

COST PRESERVING BISIMULATIONS FOR PROBABILISTIC AUTOMATA *

ANDREA TURRINI^a AND HOLGER HERMANN^b

^a State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences,
Beijing, China
e-mail address: turrini@ios.ac.cn

^b Saarland University – Computer Science, Saarbrücken, Germany
e-mail address: hermanns@cs.uni-saarland.de

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, for which we establish compositionality properties with respect to parallel composition. Furthermore, polynomial-time decision algorithms are proposed, that can be effectively used to compute reward-bounding abstractions of Markov decision processes in a compositional manner.

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*) [26]. *PAs* form the backbone model of successful model checkers such as PRISM [17] enabling the analysis of randomised concurrent systems.

In probabilistic automata, probabilistic experiments can be performed inside a transition. This embodies a clear separation between probability and nondeterminism, and is represented by transitions of the form $s \xrightarrow{a} \mu$, where s is a state, a is an action label, and μ is a probability distribution on states. Labelled transition systems are instances of this model family, obtained by restricting to Dirac distributions (assigning full probability to

2012 ACM CCS: [**Mathematics of computing**]: Probability and statistics—Stochastic processes—Markov processes; Design and analysis of algorithms—Mathematical optimization—Continuous optimization—Linear Programming; [**Software and its engineering**]: Software creation and management—Software verification and validation—Formal software verification.

Key words and phrases: Markov decision processes, formal verification, rewards, bisimulation.

* An extended abstract appeared in CONCUR 2013.

single states). Thus, foundational concepts and results of standard concurrency theory are retained in full and extend smoothly to the model of probabilistic automata. This includes notions of strong, branching and weak probabilistic bisimilarity [26].

As one of the classical concurrency theory manifestations, weak probabilistic bisimilarity is a congruence relation for parallel composition on *PAs*. In other contexts, this has enabled powerful compositional minimisation approaches to combat the state space explosion problem in explicit state verification approaches [7, 14, 20]. This is rooted in the availability of effective minimisation algorithms for weak bisimulation implemented in tools like CADP [8], MRMC [21] or sigref [30]. In the *PA* context, this avenue has not been explored, mainly because for a long time only an exponential decision algorithm for weak probabilistic bisimilarity was known [5], and it was unclear how to turn the decision algorithm into a minimisation algorithm. Lately, these two problems have been successfully attacked: A polynomial time algorithm for deciding weak probabilistic bisimilarity [15] has been devised, and has been embedded into a minimisation algorithm [10], producing in polynomial time the minimal canonical representation with respect to weak probabilistic bisimilarity for any given *PA*. Therefore, compositional minimisation can now be followed also in the context of *PAs* and *MDPs*.

MDP models are usually decorated with cost or reward 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.

We then turn our attention to strong and weak probabilistic bisimulation that accounts for costs. Costs for weak transitions are interpreted in line with the vast body of literature on *MDPs*. 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 *CPA* to the other, leading to the notion of *minor cost weak probabilistic bisimulation*.

When establishing the base properties expected from these kind of definitions, especially transitivity of minor cost weak probabilistic bisimulation turns out to be quite intricate to prove. We also show that both relations are compositional: Cost-preserving weak probabilistic bisimulation is a congruence with respect to parallel composition, and minor cost weak probabilistic bisimulation is a precongruence.

Furthermore, we provide polynomial time algorithms for all the cost related relations discussed, and 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.

The algorithmic advancement is rooted in an alternative interpretation of weak transition costs, which agrees with the original one with respect to cost expectations, but provides us with the technical assets to establish a link to the polynomial time algorithm for *PA* weak probabilistic bisimilarity. At the core of that algorithm is a polynomial number of linear programming (LP) problems, each of them checking the existence of a specific weak transition, and this is what we manage to bridge to also in the cost setting.

Organisation of the paper. After introducing preliminaries in Section 2 and probabilistic automata background in Section 3, we present cost probabilistic automata in Section 4, the strong and weak cost-preserving bisimilarities in Section 5, and the strong and weak cost-bounding bisimilarities in Section 6 where we study their properties, and exemplify the usefulness of minor cost weak bisimulation by means of a wireless channel example. We then revisit the LP problem formulation behind weak probabilistic bisimilarity in Section 7 so as to arrive at polynomial-time algorithms for all bisimilarities we introduced. Related work and possible extensions are discussed in Section 8 and we conclude the paper in Section 9 with some remarks. To keep the presentation of the paper clear, we moved all non-trivial proofs to the appendix.

Parts of this paper are based on a conference publication in CONCUR 2013 [16].

2. MATHEMATICAL PRELIMINARIES

This section recalls the basic mathematical preliminaries together with the notational conventions we adhere to in this work. Given a function $f: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$, we say that f is

- *symmetric* if, for each $x, y \in \mathbb{R}$, it holds $f(x, y) = f(y, x)$;
- *zero-preserving* if $f(0, 0) = 0$;
- *distributive* over convex combination if, for any finite sets $I, J \subseteq \mathbb{N}$, each $\{x_i \in \mathbb{R}\}_{i \in I}$, and each $\{y_j \in \mathbb{R}\}_{j \in J}$, each $\{p_i \in \mathbb{R}^{>0}\}_{i \in I}$, and each $\{q_j \in \mathbb{R}^{>0}\}_{j \in J}$, the following holds:
 - $f(x, y) = \sum_{j \in J} q_j \cdot f(x, y_j)$, and
 - $f(x, y) = \sum_{i \in I} p_i \cdot f(x_i, y)$,
 where $x = \sum_{i \in I} p_i \cdot x_i$ and $y = \sum_{j \in J} q_j \cdot y_j$;
- *monotone increasing* if, for each $x, x', y, y' \in \mathbb{R}$ with $x < x'$ and $y < y'$, the following holds:
 - $f(x, y) < f(x', y)$,
 - $f(x, y) < f(x, y')$, and
 - $f(x, y) < f(x', y')$.

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 $\rho \in \text{SubDisc}(X)$ and $Y \subseteq X$, we write $\rho(Y)$ for $\sum_{y \in Y} \rho(y)$. Given $\rho \in \text{SubDisc}(X)$, we denote by $\text{Supp}(\rho)$ the set $\{x \in X \mid \rho(x) > 0\}$, by $\rho(\perp)$ the value $1 - \rho(X)$ where $\perp \notin X$, and by δ_x , where $x \in X \cup \{\perp\}$, the *Dirac* distribution such that $\rho(y) = 1$ for $y = x$, 0 otherwise. For a sub-probability distribution ρ , we also write $\rho = \{(x, p_x) \mid x \in X\}$ where p_x is the probability of x . Given $\rho_x \in \text{SubDisc}(X)$ and $\rho_y \in \text{SubDisc}(Y)$, we denote by $\rho_x \times \rho_y$ the sub-probability distribution over $X \times Y$ defined by $\rho_x \times \rho_y(u, v) = \rho_x(u) \cdot \rho_y(v)$ for each $(u, v) \in X \times Y$. Given a finite set I of indexes, a family $\{p_i \in \mathbb{R}^{>0}\}_{i \in I}$ such that $\sum_{i \in I} p_i = 1$, and a family $\{\rho_i \in \text{SubDisc}(X)\}_{i \in I}$, we say that ρ is the *convex combination* of $\{\rho_i\}_{i \in I}$ according to $\{p_i\}_{i \in I}$, denoted by $\sum_{i \in I} p_i \cdot \rho_i$, if for each $x \in X$, $\rho(x) = \sum_{i \in I} p_i \cdot \rho_i(x)$.

Given a relation $\mathcal{R} \subseteq X \times X$, we say that \mathcal{R} is a preorder if it is reflexive and transitive. We say that \mathcal{R} is an equivalence relation if it is a symmetric preorder. Given an equivalence relation \mathcal{R} on X , we denote by X/\mathcal{R} the set of equivalence classes induced by \mathcal{R} and, for $x \in X$, by $[x]_{\mathcal{R}}$ the class $\mathcal{E} \in X/\mathcal{R}$ such that $x \in \mathcal{E}$. We denote by \mathcal{I} the *identity relation*, i.e., the equivalence relation having $[x]_{\mathcal{I}} = \{x\}$ for each $x \in X$.

Given the relations $\mathcal{R}_1 \subseteq X \times Y$ and $\mathcal{R}_2 \subseteq Y \times Z$, the *composition* of \mathcal{R}_1 and \mathcal{R}_2 , denoted by $\mathcal{R}_1 \circ \mathcal{R}_2$, is the relation $\mathcal{R} \subseteq X \times Z$ defined as $\mathcal{R} = \{(x, z) \mid \exists y \in Y. x \mathcal{R}_1 y \mathcal{R}_2 z\}$.

If \mathcal{R}_1 and \mathcal{R}_2 are equivalence relations on $X \cup Y$ and $Y \cup Z$, respectively, then $\mathcal{R}_1 \circ \mathcal{R}_2$ is the equivalence relation \mathcal{R} on $X \cup Z$ defined as the symmetric and transitive closure of $\{(x, z) \mid \exists y \in Y. x \mathcal{R}_1 y \mathcal{R}_2 z\} \cup \{(x, x') \in X \times X \mid x \mathcal{R}_1 x'\} \cup \{(z, z') \in Z \times Z \mid z \mathcal{R}_2 z'\}$.

Given the relations $\mathcal{R}_1 \subseteq W \times X$ and $\mathcal{R}_2 \subseteq Y \times Z$, the *cross-product* of \mathcal{R}_1 and \mathcal{R}_2 , denoted by $\mathcal{R}_1 \times \mathcal{R}_2$, is the relation $\mathcal{R} \subseteq (W \times Y) \times (X \times Z)$ such that $(w, y) \mathcal{R} (x, z)$ if and only if $w \mathcal{R}_1 x$ and $y \mathcal{R}_2 z$.

The lifting $\mathcal{L}(\mathcal{R})$ [19] 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* $w: X \times Y \rightarrow [0, 1]$ 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) $\sum_{x \in X} w(x, y) = \rho_Y(y)$.

The lifting of relations has some interesting properties:

- (1) $x \mathcal{R} y$ if and only if $\delta_x \mathcal{L}(\mathcal{R}) \delta_y$.
- (2) $\mathcal{R} = \emptyset$ if and only if $\mathcal{L}(\mathcal{R}) = \emptyset$.
- (3) If $\mathcal{R} \subseteq \mathcal{S}$, then $\mathcal{L}(\mathcal{R}) \subseteq \mathcal{L}(\mathcal{S})$.
- (4) If \mathcal{R} is reflexive, then $\mathcal{L}(\mathcal{R})$ is reflexive.
- (5) If \mathcal{R} is symmetric, then $\mathcal{L}(\mathcal{R})$ is symmetric.
- (6) If \mathcal{R} is transitive, then $\mathcal{L}(\mathcal{R})$ is transitive.
- (7) If $\rho_x \mathcal{L}(\mathcal{R}) \rho_y$ and $\rho_y \mathcal{L}(\mathcal{S}) \rho_z$, then $\rho_x \mathcal{L}(\mathcal{R} \circ \mathcal{S}) \rho_z$.
- (8) If $\rho_x \mathcal{L}(\mathcal{R}) \rho_y$, then $\rho_x \times \rho_z \mathcal{L}(\mathcal{R} \times \mathcal{I}) \rho_y \times \rho_z$.
- (9) Given a finite set I of indexes, a family $\{p_i \in \mathbb{R}^{>0}\}_{i \in I}$ such that $\sum_{i \in I} p_i = 1$, a family $\{\rho_{x,i} \in \text{Disc}(X)\}_{i \in I}$, and a family $\{\rho_{y,i} \in \text{Disc}(Y)\}_{i \in I}$, if $\rho_{x,i} \mathcal{L}(\mathcal{R}) \rho_{y,i}$ for each $i \in I$, then $\sum_{i \in I} p_i \cdot \rho_{x,i} \mathcal{L}(\mathcal{R}) \sum_{i \in I} p_i \cdot \rho_{y,i}$.

3. PROBABILISTIC AUTOMATA

We now recall the definition of probabilistic automata as proposed by Segala in [26] as *simple* probabilistic automata. We then review strong and weak bisimilarities on PAs together with their properties. We follow the notation used in [27].

Definition 3.1. A *Probabilistic Automaton (PA)* \mathcal{A} is a tuple (S, \bar{s}, Σ, D) , where S is a countable set of *states*, $\bar{s} \in S$ is the *start state*, Σ is a countable set of *actions*, and $D \subseteq S \times \Sigma \times \text{Disc}(S)$ is a *probabilistic transition relation*.

The set Σ is divided in two sets H and 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 τ range over internal actions. We also denote the generic elements of a probabilistic automaton \mathcal{A} by S, \bar{s}, Σ, D , and we propagate primes and indices when necessary. Thus, for example, the probabilistic automaton \mathcal{A}'_i has states S'_i , start state \bar{s}'_i , actions Σ'_i and transition relation D'_i .

A transition $tr = (s, a, \mu) \in D$, also denoted by $s \xrightarrow{a} \mu$, is said to *leave* from state s , to be *labelled* by a , and to *lead* to the *target* distribution μ , also denoted by μ_{tr} . We denote by $src(tr)$ the *source* state s and by $act(tr)$ the *action* a . We also say that s enables action a , that action a is enabled from s , and that (s, a, μ) is enabled from s . Finally, we let $D(a) = \{tr \in D \mid act(tr) = a\}$ be the set of transitions with label a .

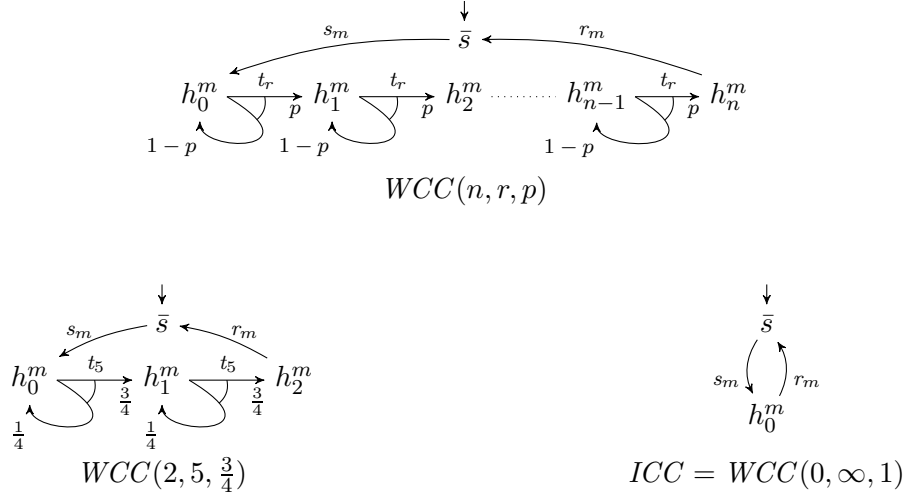


Figure 1: The wireless communication channel PA $WCC(n, r, p)$, its concrete instance $WCC(2, 5, \frac{3}{4})$, and the ideal communication channel PA $ICC = WCC(0, \infty, 1)$

If we restrict the nondeterminism in each state so that at the state enables at most one transition per action, we obtain the Markov decision process model [3, 18, 24]. This model is widely used in operations research and artificial intelligence literature to represent systems exhibiting both probabilistic and nondeterministic behaviours. Usually, the actual action labelling a transition is ignored while in the PA setting, actions are used for synchronisation on parallel composition, as we will see below.

Definition 3.2. A *Markov Decision Process (MDP)* \mathcal{M} is a tuple $(S, \bar{s}, \Sigma, P, r)$ that can be considered as a variation of a PA with a functional transition relation $P: S \times \Sigma \rightarrow \text{Disc}(S)$.

Since an MDP is a special case of PA, we adopt the same terminology and notation for both models. Given a state s , we denote by $\Sigma(s)$ the set of actions enabled by s , i.e., $\Sigma(s) = \{a \in \Sigma \mid P(s, a) \text{ is defined}\}$.

In this paper we consider only finite models, i.e., PAs or MDPs such that S and D (or P) are finite. Moreover, we assume that every state can be reached from the start state.

Example 3.3 (A wireless communication channel). As an example of a PA, consider a reliable wireless communication channel used to transmit messages belonging to the set Msg from a sender to a receiver. The wireless implementation of the communication channel is depicted in Figure 1: We graphically mark the start state \bar{s} of the automaton with an incoming arrow without source state; such arrow does not represent a transition of the automaton as other arrows do.

The PA $WCC(n, r, p)$ models a communication channel that requires n intermediate nodes (hops) to reach the receiver where the probability to transmit correctly the message from each node to the successor is p . Each intermediate node has a transmission radius r , and this parameter will become useful when determining the transmission cost in terms of power consumed, power that we aim to reduce. In this PA, the message m to transmit is obtained from the sender via the external s_m action and it is delivered to the receiver by using the external action r_m . Internal action t_r models the transmission of the message m from one node to the successor distant at most r , the transmission radius.

As a concrete example of wireless communication channel, we consider the case with two intermediate hops, a transmission radius of 5, and a success transmission probability of 0.75; the corresponding automaton is the *PA WCC*(2, 5, $\frac{3}{4}$) shown in the bottom left-hand part of Figure 1.

The ideal communication channel is modelled by the *PA WCC*(0, ∞ , 1), that is, the automaton that does not require intermediate nodes: The corresponding automaton is the *PA ICC* shown in the bottom right-hand part of Figure 1. Obviously, *ICC* models a reliable communication channel since the message is delivered with probability 1 just after having received it. \diamond

The following definition of parallel composition is just an equivalent rewriting of the definition provided in [27].

Definition 3.4. Given two *PAs* \mathcal{A}_1 and \mathcal{A}_2 , we say that \mathcal{A}_1 and \mathcal{A}_2 are *compatible* if $\Sigma_1 \cap \Sigma_2 = \emptyset = \mathbb{H}_1 \cap \mathbb{H}_2$.

Given two compatible *PAs* \mathcal{A}_1 and \mathcal{A}_2 , the *parallel composition* of \mathcal{A}_1 and \mathcal{A}_2 , denoted by $\mathcal{A}_1 \parallel \mathcal{A}_2$, is the probabilistic automaton $\mathcal{A} = (S, \bar{s}, \Sigma, D)$ where

- $S = S_1 \times S_2$,
- $\bar{s} = (\bar{s}_1, \bar{s}_2)$,
- $\Sigma = \mathbb{E} \cup \mathbb{H}$ where $\mathbb{E} = \mathbb{E}_1 \cup \mathbb{E}_2$ and $\mathbb{H} = \mathbb{H}_1 \cup \mathbb{H}_2$, and
- $((s_1, s_2), a, \mu_1 \times \mu_2) \in D$ if and only if
 - whenever $a \in \Sigma_1 \cap \Sigma_2$, $(s_1, a, \mu_1) \in D_1$ and $(s_2, a, \mu_2) \in D_2$,
 - whenever $a \in \Sigma_1 \setminus \Sigma_2$, $(s_1, a, \mu_1) \in D_1$ and $\mu_2 = \delta_{s_2}$, and
 - whenever $a \in \Sigma_2 \setminus \Sigma_1$, $(s_2, a, \mu_2) \in D_2$ and $\mu_1 = \delta_{s_1}$.

For $a \in \Sigma_1 \setminus \Sigma_2$, we denote by $(s_2, \nu_a, \delta_{s_2})$ the *apparent* internal transition corresponding to not performing any transition from s_2 in the composed transition, and similarly for $a \in \Sigma_2 \setminus \Sigma_1$.

For two compatible *PAs* \mathcal{A}_1 and \mathcal{A}_2 and their parallel composition $\mathcal{A}_1 \parallel \mathcal{A}_2$, we refer to \mathcal{A}_1 and \mathcal{A}_2 as the component automata and to $\mathcal{A}_1 \parallel \mathcal{A}_2$ as the composed automaton.

The composition of two compatible *MDPs* is not necessarily an *MDP* because the composed transition relation might become non-functional. Thus, *MDPs* are not closed under parallel composition, in contrast to *PAs*.

Example 3.5. As an example of parallel composition, consider the two automata \mathcal{A}_0 and \mathcal{A}_1 depicted in Figure 2. To keep the picture clear, we omitted the probability values since all transitions lead to a Dirac measure, as well as the τ label on internal transitions. The automaton $\mathcal{A}_0 \parallel \mathcal{A}_1$ is the fragment of the parallel composition of \mathcal{A}_0 and \mathcal{A}_1 reachable from the composed start state (\bar{s}_0, \bar{s}_1) . \diamond

3.1. Strong Probabilistic Bisimulation. Strong probabilistic bisimilarity is the base notion for our considerations [26, 27]. It uses the concept of combined transitions, defined as follows. Given a *PA* \mathcal{A} , a finite set I of indexes, a family $\{p_i \in \mathbb{R}^{>0}\}_{i \in I}$ such that $\sum_{i \in I} p_i = 1$, and a family $\{s \xrightarrow{a} \mu_i \in D\}_{i \in I}$, we say that s enables the *strong combined transition* (s, a, μ) , denoted by $s \xrightarrow{a}_C \mu$, if $\mu = \sum_{i \in I} p_i \cdot \mu_i$.

Definition 3.6. Let $\mathcal{A}_1, \mathcal{A}_2$ be two *PAs*. An equivalence relation \mathcal{R} on the disjoint union $S_1 \uplus S_2$ is a *strong probabilistic bisimulation* if, for each pair of states $s, t \in S_1 \uplus S_2$ such

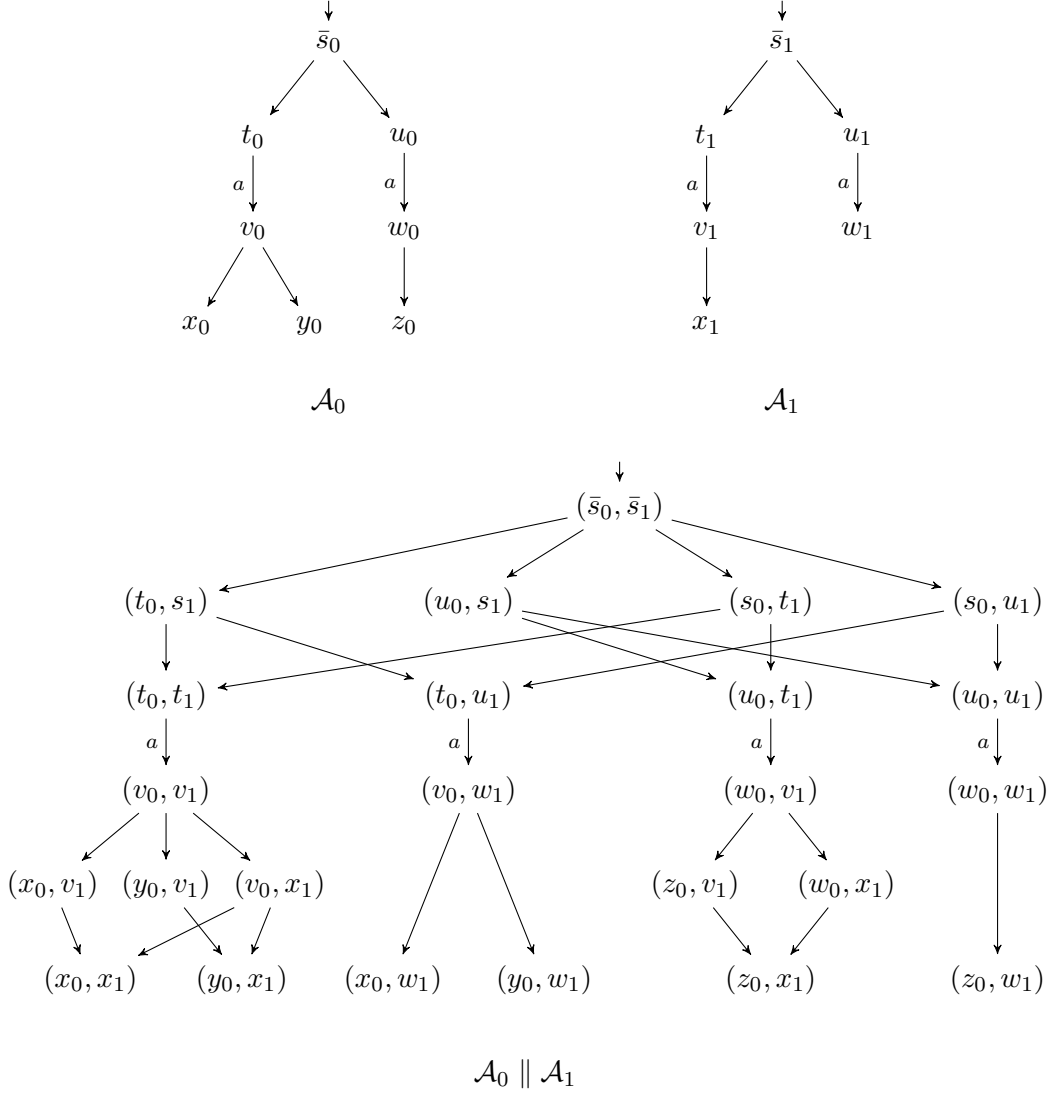


Figure 2: Example of parallel composition (fragment generated from the composed state (\bar{s}_0, \bar{s}_1))

that $s \mathcal{R} t$, if $s \xrightarrow{a} \mu_s$ for some probability distribution μ_s , then there exists μ_t such that $t \xrightarrow{a}_{\mathcal{C}} \mu_t$ and $\mu_s \mathcal{L}(\mathcal{R}) \mu_t$.

We say that \mathcal{A}_1 and \mathcal{A}_2 are strong probabilistic bisimilar if there exists a strong probabilistic bisimulation \mathcal{R} on $S_1 \uplus S_2$ such that $\bar{s}_1 \mathcal{R} \bar{s}_2$ and we say that two states s_1 and s_2 are strong probabilistic bisimilar if $s_1 \mathcal{R} s_2$. We denote strong probabilistic bisimilarity by \sim_p .

If $\xrightarrow{a}_{\mathcal{C}}$ is replaced by \xrightarrow{a} in the above definition, then *strong bisimilarity* (denoted \sim) results, as opposed to strong *probabilistic* bisimilarity as we define it here. So, the difference between these two relations is that the state t has to match the transition enabled by s by using an ordinary transition in strong bisimulation, and a combined transition in strong

probabilistic bisimulation. In the remainder of the paper, we may refer to s as the challenger, to t as the defender, and to the condition on transitions as the step condition.

It is known [26] that both strong and strong probabilistic bisimilarities are equivalence relations on the set of all PAs, and that they are the coarsest relations satisfying their respective bisimulation definitions. Furthermore, they are preserved by parallel composition, thus they are congruence relations with respect to parallel composition (and other algebraic operators such as nondeterministic choice and sequential composition).

