Cost Preserving Bisimulations for Probabilistic Automata

Holger Hermanns and Andrea Turrini
Saarland University – Computer Science, Saarbrücken, Germany

Abstract. Probabilistic automata constitute a versatile and elegant model for concurrent probabilistic systems. They are equipped with a compositional theory supporting abstraction, enabled by weak probabilistic bisimulation serving as the reference notion for summarising the effect of abstraction.

This paper considers probabilistic automata augmented with costs. It extends the notions of weak transitions in probabilistic automata in such a way that the costs incurred along a weak transition are captured. This gives rise to cost-preserving and cost-bounding variations of weak probabilistic bisimilarity. Polynomial-time decision algorithms are proposed, that can be effectively used to compute reward-bounding abstractions of Markov decision processes.

1 Introduction

Markov Decision Processes (MDPs) are mathematical models widely used in operations research, automated planning, decision support systems and related fields. In the concurrent systems context, they appear in the form of Probabilistic Automata (PAs) [18]. PAs form the backbone model of successful model checkers such as PRISM [12] enabling the analysis of randomised concurrent systems. They extend classical concurrency models in a simple yet conservative fashion, by enabling probabilistic experiments inside transitions.

As one of the classical concurrency theory manifestations, weak probabilistic bisimilarity is a congruence relation for parallel composition and hiding on PA. In other contexts, this has enabled powerful compositional minimisation approaches to combat the state space explosion problem in explicit state verification approaches [6][10][15]. With the conception of a polynomial time algorithm for deciding weak probabilistic bisimilarity [11] this avenue can now be followed also in the context of PAs and MDPs. The decision algorithm follows the usual partition refinement approach. At its core, the decision algorithm needs to check a polynomial number of linear programming (LP) problems. Each of them checks the existence of a specific weak transition. The decision algorithm can be turned into a minimisation algorithm, producing a minimal canonical representation of the PA with respect to weak probabilistic bisimilarity [7].

MDP models are usually decorated with cost or rewards structures, with the intention to minimise costs or maximise rewards along the model execution. Likewise, in tools like PRISM, PAs appear augmented with cost or reward structures. It is hence a natural question how costs can be embedded into the approach discussed above, and this is what the paper is about.
We propose Cost Probabilistic Automata (CPAs), a model where cost is any kind of quantity associated with the transitions of the automata, and we aim to minimise the cost. For instance, we can consider as the cost of a transition the power needed to transmit a message, the time spent in the computation modelled by the transition, the (monetary) risk associated with an action, the expense of some work, and so on. Costs for weak transitions are interpreted in line with the vast body of literature on MDPs, and we describe how that interpretation can be linked to the weak transition encoding as LP problems.

We then extend weak probabilistic bisimulation to also account for costs. As a strict option, we require weak transition costs to be matched exactly for bisimilar states, inducing cost-preserving weak probabilistic bisimulation. As a weaker alternative, we ask them to be bounded from one PA to the other, leading to the notion of minor cost weak probabilistic bisimulation. We provide polynomial time algorithms for both variations. Finally, we present an application of minor cost weak probabilistic bisimulation to a multi-hop wireless communication scenario where the cost structure represents transmission power which in turn depends on physical distances.

**Organisation of the Paper.** After the preliminaries in Section 2, we revisit the LP problem formulation behind weak probabilistic bisimilarity in Section 3, and we present cost probabilistic automata and relative bisimulations in Section 4 together with the wireless channel example. We discuss related work and possible extensions in Section 5, and we conclude the paper in Section 6 with some remarks.

## 2 Mathematical Preliminaries and Probabilistic Automata

For a set $X$, denote by $\text{Disc}(X)$ the set of discrete probability distributions over $X$, and by $\text{SubDisc}(X)$ the set of discrete sub-probability distributions over $X$. Given $\rho \in \text{SubDisc}(X)$, we denote by $\text{Supp}(\rho)$ the set $\{ x \in X \mid \rho(x) > 0 \}$, by $\rho(\bot)$ the value $1 - \rho(X)$ where $\bot \notin X$, and by $\delta_x$, where $x \in X \cup \{ \bot \}$, the Dirac distribution such that $\rho(y) = 1$ for $y = x$, 0 otherwise. For a sub-probability distribution $\rho$, we also write $\rho = \{(x, p_x) \mid x \in X \}$ where $p_x$ is the probability of $x$. The lifting $\mathcal{L}(\mathcal{R})$ of a relation $\mathcal{R} \subseteq X \times Y$ is defined as: for $\rho_X \in \text{Disc}(X)$ and $\rho_Y \in \text{Disc}(Y)$, $\rho_X \mathcal{L}(\mathcal{R}) \rho_Y$ holds if there exists a weighting function $w : X \times Y \to [0, 1]$ such that (1) $w(x, y) > 0$ implies $x \mathcal{R} y$, (2) $\sum_{y \in Y} w(x, y) = \rho_X(x)$, and (3) $\sum_{x \in X} w(x, y) = \rho_Y(y)$.

A Probabilistic Automaton (PA) $\mathcal{A}$ is a tuple $(S, \bar{s}, \Sigma, D)$, where $S$ is a countable set of states, $\bar{s} \in S$ is the start state, $\Sigma$ is a countable set of actions, and $D \subseteq S \times \Sigma \times \text{Disc}(S)$ is a probabilistic transition relation. The set $\Sigma$ is divided in two sets $\mathbb{H}$ and $\mathbb{E}$ of internal (hidden) and external actions, respectively; we let $s, t, u, v$, and their variants with indices range over $S$; $a, b$ range over actions; and $\tau$ range over internal actions. In this work we consider only finite PAs, i.e., PAs such that $S$ and $D$ are finite.

A Markov Decision Process (MDP) $\mathcal{M}$ is a tuple $(S, \iota, \Sigma, P, r)$ that can be considered as a variation of a PA with a functional transition relation $P : S \times \Sigma \to \text{Disc}(S)$, a start distribution $\iota \in \text{Disc}(S)$ instead of a start state, and additionally a reward function or structure $r : S \times \Sigma \to \mathbb{R}$. In this paper we consider only non-negative rewards, i.e., $r(s, a) \geq 0$ for each $(s, a) \in S \times \Sigma$, but interpret them as costs.