture Notes In Computer Science, pages 336–350. Springer-Verlag, 2001.

A Complete Proof of Theorem 4.3

The first lemma says that there must be some non-zero probability that a weak-ε transition can exit a set of pairwise non-bisimilar states.

Lemma A.1 *Given a countable set of states A, with every pair of states in A non-bisimilar, there exists $s \in A$ such that $P(s, \varepsilon, A \setminus \{s\}) < 1$.*

Proof. Let $A = \{s_i \mid i = 1, 2, \ldots\}$. We first prove that if $((\forall s_i \in A)\ [P(s_i, \varepsilon, A \setminus \{s_i\}) = 1])$ then the same statement is true for $A \setminus \{s_j\}$ in place of A, for any $s_j \in A$. This will lead to a contradiction as follows. If the statement is true with one state removed it will be true for any finite set of states removed, by induction. Now define the set A_i to be $A \setminus \{s_1, \ldots, s_i\}$. For any i we have $P(s_1, \varepsilon, A_i) = 1$ hence $P(s_1, \varepsilon, \cap_i A_i) = 1$ but $\cap_i A_i = \emptyset$ so we get the obvious contradiction $P(s_1, \varepsilon, \emptyset) = 1$. To complete the proof we need to establish the implication asserted in the second sentence of this paragraph.

Recall that we are assuming that $P(s_i, \varepsilon, A \setminus \{s_i\}) = 1$ for any s_i in A. Let C_i be the computation that induces the maximum value, namely 1, of $P(s_i, \varepsilon, A \setminus \{s_i\})$, and let P_i be the distribution induced by this computation on $A \setminus \{s_i\}$. Then for all $j \geq 1$,

$$\begin{aligned} 1 = P(s_1, \varepsilon, [A \setminus \{s_1\}]) &= P_1(s_j) + P_1(A \setminus \{s_1, s_j\}) \text{ and} \\ 1 = P(s_j, \varepsilon, [A \setminus \{s_j\}]) &= P_j(s_1) + P_j(A \setminus \{s_1, s_j\}). \end{aligned}$$

Now the computation C_j (resp C_1) was chosen to attain the maximum possible value on $A \setminus \{s_i\}$ but it need not be the one that maximises the probability for reaching $A \setminus \{s_1, s_j\}$. Thus, we have the inequality

$$\begin{aligned} & P(s_j, \varepsilon, A \setminus \{s_1, s_j\}) \\ & \quad \geq P_j(A \setminus \{s_1, s_j\}) + P_j(s_1) P(s_1, \varepsilon, A \setminus \{s_1, s_j\}) \\ & \quad \geq P_j(A \setminus \{s_1, s_j\}) + P_j(s_1)[P_1(A \setminus \{s_1, s_j\}) + P_1(s_j) P(s_j, \varepsilon, A \setminus \{s_1, s_j\})]. \end{aligned}$$

Thus since s_1 is not bisimilar to s_j, we have $P_j(s_1) P_1(s_j) < 1$, and hence

$$P(s_j, \varepsilon, A \setminus \{s_1, s_j\}) \geq \frac{P_j(A \setminus \{s_1, s_j\}) + P_j(s_1) P_1(A \setminus \{s_1, s_j\})}{1 - P_j(s_1) P_1(s_j)}.$$

But this fraction is equal to 1 because of the two equalities above. Thus it follows that $P(s_j, \varepsilon, [A \setminus \{s_1, s_j\}]) = 1$. This completes the argument. ■

Now we give the complete proof of Theorem 4.3. In the main text we had already shown that (2) implies (3) and observed that (3) implies (1) immediately.

Theorem A.2 *Given an LCMC which satisfies the property that the total of all the probabilities from any probabilistic state is 1, if states s and t in it are bisimilar then they are bisimilar according to the definition of Lee, Philippou and Sokolsky [PLS00].*

Proof. This corresponds to (1) $\Rightarrow$ (2) in the statement of Theorem 4.3.