3.2. Weak Transitions. In the setting of labelled transition systems, weak transitions are used to abstract from internal computations [23]. Intuitively, an internal weak transition is formed by an arbitrary long sequence of internal transitions, and an external weak transition is formed by an external transition preceded and followed by arbitrary long sequences of internal transitions. To lift this idea to the setting of probabilistic automata is a little intricate owed to the fact that transitions branch into distributions, and one thus has to work with tree-like objects instead of sequences, as detailed in the sequel.

An *execution fragment* of a PA \mathcal{A} is a finite or infinite sequence of alternating states and actions $\alpha = s_0 a_1 s_1 a_2 s_2 \dots$ starting from a state s_0 , also denoted by $first(\alpha)$, and, if the sequence is finite, ending with a state denoted by $last(\alpha)$, such that for each $i > 0$ there exists a transition $(s_{i-1}, a_i, \mu_i) \in D$ such that $\mu_i(s_i) > 0$. The *length* of α , denoted by $|\alpha|$, is the number of occurrences of actions in α . If α is infinite, then $|\alpha| = \infty$. Denote by $frags(\mathcal{A})$ the set of execution fragments of \mathcal{A} and by $frags^*(\mathcal{A})$ the set of finite execution fragments of \mathcal{A} . An execution fragment α is a *prefix* of an execution fragment α' , denoted by $\alpha \leq \alpha'$, if the sequence α is a prefix of the sequence α' . The *trace* $trace(\alpha)$ of α is the sub-sequence of external actions of α ; we denote by ε the empty trace and we define $trace(a) = a$ for $a \in \mathbf{E}$ and $trace(a) = \varepsilon$ for $a \in \mathbf{H}$.

A *scheduler* for a PA \mathcal{A} is a function $\sigma: frags^*(\mathcal{A}) \rightarrow \text{SubDisc}(D)$ such that for each $\alpha \in frags^*(\mathcal{A})$, $\sigma(\alpha) \in \text{SubDisc}(\{tr \in D \mid src(tr) = last(\alpha)\})$. Given a scheduler σ and a finite execution fragment α , the distribution $\sigma(\alpha)$ describes how transitions are chosen to move on from $last(\alpha)$. A scheduler σ and a state s induce a probability distribution $\mu_{\sigma,s}$ over execution fragments as follows. The basic measurable events are the cones of finite execution fragments, where the cone of α , denoted by C_α , is the set $\{\alpha' \in frags(\mathcal{A}) \mid \alpha \leq \alpha'\}$. The probability $\mu_{\sigma,s}$ of a cone C_α is defined recursively as follows:

$$\mu_{\sigma,s}(C_\alpha) = \begin{cases} 0 & \text{if } \alpha = t \text{ for a state } t \neq s, \\ 1 & \text{if } \alpha = s, \\ \mu_{\sigma,s}(C_{\alpha'}) \cdot \sum_{tr \in D(a)} \sigma(\alpha')(tr) \cdot \mu_{tr}(t) & \text{if } \alpha = \alpha'at. \end{cases}$$

Standard measure theoretical arguments ensure that $\mu_{\sigma,s}$ extends uniquely to the σ -field generated by cones. We call the resulting measure $\mu_{\sigma,s}$ a *probabilistic execution fragment* of \mathcal{A} and we say that it is generated by σ from s . Given a finite execution fragment α , we define $\mu_{\sigma,s}(\alpha)$ as $\mu_{\sigma,s}(\alpha) = \mu_{\sigma,s}(C_\alpha) \cdot \sigma(\alpha)(\perp)$, where $\sigma(\alpha)(\perp)$ is the probability of terminating the computation after α has occurred.

Definition 3.7. We say that there is a *weak combined transition* from $s \in S$ to $\mu \in \text{Disc}(S)$ labelled by $a \in \Sigma$, denoted by $s \xrightarrow{a}_C \mu$, if there exists a scheduler σ such that the following holds for the induced probabilistic execution fragment $\mu_{\sigma,s}$:

- (1) $\mu_{\sigma,s}(frags^*(\mathcal{A})) = 1$;
- (2) for each $\alpha \in frags^*(\mathcal{A})$, if $\mu_{\sigma,s}(\alpha) > 0$ then $trace(\alpha) = trace(a)$;

(3) for each state t , $\mu_{\sigma,s}(\{\alpha \in frags^*(\mathcal{A}) \mid last(\alpha) = t\}) = \mu(t)$.

In this case, we say that the weak combined transition $s \xrightarrow{a}_C \mu$ is induced by σ .

Although the definition of weak combined transitions is admittedly intricate, it is just the obvious extension of weak transitions on labelled transition systems to the setting with probabilities. We refer to Segala [27] for more details on weak combined transitions.

Example 3.8. Consider again the PA $WCC(2, 5, \frac{3}{4})$, depicted in Figure 1, and the weak combined transition $h_0^m \xrightarrow{\tau}_C \delta_{h_1^m}$. To simplify the notation, let us denote by τ the internal action t_5 of $WCC(2, 5, \frac{3}{4})$. In order to show that $h_0^m \xrightarrow{\tau}_C \delta_{h_1^m}$ is actually a weak combined transition of $WCC(2, 5, \frac{3}{4})$, we have to exhibit a scheduler σ inducing it. It is easy to verify that σ defined as: $\sigma(\alpha) = \delta_{h_0^m} \xrightarrow{\tau} \rho_0$ if $last(\alpha) = h_0^m$, δ_{\perp} otherwise, where $\rho_0 = \{(h_1^m, \frac{3}{4}), (h_0^m, \frac{1}{4})\}$, induces the transition $h_0^m \xrightarrow{\tau}_C \delta_{h_1^m}$. Consider, for instance, the probability of stopping in h_1^m , i.e., the sum of the probability of each finite execution fragment ending with h_1^m , i.e., execution fragments of the form $(h_0^m \tau)^{n+1} h_1^m (\tau h_1^m)^l$ where $l, n \in \mathbb{N}$; it is easy to derive that for $n \in \mathbb{N}$, $\mu_{\sigma, h_0^m}((h_0^m \tau)^{n+1} h_1^m) = 1 \cdot (1 \cdot \frac{1}{4})^n \cdot \frac{3}{4} \cdot 1 = (\frac{1}{4})^n \cdot \frac{3}{4}$ and that for $l, n \in \mathbb{N}$, $\mu_{\sigma, h_0^m}((h_0^m \tau)^{n+1} h_1^m (\tau h_1^m)^{l+1}) = 1 \cdot (1 \cdot \frac{1}{4})^n \cdot \frac{3}{4} \cdot (0 \cdot \frac{1}{4})^{l+1} \cdot 1 = 0$. Note that the factor 0 appearing in $\mu_{\sigma, h_0^m}((h_0^m \tau)^{n+1} h_1^m (\tau h_1^m)^{l+1})$ comes from the fact that for each α such that $last(\alpha) = h_1^m$, $\sigma(\alpha)(h^m \xrightarrow{\tau} \rho_1) = 0$, where $\rho_1 = \{(h_2^m, \frac{3}{4}), (h_1^m, \frac{1}{4})\}$. Hence we have that $\mu_{\sigma, h_0^m}(\{\alpha \in frags^*(\mathcal{A}) \mid last(\alpha) = h_1^m\}) = \mu_{\sigma, h_0^m}(\{(h_0^m \tau)^{n+1} h_1^m \mid n \in \mathbb{N}\}) + \mu_{\sigma, h_0^m}(\{(h_0^m \tau)^{n+1} h_1^m (\tau h_1^m)^{l+1} \mid l, n \in \mathbb{N}\}) = \sum_{n \in \mathbb{N}} (\frac{1}{4})^n \cdot \frac{3}{4} + 0 = \frac{3}{4} \cdot \frac{1}{1-\frac{1}{4}} = 1 = \delta_{h_1^m}(h_1^m)$.

◇

We say that there is a *hyper-transition* from $\rho \in \text{Disc}(S)$ to $\mu \in \text{Disc}(S)$ labelled by $a \in \Sigma$, denoted by $\rho \xrightarrow{a}_C \mu$, if there exists a family of weak combined transitions $\{s \xrightarrow{a}_C \mu_s\}_{s \in \text{Supp}(\rho)}$ such that $\mu = \sum_{s \in \text{Supp}(\rho)} \rho(s) \cdot \mu_s$. Given $s \xrightarrow{a} \rho$ and $\rho \xrightarrow{\tau}_C \mu$, we denote by $s \xrightarrow{a}_C \rho \xrightarrow{\tau}_C \mu$ the weak combined transition $s \xrightarrow{a}_C \mu$ obtained by concatenating $s \xrightarrow{a} \rho$ and $\rho \xrightarrow{\tau}_C \mu$ (cf. [22, Proposition 3.6]).

3.3. Weak Transition Compositions. Since we are working in a compositional setting, it will become important to discuss how weak transitions are composed via a parallel composition of PAs, respectively in what sense a weak transition of the composed system can be decomposed into component weak transitions.

Given two automata \mathcal{A}_0 and \mathcal{A}_1 , it is possible to construct a weak combined transition $(s_0, s_1) \xrightarrow{a}_C \mu_0 \times \mu_1$ for the composed automaton $\mathcal{A}_0 \parallel \mathcal{A}_1$ given two weak combined transitions $s_i \xrightarrow{a_i}_C \mu_i$ of the component automata, provided that actions a_0 and a_1 are either the same external action, or $a_i = a$ and $a_{1-i} = \tau$. The construction of the composed weak combined transition is quite easy and we illustrate it on the two automata \mathcal{A}_0 and \mathcal{A}_1 and their parallel composition $\mathcal{A}_0 \parallel \mathcal{A}_1$ shown in Figure 2.

Example 3.9. As weak combined transitions of \mathcal{A}_0 and \mathcal{A}_1 , consider the weak combined transitions $tr_0 = \bar{s}_0 \xrightarrow{a}_C \mu_0$ where $\mu_0 = \{(x_0, \frac{1}{4}), (y_0, \frac{1}{4}), (z_0, \frac{1}{2})\}$ and $tr_1 = \bar{s}_1 \xrightarrow{a}_C \mu_1$ where $\mu_1 = \{(x_1, \frac{1}{2}), (w_1, \frac{1}{2})\}$.

The expected composed weak combined transition of $\mathcal{A}_1 \parallel \mathcal{A}_2$ is $(\bar{s}_0, \bar{s}_1) \xrightarrow{a}_C \mu_{01}$ where the measure μ_{01} assigns value $\frac{1}{8}$ to the states (x_0, x_1) , (y_0, x_1) , (x_0, w_1) , and (y_0, w_1) and value $\frac{1}{4}$ to the states (z_0, x_1) and (z_0, w_1) .

It is easy to verify that both tr_1 and tr_2 are induced by the scheduler that chooses uniformly the transitions enabled by each state. More precisely, tr_0 is induced by the scheduler σ_0 defined as follows:

$$\sigma_0(\alpha) = \begin{cases} \{(\bar{s}_0 \xrightarrow{\tau} \delta_{t_0}, \frac{1}{2}), (\bar{s}_0 \xrightarrow{\tau} \delta_{u_0}, \frac{1}{2})\} & \text{if } \alpha = \bar{s}_0, \\ \{(t_0 \xrightarrow{a} \delta_{v_0}, 1)\} & \text{if } \alpha = \bar{s}_0\tau t_0, \\ \{(u_0 \xrightarrow{a} \delta_{w_0}, 1)\} & \text{if } \alpha = \bar{s}_0\tau u_0, \\ \{(v_0 \xrightarrow{\tau} \delta_{x_0}, \frac{1}{2}), (v_0 \xrightarrow{\tau} \delta_{y_0}, \frac{1}{2})\} & \text{if } \alpha = \bar{s}_0\tau t_0 a v_0, \\ \{(u_0 \xrightarrow{\tau} \delta_{z_0}, 1)\} & \text{if } \alpha = \bar{s}_0\tau u_0 a w_0, \text{ and} \\ \delta_{\perp} & \text{otherwise.} \end{cases}$$

Similarly, tr_1 is induced by the scheduler σ_1 defined as follows:

$$\sigma_1(\alpha) = \begin{cases} \{(\bar{s}_1 \xrightarrow{\tau} \delta_{t_1}, \frac{1}{2}), (\bar{s}_1 \xrightarrow{\tau} \delta_{u_1}, \frac{1}{2})\} & \text{if } \alpha = \bar{s}_1, \\ \{(t_1 \xrightarrow{a} \delta_{v_1}, 1)\} & \text{if } \alpha = \bar{s}_1\tau t_1, \\ \{(u_1 \xrightarrow{a} \delta_{w_1}, 1)\} & \text{if } \alpha = \bar{s}_1\tau u_1, \\ \{(v_1 \xrightarrow{\tau} \delta_{x_1}, 1)\} & \text{if } \alpha = \bar{s}_1\tau t_1 a v_1, \text{ and} \\ \delta_{\perp} & \text{otherwise.} \end{cases}$$

◇

For $i = 0, 1$, let σ_i be the scheduler inducing $s_i \xrightarrow{a_i} \mu_i$. Suppose that a_0 and a_1 are the same external action a . From (s_0, s_1) , extend each obtained execution fragment α_0 by scheduling the transition $(v_0, s_1) \xrightarrow{\tau} \mu_0 \times \delta_{s_1}$ with probability $\sigma_0(\alpha_0)(v_0 \xrightarrow{\tau} \mu_0)$ until no more internal transitions can be performed according to σ_0 (here we write $\sigma_0(\alpha_0)$ to mean $\sigma_0(\beta_0)$ where β_0 is α_0 where the second component of each state has been dropped).

Example 3.10. According to the above procedure, the scheduler σ_{01} that is expected to induce $(\bar{s}_0, \bar{s}_1) \xrightarrow{a} \mu_{01}$ performs the following choice:

$$\sigma_{01}((\bar{s}_0, \bar{s}_1)) = \{((\bar{s}_0, \bar{s}_1) \xrightarrow{\tau} \delta_{(t_0, \bar{s}_1)}, \frac{1}{2}), ((\bar{s}_0, \bar{s}_1) \xrightarrow{\tau} \delta_{(u_0, \bar{s}_1)}, \frac{1}{2})\}.$$

◇

When no more internal transitions can be performed according to σ_0 , extend each execution fragment α_0 with $last(\alpha_0) = (t_0, s_1)$ with the execution fragment α_1 obtained by scheduling the transitions $(t_0, v_1) \xrightarrow{\tau} \delta_{t_0} \times \mu_1$ with probability $\sigma_1(\alpha_1)(v_1 \xrightarrow{\tau} \mu_1)$ until no more internal transitions can be performed according to σ_1 .

Example 3.11. The resulting choices of the scheduler σ_{01} after this extension are the following:

$$\sigma_{01}(\alpha) = \begin{cases} \{((t_0, \bar{s}_1) \xrightarrow{\tau} \delta_{(t_0, t_1)}, \frac{1}{2}), ((t_0, \bar{s}_1) \xrightarrow{\tau} \delta_{(t_0, u_1)}, \frac{1}{2})\} & \text{if } \alpha = (\bar{s}_0, \bar{s}_1)\tau(t_0, \bar{s}_1), \\ \{((u_0, \bar{s}_1) \xrightarrow{\tau} \delta_{(u_0, t_1)}, \frac{1}{2}), ((u_0, \bar{s}_1) \xrightarrow{\tau} \delta_{(u_0, u_1)}, \frac{1}{2})\} & \text{if } \alpha = (\bar{s}_0, \bar{s}_1)\tau(u_0, \bar{s}_1). \end{cases}$$

◇

For each execution fragment $\alpha_0\alpha_1$, let (v_0, v_1) be $last(\alpha_0\alpha_1)$ and choose the transition $(v_0, v_1) \xrightarrow{a} \mu_0 \times \mu_1$ with probability equal to $\sigma_0(\alpha_0)(v_0 \xrightarrow{a} \mu_0) \cdot \sigma_1(\alpha_1)(v_1 \xrightarrow{a} \mu_1)$.

Example 3.12. The resulting choices of the scheduler σ_{01} after this extension are the following:

$$\sigma_{01}(\alpha) = \begin{cases} \{((t_0, t_1) \xrightarrow{a} \delta_{(v_0, v_1)}, 1)\} & \text{if } \alpha = (\bar{s}_0, \bar{s}_1)\tau(t_0, \bar{s}_1)\tau(t_0, t_1), \\ \{((t_0, u_1) \xrightarrow{a} \delta_{(v_0, w_1)}, 1)\} & \text{if } \alpha = (\bar{s}_0, \bar{s}_1)\tau(t_0, \bar{s}_1)\tau(t_0, u_1), \\ \{((u_0, t_1) \xrightarrow{a} \delta_{(w_0, v_1)}, 1)\} & \text{if } \alpha = (\bar{s}_0, \bar{s}_1)\tau(u_0, \bar{s}_1)\tau(u_0, t_1), \\ \{((u_0, u_1) \xrightarrow{a} \delta_{(w_0, w_1)}, 1)\} & \text{if } \alpha = (\bar{s}_0, \bar{s}_1)\tau(u_0, \bar{s}_1)\tau(u_0, u_1). \end{cases}$$

◇

Let $\alpha_0\alpha_1a(u_0, u_1)$ be one of the resulting execution fragments; extend $\alpha_0\alpha_1a(u_0, u_1)$ with the execution fragment α_{0a} obtained scheduling the transition $(v_0, u_1) \xrightarrow{\tau} \mu_0 \times \delta_{s_1}$ with probability $\sigma_0(\alpha_0a(u_0, u_1)\alpha_{0a})(v_0 \xrightarrow{\tau} \mu_0)$ until no more internal transitions can be performed according to σ_0 .

Example 3.13. The resulting choices of the scheduler σ_{01} after this extension are the following:

$$\sigma_{01}(\alpha) = \begin{cases} \{((v_0, v_1) \xrightarrow{\tau} \delta_{(x_0, v_1)}, \frac{1}{2}), ((v_0, v_1) \xrightarrow{\tau} \delta_{(y_0, v_1)}, \frac{1}{2})\} & \text{if } \alpha = (\bar{s}_0, \bar{s}_1)\tau(t_0, \bar{s}_1)\tau(t_0, t_1)a(v_0, v_1), \\ \{((v_0, w_1) \xrightarrow{\tau} \delta_{(x_0, w_1)}, \frac{1}{2}), ((v_0, w_1) \xrightarrow{\tau} \delta_{(y_0, w_1)}, \frac{1}{2})\} & \text{if } \alpha = (\bar{s}_0, \bar{s}_1)\tau(t_0, \bar{s}_1)\tau(t_0, u_1)a(v_0, w_1), \\ \{((w_0, v_1) \xrightarrow{\tau} \delta_{(z_0, v_1)}, 1)\} & \text{if } \alpha = (\bar{s}_0, \bar{s}_1)\tau(u_0, \bar{s}_1)\tau(u_0, t_1)a(w_0, v_1), \\ \{((w_0, w_1) \xrightarrow{\tau} \delta_{(z_0, w_1)}, 1)\} & \text{if } \alpha = (\bar{s}_0, \bar{s}_1)\tau(u_0, \bar{s}_1)\tau(u_0, u_1)a(w_0, w_1). \end{cases}$$

◇

When no more internal transitions can be performed according to σ_0 , extend each execution fragment $\alpha_0\alpha_1a(u_0, u_1)\alpha_{0a}$ with $last(\alpha_{0a}) = (x_0, u_1)$ with the execution fragment α_{1a} obtained by scheduling the transitions $(x_0, v_1) \xrightarrow{\tau} \delta_{x_0} \times \mu_1$ with probability equal to $\sigma_1(\alpha_1a(u_0, u_1)\alpha_{1a})(v_1 \xrightarrow{\tau} \mu_1)$ until no more internal transitions can be performed according to σ_1 . Since there may be finite execution fragments for which the resulting scheduler is still undefined, extend the scheduler by mapping such execution fragments to δ_{\perp} .

Example 3.14. The resulting choices of the scheduler σ_{01} after this extension are the following:

$$\sigma_{01}(\alpha) = \begin{cases} \{((x_0, v_1) \xrightarrow{\tau} \delta_{(x_0, x_1)}, 1)\} & \text{if } \alpha = (\bar{s}_0, \bar{s}_1)\tau(t_0, \bar{s}_1)\tau(t_0, t_1)a(v_0, v_1)\tau(x_0, v_1), \\ \{((y_0, v_1) \xrightarrow{\tau} \delta_{(y_0, x_1)}, 1)\} & \text{if } \alpha = (\bar{s}_0, \bar{s}_1)\tau(t_0, \bar{s}_1)\tau(t_0, t_1)a(v_0, v_1)\tau(y_0, v_1), \\ \{((z_0, v_1) \xrightarrow{\tau} \delta_{(z_0, x_1)}, 1)\} & \text{if } \alpha = (\bar{s}_0, \bar{s}_1)\tau(u_0, \bar{s}_1)\tau(u_0, t_1)a(w_0, v_1)\tau(z_0, v_1), \\ \delta_{\perp} & \text{otherwise.} \end{cases}$$

It is routine to verify that the scheduler σ_{01} induces $(\bar{s}_0, \bar{s}_1) \xrightarrow{a}_C \mu_{01}$ as desired. ◇

If a_0 and a_1 are not the same external action a , then the construction is similar, except for the scheduling of the transitions with label a (if external) and the extension with the subsequent internal transitions. It can be shown that this construction actually leads to the weak combined transition $(s_0, s_1) \xrightarrow{a}_C \mu_0 \times \mu_1$.

The inverse operation, that is, the decomposition of a weak combined transition of a composed automaton into weak combined transitions of the component automata, is possible as well. In fact, [26, Section 4.3.2] shows how identify two weak combined transitions, one for each component automaton, corresponding to the view that each component automaton has of the composed weak combined transition.

Definition 3.15. Given two compatible PAs \mathcal{A}_0 and \mathcal{A}_1 , let \mathcal{A} be their parallel composition $\mathcal{A}_0 \parallel \mathcal{A}_1$ and consider a weak combined transition $s \xrightarrow{a}_C \mu$ with $s = (s_0, s_1)$. The *projection* $\text{proj}_i(s \xrightarrow{a}_C \mu)$ of $s \xrightarrow{a}_C \mu$ on the component automaton \mathcal{A}_i is the weak combined transition $s_i \xrightarrow{a_i}_C \mu_i$ where $a_i = a$ if $a \in \Sigma_i$, $a_i = \tau$ otherwise, and for each $t_i \in S_i$, $\mu_i(t_i) = \sum_{t_{1-i} \in S_{1-i}} \mu(t_0, t_1)$.

Note that the action a_i labelling $s_i \xrightarrow{a_i}_C \mu_i$ depends on how \mathcal{A}_i considers the action a : if a is an action of \mathcal{A}_i (independently on whether it is internal or external), then $a_i = a$; otherwise, it means that a is an action only of \mathcal{A}_{1-i} , so the contribution of \mathcal{A}_i to $s \xrightarrow{a}_C \mu$ involves possibly only internal transitions, thus we use τ as label for $s_i \xrightarrow{a_i}_C \mu_i$.

The technical construction of $s_i \xrightarrow{a_i}_C \mu_i$ is rather involved (cf. [26, Section 4.3.2]), since it requires to manage correctly the probabilistic choices of the scheduler. Intuitively, for obtaining the projection on the component automaton \mathcal{A}_i , each execution fragment and the probabilistic execution fragment underlying the weak combined transition of the composed automaton is compressed by removing the pairs of actions and states corresponding to only a transition from the other component automaton \mathcal{A}_{1-i} , i.e., the composed transition involves an apparent transition for \mathcal{A}_i .

3.4. Weak Probabilistic Bisimulation. The above definition of weak combined transitions (Definition 3.7) naturally lead us to the definition of the weak counterpart of strong probabilistic bisimilarity, namely weak probabilistic bisimilarity [26, 27].

Definition 3.16. Let $\mathcal{A}_1, \mathcal{A}_2$ be two PAs. An equivalence relation \mathcal{R} on the disjoint union $S_1 \uplus S_2$ is a *weak probabilistic bisimulation* if, for each pair of states $s, t \in S_1 \uplus S_2$ such that $s \mathcal{R} t$, if $s \xrightarrow{a} \mu_s$ for some probability distribution μ_s , then there exists μ_t such that $t \xrightarrow{a}_C \mu_t$ and $\mu_s \mathcal{L}(\mathcal{R}) \mu_t$.

We say that \mathcal{A}_1 and \mathcal{A}_2 are weak probabilistic bisimilar if there exists a weak probabilistic bisimulation \mathcal{R} on $S_1 \uplus S_2$ such that $\bar{s}_1 \mathcal{R} \bar{s}_2$ and we say that two states s_1 and s_2 are weak probabilistic bisimilar if $s_1 \mathcal{R} s_2$. We denote weak probabilistic bisimilarity by \approx_p .

As happens for the strong case, it is known [26] that weak probabilistic bisimilarity is an equivalence relation on the set of all PAs, and that it is the coarsest relation satisfying its bisimulation definition. Furthermore, it is preserved by parallel composition, thus it is a congruence relation with respect to parallel composition.

Example 3.17. Consider any instance $WCC(n, r, p)$ and the ideal communication channel ICC . It is quite easy to verify that $ICC \approx_p WCC(n, r, p)$ for each $n \in \mathbb{N}$, $r \in \mathbb{R}^{\geq 0}$, and $p \in (0, 1]$, where the relation \mathcal{R} justifying $ICC \approx_p WCC(n, r, p)$ has for each $m \in \text{Msg}$ one class containing all h_i^m states and another class containing start states. This means, by transitivity of \approx_p , that $WCC(n, r, p) \approx_p WCC(n', r', p')$ for each possible value of $n, n' \in \mathbb{N}$, $r, r' \in \mathbb{R}^{\geq 0}$, and $p, p' \in (0, 1]$. \diamond

There exists also a notion of weak bisimulation on probabilistic automata, obtained by restricting the step condition of the weak probabilistic bisimulation to use only Dirac schedulers. This echoes the difference between strong and strong probabilistic bisimilarity. A Dirac scheduler is a scheduler σ such that for each execution fragment α , $\sigma(\alpha)$ is a Dirac distribution. The main problem with this weak bisimulation is that it is not transitive [9], opposed to weak probabilistic bisimilarity.

4. COST PROBABILISTIC AUTOMATA

We are now ready to discuss the cost augmented probabilistic automata model that will be in our focus. As already hinted at in Example 3.3, we consider as *cost* any kind of quantity associated with the transitions of the automaton \mathcal{A} . We aim to minimise these costs. We model the cost of the transitions by a function c that assigns to each transition a non-negative real value.

Definition 4.1. A *cost probabilistic automaton* (CPA) is a pair (\mathcal{A}, c) where $\mathcal{A} = (S, \bar{s}, \Sigma, D)$ is a probabilistic automaton and c , the *transition cost function*, is a total function $c: D \rightarrow \mathbb{R}^{\geq 0}$.

The above definition follows (and generalises) the standard definition of *reward structure* we find for MDPs (cf. [24, Section 2.1.3]):

Definition 4.2. A *Markov decision process with rewards* is a tuple $(S, \bar{s}, \Sigma, P, r)$ where (S, \bar{s}, Σ, P) is an MDP and $r: S \times \Sigma \rightarrow \mathbb{R}$ is a *reward function* or *structure*.

In this paper we consider only non-negative rewards, i.e., it is assumed as $r(s, a) \geq 0$ for each $(s, a) \in S \times \Sigma$. We usually interpret them as *transition costs*.

4.1. Strong Combined Transition Cost. The extension of costs from a single transition to a convex combination of transitions is straightforward: It is canonical to consider as cost the weighted sum of the costs of the transitions which are being combined. This corresponds directly to the expected reward criterion we find in the operations research literature [18].

Definition 4.3. Given a CPA (\mathcal{A}, c) , a finite set I of indexes, a family $\{p_i \in \mathbb{R}^{>0}\}_{i \in I}$ such that $\sum_{i \in I} p_i = 1$, and a family $\{s \xrightarrow{a} \mu_i \in D\}_{i \in I}$, let $s \xrightarrow{a}_{\mathcal{C}} \mu$ be the resulting strong combined transition. Then, the cost of $s \xrightarrow{a}_{\mathcal{C}} \mu$ is defined as

$$c(s \xrightarrow{a}_{\mathcal{C}} \mu) = \sum_{i \in I} p_i \cdot c(s \xrightarrow{a} \mu_i).$$

This definition will be used in the definition of cost-preserving strong probabilistic bisimilarity.

4.2. Weak Combined Transition Cost. While there is a canonical way to generalise transition costs to strong combined transitions, it is not so obvious how to faithfully extend this to weak combined transitions: There are several ways of extending the cost from a single transition to a sequence of transitions, and hence to a weak combined transition, and we elaborate on this in the sequel. A prominent possibility is to consider the weighted sum of the costs of all involved finite execution fragments. This approach matches the standard interpretation in the operations research literature for expected reward criteria [18, 24] for MDPs.

Definition 4.4. Given an MDP $\mathcal{M} = (S, \bar{s}, \Sigma, P, r)$, a finite execution fragment $\alpha = s_1 a_1 \dots s_n a_n s_{n+1} \in \text{frags}^*(\mathcal{M})$, and a policy π , let $\alpha|_i = s_1 a_1 \dots a_{i-1} s_i$ be the i -prefix of α , $r(\alpha) = \sum_{i=1}^n r(s_i, a_i)$, and $P^\pi(\alpha) = \delta_{\bar{s}}(s_1) \cdot \prod_{i=1}^n \pi(\alpha|_i)(a_i) \cdot P(s_i, a_i)(s_{i+1})$.

Then the *expected total reward with horizon N under policy π* is defined as $\mathbb{E}_N^\pi = \sum_{\alpha \in \{\alpha \in \text{frags}^*(\mathcal{M}) \mid |\alpha| = N\}} r(\alpha) \cdot P^\pi(\alpha)$.

Since probabilistic automata are a conservative extension of MDP , we extend this notion to weak transition costs by taking into account the resolution of the nondeterminism as induced by a given scheduler. This approach is similar to a radial characterisation of the cost: If we imagine the target probability distribution as the border of the execution, then each finite execution fragment is like a ray leaving the source state s and reaching such border where the execution stops. The cost of reaching the border is then the weighted sum $\sum_{\alpha \in frags^*(\mathcal{A})} c_\sigma(\alpha) \cdot \mu_{\sigma,s}(\alpha)$ of the cost $c_\sigma(\alpha)$ of each ray α that is given by the sum of the cost of each transition part of the ray weighted by the probability of such transition.

Definition 4.5. Given a $CPA (\mathcal{A}, c)$, a state s , an action a , a probability distribution μ , and a scheduler σ inducing the weak combined transition $s \xrightarrow{a}_C \mu$, we define the cost $c_\sigma(s \xrightarrow{a}_C \mu)$ of the weak combined transition $s \xrightarrow{a}_C \mu$ as

$$c_\sigma(s \xrightarrow{a}_C \mu) = \sum_{\alpha \in frags^*(\mathcal{A})} c_\sigma(\alpha) \cdot \mu_{\sigma,s}(\alpha)$$

where $c_\sigma(\alpha) = c_\sigma(\alpha') + \sum_{tr \in D(a)} c(tr) \cdot \hat{\sigma}(\alpha', t, a, tr)$ if $\alpha = \alpha'at$, 0 otherwise, and where $\hat{\sigma}: frags^*(\mathcal{A}) \times S \times \Sigma \times D \rightarrow \mathbb{R}^{\geq 0}$ is defined as:

$$\hat{\sigma}(\alpha, t, a, tr) = \begin{cases} \frac{\sigma(\alpha)(tr) \cdot \mu_{tr}(t)}{\sum_{tr \in D(a)} \sigma(\alpha)(tr) \cdot \mu_{tr}(t)} & \text{if } \sum_{tr \in D(a)} \sigma(\alpha)(tr) \cdot \mu_{tr}(t) > 0, \\ 0 & \text{otherwise.} \end{cases}$$

When the scheduler σ is clear from the context, we just write $c(s \xrightarrow{a}_C \mu)$.

In the above definition, the function $\hat{\sigma}(\alpha, t, a, tr)$ is used to normalise the contribution of each scheduled transition to the cost of the resulting finite execution fragment, so that the probabilistic effects of the choice of the scheduler and the transition are correctly managed only by $\mu_{\sigma,s}(\alpha)$. We remark that here $\mu_{\sigma,s}(\alpha) > 0$ implies that $trace(\alpha) = trace(a)$ and that $\mu_{\sigma,s}(\alpha)(\perp) > 0$, i.e., the cost of α is considered in the sum only when the computation stops. This ensures also the correctness of the definition, since it is not possible to account multiple times the same probability values. This is a particular consequence of the fact that each finite execution fragment is a measurable event, as explained in [22, Example 3.1].

When we restrict Definition 4.5 to $MDPs$, it coincides with Definition 4.4, so the definition of cost of the weak combined transition induced by a scheduler σ is a conservative extension of the definition of expected total reward with horizon N under policy π :

Proposition 4.6. *Given an $MDP \mathcal{M} = (S, \bar{s}, \Sigma, P)$ and a policy π , let \mathcal{A} be the $PA (S, \bar{s}, \Sigma, D)$ where $D = \{(s, a, P(s, a)) \mid s \in S, a \in \Sigma(s)\}$. For each $N \in \mathbb{N}$, $tr = (s, a, \mu) \in D$ and $\alpha \in frags^*(\mathcal{M})$, let $c(tr) = r(s, a)$, $\sigma(\alpha)(tr) = \pi(\alpha)(a)$ if $|\alpha| < N$, 0 otherwise, and $\bar{s} \xrightarrow{\tau}_C \mu$ be the weak combined transition of \mathcal{A} induced by the scheduler σ when all actions are considered as internal. Then, it holds that for each $N \in \mathbb{N}$,*

$$\mathbb{E}_N^\pi = c_\sigma(\bar{s} \xrightarrow{\tau}_C \mu).$$

Proof outline. The proof is based on a simple manipulation of the definition of expected total reward with horizon N under policy π . \square

Example 4.7. Consider the $CPA (WCC(2, 5, \frac{3}{4}), c)$ where c assigns cost 25 to each transition labelled by the internal action t_5 ; the weak combined transition $h_0^m \xrightarrow{t_5}_C \delta_{h_2^m}$ can be seen as the concatenation of the two transitions $h_0^m \xrightarrow{t_5}_C \delta_{h_1^m}$ and $h_1^m \xrightarrow{t_5}_C \delta_{h_2^m}$. It is routine to check that each $h_0^m \xrightarrow{tr}_C \delta_{h_1^m}$ (and similarly for $h_1^m \xrightarrow{t_5}_C \delta_{h_2^m}$) is induced by

the scheduler σ_0 such that $\sigma_0(\alpha) = \delta_{tr_0^m}$ if $last(\alpha) = h_0^m$, δ_{\perp} otherwise, where $tr_0^m = h_0^m \xrightarrow{t_5} \{(h_1^m, \frac{3}{4}), (h_0^m, \frac{1}{4})\}$. Now, consider the finite execution fragment $\alpha = (h_0^m t_5)^{n+1} h_1^m$: According to Definition 4.5, it has cost $c_{\sigma_0}(\alpha) = (n+1) \cdot 25$. The probability $\mu_{\sigma_0, h_0^m}(\alpha)$ of α is $\frac{1}{4}^n \cdot \frac{3}{4}$ while the probability of each $\alpha' \in frags^*(WCC(2, 5, \frac{3}{4})) \setminus \{(h_0^m t_5)^{n+1} h_1^m \mid n \in \mathbb{N}\}$ is 0, thus the cost of the transition $h_0^m \xrightarrow{t_5}_C \delta_{h_1^m}$ as induced by σ_0 is $c_{\sigma_0}(h_0^m \xrightarrow{t_5}_C \delta_{h_1^m}) = \sum_{n \in \mathbb{N}} (n+1) \cdot 25 \cdot \frac{1}{4}^n \cdot \frac{3}{4} = 25 \cdot \frac{3}{4} \cdot \sum_{n \in \mathbb{N}} (n+1) \cdot \frac{1}{4}^n = \frac{25 \cdot \frac{3}{4}}{\frac{1}{4}} \cdot \sum_{n \in \mathbb{N}} (n+1) \cdot \frac{1}{4}^{n+1} = \frac{25 \cdot \frac{3}{4}}{\frac{1}{4}} \cdot \frac{\frac{1}{4}}{(1-\frac{1}{4})^2} = \frac{25}{\frac{3}{4}}$, hence $h_0^m \xrightarrow{t_5}_C \delta_{h_2^m}$ has cost $2 \cdot \frac{25}{\frac{3}{4}}$.

By using a similar approach, it is easy to generalise the above result to the CPA $(WCC(n, r, p), c)$, where c assigns cost \mathbf{r} to each transition labelled by the internal action t_r , and the weak combined transition $h_0^m \xrightarrow{t_r}_C \delta_{h_n^m}$: the resulting cost for such transition is $n \cdot \frac{\mathbf{r}}{p}$. \diamond

According to Definition 4.5, we consider a weak transition cost as a radial characterisation of the cost: The cost of reaching the border, where the execution stops, is the weighted sum of the cost of each ray α that is given by the sum of the cost of each transition part of the ray weighted by the probability of such transition.

The same execution border can be reached also by an increasing sequence of concentric balls whose centre is the source state s and whose radii are the number of performed steps. In this case the overall cost is the cost of performing the first step from the start state s to the radius 1 ball plus the cost of the steps from radius 1 to radius 2 ball plus the cost of the steps from radius 2 to radius 3 ball, and so on, where the cost of each single transition is weighted by the probability of performing it. This construction can be formalised as follows:

Definition 4.8. Given a CPA (\mathcal{A}, c) , a state s , an action a , a probability distribution μ , and a scheduler σ inducing the weak combined transition $s \xrightarrow{a}_C \mu$, we define the cost $c_{\sigma}(s \xrightarrow{a}_C \mu)$ of $s \xrightarrow{a}_C \mu$ as

$$c_{\sigma}(s \xrightarrow{a}_C \mu) = \sum_{\alpha \in frags^*(\mathcal{A})} c_{\sigma, s}(\alpha)$$

where

$$c_{\sigma, s}(\alpha) = \begin{cases} \mu_{\sigma, s}(C_{\alpha'}) \cdot \sum_{tr \in D(b)} c(tr) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t) & \text{if } \alpha = \alpha'bt, \\ 0 & \text{otherwise} \end{cases}$$

As it can be expected, Definition 4.8 is an equivalent characterisation of Definition 4.5:

Proposition 4.9. Given a CPA (\mathcal{A}, c) , a state s , an action a , a probability distribution μ , and a scheduler σ inducing the weak combined transition $s \xrightarrow{a}_C \mu$, it holds that

$$c_{\sigma}^{Ray}(s \xrightarrow{a}_C \mu) = c_{\sigma}^{Ball}(s \xrightarrow{a}_C \mu)$$

where costs c_{σ}^{Ray} and c_{σ}^{Ball} are defined according to Definition 4.5 and 4.8, respectively.

Proof outline. The proof is based on a manipulation of the definition of the cost c_{σ}^{Ray} and of the probability of a cone, together with the fact that σ induces a weak combined transition. \square

The concentric ball characterisation will be taken up later in Section 7.1 where we discuss the algorithmic aspects of CPAs.

4.3. Parallel Composition on Cost Probabilistic Automata. We now extend the parallel composition operator to the cost setting. As expected, it is based on the underlying probabilistic automata definition. However, defining the cost of transitions in the parallel composition is not entirely obvious since we have to carefully define the cost of the resulting transitions.

Definition 4.10. Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , we say that (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) are *compatible* if the underlying PAs \mathcal{A}_1 and \mathcal{A}_2 are compatible.

Definition 4.11. Given two compatible CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , consider the parallel composition $\mathcal{A}_1 \parallel \mathcal{A}_2 = (S, \bar{s}, \Sigma, D)$ of \mathcal{A}_1 and \mathcal{A}_2 . Let $c: D \rightarrow \mathbb{R}^{\geq 0}$ be total; we say that c is *cost-preserving* if there exists a symmetric, zero-preserving, distributive (over convex combination) and monotone increasing function $f: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ such that, for each $((s_1, s_2), a, \mu_1 \times \mu_2) \in D$,

$$c((s_1, s_2), a, \mu_1 \times \mu_2) = \begin{cases} f(c_1(s_1, a, \mu_1), c_2(s_2, a, \mu_2)) & \text{if } a \in \Sigma_1 \cap \Sigma_2, \\ f(c_1(s_1, a, \mu_1), c_2(s_2, \nu_a, \delta_{s_2})) & \text{if } a \in \Sigma_1 \setminus \Sigma_2, \\ f(c_1(s_1, \nu_a, \delta_{s_1}), c_2(s_2, a, \mu_2)) & \text{if } a \in \Sigma_2 \setminus \Sigma_1, \text{ and} \\ 0 & \text{otherwise,} \end{cases}$$

where we extend c_i by $c_i(s_i, \nu_a, \delta_{s_i}) = 0$ for each apparent transition $(s_i, \nu_a, \delta_{s_i})$, $i = 1, 2$. We say that f is the *generator* of c and we denote it by $\text{gen}(c)$.

The generator function f describes how to compute the cost of a composed transition. A common instance of the function f is addition (+), since often the cost of the composed transition is just the sum of the cost of the component transitions. The generator function generalises this to arbitrary functions provided they are symmetric, zero-preserving, distributive (over convex combination), and monotone increasing. These are the properties needed to establish compositionality of the bisimulations we are going to introduce in the remainder of the section. The generator function allows us to obtain more general results for parallel composition than if restricting to addition.

Definition 4.12. Given two compatible CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) and a cost-preserving function c , the *parallel composition* of (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , denoted by $(\mathcal{A}_1, c_1) \parallel_c (\mathcal{A}_2, c_2)$, is the cost probabilistic automaton $(\mathcal{A}, c) = (\mathcal{A}_1 \parallel \mathcal{A}_2, c)$.

5. COST PRESERVING BISIMULATIONS

The previous section has discussed how costs can be lifted to strong combined, respectively weak combined transitions. This is now incorporated into the probabilistic bisimulations of PAs.

5.1. Strong Cost Preserving Bisimulations. Since a *CPA* is an ordinary *PA* enriched with a cost function, one might consider a naive lifting of *PA* strong (probabilistic) bisimulation, where two *CPAs* are strong (probabilistic) bisimilar if the underlying *PAs* are. However this definition obviously falls too short, since it may relate states with different cost behaviours. For this reason and following [13] we define a refined notion of bisimulation where each transition of the challenging state has to be matched by the defender state by enabling a strong (combined) transition, as in ordinary strong (probabilistic) bisimulation, and, in addition, the costs of challenging and defending transitions must agree. In other words, the extension of strong bisimulation and strong probabilistic bisimulation from *PAs* to *CPAs* enforces that additional to the original conditions, a defender must match the transition costs proposed by the challenger.

Definition 5.1. Given two *CPAs* (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , an equivalence relation \mathcal{R} on the disjoint union $S_1 \uplus S_2$ is a *strong probabilistic cost-preserving bisimulation* if for each pair of states $s, t \in S_1 \uplus S_2$ such that $s \mathcal{R} t$, if $s \xrightarrow{a} \mu_s$, then there exists μ_t such that $t \xrightarrow{a}_C \mu_t$, $\mu_s \mathcal{L}(\mathcal{R}) \mu_t$, and $c_d(t \xrightarrow{a}_C \mu_t) = c_c(s \xrightarrow{a} \mu_s)$ where c_d and c_c are the cost functions of the defender and the challenger *CPA*, respectively.

Two *CPAs* (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) are strong probabilistic cost-preserving bisimilar if there exists a strong probabilistic cost-preserving bisimulation \mathcal{R} on $S_1 \uplus S_2$ such that $\bar{s}_1 \mathcal{R} \bar{s}_2$. We denote strong probabilistic cost-preserving bisimilarity by \sim_p .

By using this definition of bisimulation, we have that states enabling transitions with different cost are no more bisimilar, since they do not respect cost constraints.

As in the pure probabilistic setting (Definition 3.6), the above definition can be twisted to give rise to a *strong cost-preserving bisimilarity*, denoted \sim , by replacing all occurrences of \xrightarrow{a}_C by \xrightarrow{a} . It is then trivial to show that two strong cost-preserving bisimilar *CPAs* are also strong probabilistic cost-preserving bisimilar.

5.2. Properties of Strong Cost Preserving Bisimilarities. The relations we have defined above inherit all the relevant properties from their cost-insensitive counterparts.

Proposition 5.2. *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , if $(\mathcal{A}_1, c_1) \sim (\mathcal{A}_2, c_2)$, then $(\mathcal{A}_1, c_1) \sim_p (\mathcal{A}_2, c_2)$.*

Proof. The result is immediate, since each transition is also a combined transition; this implies that the relation \mathcal{R} justifying $(\mathcal{A}_1, c_1) \sim (\mathcal{A}_2, c_2)$ is also a strong probabilistic cost-preserving bisimulation, thus $(\mathcal{A}_1, c_1) \sim_p (\mathcal{A}_2, c_2)$. \square

Another important property of both strong and strong probabilistic cost-preserving bisimilarities is that they are equivalence relations on the set of *CPAs*.

Proposition 5.3. *Strong and strong probabilistic cost-preserving bisimilarities are equivalence relations on the set of CPAs.*

Proof outline. Reflexivity and symmetry are trivial as they follow immediately from reflexivity and symmetry of the underlying equivalence relations. Transitivity is based on the equivalence relation witnessing the transitivity of the underlying bisimulation for *PAs*: with a simple manipulation of combined transitions, it is shown that the costs are preserved. \square

Strong probabilistic cost-preserving bisimilarity is the coarsest strong cost-preserving probabilistic bisimulation, and similarly for strong cost-preserving bisimilarity. As one may expect, both strong and strong probabilistic cost-preserving bisimilarities are preserved by parallel composition.

Proposition 5.4. *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , if $(\mathcal{A}_1, c_1) \sim_p (\mathcal{A}_2, c_2)$, then for each (\mathcal{A}_3, c_3) compatible with both (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) and each pair of cost-preserving functions c_l and c_r with $\text{gen}(c_l) = \text{gen}(c_r)$, $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3) \sim_p (\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$, and similarly for \sim .*

Proof outline. The proof is based on the relation justifying the compositionality of the underlying bisimulation between PAs; the preservation of the cost of the transitions is based on the properties of the cost-preserving functions c_l and c_r with $\text{gen}(c_l) = \text{gen}(c_r)$. \square

Since we have shown that both strong and strong probabilistic cost-preserving bisimilarities are equivalence relations, and are preserved by parallel composition, they are congruences.

In the previous proposition, as well as in the following Propositions 5.9, 6.4 and 6.10, we make use of the condition $\text{gen}(c_l) = \text{gen}(c_r)$ about the generators of the cost functions for the parallel composition, so we assume the same generator to be used on both sides. This condition is needed to transfer the cost of a composed transition in one automaton to an equivalent composed transition in the other automaton.

For simplicity, let us consider \sim and an action $a \in \Sigma_1 \cap \Sigma_3$. Given $\mathcal{A}_1 \sim \mathcal{A}_2$, we know that $\mathcal{A}_1 \parallel \mathcal{A}_3 \sim \mathcal{A}_2 \parallel \mathcal{A}_3$ is justified by the relation $\mathcal{R} = \mathcal{R}_{12} \times \mathcal{I}_3$ where \mathcal{R}_{12} is the strong bisimulation between \mathcal{A}_1 and \mathcal{A}_2 and \mathcal{I}_3 is the identity relation on S_3 . Given $(s_1, s_3) \mathcal{R} (s_2, s_3)$ and a challenging transition $(s_1, s_3) \xrightarrow{a} \mu_1 \times \mu_3$, the defender transition is $(s_2, s_3) \xrightarrow{a} \mu_2 \times \mu_3$; $s_2 \xrightarrow{a} \mu_2$ is the defender transition when s_2 is challenged by s_1 in the step condition of \mathcal{R}_{12} with the transition $s_1 \xrightarrow{a} \mu_1$.

The construction for the strong cost-preserving bisimulation is essentially the same; the only difference is that we have also to check the equality of the cost. To prove the equality, by definition we know that $c_l((s_1, s_3) \xrightarrow{a} \mu_1 \times \mu_3) = \text{gen}(c_l)(c_1(s_1 \xrightarrow{a} \mu_1), c_3(s_3 \xrightarrow{a} \mu_3))$ and that $c_r((s_2, s_3) \xrightarrow{a} \mu_2 \times \mu_3) = \text{gen}(c_r)(c_2(s_2 \xrightarrow{a} \mu_2), c_3(s_3 \xrightarrow{a} \mu_3))$. Moreover, from $(\mathcal{A}_1, c_1) \sim (\mathcal{A}_2, c_2)$, we know that $c_1(s_1 \xrightarrow{a} \mu_1) = c_2(s_2 \xrightarrow{a} \mu_2)$.

This means that $c_l((s_1, s_3) \xrightarrow{a} \mu_1 \times \mu_3) = \text{gen}(c_l)(c_1(s_1 \xrightarrow{a} \mu_1), c_3(s_3 \xrightarrow{a} \mu_3))$ is actually equal to $\text{gen}(c_l)(c_2(s_2 \xrightarrow{a} \mu_2), c_3(s_3 \xrightarrow{a} \mu_3))$. The condition $\text{gen}(c_l) = \text{gen}(c_r)$ allows us to derive $\text{gen}(c_l)(c_2(s_2 \xrightarrow{a} \mu_2), c_3(s_3 \xrightarrow{a} \mu_3)) = \text{gen}(c_r)(c_2(s_2 \xrightarrow{a} \mu_2), c_3(s_3 \xrightarrow{a} \mu_3)) = c_r((s_2, s_3) \xrightarrow{a} \mu_2 \times \mu_3)$, as needed. If we omit such condition, then there is no way to establish the equality of the costs of the two composed transitions.

5.3. Weak Cost Preserving Bisimulations. We now discuss options for weak bisimulations on CPAs, so as to ignore internal computations as long as these do not change the visible behaviour of the system. As we have mentioned towards the end of Section 3.4, weak bisimilarity on PAs is not transitive, and therefore we base our considerations solely on weak probabilistic bisimilarity.

We follow the intuition of the strong cost-preserving setting discussed in Section 5.1 and introduce a refined notion of weak probabilistic bisimulation where each transition $s \xrightarrow{a} \mu_s$ of the challenging state s has to be matched by the defender state t by enabling a weak combined transition $t \xrightarrow{a}_C \mu_t$ such that $\mu_s \mathcal{L}(\mathcal{R}) \mu_t$ as in ordinary weak probabilistic bisimulation, and, in addition, the costs of challenging and defending transitions must agree.

Definition 5.5. Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , an equivalence relation \mathcal{R} on the disjoint union $S_1 \uplus S_2$ is a *weak probabilistic cost-preserving bisimulation* if for each pair of states $s, t \in S_1 \uplus S_2$ such that $s \mathcal{R} t$, if $s \xrightarrow{a} \mu_s$, then there exists μ_t such that $t \xrightarrow{a}_C \mu_t$, $\mu_s \mathcal{L}(\mathcal{R}) \mu_t$, and $c_d(t \xrightarrow{a}_C \mu_t) = c_c(s \xrightarrow{a} \mu_s)$ where c_d and c_c are the cost functions of the defender and the challenger CPA, respectively.

Two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) are weak probabilistic cost-preserving bisimilar if there exists a weak probabilistic cost-preserving bisimulation \mathcal{R} on $S_1 \uplus S_2$ such that $\bar{s}_1 \mathcal{R} \bar{s}_2$. We denote weak probabilistic cost-preserving bisimilarity by \approx_p .

5.4. Properties of Weak Cost-Preserving Probabilistic Bisimilarity. A first property is that strong probabilistic cost-preserving bisimilarity is a special case of weak cost-preserving probabilistic bisimilarity. This result is straightforward, as every strong combined transition is also a weak combined transition.

Proposition 5.6. *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , if $(\mathcal{A}_1, c_1) \sim_p (\mathcal{A}_2, c_2)$, then $(\mathcal{A}_1, c_1) \approx_p (\mathcal{A}_2, c_2)$.*

Another desirable property is that weak probabilistic cost-preserving bisimilarity is an equivalence relation on the set of CPAs.

Proposition 5.7. *Weak probabilistic cost-preserving bisimilarity is an equivalence relation on the set of CPAs.*

Proof outline. Similarly to the proof of Proposition 5.3, reflexivity and symmetry are trivial as they follow immediately from reflexivity and symmetry of the underlying equivalence relations. Transitivity is based on the equivalence relation witnessing the transitivity of the underlying bisimulation for PAs: with a simple manipulation of the matching combined transitions, it is shown that the costs according to Definition 4.8 are preserved. \square

In order to show that weak probabilistic cost-preserving bisimilarity is preserved by parallel composition, we state a preliminary result concerning the cost of weak combined transitions of the parallel composition of two PAs.

Lemma 5.8. *Given two compatible CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) and a cost-preserving function c , let $(s_1, s_2) \xrightarrow{a}_C \mu_1 \times \mu_2$ be a weak combined transition of $(\mathcal{A}_1, c_1) \parallel_c (\mathcal{A}_2, c_2)$. Then, $c((s_1, s_2) \xrightarrow{a}_C \mu_1 \times \mu_2) = \text{gen}(c)(c_1(s_1 \xrightarrow{a}_C \mu_1), c_2(s_2 \xrightarrow{a}_C \mu_2))$, where for $i = 1, 2$, $s_i \xrightarrow{a}_C \mu_i = \text{proj}_i((s_1, s_2) \xrightarrow{a}_C \mu_1 \times \mu_2)$.*

Proof outline. The result follows from Definition 4.8, the properties of $\text{gen}(c)$, and simple manipulation of the terms of summations. \square

This lemma puts us in the position to show that weak probabilistic cost-preserving bisimulation is preserved by parallel composition.