We show that the relation $\approx$, which is our notion of weak bisimulation, satisfies both the conditions of Definition 4.1. Let us recall this definition:

> An equivalence relation $R \subseteq S \times S$ is a PLS-weak bisimulation iff whenever sRt, then
>
> - if $s \in K_n, \alpha \in \texttt{Act}$ and $(s, \alpha, s') \in \longrightarrow$, then there exists a computation C such that $P^C(t, \alpha, [s']) = 1$.
> - if $s \in K_p$, there exists a computation C such that for all $M \in K/R - [s]_R$, $Q_R(s, M) = P^C(t, \varepsilon, M)$.
>
> Q_R is the probability distribution from $s \in K_p$ "normalized" by weighting by the probability of exiting $[s]_R$. Let $s \rightarrow_p Q$. Then:
>
> $$Q_R(s, M) \quad = \quad \begin{cases} Q(M), \text{ if } Q([s]_R) = 1 \\ \frac{Q(M)}{1 - Q([s]_R)}, \text{ otherwise} \end{cases}$$

Let $s, t \in K$, with $s \approx t$. The first condition is satisfied easily: If $(s, \alpha, s') \in \longrightarrow$, $P(s, \alpha, [s']) = 1$. Since $s \approx t$, $P(t, \alpha, [s']) = 1$, and using Proposition 3.8, we have an α-computation C such that $P^C(t, \alpha, [s']) = 1$.

For the second condition, assume $s \in K_p$. Let s_i, $i = 1, \ldots, s_n$ be the targets of the probabilistic transition from s that are not $\approx$-related to s. If there are no such states, the condition is satisfied trivially because there are no Ms of the type described above. Our proof proceeds in the following steps. For $t \in K_n$, we show that there exists $t' \approx t$ such that $(t, \tau, t') \in \longrightarrow$, thus reducing this case to the case when t is probabilistic. For $t \in K_p$, we show that the targets of the probabilistic transition from t are precisely the ones of s with identical "normalized" probabilities.

• *Case* $t \in K_n$: we will show that there exists $t' \approx t$ such that $(t, \tau, t') \in \longrightarrow$. Let E be a $\approx$-closed set that does not contain $[t]$. Then, by Proposition 3.8, there is a state t_E belonging to the targets of τ-transitions from t such that $P(t, \varepsilon, E) = P(t_E, \varepsilon, E)$. We will consider the analogous statement with the set A for E where $A = [s_1] \cup \cdots \cup [s_n]$ which is $\approx$-closed and does not contain s, and hence not $[t]$. We will show that the corresponding state t_A satisfies $t_A \approx t$; this will turn out to be the state t' that we are looking for. Now $P(t_A, \varepsilon, A) = P(t, \varepsilon, A) = P(s, \varepsilon, A) = 1$. The first equality is from the definition of t_A and the second holds because $s \approx t$. The last equality follows from the following little calculation, suppose that $s \longrightarrow_p Q$ then:

$$P(s, \varepsilon, A) = Q(A) + Q([s])P(s, \varepsilon, A).$$

This holds because from s one either goes to A or to a state bisimilar to s. From this we get

$$P(s, \varepsilon, A) = \frac{Q(A)}{1 - Q([s])}$$

which is 1 since $Q(A) + Q([s]) = 1$.

Since $(t, \tau, t_E) \in \longrightarrow$, we have $1 = P(t, \varepsilon, [t_E]) = P(s, \varepsilon, [t_E])$, it follows that $P(s_i, \varepsilon, [t_E]) = 1$ for all s_i, and hence for every element of A. Thus $P(t_A, \varepsilon, [t_E]) = 1$ and hence $P(t_A, \varepsilon, E) \geq P(t_E, \varepsilon, E)$, which, combined with $P(t, \varepsilon, E) \geq P(t_A, \varepsilon, E)$, implies

$$P(t_A, \varepsilon, E) = P(t, \varepsilon, E) \text{ for any } \approx\text{-closed } E \text{ not containing } [t].$$

The case where E contains $[t]$ is trivially handled by including the pair (t, t_A) in the bisimulation relation. Thus $t_A \approx t$ and the computation is the one that goes from t to t_A and then continues as given by the following case for state t_A.