Proposition 5.9. *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , if $(\mathcal{A}_1, c_1) \approx_p (\mathcal{A}_2, c_2)$, then for each (\mathcal{A}_3, c_3) compatible with both (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) and each pair of cost-preserving functions c_l and c_r with $\text{gen}(c_l) = \text{gen}(c_r)$, $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3) \approx_p (\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$.*

Proof outline. The result is based on the relation justifying the compositionality of the underlying weak probabilistic bisimulation on PAs; the equality of the costs is shown by Lemma 5.8. \square

Since we have shown that weak probabilistic cost-preserving bisimulation is an equivalence relation and that it is preserved by parallel composition, it is a congruence.

6. COST BOUNDING BISIMULATIONS

The definitions of strong, strong probabilistic, and especially weak probabilistic cost-preserving bisimulation allow us to relate different-size CPAs that have the same behaviour and the same cost. Since we are working in a setting where our aim is to minimise the cost while preserving the behaviour of a CPA (\mathcal{A}, c) , we will now discuss how to relax the cost equality in the bisimulation definitions so that one automaton has a cost that is at most the cost of the other one. We first consider the strong case, and then discuss the weak setting.

6.1. Strong Cost Bounding Bisimulations. We first consider an extension of the strong (probabilistic) cost-preserving bisimulation. The central idea is to require that the cost of the defender transition is at most the cost of the challenger transition.

Definition 6.1. Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , an equivalence relation \mathcal{R} on the disjoint union $S_1 \uplus S_2$ is a *minor cost strong probabilistic bisimulation* from (\mathcal{A}_1, c_1) to (\mathcal{A}_2, c_2) if \mathcal{R} is a strong probabilistic bisimulation for \mathcal{A}_1 and \mathcal{A}_2 and for each $s_2 \xrightarrow{a} \mu_2 \in D_2$ and each $s_1 \in S_1$ such that $s_2 \mathcal{R} s_1$, there exists μ_1 such that $s_1 \xrightarrow{a}_C \mu_1$, $\mu_2 \mathcal{L}(\mathcal{R}) \mu_1$, and $c_1(s_1 \xrightarrow{a}_C \mu_1) \leq c_2(s_2 \xrightarrow{a} \mu_2)$.

We say that (\mathcal{A}_1, c_1) is minor cost strong probabilistic bisimilar to (\mathcal{A}_2, c_2) if there exists a minor cost strong probabilistic bisimulation \mathcal{R} such that $\bar{s}_2 \mathcal{R} \bar{s}_1$. We denote minor cost strong probabilistic bisimilarity from (\mathcal{A}_1, c_1) to (\mathcal{A}_2, c_2) by $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$ and we say that (\mathcal{A}_1, c_1) is in minor cost strong probabilistic bisimilarity with (\mathcal{A}_2, c_2) .

Similar to Definitions 3.6 and 5.1, the above definition can be twisted to give rise to a *minor cost strong bisimilarity* (denoted \lesssim), obtained by replacing all occurrences of \xrightarrow{a}_C by $\xrightarrow{a}_>$.

6.1.1. Properties of Strong Cost Bounding Bisimilarities. We now discuss the core properties of the above relations.

As in the cost-preserving case, it is straightforward to show that two minor cost strong bisimilar CPAs are also minor cost strong probabilistic bisimilar.

Proposition 6.2. *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , if $(\mathcal{A}_1, c_1) \lesssim (\mathcal{A}_2, c_2)$, then $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$.*

Proof. The result is immediate, since each transition is also a combined transition. \square

Both minor cost strong and strong probabilistic bisimilarities are reflexive and transitive relations, but not symmetric, of course. Thus they are preorders.

Proposition 6.3. *Minor cost strong and strong probabilistic bisimilarities are preorders on the set of CPAs.*

Proof outline. Similarly to the proof of Proposition 5.3, reflexivity is trivial as it follows immediately from reflexivity of the underlying equivalence relations. Transitivity is based on the relation witnessing the transitivity of the underlying bisimulation for PAs: with a simple manipulation of the matching transitions, it is shown that the costs are correctly bounded. \square

Minor cost strong probabilistic bisimilarity is the coarsest minor cost strong probabilistic bisimulation, and similarly for minor cost strong bisimilarity. As we will discuss below, both minor cost strong and strong probabilistic bisimulations are preserved by parallel composition.

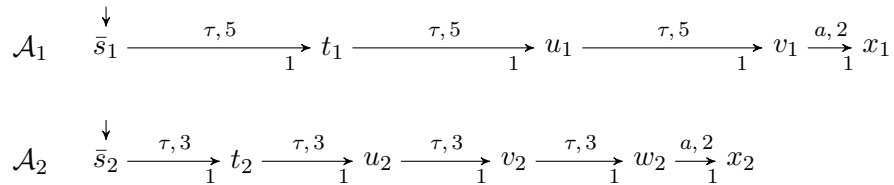
Proposition 6.4. *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , if $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$, then for each (\mathcal{A}_3, c_3) compatible with both (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) and each pair of cost-preserving functions c_l and c_r with $\text{gen}(c_l) = \text{gen}(c_r)$, $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3) \lesssim_p (\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$, and similarly for \lesssim .*

Proof outline. The proof is based on the relation justifying the compositionality of the underlying bisimulation between PAs; as for Proposition 5.4, the bounding of the cost of the transitions is based on the properties of the cost-preserving functions c_l and c_r with $\text{gen}(c_l) = \text{gen}(c_r)$. \square

Since we have shown that both minor cost strong and strong probabilistic bisimulations are preorders and they are preserved by parallel composition, they are precongruences.

6.2. Weak Cost Bounding Bisimulations. Also in the weak setting, we will relax the cost equality condition from Definition 5.5 by instead requiring that the cost of the defender matching transition is at most the cost of the challenger transition. Despite the simplicity of this idea, the formal definition is quite involved since we have to consider properly the cost of internal transitions.

To shed some light on this, consider an automaton \mathcal{A}_1 performing three internal steps $\bar{s}_1 \xrightarrow{\tau} \delta_{t_1}$, $t_1 \xrightarrow{\tau} \delta_{u_1}$, and $u_1 \xrightarrow{\tau} \delta_{v_1}$ where each step has cost 5 followed by an external step $v_1 \xrightarrow{a} \delta_{x_1}$ with cost 2 and an automaton \mathcal{A}_2 that performs four steps $\bar{s}_2 \xrightarrow{\tau} \delta_{t_2}$, $t_2 \xrightarrow{\tau} \delta_{u_2}$, $u_2 \xrightarrow{\tau} \delta_{v_2}$, and $v_2 \xrightarrow{\tau} \delta_{w_2}$ each with cost 3 followed by an external step $w_2 \xrightarrow{a} \delta_{x_2}$ with cost 2. Graphically, the two automata appear as follows, where we put the action and the cost as label of the transitions. The length of the arrows is proportional to the cost of the transition.



An external observer is able to recognise that the behaviour of \mathcal{A}_1 is more expensive than the one of \mathcal{A}_2 since the overall cost is 17 for the former, 14 for the latter. However, from a state-based bisimulation point of view, \mathcal{A}_2 is not always cheaper than \mathcal{A}_1 : Let $\{\{\bar{s}_1, \bar{s}_2\}, \{t_1, t_2\}, \{u_1, u_2\}, \{v_1, v_2, w_2\}, \{x_1, x_2\}\}$ be the equivalence classes of \mathcal{R} ; it is easy to verify that \mathcal{R} is a weak probabilistic bisimulation between \mathcal{A}_1 and \mathcal{A}_2 : When \mathcal{A}_1 performs $\bar{s}_1 \xrightarrow{\tau} \delta_{t_1}$ with cost 5, \mathcal{A}_2 replies with $\bar{s}_2 \xrightarrow{\tau} \delta_{t_2}$ with cost $3 \leq 5$ and $t_1 \mathcal{R} t_2$. Note that \mathcal{A}_2 can not perform the subsequent transition $t_2 \xrightarrow{\tau} \delta_{u_2}$ since in this case the overall cost would be $6 \not\leq 5$. The same happens for transitions $t_1 \xrightarrow{\tau} \delta_{u_1}$ and $u_1 \xrightarrow{\tau} \delta_{v_1}$ that are matched by $t_2 \xrightarrow{\tau} \delta_{u_2}$ and $u_2 \xrightarrow{\tau} \delta_{v_2}$, respectively. Since \mathcal{A}_1 now performs $v_1 \xrightarrow{a} \delta_{x_1}$ with cost 2, v_2 is not able to match this transition with a cost at most 2: In order to match the transition, \mathcal{A}_2 has to perform both transitions $v_2 \xrightarrow{\tau} \delta_{w_2}$ and $w_2 \xrightarrow{a} \delta_{x_2}$ whose cost is $5 \not\leq 2$.

These considerations indicate that internal challenger transitions should not be considered separately but as a whole, so in order to abstract away from costs of single challenger internal transitions while preserving the overall cost, we consider for the challenger the cost of reaching the border states, i.e., states where the automaton performs an external action or exhibits a different behaviour by changing the current class as induced by the weak bisimulation relation.

Definition 6.5. Given a PA \mathcal{A} and an equivalence relation \mathcal{R} over S , we say that a state s is a *border state* if there exists $s \xrightarrow{a} \mu \in D$ such that either $\mu([s]_{\mathcal{R}}) < 1$ or $a \in \mathbf{E}$.

We denote the set of all border states with respect to \mathcal{R} by $\mathcal{B}(\mathcal{R})$.

Definition 6.6. Let (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) be two CPAs. Let \mathcal{W} be an equivalence relation on the disjoint union $S_1 \uplus S_2$ and $\mathcal{C} \subseteq \mathcal{W} \cap (S_2 \times S_1)$ such that for each $s_2 \in S_2$ there exists $s_1 \in S_1$ such that $s_2 \mathcal{C} s_1$. Then we say that $(\mathcal{W}, \mathcal{C})$ is a *minor cost weak probabilistic bisimulation* from (\mathcal{A}_1, c_1) to (\mathcal{A}_2, c_2) if \mathcal{W} is a weak probabilistic bisimulation for \mathcal{A}_1 and \mathcal{A}_2 and for each $s_2 \xrightarrow{a} \mu_2 \in D_2$ and each $s_1 \in S_1$ such that $s_2 \mathcal{C} s_1$,

- (1) if there exists $\rho_2 \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_2)$ such that $\mu_2 \xrightarrow{\tau}_{\mathcal{C}} \rho_2$, then there exists $\rho_1 \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_1)$ such that
 - (a) $s_1 \xrightarrow{a}_{\mathcal{C}} \rho_1$,
 - (b) $\rho_2 \mathcal{L}(\mathcal{C}) \rho_1$,
 - (c) $c_1(s_1 \xrightarrow{a}_{\mathcal{C}} \rho_1) \leq c_2(s_2 \xrightarrow{a} \mu_2 \xrightarrow{\tau}_{\mathcal{C}} \rho_2)$, and
 - (d) $\min\{c_2(\mu_2 \xrightarrow{\tau}_{\mathcal{C}} \rho) \mid \rho \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_2)\} = c_2(\mu_2 \xrightarrow{\tau}_{\mathcal{C}} \rho_2)$; or
- (2) if there does not exist $\rho_2 \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_2)$ such that $\mu_2 \xrightarrow{\tau}_{\mathcal{C}} \rho_2$, then there exists $\mu_1 \in \text{Disc}(S_1)$ such that $s_1 \xrightarrow{a}_{\mathcal{C}} \mu_1$, $\mu_2 \mathcal{L}(\mathcal{C}) \mu_1$, and $c_1(s_1 \xrightarrow{a}_{\mathcal{C}} \mu_1) \leq c_2(s_2 \xrightarrow{a} \mu_2)$.

We say that (\mathcal{A}_1, c_1) is minor cost weak probabilistic bisimilar to (\mathcal{A}_2, c_2) if there exists a minor cost weak probabilistic bisimulation $(\mathcal{W}, \mathcal{C})$ such that $\bar{s}_2 \mathcal{C} \bar{s}_1$. We denote minor cost weak probabilistic bisimilarity from (\mathcal{A}_1, c_1) to (\mathcal{A}_2, c_2) by $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$ and we say that (\mathcal{A}_1, c_1) is in minor cost weak probabilistic bisimilarity with (\mathcal{A}_2, c_2) .

6.3. Properties of Minor Cost Weak Probabilistic Bisimilarity. A first property is that minor cost strong probabilistic bisimilarity is a special case of minor cost weak probabilistic bisimilarity. This result is rather easy, as every strong combined transition is also a weak combined transition.

Proposition 6.7. *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , if $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$, then $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$.*

Proof outline. The proof is based on the relation \mathcal{W} justifying $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$; the relation \mathcal{C} is constructed as $\mathcal{W} \cap (S_2 \times S_1)$ and the pair $(\mathcal{W}, \mathcal{C})$ shows $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$. \square

The proof idea can be reused to show that weak probabilistic cost-preserving bisimilarity is a special case of minor cost weak probabilistic bisimilarity.

Proposition 6.8. *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , if $(\mathcal{A}_1, c_1) \approx_p (\mathcal{A}_2, c_2)$, then $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$.*

Proof. The proof is a literal recapitulation of the proof of Proposition 6.7; the only difference being that when we match $t_2 \xrightarrow{b} \gamma_2$, instead of using the strong combined transition $t_1 \xrightarrow{b}_{\mathcal{C}} \gamma_1$, we use the corresponding weak combined transition $t_1 \xrightarrow{b}_{\mathcal{C}} \gamma_1$ that has cost $c_1(t_1 \xrightarrow{b}_{\mathcal{C}} \gamma_1) = c_2(t_2 \xrightarrow{b} \gamma_2)$. So it is immediate to derive that $c_1(s_1 \xrightarrow{a}_{\mathcal{C}} \rho_1) \leq$

$c_2(s_2 \xrightarrow{a} \mu_2 \xrightarrow{\tau} \rho_2)$ (as it is indeed $c_1(s_1 \xrightarrow{a} \rho_1) = c_2(s_2 \xrightarrow{a} \mu_2 \xrightarrow{\tau} \rho_2)$). Similarly for the challenging $s_2 \xrightarrow{a} \mu_2$ when μ_2 can not be extended to reach the border. \square

As for the strong case, we have that minor cost weak probabilistic bisimilarity is reflexive and transitive, thus it is a preorder. The proof of transitivity is not trivial.

Proposition 6.9. *Minor cost weak probabilistic bisimilarity is a preorder on the set of CPAs.*

Proof outline. The proof is rather involved and it is based on relations $\mathcal{W}_{31} = \mathcal{W}_{32} \circ \mathcal{W}_{21}$ and $\mathcal{C}_{31} = \mathcal{C}_{32} \circ \mathcal{C}_{21}$ obtained as composition of the relations $(\mathcal{W}_{21}, \mathcal{C}_{21})$ and $(\mathcal{W}_{32}, \mathcal{C}_{32})$ justifying $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$ and $(\mathcal{A}_2, c_2) \lesssim_p (\mathcal{A}_3, c_3)$, respectively. By manipulating the definition of weak combined transition, the properties of $(\mathcal{W}_{21}, \mathcal{C}_{21})$ and $(\mathcal{W}_{32}, \mathcal{C}_{32})$ allow us to show that $(\mathcal{W}_{31}, \mathcal{C}_{31})$ is a witness for $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_3, c_3)$. \square

Minor cost weak probabilistic bisimilarity is the coarsest minor cost weak probabilistic bisimulation. And as desired, minor cost weak probabilistic bisimulation is preserved by parallel composition.

Proposition 6.10. *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , if $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$, then for each (\mathcal{A}_3, c_3) compatible with both (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) and each pair of cost-preserving functions c_l and c_r with $\text{gen}(c_l) = \text{gen}(c_r)$, $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3) \lesssim_p (\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$.*

Proof outline. The proof is based on the pair of relations $\mathcal{W}_p = \mathcal{W} \times \mathcal{I}$ and $\mathcal{C}_p = \mathcal{C} \times \mathcal{I}$ obtained as cross-product of the relations $(\mathcal{W}, \mathcal{C})$ and $(\mathcal{I}, \mathcal{I})$ justifying $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$ and $(\mathcal{A}_3, c_3) \lesssim_p (\mathcal{A}_3, c_3)$, respectively. By manipulating the definition of weak combined transition, the properties of $(\mathcal{W}, \mathcal{C})$ and of c_l and c_r allow us to show that $(\mathcal{W}_p, \mathcal{C}_p)$ is a witness for $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3) \lesssim_p (\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$. \square

Since we have shown that minor cost weak probabilistic bisimilarity is a preorder and it is preserved by parallel composition, it is a precongruence.

6.4. The Cost of the Wireless Communication Channel. We now apply the minor cost weak probabilistic bisimulation to the reliable wireless communication channel introduced in Section 2 and depicted in Figure 1, page 5. As cost, we consider the function c that assigns cost 1 to transitions labelled by s_m or r_m and cost r^2 to transitions labelled by t_r . We use value 1 to represent a constant power consumption relative to sending/receiving message actions and value r^2 to model the energy, quadratic on the transmission radius, required to transmit a message via wireless.

As a concrete example, consider the two instances $\mathcal{A}_{23} = WCC(2, 3, \frac{1}{2})$ and $\mathcal{A}_{32} = WCC(3, 2, \frac{1}{2})$ of the wireless communication channel connecting sender and receiver that are at distance 6. To avoid name collisions, we rename the states h_j^m of $WCC(3, 2, \frac{1}{2})$ to k_j^m for $0 \leq j \leq 3$. It is easy to verify that the equivalence relation \mathcal{W} whose classes are $\{\bar{s}_{23}, \bar{s}_{32}\}$ and $\{h_i^m, k_j^m \mid 0 \leq i \leq 2, 0 \leq j \leq 3\}$ for each $m \in \text{Msg}$ justifies $\mathcal{A}_{23} \approx_p \mathcal{A}_{32}$, so consider the two CPAs (\mathcal{A}_{23}, c) and (\mathcal{A}_{32}, c) . We suspect that $(\mathcal{A}_{32}, c) \lesssim_p (\mathcal{A}_{23}, c)$, but not the reverse, since intuitively (\mathcal{A}_{23}, c) has overall cost 26 for sending and receiving a single message while (\mathcal{A}_{32}, c) has overall cost 38. In order to show $(\mathcal{A}_{32}, c) \lesssim_p (\mathcal{A}_{23}, c)$, we have to find a suitable relation \mathcal{C} that, together with \mathcal{W} , satisfies the conditions of Definition 6.6. A suitable relation is $\mathcal{C} = \{(\bar{s}_{23}, \bar{s}_{32})\} \cup \bigcup_{m \in \text{Msg}} \{(h_i^m, k_3^m) \mid 0 \leq i \leq 2\}$: Consider the pair $(\bar{s}_{23}, \bar{s}_{32})$ and the only available transition $\bar{s}_{23} \xrightarrow{s_m} \delta_{h_0^m}$. Since $\mathcal{B}(\mathcal{W}) = \{\bar{s}_{23}, \bar{s}_{32}\} \cup \{h_2^m, k_3^m \mid m \in \text{Msg}\}$,

the only possible $\rho_{23} \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_{23})$ such that $\delta_{h_0^m} \xrightarrow{\tau} \rho_{23}$ is $\rho_{23} = \delta_{h_2^m}$. In order to match such transition, \bar{s}_{32} enables the weak transition $\bar{s}_{32} \xrightarrow{s_m} \delta_{k_3^m}$ that satisfies $\delta_{h_2^m} \mathcal{L}(\mathcal{C}) \delta_{k_3^m}$. The last condition we have to verify is that $c(\bar{s}_{32} \xrightarrow{s_m} \delta_{k_3^m}) \leq c(\bar{s}_{23} \xrightarrow{s_m} \delta_{h_0^m} \xrightarrow{\tau} \delta_{h_2^m})$; this constraint is satisfied since $c(\bar{s}_{32} \xrightarrow{s_m} \delta_{k_3^m}) = 25$ while $c(\bar{s}_{23} \xrightarrow{s_m} \delta_{h_0^m} \xrightarrow{\tau} \delta_{h_2^m}) = 37$. It is routine to check the remaining pairs of states, thus $(\mathcal{A}_{32}, c) \lesssim_p (\mathcal{A}_{23}, c)$.

Now, assume $(\mathcal{A}_{23}, c) \lesssim_p (\mathcal{A}_{32}, c)$: By definition, it must hold that $\bar{s}_{32} \mathcal{C} \bar{s}_{23}$, so consider the transition $\bar{s}_{32} \xrightarrow{s_m} \delta_{k_0^m}$. For sure k_3^m and h_2^m are border states, as well as \bar{s}_{32} and \bar{s}_{23} . Moreover, \bar{s}_{32} and \bar{s}_{23} can not be related by \mathcal{W} to any other state as they are the only states performing s_m . Suppose that these are the only border states; this implies that $\bar{s}_{32} \xrightarrow{s_m} \delta_{k_0^m}$ has to be extended to $\bar{s}_{32} \xrightarrow{s_m} \delta_{k_0^m} \xrightarrow{\tau} \delta_{k_3^m}$ whose cost is 25. The only possibility for \bar{s}_{23} to match such transition while respecting the cost constraint is to perform the weak combined transition $\bar{s}_{23} \xrightarrow{s_m} \delta_{h_i^m}$ with $i = 0$ or $i = 1$ and $k_3^m \mathcal{C} h_i^m$. Note that we can not use $\bar{s}_{23} \xrightarrow{s_m} \delta_{h_2^m}$ since its cost is $37 \not\leq 25$. Independently on the chosen i , since $k_3^m \mathcal{C} h_i^m$ and $k_3^m \xrightarrow{r_m} \delta_{\bar{s}_{32}}, h_i^m$ has to perform the weak combined transition $h_i^m \xrightarrow{r_m} \delta_{\bar{s}_{23}}$ whose cost is $1 + 18 \cdot (2 - i) \not\leq 1$, so the condition is not satisfied. By applying the same approach to the case where we consider other states as border states, we can derive a similar failure, thus there does not exist any suitable cost relation \mathcal{C} with $\bar{s}_{32} \mathcal{C} \bar{s}_{23}$, hence $(\mathcal{A}_{23}, c) \not\lesssim_p (\mathcal{A}_{32}, c)$.

7. DECISION ALGORITHMS FOR BISIMULATIONS VIA LINEAR PROGRAMMING

In the previous sections we have discussed foundational properties of the cost probabilistic bisimulation variations. This section develops polynomial time decision algorithms for them. We focus on the intricacies faced when deciding the weak relations, algorithms for the strong relations are derived later.

To start with we revisit the ideas underlying the equivalence of weak transitions and linear programming problems, as developed in [15], and then extend this to the cost setting. At its core, and inspired by network flow problems, is the observation that one can view a transition $t \xrightarrow{\tau} \mu_t$ of the PA \mathcal{A} as a *flow* where the initial probability mass δ_t flows and splits along internal transitions according to (i) the transition target distributions and (ii) the scheduler resolutions of the nondeterminism occurring along the weak transition. Similarly, for $a \neq \tau$, i.e., $a \in \mathbf{E}$, one can view $t \xrightarrow{a} \mu_t$ as a *flow* flowing along internal transitions and exactly one transition with label a for each stream, again splitting in accordance with the transition target distributions and the scheduler resolutions of the nondeterminism.

From this observation one can derive an LP problem $t \xrightarrow{a} \mathcal{L}(\mathcal{E})\mu$, proposed in [15], used to validate or refute the existence of a weak combined transition $t \xrightarrow{a} \mu_t$ such that $\mu \mathcal{L}(\mathcal{E}) \mu_t$. Here it is assumed that \mathcal{E} is an equivalence relation on S ; but we can extend it to an arbitrary relation $\mathcal{R} \subseteq S \times S$ as follows: Checking that there exists μ_t such that $t \xrightarrow{a} \mu_t$ and $\mu \mathcal{L}(\mathcal{R}) \mu_t$ is equivalent, by properties of $\mathcal{L}(\cdot)$, to finding distributions μ_t and μ'_t such that $t \xrightarrow{a} \mu_t, \mu_t \mathcal{L}(\mathcal{I}) \mu'_t$, and $\mu \mathcal{L}(\mathcal{R}) \mu'_t$, where \mathcal{I} is the identity relation on S . Since verifying $\mu \mathcal{L}(\mathcal{R}) \mu'_t$ is itself equivalent [2, Lemma 5.1] to solving a maximum flow problem, such a flow problem can be merged with the $t \xrightarrow{a} \mathcal{L}(\mathcal{I})\mu'_t$ LP problem. This abstracts from the actual distribution μ'_t , so as to extend it to a binary relation \mathcal{R} , as we formalise in the sequel.

For a PA $\mathcal{A} = (S, \bar{s}, \Sigma, D)$ and $\mathcal{R} \subseteq S \times S$, for $a \in \mathbf{E}$, the network $G(t, a, \mathcal{R}) = (V, E)$ has the set of vertices $V = \{\Delta, \blacktriangledown\} \cup S \cup S^{tr} \cup S_a \cup S_a^{tr} \cup S_{\mathcal{R}}$ where

$$S^{tr} = \{v^{tr} \mid v \in S, tr = v \xrightarrow{b} \rho \in D, b \in \{a, \tau\}\},$$

$$\begin{aligned}
 S_a &= \{v_a \mid v \in S\}, \\
 S_a^{tr} &= \{v_a^{tr} \mid v^{tr} \in S^{tr}\}, \text{ and} \\
 S_{\mathcal{R}} &= \{s_{\mathcal{R}} \mid s \in S\}
 \end{aligned}$$

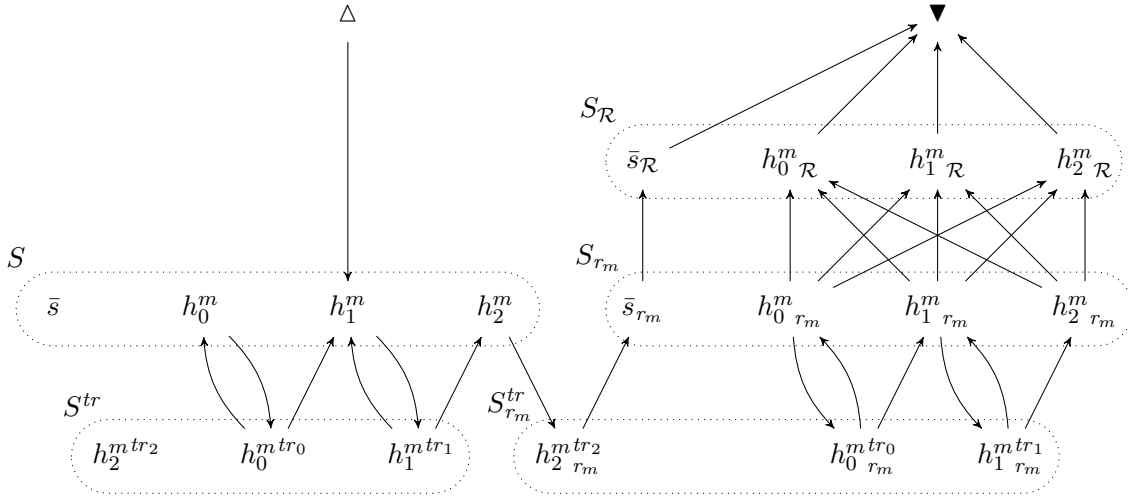
and the set of arcs is

$$\begin{aligned}
 E &= \{(\Delta, t)\} \cup \{(v_a, u_{\mathcal{R}}), (u_{\mathcal{R}}, \nabla) \mid u, v \in S, v \mathcal{R} u\} \\
 &\cup \{(v, v^{tr}), (v^{tr}, v'), (v_a, v_a^{tr}), (v_a^{tr}, v'_a) \mid tr = v \xrightarrow{\tau} \rho \in D, v' \in \text{Supp}(\rho)\} \\
 &\cup \{(v, v_a^{tr}), (v_a^{tr}, v'_a) \mid tr = v \xrightarrow{a} \rho \in D, v' \in \text{Supp}(\rho)\}.
 \end{aligned}$$

When instead $a \in \mathbb{H}$, the definition is simpler: $V = \{\Delta, \nabla\} \cup S \cup S^{tr} \cup S_{\mathcal{R}}$ and

$$\begin{aligned}
 E &= \{(\Delta, t)\} \cup \{(v, u_{\mathcal{R}}), (u_{\mathcal{R}}, \nabla) \mid u, v \in S, v \mathcal{R} u\} \\
 &\cup \{(v, v^{tr}), (v^{tr}, v') \mid tr = v \xrightarrow{\tau} \rho \in D, v' \in \text{Supp}(\rho)\}.
 \end{aligned}$$

Example 7.1. As an example of the construction of the network, consider the automaton $WCC(2, 5, \frac{3}{4})$ depicted in Figure 1, the state h_1^m , the action r_m , and the equivalence relation \mathcal{R} on states whose induced classes are $\{\bar{s}\}$ and $\{h_0^m, h_1^m, h_2^m\}$ for each message m . Denote the transitions of the automaton $WCC(2, 5, \frac{3}{4})$ as follows: $tr_s = \bar{s} \xrightarrow{s_m} \delta_{h_0^m}$, $tr_0 = h_0^m \xrightarrow{t_5} \{(h_0^m, \frac{1}{4}), (h_1^m, \frac{3}{4})\}$, $tr_1 = h_1^m \xrightarrow{t_5} \{(h_1^m, \frac{1}{4}), (h_2^m, \frac{3}{4})\}$, and $tr_2 = h_2^m \xrightarrow{r_m} \delta_{\bar{s}}$. The network $G(h_1^m, r_m, \mathcal{R})$ is as follows:



In the network we have also highlighted the different sets of vertices obtained from the states and the transitions of the automaton, by surrounding them with dotted lines. \diamond

As in [15], this network $G(t, a, \mathcal{R})$ and the associated maximum flow problem can not be used directly to encode a weak combined transition since it is not possible to force the flow to split proportional to the transition probability distributions. Instead an ordinary LP problem can be derived from the network, which is enriched with additional constraints called *balancing factors*. A balancing factor models a probabilistic choice and ensures a balance between flows that leave a vertex so as to respect the probability values in a probabilistic choice, i.e., when leaving a vertex $v \in S^{tr} \cup S_a^{tr}$.

Definition 7.2 (cf. [15, Definition 6]). Given a PA \mathcal{A} , $\mathcal{R} \subseteq S \times S$, $\mu \in \text{Disc}(S)$, and $t \in S$, for $a \in \mathbb{E}$ we define the $t \xrightarrow{a} \mathcal{L}(\mathcal{R})\mu$ LP problem associated to the network graph $(V, E) = G(t, a, \mathcal{R})$ as follows:

$$\begin{aligned}
& \max \sum_{(x,y) \in E} -f_{x,y} \\
& \text{under constraints} \\
& f_{u,v} \geq 0 \quad \text{for each } (u,v) \in E \\
& f_{\Delta,t} = 1 \\
& f_{v_{\mathcal{R}}, \blacktriangledown} = \mu(v) \quad \text{for each } v \in S_{\mathcal{R}} \\
& \sum_{u \in \{x \mid (x,v) \in E\}} f_{u,v} - \sum_{u \in \{y \mid (v,y) \in E\}} f_{v,u} = 0 \quad \text{for each } v \in V \setminus \{\Delta, \blacktriangledown\} \\
& f_{v^{tr}, v'} - \rho(v') f_{v, v^{tr}} = 0 \quad \text{for each } tr = v \xrightarrow{\tau} \rho \in D \text{ and } v' \in \text{Supp}(\rho) \\
& f_{v_a^{tr}, v'_a} - \rho(v') f_{v_a, v_a^{tr}} = 0 \quad \text{for each } tr = v \xrightarrow{\tau} \rho \in D \text{ and } v' \in \text{Supp}(\rho) \\
& f_{v_a^{tr}, v'_a} - \rho(v') f_{v, v_a^{tr}} = 0 \quad \text{for each } tr = v \xrightarrow{a} \rho \in D \text{ and } v' \in \text{Supp}(\rho)
\end{aligned}$$

When $a \in \mathbb{H}$, the LP problem $t \xrightarrow{\tau} \mathcal{L}(\mathcal{R})\mu$ associated to $G(t, \tau, \mathcal{R})$ is defined as above without the last two groups of constraints.

Example 7.3. Consider again the automaton $WCC(2, 5, \frac{3}{4})$ depicted in Figure 1, the state h_1^m , the action r_m , and the equivalence relation \mathcal{R} on states whose induced classes are $\{\bar{s}\}$ and $\{h_0^m, h_1^m, h_2^m\}$ for each message m . We have seen in the Example 7.1 the network $G(h_1^m, r_m, \mathcal{R})$. Consider the probability measure $\delta_{\bar{s}}$ and denote the transitions of the automaton $WCC(2, 5, \frac{3}{4})$ as follows: $tr_s = \bar{s} \xrightarrow{sm} \delta_{h_0^m}$, $tr_0 = h_0^m \xrightarrow{t_5} \{(h_0^m, \frac{1}{4}), (h_1^m, \frac{3}{4})\}$, $tr_1 = h_1^m \xrightarrow{t_5} \{(h_1^m, \frac{1}{4}), (h_2^m, \frac{3}{4})\}$, and $tr_2 = h_2^m \xrightarrow{r_m} \delta_{\bar{s}}$.

Besides the constraints for the non-negativity of the variables, the $h_1^m \xrightarrow{r_m} \mathcal{L}(\mathcal{R})\delta_{\bar{s}}$ LP problem associated to the network $G(h_1^m, r_m, \mathcal{R})$ has the following constraints:

- initial flow and challenging probabilities:

$$f_{\Delta, h_1^m} = 1 \quad f_{\bar{s}_{\mathcal{R}}, \blacktriangledown} = 1 \quad f_{h_0^m_{\mathcal{R}}, \blacktriangledown} = 0 \quad f_{h_2^m_{\mathcal{R}}, \blacktriangledown} = 0 \quad f_{h_2^m_{\mathcal{R}}, \blacktriangledown} = 0$$

- conservation of the flow for vertices in S :

$$\begin{aligned}
f_{h_0^{mtr_0}, h_0^m} - f_{h_0^m, h_0^{mtr_0}} &= 0 & f_{\Delta, h_1^m} + f_{h_0^{mtr_0}, h_1^m} + f_{h_1^{mtr_1}, h_1^m} - f_{h_1^m, h_1^{mtr_1}} &= 0 \\
f_{h_1^{mtr_1}, h_2^m} - f_{h_2^m, h_2^{mtr_2}} &= 0
\end{aligned}$$

- conservation of the flow for vertices in S^{tr} :

$$f_{h_0^m, h_0^{mtr_0}} - f_{h_0^{mtr_0}, h_0^m} - f_{h_0^{mtr_0}, h_1^m} = 0 \quad f_{h_1^m, h_1^{mtr_1}} - f_{h_1^{mtr_1}, h_1^m} - f_{h_1^{mtr_1}, h_2^m} = 0$$

- conservation of the flow for vertices in S_{r_m} :

$$\begin{aligned}
f_{h_2^{mtr_2}, \bar{s}_{r_m}} - f_{\bar{s}_{r_m}, h_2^m} &= 0 \\
f_{h_0^{mtr_0}, h_0^m} - f_{h_0^m, h_0^{mtr_0}} - f_{h_0^m, h_0^m} - f_{h_0^m, h_0^m} - f_{h_0^m, h_2^m} &= 0 \\
f_{h_0^{mtr_0}, h_1^m} + f_{h_1^{mtr_1}, h_1^m} - f_{h_1^m, h_1^{mtr_1}} - f_{h_1^m, h_0^m} - f_{h_1^m, h_1^m} - f_{h_1^m, h_2^m} &= 0 \\
f_{h_1^{mtr_1}, h_2^m} - f_{h_2^m, h_0^m} - f_{h_2^m, h_0^m} - f_{h_2^m, h_2^m} &= 0
\end{aligned}$$

- conservation of the flow for vertices in S^{tr} :

$$\begin{aligned}
f_{h_0^m, h_0^{mtr_0}} - f_{h_0^{mtr_0}, h_0^m} - f_{h_0^{mtr_0}, h_1^m} &= 0 \\
f_{h_1^m, h_1^{mtr_1}} - f_{h_1^{mtr_1}, h_1^m} - f_{h_1^{mtr_1}, h_2^m} &= 0 \\
f_{h_2^m, h_2^{mtr_2}} - f_{h_2^{mtr_2}, \bar{s}_{r_m}} &= 0
\end{aligned}$$

- conservation of the flow for vertices in $S_{\mathcal{R}}$:

$$\begin{aligned}
 f_{\bar{s}_{r_m}, \bar{s}_{\mathcal{R}}} - f_{\bar{s}_{\mathcal{R}}, \blacktriangledown} &= 0 \\
 f_{h_0^m, h_0^m} + f_{h_1^m, h_0^m} + f_{h_2^m, h_0^m} - f_{h_0^m, \blacktriangledown} &= 0 \\
 f_{h_0^m, h_1^m} + f_{h_1^m, h_1^m} + f_{h_2^m, h_1^m} - f_{h_1^m, \blacktriangledown} &= 0 \\
 f_{h_0^m, h_2^m} + f_{h_1^m, h_2^m} + f_{h_2^m, h_2^m} - f_{h_2^m, \blacktriangledown} &= 0
 \end{aligned}$$

- balancing constraints:

$$\begin{aligned}
 f_{h_0^m, h_0^m} - \frac{1}{4} \cdot f_{h_0^m, h_0^m} &= 0 & f_{h_0^m, h_1^m} - \frac{3}{4} \cdot f_{h_0^m, h_0^m} &= 0 \\
 f_{h_1^m, h_1^m} - \frac{1}{4} \cdot f_{h_1^m, h_1^m} &= 0 & f_{h_1^m, h_2^m} - \frac{3}{4} \cdot f_{h_1^m, h_1^m} &= 0 \\
 f_{h_2^m, h_2^m} - 1 \cdot f_{h_2^m, h_2^m} &= 0 & & \\
 f_{h_0^m, h_0^m} - \frac{1}{4} \cdot f_{h_0^m, h_0^m} &= 0 & f_{h_0^m, h_1^m} - \frac{3}{4} \cdot f_{h_0^m, h_0^m} &= 0 \\
 f_{h_1^m, h_1^m} - \frac{1}{4} \cdot f_{h_1^m, h_1^m} &= 0 & f_{h_1^m, h_2^m} - \frac{3}{4} \cdot f_{h_1^m, h_1^m} &= 0
 \end{aligned}$$

A solution that maximises the objective function assigns value 0 to all variables except for the following variables:

$$\begin{aligned}
 f_{\Delta, h_1^m} &= 1 & f_{h_1^m, h_1^m} &= \frac{4}{3} & f_{h_1^m, h_1^m} &= \frac{1}{3} & f_{h_1^m, h_2^m} &= 1 \\
 f_{h_2^m, h_2^m} &= 1 & f_{h_2^m, \bar{s}_{r_m}} &= 1 & f_{\bar{s}_{r_m}, \bar{s}_{\mathcal{R}}} &= 1 & f_{\bar{s}_{\mathcal{R}}, \blacktriangledown} &= 1
 \end{aligned}$$

It is not uncommon to have variables with value greater than 1, as happens for $f_{h_1^m, h_1^m}$, in particular when such variables correspond to edges in a cycle. \diamond

The LP problem $t \xrightarrow{a}_{\mathcal{C}} \mathcal{L}(\mathcal{R})\mu$ is equivalent to a weak combined transition modulo \mathcal{R} , in the sense that any feasible solution of the LP problem is enough to establish the transition (cf. [15, Theorem 8]). So the objective function has no impact, and this gives us room to use for instance some $\min \sum_{(x,y) \in E} c_{x,y} \cdot f_{x,y}$ as objective function. In this way, a weak transition can also be seen as a minimum cost flow problem plus balancing constraints. In the sequel we explore how to use the objective function to compute and minimise the cost of performing a weak combined transition.

7.1. Incorporating Transition Costs. In order to extend our computational approach to costs we revisit the concentric ball characterisation of weak transition cost from Definition 4.8. This is worthwhile when the weak combined transition is induced by a *determinate* scheduler [5], that is, a scheduler σ such that for each pair of finite execution fragments α, α' , if $\text{trace}(\alpha) = \text{trace}(\alpha')$ and $\text{last}(\alpha) = \text{last}(\alpha')$, then $\sigma(\alpha) = \sigma(\alpha')$. Under these schedulers, the resolution of the nondeterminism is the same for all finite execution fragments having the same trace and the same final state, so we can rearrange addends and factors in Definition 4.8 in order to express the cost of the weak combined transition as the sum of the cost of each transition weighted by the sum of the probabilities of performing it regardless of the ball we are considering. Since there is a strict relation between probabilities of reaching a state and the flow entering the corresponding vertex (cf. [15, Corollary 2]), the overall resulting cost is the sum of the cost of each transition $tr = u \xrightarrow{b} \rho$ multiplied by the flow from u to u^{tr} . This consideration is the base for the following definition, where we encode the transition costs in the LP problem as coefficients of the objective function.

Definition 7.4. Given a CPA (\mathcal{A}, c) , a binary relation \mathcal{R} on S , a probability distribution $\mu \in \text{Disc}(S)$, and a state $t \in S$, for action $a \in \mathbf{E}$ we define the min-cost LP problem $\min_c t \xrightarrow{a}_{\mathcal{C}} \mathcal{L}(\mathcal{R})\mu$ associated to the network $G(t, a, \mathcal{R})$ as follows.

$$\begin{aligned}
& \min \sum_{(x,y) \in E} c_f(x,y) \cdot f_{x,y} \\
& \text{under constraints} \\
& f_{u,v} \geq 0 \quad \text{for each } (u,v) \in E \\
& f_{\Delta,t} = 1 \\
& f_{v_{\mathcal{R}},\blacktriangledown} = \mu(v) \quad \text{for each } v \in S_{\mathcal{R}} \\
& \sum_{u \in \{x \mid (x,v) \in E\}} f_{u,v} - \sum_{u \in \{y \mid (v,y) \in E\}} f_{v,u} = 0 \quad \text{for each } v \in V \setminus \{\Delta, \blacktriangledown\} \\
& f_{v^{tr},v'} - \rho(v') f_{v,v^{tr}} = 0 \quad \text{for each } tr = v \xrightarrow{\tau} \rho \in D \text{ and } v' \in \text{Supp}(\rho) \\
& f_{v_a^{tr},v'_a} - \rho(v') f_{v_a,v_a^{tr}} = 0 \quad \text{for each } tr = v \xrightarrow{\tau} \rho \in D \text{ and } v' \in \text{Supp}(\rho) \\
& f_{v_a^{tr},v'_a} - \rho(v') f_{v,v_a^{tr}} = 0 \quad \text{for each } tr = v \xrightarrow{a} \rho \in D \text{ and } v' \in \text{Supp}(\rho)
\end{aligned}$$

where $c_f: E \rightarrow \mathbb{R}^{\geq 0}$ is a total function defined as follows:

$$c_f(x,y) = \begin{cases} c(tr) & \text{if } tr = v \xrightarrow{\tau} \rho, x = v, y = v^{tr}, \\ c(tr) & \text{if } tr = v \xrightarrow{\tau} \rho, x = v_a, y = v_a^{tr}, \\ c(tr) & \text{if } tr = v \xrightarrow{a} \rho, x = v, y = v_a^{tr}, \\ 0 & \text{otherwise.} \end{cases}$$

If $\min_c t \xrightarrow{a} \mathcal{L}(\mathcal{R})\mu$ has an optimal solution f^o , then we denote by \mathfrak{C} the minimum cost $\mathfrak{C} = \sum_{(x,y) \in E} c_f(x,y) \cdot f_{x,y}^o$.

When $a \in \mathbb{H}$, the min-cost LP problem $\min_c t \xrightarrow{\tau} \mathcal{L}(\mathcal{R})\mu$ associated to the network $G(t, \tau, \mathcal{R})$ is defined as above without the last two groups of constraints.

A first obvious result is that $\min_c t \xrightarrow{a} \mathcal{L}(\mathcal{R})\mu$ is feasible if and only if $t \xrightarrow{a} \mathcal{L}(\mathcal{R})\mu$ is feasible, since the only difference between the two problems is the objective function that does not affect the feasibility of an LP problem:

Proposition 7.5. *Given a CPA (\mathcal{A}, c) , $\mathcal{R} \subseteq S \times S$, $a \in \Sigma$, $\mu \in \text{Disc}(S)$, and $t \in S$, the minimisation LP problem $\min_c t \xrightarrow{a} \mathcal{L}(\mathcal{R})\mu$ has a feasible solution f^* if and only if f^* is a feasible solution of the LP problem $t \xrightarrow{a} \mathcal{L}(\mathcal{R})\mu$.*

Proof. The equivalence holds since $\min_c t \xrightarrow{a} \mathcal{L}(\mathcal{R})\mu$ and $t \xrightarrow{a} \mathcal{L}(\mathcal{R})\mu$ have the same set of constraints. \square

Similarly, as generating and checking the existence of a valid solution of the LP problem $t \xrightarrow{a} \mathcal{L}(\mathcal{R})\mu$ is polynomial in $N = \max\{|S|, |D|\}$ (cf. [15, Theorem 7]), the same holds for $\min_c t \xrightarrow{a} \mathcal{L}(\mathcal{R})\mu$:

Corollary 7.6. *Given a CPA (\mathcal{A}, c) , $\mathcal{R} \subseteq S \times S$, $a \in \Sigma$, $\mu \in \text{Disc}(S)$, and $t \in S$, generating and checking the existence of a valid solution of the minimisation LP problem $\min_c t \xrightarrow{a} \mathcal{L}(\mathcal{R})\mu$ is polynomial in $N = \max\{|S|, |D|\}$.*

Proof. The result follows immediately from Proposition 7.5 and [15, Theorem 7]. \square

Since $t \xrightarrow{a} \mathcal{L}(\mathcal{R})\mu$ is feasible if and only if there exists a scheduler σ that induces $t \xrightarrow{a} \mu_t$ such that $\mu \mathcal{L}(\mathcal{R}) \mu_t$, we may expect a similar result regarding costs, that is, $\min_c t \xrightarrow{a} \mathcal{L}(\mathcal{R})\mu$ is feasible with optimal value \mathfrak{C} if and only if there exists a scheduler σ that induces $t \xrightarrow{a} \mu_t$ such that $\mu \mathcal{L}(\mathcal{R}) \mu_t$ and $c(t \xrightarrow{a} \mu_t) = \mathfrak{C}$. Note that in general it is not possible to obtain such a result: There can be different ways to resolve nondeterminism, i.e., different schedulers, that induce the same weak combined transition but with different costs. Thus we can not talk about *the cost* of a weak combined transition, but of the cost

of the weak combined transition *as induced by the scheduler* σ . For instance, consider the automaton \mathcal{A} depicted on the right whose transitions are $tr_1 = \bar{s} \xrightarrow{a} \delta_t$, $tr_2 = \bar{s} \xrightarrow{\tau} \delta_v$, and $tr_3 = v \xrightarrow{a} \delta_t$, each one with cost 1.

It is straightforward to check that the scheduler σ_1 such that $\sigma_1(\bar{s}) = \delta_{tr_1}$ and $\sigma_1(\alpha) = \delta_\perp$ for each finite execution fragment $\alpha \neq \bar{s}$ induces the weak combined transition $\bar{s} \xrightarrow{a}_C \delta_t$ whose cost is 1; the same transition is induced also by the scheduler σ_2 defined as $\sigma_2(\bar{s}) = \delta_{tr_2}$, $\sigma_2(\bar{s}\tau v) = \delta_{tr_3}$, and $\sigma_2(\alpha) = \delta_\perp$ for each other finite execution fragment α . However the cost as induced by σ_2 is $c_{\sigma_2}(\bar{s} \xrightarrow{a}_C \delta_t) = 2 \neq 1 = c_{\sigma_1}(\bar{s} \xrightarrow{a}_C \delta_t)$; it is easy to show that $1 \leq c_\sigma(\bar{s} \xrightarrow{a}_C \delta_t) \leq 2$ for each scheduler σ inducing $\bar{s} \xrightarrow{a}_C \delta_t$. Note that there are uncountably many such schedulers, each one corresponding to a different resolution of the choice between $tr_1 = \bar{s} \xrightarrow{a} \delta_t$ and $tr_2 = \bar{s} \xrightarrow{\tau} \delta_v$: In general, we can denote such choice as the distribution $\{(tr_1, p), (tr_2, 1 - p)\}$ where $p \in [0, 1]$.

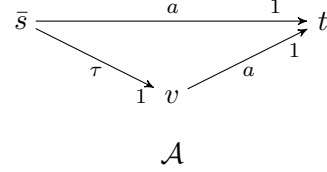


Figure 3: An automaton

The cost given by a scheduler and the value of the objective function of the corresponding LP problem are however related:

Theorem 7.7. *Given a CPA (\mathcal{A}, c) , $\mathcal{R} \subseteq S \times S$, $a \in \Sigma$, $\mu \in \text{Disc}(S)$, and $t \in S$, consider the $\min_c t \xrightarrow{a}_C \mathcal{L}(\mathcal{R})\mu$ LP problem. The following implications hold:*

- (1) *If there exists a scheduler σ for \mathcal{A} that induces $t \xrightarrow{a}_C \mu_t$ such that $\mu \mathcal{L}(\mathcal{R}) \mu_t$, then $\min_c t \xrightarrow{a}_C \mathcal{L}(\mathcal{R})\mu$ has an optimal solution f^o such that $\mathfrak{C} \leq c(t \xrightarrow{a}_C \mu_t)$.*
- (2) *If $\min_c t \xrightarrow{a}_C \mathcal{L}(\mathcal{R})\mu$ has an optimal solution f^o , then there exists a scheduler σ for \mathcal{A} that induces $t \xrightarrow{a}_C \mu_t$ such that $\mu \mathcal{L}(\mathcal{R}) \mu_t$ and $c(t \xrightarrow{a}_C \mu_t) = \mathfrak{C}$.*

Proof outline. The proof is mainly based on the proof of [15, Theorem 8]; the relations between the optimal value and the cost of the weak combined transition are shown by manipulating the summations in the objective function of the LP problem and of the cost of the weak combined transition, together with the fact that the flow incoming a vertex v in $S \cup S_a$ equals the sum of the probabilities of the cones of finite execution fragment ending with the state v . \square

As immediate corollaries we have that the cost given by the optimal solution of the $\min_c t \xrightarrow{a}_C \mathcal{L}(\mathcal{R})\mu$ LP problem corresponds to the minimum cost induced by any scheduler inducing $t \xrightarrow{a}_C \mu_t$ and that finding such minimum is polynomial.

Corollary 7.8. *Given a CPA (\mathcal{A}, c) , $\mathcal{R} \subseteq S \times S$, $a \in \Sigma$, $\mu \in \text{Disc}(S)$, and $t \in S$ such that there exists $t \xrightarrow{a}_C \mu_t$ with $\mu \mathcal{L}(\mathcal{R}) \mu_t$, the LP problem $\min_c t \xrightarrow{a}_C \mathcal{L}(\mathcal{R})\mu$ has minimum cost $\mathfrak{C} = \min\{c_\sigma(t \xrightarrow{a}_C \mu_t) \mid \sigma \text{ induces } t \xrightarrow{a}_C \mu_t \text{ such that } \mu \mathcal{L}(\mathcal{R}) \mu_t\}$.*

Proof. Let σ be a scheduler that induces the transition $t \xrightarrow{a}_C \mu_t$ with $\mu \mathcal{L}(\mathcal{R}) \mu_t$ such that $c_\sigma(t \xrightarrow{a}_C \mu_t) = \min\{c_\sigma(t \xrightarrow{a}_C \mu_t) \mid \sigma \text{ induces } t \xrightarrow{a}_C \mu_t\}$. By Theorem 7.7, we can derive that $\min_c t \xrightarrow{a}_C \mathcal{L}(\mathcal{R})\mu$ has an optimal solution f^o such that $\mathfrak{C} = \sum_{(x,y) \in E} c_f((x,y)) \cdot f_{x,y}^o \leq c_\sigma(t \xrightarrow{a}_C \mu_t)$, hence $\mathfrak{C} \leq \min\{c_\sigma(t \xrightarrow{a}_C \mu_t) \mid \sigma \text{ induces } t \xrightarrow{a}_C \mu_t \text{ such that } \mu \mathcal{L}(\mathcal{R}) \mu_t\}$.