• *Case* $t \in K_p$: If t has a probability 1 transition to another state, then it is bisimilar to that state, reducing us to the case above. This process stops at some point because t_A is given by the computation of Proposition 3.8, and hence it is a probabilistic state that does not have probability 1 to go back to t. This means that we build up the desired computation by travelling through bisimilar states until we reach a (probabilistic) state t' that does not have probability 1 to some other state. This must happen because s has non-bisimilar successors and $s \approx t$. We then append the computation that we built to the one for t', as shown below.

Otherwise, let Q_s and Q_t be the normalized probability distributions arising from probabilistic transitions at s and t; that is, the supports of Q_s and Q_t are disjoint from $[s]$ and for any set M, $Q_s(M) = \frac{P_s(M)}{1 - P_s([s])}$ where $s \longrightarrow_p P_s$, similarly for t (recall that $P_s([s])$ and $P_t([t])$ are not 1). Let

$s_1, s_2, \ldots$ be states in the targets of Q_s or Q_t (in the union of their support) such that $Q_s([s_i]) = Q_t([s_i])$ for all $i \geq 1$. Let U be the set containing the remaining states in the union of the supports of Q_s and Q_t. This set is also countable. We will show that U is empty.

Let $A = \cup_{u \in U}[u]$[9]. Then, since $s \notin A$,

$$\begin{aligned} P(s, \varepsilon, A) &= \sum_{u \in U} Q_s(u) + \sum_{s_j} Q_s(s_j) P(s_j, \varepsilon, A) \\ &= \sum_{u \in U} Q_s(u) + \sum_{[s_j]} Q_s([s_j]) P(s_j, \varepsilon, A) \quad \text{by definition of } \approx . \end{aligned}$$

Using the same equality on the t side, and using $P(s, \varepsilon, A) = P(t, \varepsilon, A)$, since A is $\approx$-closed, and also $Q_s([s_i]) = Q_t([s_i])$, we have that $0 = \sum_{u \in U}(Q_s(u) - Q_t(u))$.

Now by Lemma A.1 there exists $u \in A$ such that $P(u, \varepsilon, A \setminus \{u\}) < 1$, and hence there is some $u_0 \in [u] \cap U$ such that $P(u_0, \varepsilon, A \setminus [u_0]) < 1$. Now

$$P(s, \varepsilon, A \backslash [u_0]) = \sum_{u \in U \backslash [u_0]} Q_s(u) + Q_s([u_0]) P(u_0, \varepsilon, A \backslash [u_0]) + \sum_{[s_j]} Q_s([s_j]) P(s_j, \varepsilon, A \backslash [u_0]).$$

By using a similar equality for t, and because $A \setminus [u_0]$ is $\approx$-closed, we obtain

$$0 = \sum_{u \in U \backslash [u_0]} (Q_s(u) - Q_t(u)) + (Q_s([u_0]) - Q_t([u_0])) P(u_0, \varepsilon, A \setminus [u_0]).$$

Subtracting this equation from the previous equation, we have $(Q_s([u_0]) - Q_t([u_0]))(1 - P(u_0, \varepsilon, A \setminus [u_0]) = 0$, which means that $Q_s([u_0]) = Q_t([u_0])$, as $P(u_0, \epsilon, A \setminus [u_0]) < 1$. This is a contradiction to the fact that $u_0 \in U$, and hence U is empty. The emptiness of U implies that s and t have the same transition probabilities to any equivalence class.

In order to complete the proof we need to construct a computation C starting from t. This construction proceeds as follows. At t, all successors of t that are not in $[t]$ will be leaves at depth 1 in C, and then we append to successors of t that are in $[t]$ the computation given by the arguments above. Of course $Q_s([s]) = Q_t([s]) \neq 1$ and hence $\frac{Q_s(M)}{1 - Q_s([s])} = \frac{Q_t(M)}{1 - Q_t([s])} = P^C(t, \varepsilon, M)$.

∎

[9] If it is empty, we are done.