Suppose, for the sake of contradiction, that there exists a scheduler σ inducing $t \xrightarrow{a}_C \mu_t$ such that $\mu \mathcal{L}(\mathcal{R}) \mu_t$ such that $c_\sigma(t \xrightarrow{a}_C \mu_t) < \mathfrak{C}$. Hence, by Theorem 7.7, we have that $\min_c t \xrightarrow{a}_C \mathcal{L}(\mathcal{R})\mu$ has a solution f^* , that is induced by the scheduler σ , such that $\sum_{(x,y) \in E} c_f((x,y)) \cdot f_{x,y}^* \leq c_\sigma(t \xrightarrow{a}_C \mu_t) < \mathfrak{C}$, but this contradicts the fact that f^o is optimal. Thus for each scheduler σ' , $c_{\sigma'}(t \xrightarrow{a}_C \mu_t) \geq \mathfrak{C}$. Theorem 7.7 implies also that there exists a

scheduler σ' inducing the transition $t \xrightarrow{a}_C \mu_t$ such that $\mu \mathcal{L}(\mathcal{R}) \mu_t$ and that $c_{\sigma'}(t \xrightarrow{a}_C \mu_t) = \mathfrak{C}$, thus $\mathfrak{C} = \min\{c_\sigma(t \xrightarrow{a}_C \mu_t) \mid \sigma \text{ induces } t \xrightarrow{a}_C \mu_t \text{ such that } \mu \mathcal{L}(\mathcal{R}) \mu_t\}$. \square

Corollary 7.9. *Given a CPA (\mathcal{A}, c) , $\mathcal{R} \subseteq S \times S$, $a \in \Sigma$, $\mu \in \text{Disc}(S)$, and $t \in S$, finding $\min\{c_\sigma(t \xrightarrow{a}_C \mu_t) \mid \sigma \text{ induces } t \xrightarrow{a}_C \mu_t \text{ such that } \mu \mathcal{L}(\mathcal{R}) \mu_t\}$ is polynomial in $N = \max\{|S|, |D|\}$.*

Proof. The result follows immediately from Corollaries 7.8 and 7.6. \square

Extending the above results to hyper-transitions of the CPA (\mathcal{A}, c) is straightforward, since we can consider each hyper-transition $\rho \xrightarrow{a}_C \mu$ as the weak combined transition $h \xrightarrow{a}_C \mu$ in the CPA (\mathcal{A}', c') that is (\mathcal{A}, c) enriched with the fresh state h and the transition $h \xrightarrow{\tau} \rho$ whose cost is set to 0.

7.2. Deciding Cost Bisimulations. We now show how we can decide in polynomial time the cost bisimulations we have presented for CPA in Section 4. We commence our discussion with the most intricate relation, minor cost weak probabilistic bisimulation, and then move on to the simpler weak probabilistic cost-preserving bisimulation. Only after that we consider the strong probabilistic cost relations and finally the strong cost relations.

7.2.1. Deciding Minor Cost Weak Probabilistic Bisimulation. In order to algorithmically decide whether $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$, we extend the polynomial decision procedure QUOTIENT that establishes whether $\mathcal{A}_1 \approx_p \mathcal{A}_2$ holds [15], to the MINORCOST algorithm depicted in Figure 4 that computes $(\mathcal{W}, \mathcal{C})$ justifying $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$: We first compute $\mathcal{W} = \text{QUOTIENT}(\mathcal{A}_1, \mathcal{A}_2)$ and then we consider as candidate cost relation $\mathcal{C} = \mathcal{C}'$ all pairs $s_2 \mathcal{W} s_1$ with $s_2 \in S_2$ and $s_1 \in S_1$. In the main loop of MINORCOST we repeatedly refine \mathcal{C} by removing all pairs that do not satisfy the conditions of Definition 6.6: If a check fails, we remove the offending pair (s_2, s_1) from \mathcal{C}' .

On termination of the loop, \mathcal{C} contains only pairs satisfying Definition 6.6, so deciding whether $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$ reduces to check whether $\bar{s}_2 \mathcal{C} \bar{s}_1$ and whether for each $s_2 \in S_2$ there exists $s_1 \in S_1$ such that $s_2 \mathcal{C} s_1$.

Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , let $N = \max\{|S_1 \uplus S_2|, |D_1 \uplus D_2|\}$. Computing $\mathcal{W} = \text{QUOTIENT}(\mathcal{A}_1, \mathcal{A}_2)$ is polynomial in N (cf. [15, Theorem 11]), say $P(N)$; in the worst case, that occurs when we remove all pairs from \mathcal{C} , the main loop of MINORCOST is performed at most N^2 times; according to Theorem 7.7 and its corollaries, finding $\rho_2 \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_2)$ such that $\mu_2 \xrightarrow{\tau}_C \rho_2$ and $c_2(\mu_2 \xrightarrow{\tau}_C \rho_2) = \min\{c_2(\mu_2 \xrightarrow{\tau}_C \rho) \mid \rho \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_2)\}$ is polynomial in N , say $R(N)$, by solving the $\min_c \mu_2 \xrightarrow{\tau}_C \mathcal{L}(\mathcal{B})\delta_{b_2}$ LP problem where $b_2 \in \mathcal{B}(\mathcal{W}) \cap S_2$ and \mathcal{B} is the reflexive, symmetric, and transitive closure of $\mathcal{B}(\mathcal{W})$. Similarly, $R(N)$ is also the complexity of either finding $\rho_1 \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_1)$ such that $s_1 \xrightarrow{a}_C \rho_1$, $\rho_2 \mathcal{L}(\mathcal{C}) \rho_1$, and $c_1(s_1 \xrightarrow{a}_C \rho_1) \leq c_2(s_2 \xrightarrow{a} \mu_2 \xrightarrow{\tau}_C \rho_2)$, or finding $\mu_1 \in \text{Disc}(S_1)$ such that $s_1 \xrightarrow{a}_C \mu_1$, $\mu_2 \mathcal{L}(\mathcal{C}) \mu_1$, and $c_1(s_1 \xrightarrow{a}_C \mu_1) \leq c_2(s_2 \xrightarrow{a} \mu_2)$. This implies that the total complexity of MINORCOST is $P(N) + N^2 \cdot 2R(N)$.

Theorem 7.10. *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , checking $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$ is polynomial in $N = \max\{|S_1 \uplus S_2|, |D_1 \uplus D_2|\}$.*

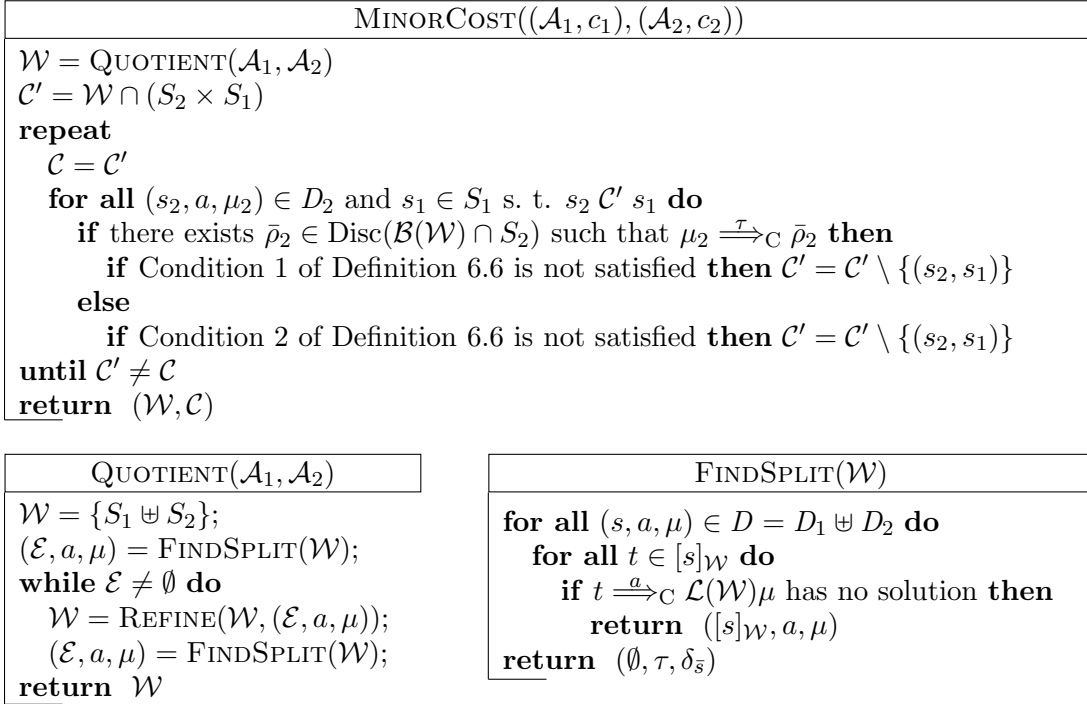


Figure 4: Minor cost weak bisimulation decision procedure

7.2.2. *Deciding Weak Probabilistic Cost-Preserving Bisimulation.* Regarding weak probabilistic cost-preserving bisimulation, the algorithm is actually simpler, since in order to check for the existence of weak combined transitions with a given cost \mathbf{c} , it is enough to add the new constraint $\sum_{(x,y) \in E} c_f((x,y)) \cdot f_{x,y} = \mathbf{c}$ to the $\min_c t \xrightarrow{a}_{\mathcal{C}} \mathcal{L}(\mathcal{R})\mu$ LP problem. This allows us to check in polynomial time whether two CPAs are weak probabilistic cost-preserving bisimilar: We compute QUOTIENT on the two CPAs where we have replaced in procedure FINDSPLIT the test for feasibility of $t \xrightarrow{a}_{\mathcal{C}} \mathcal{L}(\mathcal{W})\mu$ with the test for feasibility of $\min_c t \xrightarrow{a}_{\mathcal{C}} \mathcal{L}(\mathcal{R})\mu$ extended with the constraint $\sum_{(x,y) \in E} c_f((x,y)) \cdot f_{x,y} = c(s \xrightarrow{a} \mu_s)$.

Theorem 7.11. *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , checking $(\mathcal{A}_1, c_1) \approx_p (\mathcal{A}_2, c_2)$ is polynomial in $N = \max\{|S_1 \uplus S_2|, |D_1 \uplus D_2|\}$.*

7.2.3. *Deciding Strong Probabilistic Bisimulations on CPAs.* We now consider the decision algorithm for both minor cost strong probabilistic bisimulation and strong probabilistic cost-preserving bisimulation for the two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) ; by definition, the only difference between the two bisimulations is that the former requires that one cost is at most the other, while the latter requires that the two costs are the same. The remaining requirements are precisely those of strong probabilistic bisimulation on PAs, so we use its decision procedure as base: The main procedure is again QUOTIENT where we have replaced in procedure FINDSPLIT the test for feasibility of $t \xrightarrow{a}_{\mathcal{C}} \mathcal{L}(\mathcal{W})\mu$ with the test for feasibility of the following LP problem [31]: Let S be the disjoint union of the states of the two PAs; $t \xrightarrow{a} \mu_1, t \xrightarrow{a} \mu_2, \dots, t \xrightarrow{a} \mu_k$ be an enumeration of the transitions enabled by t with label a (we assume that $k > 0$, otherwise the test trivially fails); $\mathcal{W}(u) = \{u \in S \mid u \mathcal{W} v\}$;

and $\mathcal{W}^{-1}(v) = \{u \in S \mid u \mathcal{W} v\}$.

$$\begin{aligned} \sum_{i=1}^k p_i &= 1 && \text{for each } i \in \{1, \dots, k\} \\ 0 \leq p_i &\leq 1 && \text{for each } i \in \{1, \dots, k\} \\ 0 \leq f_{u,v} &\leq 1 && \text{for each } (u, v) \in \mathcal{W} \\ \mu(u) &= \sum_{v \in \mathcal{W}(u)} f_{u,v} && \text{for each } u \in S \\ \sum_{u \in \mathcal{W}^{-1}(v)} f_{u,v} &= \sum_{i=1}^k p_i \cdot \mu_i(v) && \text{for each } v \in S \end{aligned}$$

By [31, Lemma 5.3.1], we have that the above LP problem is feasible if and only if t enables a strong combined transition $t \xrightarrow{a}_C \mu_t$ such that $\mu \mathcal{L}(\mathcal{W}) \mu_t$; such strong combined transition $t \xrightarrow{a}_C \mu_t$ is actually obtained by combining each transition $t \xrightarrow{a} \mu_i$ with weight p_i if $p_i > 0$. It is immediate to see that the size of the above LP problem is polynomial in $N = \max\{|S_1 \uplus S_2|, |D_1 \uplus D_2|\}$, so it can be solved in polynomial time.

Extending this approach to the cost bisimulations we have presented is now trivial: it is enough to add the constraint $\sum_{i=1}^k p_i \cdot c_d(t \xrightarrow{a} \mu_i) = c_c(s \xrightarrow{a} \mu)$ when we decide the strong probabilistic cost-preserving bisimulation, or the constraint $\sum_{i=1}^k p_i \cdot c_d(t \xrightarrow{a} \mu_i) \leq c_c(s \xrightarrow{a} \mu)$ when we decide the minor cost strong probabilistic bisimulation. Obviously these two constraints do not change the complexity class of the LP problem, so we can check in polynomial time the strong probabilistic bisimulations on CPAs.

Theorem 7.12. *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , checking $(\mathcal{A}_1, c_1) \sim_p (\mathcal{A}_2, c_2)$ and $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$ is polynomial in $N = \max\{|S_1 \uplus S_2|, |D_1 \uplus D_2|\}$.*

7.2.4. Deciding Strong Bisimulations on CPAs. The last decision algorithm we propose allows us to check the minor cost strong bisimulation and the strong cost-preserving bisimulation for the two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) . As for the probabilistic case, the only difference between the two bisimulations is that the former asks that one cost is at most the other, while the latter asks them to be the same. The remaining requirements again coincide with strong bisimulation on PAs. This makes the algorithm straightforward, the main procedure is, as before, QUOTIENT where we replace in procedure FINDSPLIT the test for feasibility of $t \xrightarrow{a}_C \mathcal{L}(\mathcal{W})\mu$ with a test iterating over all transitions enabled by t with label a ; if among them we find a transition $t \xrightarrow{a} \mu_t$ such that $\mu \mathcal{L}(\mathcal{W}) \mu_t$, then the test succeeds, otherwise it fails. Since deciding $\mu \mathcal{L}(\mathcal{W}) \mu_t$ is equivalent to solving a network flow problem [2] whose number of arcs is quadratic in $|S_1 \uplus S_2|$, this iteration has a complexity that is in $\mathcal{O}(|D_t| \cdot |S_1 \uplus S_2|)$ (where D_t is the set of transitions with label a enabled by t). The cost condition can be incorporated in the obvious and straightforward manner: Before verifying $\mu \mathcal{L}(\mathcal{W}) \mu_t$ for the current $t \xrightarrow{a} \mu_t$, we check that $c_d(t \xrightarrow{a} \mu_t) \leq c_c(s \xrightarrow{a} \mu)$ provided we are computing the minor cost strong bisimulation, or that $c_d(t \xrightarrow{a} \mu_t) = c_c(s \xrightarrow{a} \mu)$ in case we are computing the strong cost-preserving bisimulation. This additional check does not increase the overall complexity, so also deciding the strong bisimulations on CPAs can be done in polynomial time.

Theorem 7.13. *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , checking $(\mathcal{A}_1, c_1) \sim (\mathcal{A}_2, c_2)$ and $(\mathcal{A}_1, c_1) \lesssim (\mathcal{A}_2, c_2)$ is polynomial in $N = \max\{|S_1 \uplus S_2|, |D_1 \uplus D_2|\}$.*

In summary we have devised polynomial time algorithms for all the cost related relations presented in this paper. For the strong relations, these are essentially echoing the strong (probabilistic) bisimulation decision algorithms [2, 31] for PAs, though presented in our setting. The central innovation and contribution of our algorithmic work arguably lies

in the treatment of weak transition costs, embodied in weak cost-preserving probabilistic bisimulation and minor cost weak probabilistic bisimulation.

8. DISCUSSION

This section puts our work in the greater context of related work and also discusses several other options to follow.

Givan, Dean and Greig [13] have introduced the idea of strong bisimilarity for *MDPs* with state and transition costs, together with algorithms for minimisation to the quotient model. The minimisation with respect to weak probabilistic bisimulation on *PA* has lately been discussed [10], and it remains to be investigated how the minimisation can be applied for the minor cost approach meaningfully. For the cost-preserving bisimilarity, the adaptations are straightforward, so we can indeed minimise with respect to weak transition costs.

In the present work we focus our attention on the minimisation of the cost, which is quite natural. The converse, i.e., maximisation of the cost, is hardly definable whenever the model contains cycles possibly occurring with probability 1. In such cases, the maximum cost is infinite, obtained by performing such cycles forever. Fairness arguments might be added to enforce to eventually leave such cycles almost surely, but still do not help since it does not change the cost suprema. This seems to indicate that we need to intertwine arguments about almost sure cycle termination with finite expectation in order to overcome this problem.

It is also interesting to discuss how state costs can be handled. Indeed it is possible to turn state costs to transition costs by moving them on incoming or outgoing transitions. The concrete choice makes a difference, because the labels of incoming and outgoing transitions generally differ. If already transitions costs were present prior to the move, we end up with a second cost structure. Multiple cost structures can indeed also be integrated into our setting rather easily, one just needs to take the minor cost for all structures in the decision problem.

For *MDPs*, multiple reward structures have been investigated [11] in the context of model checking, and our approach naturally combines with that. Chatterjee, Majumdar, and Henzinger [6] investigated them in a setting with discounting. In fact, our polynomial time LP approach can be extended to compute the minimum cost of discounted weak combined transitions, if we can assume a polynomially bounded number of internal steps. Conversely, one can compute an upper bound on discounted but non-polynomially bounded weak combined transitions in polynomial time.

If discounting is integrated into the weak bisimulation definitions we propose, this however induces difficult-to-grasp equalities. This is because sequences of internal transitions of different length are abstracted away by weak bisimilarity, but they would imply different discounts. For similar reasons, our cost model does by itself not talk about traces. As long as internal transitions carry nonzero costs, the definition of the cost of a weak trace is not obvious. Even if two execution fragments have the same trace, i.e., the same sequence of visible actions, different execution fragments usually have different costs when they involve different internal transitions, in particular after the last external action of the trace. Moreover, even if the execution fragment does not involve internal transitions, it can have different costs as resulting by the resolution of probabilistic and nondeterministic choices, the latter performed by the scheduler.

Still, cost-preserving bisimilarity implies equal trace costs, and if (\mathcal{A}_1, c_1) is in minor cost weak probabilistic bisimilarity with (\mathcal{A}_2, c_2) , then the trace costs of (\mathcal{A}_1, c_1) are bounded from above by (\mathcal{A}_2, c_2) . Trace costs appear central in many cost related formalisms not involving probabilities, such as weighted timed and energy automata [4, 25], though without (internal) actions playing a dedicated role here, so it is worth to investigate trace costs in the *CPA* model as well.

While minor cost weak bisimilarity is implicitly asymmetric, we have still formulated it as an equivalence relation. The wireless communication channel example has demonstrated that this approach is undoubtedly useful. Yet, it seems worthwhile to also take inspiration from simulation and simulation distance approaches [1, 29] in this matter. Another avenue that is worth exploring is to consider branching instead of weak bisimulation [28], not only because our notion of border states fits particularly well to that setting [12]. We conjecture that requiring the underlying weak bisimulations to be branching is enough to achieve the corresponding transitivity and compositional properties, as well as polynomial decision algorithms.

9. CONCLUDING REMARKS

In this paper we have presented an extension of Probabilistic Automata to Cost Probabilistic Automata and we have proposed cost related strong and weak probabilistic bisimulations on these models. We have distinguished cost-preserving and cost-bounding variations. The latter is based on the idea that the defender matches a transition with a cost that is bounded by at most the cost of the challenger. We have exemplified the use of this idea with a power sensitive wireless sensing example.

The bisimilarities introduced are precongruences, respectively congruences, with respect to parallel composition of *CPAs*, which sets the foundational ground for compositional construction and minimization approaches of *CPAs*. Especially in the minor cost weak setting the proofs to establish transitivity and precongruence are all but straightforward.

Moreover we have shown how to compute in polynomial time the minimum cost for each transition, and hence to decide each of the relations. With this, it is practically possible to roll out compositional construction and minimisation techniques to operations research, automated planning, and decision support applications. This is because, the *CPA* model encompasses *MDPs*, so the results apply readily to these models as well.

Acknowledgements. This work is supported by the DFG/NWO bilateral research programme ROCKS, by the DFG as part of the SFB/TR 14 AVACS, by the EU FP7 Programme under grant agreement no. 295261 (MEALS) and 318490 (SENSATION), and by the CAS/SAFEA International Partnership Program for Creative Research Teams. Part of this work has been done while Andrea Turrini was at Saarland University supported by the Cluster of Excellence “Multimodal Computing and Interaction” (MMCI), part of the German Excellence Initiative.

APPENDIX A. PROOFS OF THE RESULTS ENUNCIATED IN THE PAPER

In this appendix we provide the detailed non-trivial proofs for the results enunciated in the main part of the paper.

Result A.1 (Proposition 4.6). *Given an MDP $\mathcal{M} = (S, \bar{s}, \Sigma, P)$ and a policy π , let \mathcal{A} be the PA (S, \bar{s}, Σ, D) where $D = \{(s, a, P(s, a)) \mid s \in S, a \in \Sigma(s)\}$. For each $N \in \mathbb{N}$, $tr = (s, a, \mu) \in D$ and $\alpha \in \text{frags}^*(\mathcal{M})$, let $c(tr) = r(s, a)$, $\sigma(\alpha)(tr) = \pi(\alpha)(a)$ if $|\alpha| < N$, 0 otherwise, and $\bar{s} \xrightarrow{\tau}_C \mu$ be the weak combined transition of \mathcal{A} induced by the scheduler σ when all actions are considered as internal. Then, it holds that for each $N \in \mathbb{N}$,*

$$\mathbb{E}_N^\pi = c_\sigma(\bar{s} \xrightarrow{\tau}_C \mu).$$

Proof. Given $\alpha = s_1 a_1 s_2 a_2 \dots$, let $\text{frags}^*(\mathcal{M}, N)$ be $\{\alpha \in \text{frags}^*(\mathcal{M}) \mid |\alpha| = N\}$, α^i be the action a_i , and α_i be the state s_i .

$$\begin{aligned} \mathbb{E}_N^\pi &= \sum_{\alpha \in \text{frags}^*(\mathcal{M}, N)} r(\alpha) \cdot P^\pi(\alpha) \\ &= \sum_{\alpha \in \text{frags}^*(\mathcal{M}, N)} \left(\sum_{i=1}^N r(\alpha_i, \alpha^i) \right) \cdot \left(\delta_{\bar{s}}(\alpha_1) \cdot \prod_{i=1}^N \pi(\alpha|i)(\alpha^i) \cdot P(\alpha_i, \alpha^i)(\alpha_{i+1}) \right) \\ &= \sum_{\alpha \in \text{frags}^*(\mathcal{M}, N)} \delta_{\bar{s}}(\alpha_1) \cdot \left(\sum_{i=1}^N r(\alpha_i, \alpha^i) \right) \cdot \prod_{i=1}^N \pi(\alpha|i)(\alpha^i) \cdot P(\alpha_i, \alpha^i)(\alpha_{i+1}) \\ &= \sum_{s \in S} \sum_{\alpha \in \{\alpha \in \text{frags}^*(\mathcal{M}, N) \mid \alpha_1 = s\}} \delta_{\bar{s}}(s) \cdot \left(\sum_{i=1}^N r(\alpha_i, \alpha^i) \right) \cdot \prod_{i=1}^N \pi(\alpha|i)(\alpha^i) \cdot P(\alpha_i, \alpha^i)(\alpha_{i+1}) \\ &= \sum_{s \in S} \delta_{\bar{s}}(s) \cdot \sum_{\alpha \in \{\alpha \in \text{frags}^*(\mathcal{M}, N) \mid \alpha_1 = s\}} \left(\sum_{i=1}^N r(\alpha_i, \alpha^i) \right) \cdot \prod_{i=1}^N \pi(\alpha|i)(\alpha^i) \cdot P(\alpha_i, \alpha^i)(\alpha_{i+1}) \\ &= \sum_{\alpha \in \{\alpha \in \text{frags}^*(\mathcal{M}, N) \mid \alpha_1 = \bar{s}\}} \left(\sum_{i=1}^N r(\alpha_i, \alpha^i) \right) \cdot \prod_{i=1}^N \sigma(\alpha|i)(tr_i) \cdot \mu_{tr_i}(\alpha_{i+1}) \end{aligned}$$

where $tr_i = (\alpha_i, \alpha^i, P(\alpha_i, \alpha^i))$

$$\begin{aligned} &= \sum_{\alpha \in \{\alpha \in \text{frags}^*(\mathcal{M}, N) \mid \alpha_1 = \bar{s}\}} \left(\sum_{i=1}^N c(tr_i) \right) \cdot \prod_{i=1}^N \sigma(\alpha|i)(tr_i) \cdot \mu_{tr_i}(\alpha_{i+1}) \\ &\stackrel{\dagger}{=} \sum_{\alpha \in \{\alpha \in \text{frags}^*(\mathcal{M}, N) \mid \alpha_1 = \bar{s}\}} c_\sigma(\alpha) \cdot \prod_{i=1}^N \sigma(\alpha|i)(tr_i) \cdot \mu_{tr_i}(\alpha_{i+1}) \\ &\stackrel{\ddagger}{=} \sum_{\alpha \in \{\alpha \in \text{frags}^*(\mathcal{M}, N) \mid \alpha_1 = \bar{s}\}} c_\sigma(\alpha) \cdot \mu_{\sigma, \bar{s}}(\alpha) \\ &\stackrel{\S}{=} \sum_{\alpha \in \{\alpha \in \text{frags}^*(\mathcal{M}, N) \mid \alpha_1 = \bar{s}\}} c_\sigma(\alpha) \cdot \mu_{\sigma, \bar{s}}(\alpha) + \sum_{\alpha \in \{\alpha \in \text{frags}^*(\mathcal{M}) \mid |\alpha| \neq N \wedge \alpha_1 = \bar{s}\}} c_\sigma(\alpha) \cdot \mu_{\sigma, \bar{s}}(\alpha) \\ &= \sum_{\alpha \in \{\alpha \in \text{frags}^*(\mathcal{M}) \mid \alpha_1 = \bar{s}\}} c_\sigma(\alpha) \cdot \mu_{\sigma, \bar{s}}(\alpha) \end{aligned}$$

$$\begin{aligned}
&= \sum_{\alpha \in \{\alpha \in \text{frags}^*(\mathcal{M}) \mid \alpha_1 = \bar{s}\}} c_\sigma(\alpha) \cdot \mu_{\sigma, \bar{s}}(\alpha) + \sum_{\alpha \in \{\alpha \in \text{frags}^*(\mathcal{M}) \mid \alpha_1 \neq \bar{s}\}} c_\sigma(\alpha) \cdot \mu_{\sigma, \bar{s}}(\alpha) \\
&= \sum_{\alpha \in \text{frags}^*(\mathcal{M})} c_\sigma(\alpha) \cdot \mu_{\sigma, \bar{s}}(\alpha) \\
&= c_\sigma(s \xrightarrow{\tau} c \mu),
\end{aligned}$$

where equalities marked by †, ‡, and § are justified as follows:

- **Equality †:** we prove that $\mu_{\sigma, \bar{s}}(C_\alpha) = \prod_{i=1}^n \sigma(\alpha|i)(tr_i) \cdot \mu_{tr_i}(\alpha_{i+1})$ by induction on the length $n \leq N$ of α . If $n = 0$, then $\mu_{\sigma, \bar{s}}(C_\alpha) = \mu_{\sigma, \bar{s}}(C_s) = 1 = \prod_{i=1}^0 \sigma(\alpha|i)(tr_i) \cdot \mu_{tr_i}(\alpha_{i+1}) = \prod_{i=1}^n \sigma(\alpha|i)(tr_i) \cdot \mu_{tr_i}(\alpha_{i+1})$; if $n > 0$, then $\alpha = \alpha'at$ for some action a and state t , therefore

$$\begin{aligned}
\mu_{\sigma, \bar{s}}(C_\alpha) &= \mu_{\sigma, \bar{s}}(C_{\alpha'at}) \\
&= \mu_{\sigma, \bar{s}}(C_{\alpha'}) \cdot \sum_{tr \in D(a)} \sigma(\alpha')(tr) \cdot \mu_{tr}(t) \\
&= \left(\prod_{i=1}^{n-1} \sigma(\alpha|i)(tr_i) \cdot \mu_{tr_i}(\alpha_{i+1}) \right) \cdot \sum_{tr \in D(a)} \sigma(\alpha')(tr) \cdot \mu_{tr}(t)
\end{aligned}$$

by inductive hypothesis

$$= \left(\prod_{i=1}^{n-1} \sigma(\alpha|i)(tr_i) \cdot \mu_{tr_i}(\alpha_{i+1}) \right) \cdot \sigma(\alpha|n)(tr_n) \cdot \mu_{tr_n}(t)$$

where $tr_n = (\alpha'|n, a, P(\alpha'|n, a))$

$$\begin{aligned}
&= \left(\prod_{i=1}^{n-1} \sigma(\alpha|i)(tr_i) \cdot \mu_{tr_i}(\alpha_{i+1}) \right) \cdot \sigma(\alpha|n)(tr_n) \cdot \mu_{tr_n}(\alpha_{n+1}) \\
&= \prod_{i=1}^n \sigma(\alpha|i)(tr_i) \cdot \mu_{tr_i}(\alpha_{i+1})
\end{aligned}$$

where $\sum_{tr \in D(a)} \sigma(\alpha')(tr) \cdot \mu_{tr}(t)$ reduces to $\sigma(\alpha|n)(tr_n) \cdot \mu_{tr_n}(t)$ with $\alpha|n = \alpha'$ since by definition of MDP, there is only the transition tr_n with label a from state $\alpha|n$, thus for each other $tr \in D(a)$, $\sigma(\alpha')(tr) = 0$.

- **Equality ‡:** we assume, without loss of generality, that $\mu_{\sigma, \bar{s}}(C_\alpha) > 0$ and we prove that $c_\sigma(\alpha) = \sum_{i=1}^N c(tr_i)$ by induction on the length N of α . If $N = 0$, then $c_\sigma(\alpha) = 0 = \sum_{i=1}^0 c(tr_i) = \sum_{i=1}^N c(tr_i)$; if $N > 0$, then $\alpha = \alpha'at$ for some action a and state t , therefore

$$\begin{aligned}
c_\sigma(\alpha) &= c_\sigma(\alpha') + \sum_{tr_N \in D(a)} c(tr_N) \cdot \hat{\sigma}(\alpha', t, a, tr_N) \\
&= \left(\sum_{i=1}^{N-1} c(tr_i) \right) + \sum_{tr \in D(a)} c(tr) \cdot \hat{\sigma}(\alpha', t, a, tr)
\end{aligned}$$

by inductive hypothesis

$$= \sum_{i=1}^{N-1} c(tr_i) + c(tr_N) \cdot 1$$

where $tr_N = (\alpha' \downarrow N, a, P(\alpha' \downarrow N, a))$

$$= \sum_{i=1}^N c(tr_i)$$

since $\widehat{\sigma}(\alpha', t, a, tr)$ is 0 for each $tr \in D(a)$ that is different from tr_N , as $src(tr) \neq \alpha' \downarrow N$, thus $\sigma(\alpha')(tr) = 0$. When we consider the transition tr_N , we have that

$$\widehat{\sigma}(\alpha', t, a, tr_N) = \frac{\sigma(\alpha')(tr_N) \cdot \mu_{tr_N}(t)}{\sum_{tr \in D(a)} \sigma(\alpha')(tr) \cdot \mu_{tr}(t)} = \frac{\sigma(\alpha')(tr_N) \cdot \mu_{tr_N}(t)}{\sigma(\alpha')(tr_N) \cdot \mu_{tr_N}(t)} = 1,$$

thus $\sum_{tr \in D(a)} c(tr) \cdot \widehat{\sigma}(\alpha', t, a, tr) = c(tr_N)$ holds. Note that $\mu_{tr_N}(t) = P(\alpha' \downarrow N, a)(t) > 0$ follows by definition of finite execution fragment for $\alpha = \alpha'at$.

- **Equality §:** when $|\alpha| < N$, by definition of σ it follows that $\sigma(\alpha)(\perp) = 0$, hence $\mu_{\sigma, \bar{s}}(\alpha) = \mu_{\sigma, \bar{s}}(C_\alpha) \cdot \sigma(\alpha)(\perp) = 0$. When $|\alpha| > N$, by definition of σ it follows that $\sigma(\alpha \downarrow N)(\perp) = 1$, thus $\mu_{\sigma, \bar{s}}(C_\alpha) = 0$, hence $\mu_{\sigma, \bar{s}}(\alpha) = \mu_{\sigma, \bar{s}}(C_\alpha) \cdot \sigma(\alpha)(\perp) = 0$.

This concludes the proof that $\mathbb{E}_N^\pi = c_\sigma(s \xrightarrow{\tau}_C \mu)$. \square

Result A.2 (Proposition 4.9). *Given a CPA (\mathcal{A}, c) , a state s , an action a , a probability distribution μ , and a scheduler σ inducing the weak combined transition $s \xrightarrow{a}_C \mu$, it holds that*

$$c_\sigma^{Ray}(s \xrightarrow{a}_C \mu) = c_\sigma^{Ball}(s \xrightarrow{a}_C \mu)$$

where costs c_σ^{Ray} and c_σ^{Ball} are defined according to Definition 4.5 and 4.8, respectively.

Proof. As preliminary result, we show by induction that for each execution fragment $\alpha \in frags^*(\mathcal{A})$, if $\mu_{\sigma, s}(C_\alpha) > 0$, then $c_\sigma^{Ray}(\alpha) = \sum_{\alpha' \leq \alpha} \frac{c_{\sigma, s}^{Ball}(\alpha')}{\mu_{\sigma, s}(C_{\alpha'})}$. Note that $\mu_{\sigma, s}(C_\alpha) > 0$ implies that for each execution fragment α' such that $\alpha' \leq \alpha$, $\mu_{\sigma, s}(C_{\alpha'}) > 0$ as well.

Fix a generic execution fragment $\alpha \in frags^*(\mathcal{A})$ such that $\mu_{\sigma, s}(C_\alpha) > 0$; if $|\alpha| = 0$, then $c_\sigma^{Ray}(\alpha) = 0 = c_{\sigma, s}^{Ball}(\alpha) = \frac{c_{\sigma, s}^{Ball}(\alpha)}{\mu_{\sigma, s}(C_\alpha)} = \sum_{\phi \leq \alpha} \frac{c_{\sigma, s}^{Ball}(\phi)}{\mu_{\sigma, s}(C_\phi)}$.

Now, suppose that $|\alpha| > 0$; this implies that there exists α', a, t such that $\alpha = \alpha'at$. Thus,

$$\begin{aligned} c_\sigma^{Ray}(\alpha) &= c_\sigma^{Ray}(\alpha') + \sum_{tr \in D(a)} c(tr) \cdot \widehat{\sigma}(\alpha', t, a, tr) \\ &= c_\sigma^{Ray}(\alpha') + \sum_{tr \in D(a)} c(tr) \cdot \frac{\sigma(\alpha')(tr) \cdot \mu_{tr}(t)}{\sum_{tr' \in D(a)} \sigma(\alpha')(tr') \cdot \mu_{tr'}(t)} \\ &= c_\sigma^{Ray}(\alpha') + \sum_{tr \in D(a)} c(tr) \cdot \frac{\sigma(\alpha')(tr) \cdot \mu_{tr}(t)}{\sum_{tr' \in D(a)} \sigma(\alpha')(tr') \cdot \mu_{tr'}(t)} \\ &= c_\sigma^{Ray}(\alpha') + \sum_{tr \in D(a)} \frac{\mu_{\sigma, s}(C_{\alpha'})}{\mu_{\sigma, s}(C_{\alpha'})} \cdot c(tr) \cdot \frac{\sigma(\alpha')(tr) \cdot \mu_{tr}(t)}{\sum_{tr' \in D(a)} \sigma(\alpha')(tr') \cdot \mu_{tr'}(t)} \end{aligned}$$

$$\begin{aligned}
&= c_{\sigma}^{Ray}(\alpha') + \mu_{\sigma,s}(C_{\alpha'}) \cdot \sum_{tr \in D(a)} \frac{c(tr) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t)}{\mu_{\sigma,s}(C_{\alpha'}) \cdot \sum_{tr' \in D(a)} \sigma(\alpha')(tr') \cdot \mu_{tr'}(t)} \\
&= c_{\sigma}^{Ray}(\alpha') + \frac{\mu_{\sigma,s}(C_{\alpha'}) \cdot \sum_{tr \in D(a)} c(tr) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t)}{\mu_{\sigma,s}(C_{\alpha'}) \cdot \sum_{tr' \in D(a)} \sigma(\alpha')(tr') \cdot \mu_{tr'}(t)} \\
&= c_{\sigma}^{Ray}(\alpha') + \frac{c_{\sigma,s}^{Ball}(\alpha)}{\mu_{\sigma,s}(C_{\alpha})} \\
&= \frac{c_{\sigma,s}^{Ball}(\alpha)}{\mu_{\sigma,s}(C_{\alpha})} + \sum_{\phi \leq \alpha'} \frac{c_{\sigma,s}^{Ball}(\phi)}{\mu_{\sigma,s}(C_{\phi})} \\
&= \sum_{\phi \leq \alpha} \frac{c_{\sigma,s}^{Ball}(\phi)}{\mu_{\sigma,s}(C_{\phi})}.
\end{aligned}$$

It is interesting to note that whenever $\mu_{\sigma,s}(C_{\alpha}) = 0$, then $c_{\sigma,s}^{Ball}(\alpha) = 0$ by definition as well as $c_{\sigma}^{Ray}(\alpha) \cdot \mu_{\sigma,s}(C_{\alpha})$.

Now we are ready to prove the statement of the proposition:

$$\begin{aligned}
c_{\sigma}^{Ray}(s \xrightarrow{a}_{\mathbb{C}} \mu) &= \sum_{\alpha \in frags^*(\mathcal{A})} c_{\sigma}^{Ray}(\alpha) \cdot \mu_{\sigma,s}(\alpha) \\
&= \sum_{\alpha \in frags^*(\mathcal{A})} \mu_{\sigma,s}(C_{\alpha}) \cdot \sigma(\alpha)(\perp) \cdot c_{\sigma}^{Ray}(\alpha) \\
&= \sum_{\alpha \in frags^*(\mathcal{A})} \mu_{\sigma,s}(C_{\alpha}) \cdot \sigma(\alpha)(\perp) \cdot \sum_{\phi \leq \alpha} \frac{c_{\sigma,s}^{Ball}(\phi)}{\mu_{\sigma,s}(C_{\phi})} \\
&= \sum_{\alpha \in frags^*(\mathcal{A})} \sum_{\phi \leq \alpha} \frac{c_{\sigma,s}^{Ball}(\phi)}{\mu_{\sigma,s}(C_{\phi})} \cdot \mu_{\sigma,s}(C_{\alpha}) \cdot \sigma(\alpha)(\perp) \\
&= \sum_{\phi \in frags^*(\mathcal{A})} \sum_{\alpha \in C_{\phi}} \frac{c_{\sigma,s}^{Ball}(\phi)}{\mu_{\sigma,s}(C_{\phi})} \cdot \mu_{\sigma,s}(C_{\alpha}) \cdot \sigma(\alpha)(\perp) \\
&= \sum_{\phi \in frags^*(\mathcal{A})} \frac{c_{\sigma,s}^{Ball}(\phi)}{\mu_{\sigma,s}(C_{\phi})} \cdot \sum_{\alpha \in C_{\phi}} \mu_{\sigma,s}(C_{\alpha}) \cdot \sigma(\alpha)(\perp) \\
&\stackrel{\dagger}{=} \sum_{\phi \in frags^*(\mathcal{A})} \frac{c_{\sigma,s}^{Ball}(\phi)}{\mu_{\sigma,s}(C_{\phi})} \cdot \mu_{\sigma,s}(C_{\phi}) \\
&= \sum_{\phi \in frags^*(\mathcal{A})} c_{\sigma,s}^{Ball}(\phi) \\
&= c_{\sigma,s}^{Ball}(s \xrightarrow{a}_{\mathbb{C}} \mu),
\end{aligned}$$

where $\stackrel{\dagger}{=}$ is justified by the definition of the probability of a cone and the fact that σ induces a weak transition. \square

Result A.3 (Proposition 5.3). *Strong and strong probabilistic cost-preserving bisimilarities are equivalence relations on the set of CPAs.*

Proof. It is trivial to show that both strong and strong probabilistic cost-preserving bisimulations are reflexive and symmetric, since the identity relation \mathcal{I} suffices for reflexivity and the symmetry of the underlying equivalence relation is the base for the symmetry. Transitivity is more interesting, that is, given three CPAs (\mathcal{A}_1, c_1) , (\mathcal{A}_2, c_2) , and (\mathcal{A}_3, c_3) such that $(\mathcal{A}_1, c_1) \sim_p (\mathcal{A}_2, c_2)$ and $(\mathcal{A}_2, c_2) \sim_p (\mathcal{A}_3, c_3)$, then $(\mathcal{A}_1, c_1) \sim_p (\mathcal{A}_3, c_3)$ (and similarly for \sim); we provide the proof for strong probabilistic cost-preserving bisimulation since the proof for strong cost-preserving bisimulation is essentially the same, where the involved combined transitions are just ordinary transitions and the families are just singletons.

Since $(\mathcal{A}_1, c_1) \sim_p (\mathcal{A}_2, c_2)$ and $(\mathcal{A}_2, c_2) \sim_p (\mathcal{A}_3, c_3)$, it follows that $\mathcal{A}_1 \sim_p \mathcal{A}_2$ and $\mathcal{A}_2 \sim_p \mathcal{A}_3$, since the step condition of strong probabilistic cost-preserving bisimulation is the step condition of strong probabilistic bisimulation extended with a constraint on the cost of the involved transitions. Let \mathcal{R}_{12} and \mathcal{R}_{23} be the corresponding relations. By transitivity of strong probabilistic bisimulation on PAs [26], we have that $\mathcal{A}_1 \sim_p \mathcal{A}_3$ and this is justified by $\mathcal{R}_{13} = \mathcal{R}_{12} \circ \mathcal{R}_{23}$. We claim that \mathcal{R}_{13} is also a strong probabilistic cost-preserving bisimulation; to show this claim, we need to check that for each $(s_1, s_3) \in \mathcal{R}_{13}$ and $s_1 \xrightarrow{a} \mu_1$, there exists $s_3 \xrightarrow{a}_{\mathcal{C}} \mu_3$ such that $\mu_1 \mathcal{L}(\mathcal{R}_{13}) \mu_3$ and $c_3(s_3 \xrightarrow{a}_{\mathcal{C}} \mu_3) = c_1(s_1 \xrightarrow{a} \mu_1)$.

Let $(s_1, s_3) \in \mathcal{R}_{13}$ and $s_1 \xrightarrow{a} \mu_1$. Suppose that $s_1 \in S_1$ and $s_3 \in S_3$; the symmetric case is analogous while the case where both states belong to the same automaton is trivial. By definition of \mathcal{R}_{13} , we know that there exists $s_2 \in S_2$ such that $s_1 \mathcal{R}_{12} s_2 \mathcal{R}_{23} s_3$; moreover, by $(\mathcal{A}_1, c_1) \sim_p (\mathcal{A}_2, c_2)$, there exists $s_2 \xrightarrow{a}_{\mathcal{C}} \mu_2$ such that $\mu_1 \mathcal{L}(\mathcal{R}_{12}) \mu_2$ and $c_2(s_2 \xrightarrow{a}_{\mathcal{C}} \mu_2) = c_1(s_1 \xrightarrow{a} \mu_1)$. Let $\{s_2 \xrightarrow{a} \mu_{2,i}\}_{i \in I}$ and $\{p_i\}_{i \in I}$ be the families of transitions and weights generating $s_2 \xrightarrow{a}_{\mathcal{C}} \mu_2$. Since $s_2 \mathcal{R}_{23} s_3$ and $(\mathcal{A}_2, c_2) \sim_p (\mathcal{A}_3, c_3)$, for each $i \in I$ there exists $s_3 \xrightarrow{a}_{\mathcal{C}} \mu_{3,i}$ such that $\mu_{2,i} \mathcal{L}(\mathcal{R}_{23}) \mu_{3,i}$ and $c_3(s_3 \xrightarrow{a}_{\mathcal{C}} \mu_{3,i}) = c_2(s_2 \xrightarrow{a} \mu_{2,i})$. Let $s_3 \xrightarrow{a}_{\mathcal{C}} \mu_3$ be the strong combined transition such that $\mu_3 = \sum_{i \in I} p_i \cdot \mu_{3,i}$. By properties of the lifting $\mathcal{L}(\cdot)$, it is immediate to see that $\mu_1 \mathcal{L}(\mathcal{R}_{13}) \mu_3$; for the cost of the strong combined transition, we have: $c_3(s_3 \xrightarrow{a}_{\mathcal{C}} \mu_3) = \sum_{i \in I} p_i \cdot c_3(s_3 \xrightarrow{a}_{\mathcal{C}} \mu_{3,i}) = \sum_{i \in I} p_i \cdot c_2(s_2 \xrightarrow{a} \mu_{2,i}) = c_2(s_2 \xrightarrow{a}_{\mathcal{C}} \mu_2) = c_1(s_1 \xrightarrow{a} \mu_1)$, as required.

This completes the proof that \mathcal{R}_{13} is a strong probabilistic cost-preserving bisimulation, thus $(\mathcal{A}_1, c_1) \sim_p (\mathcal{A}_3, c_3)$. \square

Result A.4 (Proposition 5.4). *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , if $(\mathcal{A}_1, c_1) \sim_p (\mathcal{A}_2, c_2)$, then for each (\mathcal{A}_3, c_3) compatible with both (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) and each pair of cost-preserving functions c_l and c_r with $\text{gen}(c_l) = \text{gen}(c_r)$, $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3) \sim_p (\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$, and similarly for \sim .*

Proof. We detail here the proof for strong probabilistic cost-preserving bisimilarity. The one for strong cost-preserving bisimilarity is a simplification. Denoting by S_{12} the set $S_1 \uplus S_2$, let \mathcal{R} be the equivalence relation on S_{12} justifying $(\mathcal{A}_1, c_1) \sim_p (\mathcal{A}_2, c_2)$ and \mathcal{R}_p be the strong probabilistic bisimulation justifying $\mathcal{A}_1 \parallel \mathcal{A}_3 \sim_p \mathcal{A}_2 \parallel \mathcal{A}_3$. Note that $\mathcal{R}_p = \mathcal{R} \times \mathcal{I}$, where \mathcal{I} is the identity relation on S_3 . The existence of \mathcal{R}_p is ensured by the fact that $\mathcal{A}_1 \sim_p \mathcal{A}_2$ and strong probabilistic bisimilarity on probabilistic automata is preserved by parallel composition. We now show that \mathcal{R}_p is a strong probabilistic cost-preserving bisimulation between $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3)$ and $(\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$. The fact that \mathcal{R}_p is an equivalence relation follows directly from being a strong probabilistic bisimulation; moreover, this implies $(\bar{s}_1, \bar{s}_3) \mathcal{R}_p (\bar{s}_2, \bar{s}_3)$ as well.

So, consider a pair of states $(s_1, s_3) \mathcal{R}_p (s_2, s_3)$ and suppose that $(s_1, s_3) \xrightarrow{a} \mu_1 \times \mu_3$. Just for simplicity, assume that $s_1 \in S_1$ and $s_2 \in S_2$; the remaining cases are essentially the same. There are three cases:

Case $a \in \Sigma_3 \setminus \Sigma_1$: In this case, $\mu_1 = \delta_{s_1}$ and the transition $(s_1, s_3) \xrightarrow{a} \mu_1 \times \mu_3$ can be matched by (s_2, s_3) via the transition $(s_2, s_3) \xrightarrow{a} \delta_{s_2} \times \mu_3$ that is trivially also a strong combined transition; it is immediate to see that $\delta_{s_1} \times \mu_3 \mathcal{L}(\mathcal{R}_p) \delta_{s_2} \times \mu_3$. For the cost, we have that

$$\begin{aligned} c_r((s_2, s_3) \xrightarrow{a} \delta_{s_2} \times \mu_3) &= \text{gen}(c_r)(c_2(s_2 \xrightarrow{\nu_a} \delta_{s_2}), c_3(s_3 \xrightarrow{a} \mu_3)) \\ &= \text{gen}(c_r)(0, c_3(s_3 \xrightarrow{a} \mu_3)) \\ &= \text{gen}(c_l)(c_1(s_1 \xrightarrow{\nu_a} \delta_{s_1}), c_3(s_3 \xrightarrow{a} \mu_3)) \\ &= c_l((s_1, s_3) \xrightarrow{a} \delta_{s_1} \times \mu_3), \end{aligned}$$

as required.

Case $a \in \Sigma_1 \setminus \Sigma_3$: In this case, $\mu_3 = \delta_{s_3}$. Since $(\mathcal{A}_1, c_1) \sim_p (\mathcal{A}_2, c_2)$, it follows that there exists $s_2 \xrightarrow{a} \mu_2$ such that $\mu_1 \mathcal{L}(\mathcal{R}) \mu_2$, and $c_2(s_2 \xrightarrow{a} \mu_2) = c_1(s_1 \xrightarrow{a} \mu_1)$. This implies that there exists the strong combined transition $(s_2, s_3) \xrightarrow{a} \mu_2 \times \delta_{s_3}$ such that $\mu_1 \times \delta_{s_3} \mathcal{L}(\mathcal{R}_p) \mu_2 \times \delta_{s_3}$. For the cost of such transition, we have that

$$\begin{aligned} c_r((s_2, s_3) \xrightarrow{a} \mu_2 \times \delta_{s_3}) &= \text{gen}(c_r)(c_2(s_2 \xrightarrow{a} \mu_2), c_3(s_3 \xrightarrow{\nu_a} \delta_{s_3})) \\ &= \text{gen}(c_r)(c_2(s_2 \xrightarrow{a} \mu_2), 0) \\ &= \text{gen}(c_r)(c_1(s_1 \xrightarrow{a} \mu_1), 0) \\ &= \text{gen}(c_l)(c_1(s_1 \xrightarrow{a} \mu_1), 0) \\ &= \text{gen}(c_l)(c_1(s_1 \xrightarrow{a} \mu_1), c_3(s_3 \xrightarrow{\nu_a} \delta_{s_3})) \\ &= c_l((s_1, s_3) \xrightarrow{a} \mu_1 \times \delta_{s_3}), \end{aligned}$$

as required.

Case $a \in \Sigma_3 \cap \Sigma_1$: In this case, we have that $(s_1, s_3) \xrightarrow{a} \mu_1 \times \mu_3$ is generated in the parallel composition by the two transitions $s_2 \xrightarrow{a} \mu_2$ and $s_3 \xrightarrow{a} \mu_3$. Since by hypothesis we have that $(\mathcal{A}_1, c_1) \sim_p (\mathcal{A}_2, c_2)$, it follows that there exists $s_2 \xrightarrow{a} \mu_2$ such that $\mu_1 \mathcal{L}(\mathcal{R}) \mu_2$, and $c_2(s_2 \xrightarrow{a} \mu_2) = c_1(s_1 \xrightarrow{a} \mu_1)$. Let $\{s_2 \xrightarrow{a} \mu_{2,i}\}_{i \in I}$ and $\{p_i\}_{i \in I}$ be such that $\sum_{i \in I} p_i \cdot \mu_{2,i} = \mu_2$. Let $(s_2, s_3) \xrightarrow{a} \mu_2 \times \mu_3$ be the strong combined transition for $\mathcal{A}_2 \times \mathcal{A}_3$ obtained from $\{(s_2, s_3) \xrightarrow{a} \mu_{2,i} \times \mu_3\}_{i \in I}$ and $\{p_i\}_{i \in I}$, i.e., $\mu_2 \times \mu_3 = \sum_{i \in I} p_i \cdot \mu_{2,i} \times \mu_3$.

It is immediate to see that $\mu_1 \times \mu_3 \mathcal{L}(\mathcal{R}_p) \mu_2 \times \mu_3$; for the cost of such strong combined transition, we have that

$$\begin{aligned} c_r((s_2, s_3) \xrightarrow{a} \mu_2 \times \mu_3) &= \sum_{i \in I} p_i \cdot c_r((s_2, s_3) \xrightarrow{a} \mu_{2,i} \times \mu_3) \\ &= \sum_{i \in I} p_i \cdot \text{gen}(c_r)(c_2(s_2 \xrightarrow{a} \mu_{2,i}), c_3(s_3 \xrightarrow{a} \mu_3)) \\ &= \text{gen}(c_r)(\sum_{i \in I} p_i \cdot c_2(s_2 \xrightarrow{a} \mu_{2,i}), c_3(s_3 \xrightarrow{a} \mu_3)) \\ &= \text{gen}(c_r)(c_2(s_2 \xrightarrow{a} \mu_2), c_3(s_3 \xrightarrow{a} \mu_3)) \\ &= \text{gen}(c_l)(c_2(s_2 \xrightarrow{a} \mu_2), c_3(s_3 \xrightarrow{a} \mu_3)) \\ &= \text{gen}(c_l)(c_1(s_1 \xrightarrow{a} \mu_1), c_3(s_3 \xrightarrow{a} \mu_3)) \\ &= c_l((s_1, s_3) \xrightarrow{a} \mu_1 \times \mu_3), \end{aligned}$$

as required.

This completes the proof that \mathcal{R}_p is a strong probabilistic cost-preserving bisimulation, thus $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3) \sim_p (\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$. \square

Result A.5 (Proposition 5.7). *Weak probabilistic cost-preserving bisimilarity is an equivalence relation on the set of CPAs.*

Proof. Reflexivity and symmetry are straightforward, because $\mathcal{I} \subseteq \approx_p$ and because the underlying equivalence relation is symmetric. Transitivity needs a more detailed account: Given three CPAs (\mathcal{A}_1, c_1) , (\mathcal{A}_2, c_2) , and (\mathcal{A}_3, c_3) , if $(\mathcal{A}_1, c_1) \approx_p (\mathcal{A}_2, c_2)$ and $(\mathcal{A}_2, c_2) \approx_p (\mathcal{A}_3, c_3)$, then $(\mathcal{A}_1, c_1) \approx_p (\mathcal{A}_3, c_3)$.

Since $(\mathcal{A}_1, c_1) \approx_p (\mathcal{A}_2, c_2)$ and $(\mathcal{A}_2, c_2) \approx_p (\mathcal{A}_3, c_3)$, it follows that $\mathcal{A}_1 \approx_p \mathcal{A}_2$ and $\mathcal{A}_2 \approx_p \mathcal{A}_3$, thus, by transitivity of weak probabilistic bisimulation on PAs [26], we have that $\mathcal{A}_1 \approx_p \mathcal{A}_3$. Let \mathcal{R} be the corresponding equivalence relation. We claim that \mathcal{R} is also a weak probabilistic cost-preserving bisimulation; the only remaining thing we need to check is that for each $(s_1, s_3) \in \mathcal{R}$ and $s_1 \xrightarrow{a} \mu_1$, there exists $s_3 \xrightarrow{a} \mu_3$ such that $\mu_1 \mathcal{L}(\mathcal{R}) \mu_3$ and $c_3(s_3 \xrightarrow{a} \mu_3) = c_1(s_1 \xrightarrow{a} \mu_1)$. The existence of $s_3 \xrightarrow{a} \mu_3$ such that $\mu_1 \mathcal{L}(\mathcal{R}) \mu_3$ is ensured again by the fact that \mathcal{R} justifies $\mathcal{A}_1 \approx_p \mathcal{A}_3$; moreover, $s_3 \xrightarrow{a} \mu_3$ is essentially obtained from $s_2 \xrightarrow{a} \mu_2$ (induced by some scheduler σ_2 and used to match $s_1 \xrightarrow{a} \mu_1$ when checking the step condition for the pair (s_1, s_2) in $\mathcal{A}_1 \approx_p \mathcal{A}_2$) by replacing each transition $s_2' \xrightarrow{b} \mu_2'$ chosen by the scheduler σ_2 by the corresponding matching $s_3' \xrightarrow{b} \mu_3'$ where $b \in \{a, \tau\}$. Since by hypothesis $c_2(s_2' \xrightarrow{b} \mu_2') = c_3(s_3' \xrightarrow{b} \mu_3')$, it is immediate to derive that indeed $c_3(s_3 \xrightarrow{a} \mu_3) = c_1(s_1 \xrightarrow{a} \mu_1)$ by using the definition of cost provided in Definition 4.8.

This completes the proof that \mathcal{R}_p is a weak probabilistic cost-preserving bisimulation, thus $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3) \approx_p (\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$. \square

Result A.6 (Lemma 5.8). *Given two compatible CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) and a cost preserving*

function c , let $(s_1, s_2) \xrightarrow{a} \mu_1 \times \mu_2$ be a weak combined transition of $(\mathcal{A}_1, c_1) \parallel_c (\mathcal{A}_2, c_2)$. Then, $c((s_1, s_2) \xrightarrow{a} \mu_1 \times \mu_2) = \text{gen}(c)(c_1(s_1 \xrightarrow{a} \mu_1), c_2(s_2 \xrightarrow{a} \mu_2))$, where for $i = 1, 2$, $s_i \xrightarrow{a} \mu_i = \text{proj}_i((s_1, s_2) \xrightarrow{a} \mu_1 \times \mu_2)$.

Proof. Let $\mathcal{A} = \mathcal{A}_1 \parallel \mathcal{A}_2$ and σ be the scheduler inducing $(s_1, s_2) \xrightarrow{a} \mu_1 \times \mu_2$; by definition 4.8, $c((s_1, s_2) \xrightarrow{a} \mu_1 \times \mu_2) = \sum_{\alpha \in \text{frags}^*(\mathcal{A})} c_{\sigma, (s_1, s_2)}(\alpha)$ where

$$c_{\sigma, (s_1, s_2)}(\alpha) = \begin{cases} \mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sum_{tr \in D(b)} c(tr) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2) & \text{if } \alpha = \alpha'b(t_1, t_2), \\ 0 & \text{otherwise.} \end{cases}$$

Suppose that for each $\alpha \in \text{frags}^*(\mathcal{A})$, $c_{\sigma, (s_1, s_2)}(\alpha) = \text{gen}(c)(c_{1\sigma, (s_1, s_2)}(\alpha), c_{2\sigma, (s_1, s_2)}(\alpha))$; this implies that $\sum_{\alpha \in \text{frags}^*(\mathcal{A})} c_{\sigma, (s_1, s_2)}(\alpha) = \sum_{\alpha \in \text{frags}^*(\mathcal{A})} \text{gen}(c)(c_{1\sigma, (s_1, s_2)}(\alpha), c_{2\sigma, (s_1, s_2)}(\alpha))$. Since $\text{gen}(c)$ is distributive by hypothesis, we can move the summation inside $\text{gen}(c)$, that is, $\sum_{\alpha \in \text{frags}^*(\mathcal{A})} c_{\sigma, (s_1, s_2)}(\alpha) = \text{gen}(c)(\sum_{\alpha \in \text{frags}^*(\mathcal{A})} c_{1\sigma, (s_1, s_2)}(\alpha), \sum_{\alpha \in \text{frags}^*(\mathcal{A})} c_{2\sigma, (s_1, s_2)}(\alpha))$, i.e., $c((s_1, s_2) \xrightarrow{a} \mu_1 \times \mu_2) = \text{gen}(c)(c_1(s_1 \xrightarrow{a} \mu_1), c_2(s_2 \xrightarrow{a} \mu_2))$, as required.

We now show that $c_{\sigma, (s_1, s_2)}(\alpha) = \text{gen}(c)(c_{1\sigma, (s_1, s_2)}(\alpha), c_{2\sigma, (s_1, s_2)}(\alpha))$ holds for each $\alpha \in \text{frags}^*(\mathcal{A})$. Suppose that $\alpha = t$; this case is obvious, since $c_{\sigma, (s_1, s_2)}(t) = 0 = \text{gen}(c)(0, 0) = \text{gen}(c)(c_{1\sigma, (s_1, s_2)}(t), c_{2\sigma, (s_1, s_2)}(t))$; the equality $0 = \text{gen}(c)(0, 0)$ follows by the property of being zero-preserving.

Suppose that $\alpha = \alpha'b(t_1, t_2)$ for some $\alpha' \in \text{frags}^*(\mathcal{A})$, $b \in \Sigma$, and $(t_1, t_2) \in S$. There are three cases:

Case $b \in \Sigma_1 \setminus \Sigma_2$: In this case, each transition $tr = (v_1, v_2) \xrightarrow{b} \mu_1 \times \mu_2 \in D(b)$ has been obtained by combining $v_1 \xrightarrow{b} \mu_1$ and the apparent transition $v_2 \xrightarrow{\nu_b} \delta_{v_2}$ where $\mu_2 = \delta_{v_2}$; its cost is by definition $c(tr) = \text{gen}(c)(c_1(v_1 \xrightarrow{b} \mu_1), 0)$. This means that

$$\begin{aligned} & c_{\sigma, (s_1, s_2)}(\alpha) \\ &= \mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sum_{tr \in D(b)} c(tr) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2) \\ &= \sum_{tr \in D(b)} \mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2) \cdot c(tr) \end{aligned}$$

by reordering of summations and products,

$$= \sum_{tr \in D(b)} \mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2) \cdot \text{gen}(c)(c_1(v_1 \xrightarrow{b} \mu_1), c_2(v_2 \xrightarrow{\nu_b} \delta_{v_2}))$$

by definition of c ,

$$= \text{gen}(c) \left(\sum_{tr \in D(b)} \mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2) \cdot c_1(v_1 \xrightarrow{b} \mu_1), 0 \right)$$

by distributivity of $\text{gen}(c)$ and $c_2(v_2 \xrightarrow{\nu_b} \delta_{v_2}) = 0$,

$$= \text{gen}(c) \left(\mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sum_{tr \in D(b)} c_1(v_1 \xrightarrow{b} \mu_1) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2), 0 \right)$$

by reordering of summations and products,

$$= \text{gen}(c)(c_{1\sigma, (s_1, s_2)}(\alpha), c_{2\sigma, (s_1, s_2)}(\alpha)).$$

The last equality comes from the fact that by definition, $c_2(v_2 \xrightarrow{\nu_b} \delta_{v_2}) = 0$, thus

$$\begin{aligned} 0 &= 0 \cdot \mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sum_{tr \in D(b)} \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2) \\ &= c_2(v_2 \xrightarrow{\nu_b} \delta_{v_2}) \cdot \mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sum_{tr \in D(b)} \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2) \\ &= \mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sum_{tr \in D(b)} c_2(v_2 \xrightarrow{\nu_b} \delta_{v_2}) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2) \\ &= c_{2\sigma, (s_1, s_2)}(\alpha). \end{aligned}$$

Case $b \in \Sigma_2 \setminus \Sigma_1$: This case is symmetric to the previous case.

Case $b \in \Sigma_1 \cap \Sigma_2$: In this case, every transition $tr = (v_1, v_2) \xrightarrow{b} \mu_1 \times \mu_2 \in D(b)$ has been obtained by combining $v_1 \xrightarrow{b} \mu_1$ and $v_2 \xrightarrow{b} \mu_2$; its cost, by definition, is $c(tr) = \text{gen}(c)(c_1(v_1 \xrightarrow{b} \mu_1), c_2(v_2 \xrightarrow{b} \mu_2))$. This means that

$$\begin{aligned} & c_{\sigma, (s_1, s_2)}(\alpha) \\ &= \mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sum_{tr \in D(b)} c(tr) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2) \end{aligned}$$

$$= \sum_{tr \in D(b)} \mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2) \cdot c(tr)$$

by reordering of summations and products,

$$= \sum_{tr \in D(b)} \mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2) \cdot \text{gen}(c)(c_1(v_1 \xrightarrow{b} \mu_1), c_2(v_2 \xrightarrow{b} \mu_2))$$

by definition of c ,

$$= \text{gen}(c) \left(\sum_{tr \in D(b)} \mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2) \cdot c_1(v_1 \xrightarrow{b} \mu_1), \right. \\ \left. \sum_{tr \in D(b)} \mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2) \cdot c_2(v_2 \xrightarrow{b} \mu_2) \right)$$

by distributivity of $\text{gen}(c)$,

$$= \text{gen}(c) \left(\mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sum_{tr \in D(b)} c_1(v_1 \xrightarrow{b} \mu_1) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2), \right. \\ \left. \mu_{\sigma, (s_1, s_2)}(C_{\alpha'}) \cdot \sum_{tr \in D(b)} c_2(v_2 \xrightarrow{b} \mu_2) \cdot \sigma(\alpha')(tr) \cdot \mu_{tr}(t_1, t_2) \right)$$

by reordering of summations and products,

$$= \text{gen}(c)(c_{1_{\sigma, (s_1, s_2)}}(\alpha), c_{2_{\sigma, (s_1, s_2)}}(\alpha))$$

by definition of $c_{1_{\sigma, (s_1, s_2)}}(\alpha)$ and $c_{2_{\sigma, (s_1, s_2)}}(\alpha)$.

This completes the proof that $c_{\sigma, (s_1, s_2)}(\alpha) = \text{gen}(c)(c_{1_{\sigma, (s_1, s_2)}}(\alpha), c_{2_{\sigma, (s_1, s_2)}}(\alpha))$ holds for each $\alpha \in \text{frags}^*(\mathcal{A})$, hence $c((s_1, s_2) \xrightarrow{a} \mu_1 \times \mu_2) = \text{gen}(c)(c_1(s_1 \xrightarrow{a} \mu_1), c_2(s_2 \xrightarrow{a} \mu_2))$. \square

Result A.7 (Proposition 5.9). *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , if $(\mathcal{A}_1, c_1) \approx_p (\mathcal{A}_2, c_2)$, then for each (\mathcal{A}_3, c_3) compatible with both (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) and each pair of cost-preserving functions c_l and c_r with $\text{gen}(c_l) = \text{gen}(c_r)$, $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3) \approx_p (\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$.*

Proof. Denoted by S_{12} the set $S_1 \uplus S_2$, let \mathcal{R} be the equivalence relation on S_{12} justifying $(\mathcal{A}_1, c_1) \approx_p (\mathcal{A}_2, c_2)$ and \mathcal{R}_p be a weak probabilistic bisimulation justifying $\mathcal{A}_1 \parallel \mathcal{A}_3 \approx_p \mathcal{A}_2 \parallel \mathcal{A}_3$. Note that $\mathcal{R}_p = \mathcal{R} \times \mathcal{I}$, where \mathcal{I} is the identity relation on S_3 . The existence of \mathcal{R}_p is ensured by the fact that $\mathcal{A}_1 \approx_p \mathcal{A}_2$ and that weak probabilistic bisimilarity on probabilistic automata is preserved by parallel composition. We now show that \mathcal{R}_p is a weak probabilistic cost-preserving bisimulation between $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3)$ and $(\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$. The fact that \mathcal{R}_p is an equivalence relation follows directly from being a weak probabilistic bisimulation; moreover, this implies $(\bar{s}_1, \bar{s}_3) \mathcal{R}_p (\bar{s}_2, \bar{s}_3)$ as well.

So, consider a pair of states $(s_1, s_3) \mathcal{R}_p (s_2, s_3)$ and suppose that $(s_1, s_3) \xrightarrow{a} \mu_1 \times \mu_3$. Just for simplicity, assume that $s_1 \in S_1$ and $s_2 \in S_2$; the remaining cases are essentially the same. There are three cases:

Case $a \in \Sigma_3 \setminus \Sigma_1$: In this case, $\mu_1 = \delta_{s_1}$ and the transition $(s_1, s_3) \xrightarrow{a} \mu_1 \times \mu_3$ can be matched by (s_2, s_3) via the transition $(s_2, s_3) \xrightarrow{a} \delta_{s_2} \times \mu_3$ that is trivially also a weak combined transition; it is immediate to see that $\delta_{s_1} \times \mu_3 \mathcal{L}(\mathcal{R}_p) \delta_{s_2} \times \mu_3$. For the cost, we have that

$$\begin{aligned} c_r((s_2, s_3) \xrightarrow{a} \delta_{s_2} \times \mu_3) &= \text{gen}(c_r)(c_2(s_2 \xrightarrow{\nu_a} \delta_{s_2}), c_3(s_3 \xrightarrow{a} \mu_3)) \\ &= \text{gen}(c_r)(0, c_3(s_3 \xrightarrow{a} \mu_3)) \\ &= \text{gen}(c_l)(0, c_3(s_3 \xrightarrow{a} \mu_3)) \\ &= \text{gen}(c_l)(c_1(s_1 \xrightarrow{\nu_a} \delta_{s_1}), c_3(s_3 \xrightarrow{a} \mu_3)) \\ &= c_l((s_1, s_3) \xrightarrow{a} \delta_{s_1} \times \mu_3), \end{aligned}$$

as required.

Case $a \in \Sigma_1 \setminus \Sigma_3$: In this case, $\mu_3 = \delta_{s_3}$. Since $(\mathcal{A}_1, c_1) \approx_p (\mathcal{A}_2, c_2)$, it follows that there exists $s_2 \xrightarrow{a} c_2 \mu_2$ such that $\mu_1 \mathcal{L}(\mathcal{R}) \mu_2$, and $c_2(s_2 \xrightarrow{a} c_2 \mu_2) = c_1(s_1 \xrightarrow{a} \mu_1)$. This implies that there exists a weak combined transition $(s_2, s_3) \xrightarrow{a} c_2 \mu_2 \times \delta_{s_3}$ such that $\mu_1 \times \delta_{s_3} \mathcal{L}(\mathcal{R}_p) \mu_2 \times \delta_{s_3}$. By Lemma 5.8, we have that

$$\begin{aligned} c_r((s_2, s_3) \xrightarrow{a} c_2 \mu_2 \times \delta_{s_3}) &= \text{gen}(c_r)(c_2(s_2 \xrightarrow{a} c_2 \mu_2), c_3(s_3 \xrightarrow{\tau} \delta_{s_3})) \\ &= \text{gen}(c_r)(c_2(s_2 \xrightarrow{a} c_2 \mu_2), 0) \\ &= \text{gen}(c_r)(c_1(s_1 \xrightarrow{a} \mu_1), 0) \\ &= \text{gen}(c_l)(c_1(s_1 \xrightarrow{a} \mu_1), 0) \\ &= \text{gen}(c_l)(c_1(s_1 \xrightarrow{a} \mu_1), c_3(s_3 \xrightarrow{\nu_a} \delta_{s_3})) \\ &= c_l((s_1, s_3) \xrightarrow{a} \mu_1 \times \delta_{s_3}), \end{aligned}$$

as required.

Case $a \in \Sigma_3 \cap \Sigma_1$: In this case, we have that $(s_1, s_3) \xrightarrow{a} \mu_1 \times \mu_3$ is a transition of the composed automaton obtained by combining the two transitions $s_2 \xrightarrow{a} \mu_2$ and $s_3 \xrightarrow{a} \mu_3$. Since $(\mathcal{A}_1, c_1) \approx_p (\mathcal{A}_2, c_2)$, it follows that there exists $s_2 \xrightarrow{a} c_2 \mu_2$ induced by the scheduler σ such that $\mu_1 \mathcal{L}(\mathcal{R}) \mu_2$, and $c_2(s_2 \xrightarrow{a} c_2 \mu_2) = c_1(s_1 \xrightarrow{a} \mu_1)$. Let $(s_2, s_3) \xrightarrow{a} c_2 \mu_2 \times \mu_3$ be the weak combined transition for $\mathcal{A}_2 \times \mathcal{A}_3$ obtained by the scheduler σ_p that mimics σ for all internal transitions of \mathcal{A}_2 used in $s_2 \xrightarrow{a} c_2 \mu_2$ and schedules $(t_2, s_3) \xrightarrow{a} \rho_2 \times \mu_3$ on an execution fragment α with the same probability of scheduling $t_2 \xrightarrow{a} \rho_2$ given by σ on $\text{proj}_2(\alpha)$. It is worthwhile to note that $\text{proj}_3((s_2, s_3) \xrightarrow{a} c_2 \mu_2 \times \mu_3) = s_3 \xrightarrow{a} c_3 \mu_3 = s_3 \xrightarrow{a} \mu_3$. It is immediate to see that $\mu_1 \times \mu_3 \mathcal{L}(\mathcal{R}_p) \mu_2 \times \mu_3$; for the cost of such weak combined transition, by Lemma 5.8, we have that

$$\begin{aligned} c_r((s_2, s_3) \xrightarrow{a} c_2 \mu_2 \times \mu_3) &= \text{gen}(c_r)(c_2(s_2 \xrightarrow{a} c_2 \mu_2), c_3(s_3 \xrightarrow{a} c_3 \mu_3)) \\ &= \text{gen}(c_r)(c_2(s_2 \xrightarrow{a} c_2 \mu_2), c_3(s_3 \xrightarrow{a} \mu_3)) \\ &= \text{gen}(c_r)(c_1(s_1 \xrightarrow{a} \mu_1), c_3(s_3 \xrightarrow{a} \mu_3)) \\ &= \text{gen}(c_l)(c_1(s_1 \xrightarrow{a} \mu_1), c_3(s_3 \xrightarrow{a} \mu_3)) \\ &= c_l((s_1, s_3) \xrightarrow{a} c_1 \mu_1 \times \mu_3), \end{aligned}$$

as required.

This completes the proof that \mathcal{R}_p is a weak probabilistic cost-preserving bisimulation, thus $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3) \approx_p (\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$. \square

Result A.8 (Proposition 6.3). *Minor cost strong and strong probabilistic bisimilarities are preorders on the set of CPAs.*

Proof. It is trivial to show that both bisimulations are reflexive, so we concentrate on transitivity, that is, given three CPAs (\mathcal{A}_1, c_1) , (\mathcal{A}_2, c_2) , and (\mathcal{A}_3, c_3) , if $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$ and $(\mathcal{A}_2, c_2) \lesssim_p (\mathcal{A}_3, c_3)$, then $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_3, c_3)$, and similarly for \lesssim . As for Proposition 5.3, we provide the proof only for minor cost strong probabilistic bisimulation; the proof for minor cost strong bisimulation is essentially the same, except that the involved combined transitions are just ordinary transitions and that families are just singletons.

Since $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$ and $(\mathcal{A}_2, c_2) \lesssim_p (\mathcal{A}_3, c_3)$, it follows by definition that $\mathcal{A}_2 \sim_p \mathcal{A}_1$ and $\mathcal{A}_3 \sim_p \mathcal{A}_2$. Let \mathcal{R}_{21} and \mathcal{R}_{32} be the corresponding relations. By transitivity of strong probabilistic bisimulation on PAs [26], we have that $\mathcal{A}_3 \sim_p \mathcal{A}_1$ and this is justified by $\mathcal{R}_{31} = \mathcal{R}_{32} \circ \mathcal{R}_{21}$. We claim that \mathcal{R}_{31} is also a minor cost strong probabilistic bisimulation; to show this claim, we need to check that for each $s_3 \xrightarrow{a} \mu_3 \in D_3$ and $s_1 \in S_1$ such that $s_3 \mathcal{R}_{31} s_1$, there exists $s_1 \xrightarrow{a}_C \mu_1$ such that $\mu_3 \mathcal{L}(\mathcal{R}_{31}) \mu_1$ and $c_1(s_1 \xrightarrow{a}_C \mu_1) \leq c_3(s_3 \xrightarrow{a} \mu_3)$.

Let $s_3 \xrightarrow{a} \mu_3$ and $s_3 \mathcal{R}_{31} s_1$ with $s_1 \in S_1$. By definition of \mathcal{R}_{31} , we know that there exists $s_2 \in S_2$ such that $s_3 \mathcal{R}_{32} s_2 \mathcal{R}_{21} s_1$; moreover, by $(\mathcal{A}_2, c_2) \lesssim_p (\mathcal{A}_3, c_3)$, there exists $s_2 \xrightarrow{a}_C \mu_2$ such that $\mu_3 \mathcal{L}(\mathcal{R}_{32}) \mu_2$ and $c_2(s_2 \xrightarrow{a}_C \mu_2) \leq c_3(s_3 \xrightarrow{a} \mu_3)$. Let $\{s_2 \xrightarrow{a} \mu_{2,i}\}_{i \in I}$ and $\{p_i\}_{i \in I}$ be the families of transitions and weights generating $s_2 \xrightarrow{a}_C \mu_2$. Since $s_2 \mathcal{R}_{21} s_1$ and $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$, for each $i \in I$ there exists $s_1 \xrightarrow{a}_C \mu_{1,i}$ such that $\mu_{2,i} \mathcal{L}(\mathcal{R}_{21}) \mu_{1,i}$ and $c_1(s_1 \xrightarrow{a}_C \mu_{1,i}) \leq c_2(s_2 \xrightarrow{a} \mu_{2,i})$. Let $s_1 \xrightarrow{a}_C \mu_1$ be the strong combined transition such that $\mu_1 = \sum_{i \in I} p_i \cdot \mu_{1,i}$. By properties of the lifting $\mathcal{L}(\cdot)$, it is immediate to see that $\mu_3 \mathcal{L}(\mathcal{R}_{31}) \mu_1$; for the cost of the strong combined transition, we have: $c_1(s_1 \xrightarrow{a}_C \mu_1) = \sum_{i \in I} p_i \cdot c_1(s_1 \xrightarrow{a}_C \mu_{1,i}) \leq \sum_{i \in I} p_i \cdot c_2(s_2 \xrightarrow{a} \mu_{2,i}) = c_2(s_2 \xrightarrow{a}_C \mu_2) \leq c_3(s_3 \xrightarrow{a} \mu_3)$, as required.

This completes the proof that \mathcal{R}_{31} is a minor cost strong probabilistic bisimulation, thus $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_3, c_3)$. \square

Result A.9 (Proposition 6.4). *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , if $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$, then for each (\mathcal{A}_3, c_3) compatible with both (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) and each pair of cost-preserving functions c_l and c_r with $\text{gen}(c_l) = \text{gen}(c_r)$, $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3) \lesssim_p (\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$, and similarly for \lesssim .*

Proof. We provide the proof for minor cost strong probabilistic bisimulation, the one for minor cost strong bisimulation is again a simplification thereof. Denoted by S_{12} the set $S_1 \uplus S_2$, let \mathcal{R} be the equivalence relation on S_{12} justifying $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$ and \mathcal{R}_p be the strong probabilistic bisimulation justifying $\mathcal{A}_1 \parallel \mathcal{A}_3 \sim_p \mathcal{A}_2 \parallel \mathcal{A}_3$. Note that $\mathcal{R}_p = \mathcal{R} \times \mathcal{I}$, where \mathcal{I} is the identity relation on S_3 . The existence of \mathcal{R}_p is ensured by the fact that $\mathcal{A}_1 \sim_p \mathcal{A}_2$ and that strong probabilistic bisimilarity on probabilistic automata is preserved by parallel composition. We now show that \mathcal{R}_p is a minor cost strong probabilistic bisimulation from $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3)$ to $(\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$. The fact that \mathcal{R}_p is an equivalence relation follows directly from being a strong probabilistic bisimulation; moreover, this implies $(\bar{s}_1, \bar{s}_3) \mathcal{R}_p (\bar{s}_2, \bar{s}_3)$ as well.

So, consider $(s_2, s_3) \xrightarrow{a} \mu_2 \times \mu_3 \in D_{2,3}$ and $(s_1, s_3) \in S_{1,3}$ with $(s_2, s_3) \mathcal{R}_p (s_1, s_3)$. There are three cases:

Case $a \in \Sigma_3 \setminus \Sigma_2$: In this case, $\mu_2 = \delta_{s_2}$ and the transition $(s_2, s_3) \xrightarrow{a} \mu_2 \times \mu_3$ can be matched by (s_1, s_3) via the transition $(s_1, s_3) \xrightarrow{a} \delta_{s_1} \times \mu_3$ that is trivially also a strong combined transition; it is immediate to see that $\delta_{s_2} \times \mu_3 \mathcal{L}(\mathcal{R}_p) \delta_{s_1} \times \mu_3$. For the cost,

we have that

$$\begin{aligned}
c_r((s_1, s_3) \xrightarrow{a} \delta_{s_1} \times \mu_3) &= \text{gen}(c_r)(c_1(s_1 \xrightarrow{\nu_a} \delta_{s_1}), c_3(s_3 \xrightarrow{a} \mu_3)) \\
&= \text{gen}(c_r)(0, c_3(s_3 \xrightarrow{a} \mu_3)) \\
&= \text{gen}(c_l)(0, c_3(s_3 \xrightarrow{a} \mu_3)) \\
&= \text{gen}(c_l)(c_2(s_2 \xrightarrow{\nu_a} \delta_{s_2}), c_3(s_3 \xrightarrow{a} \mu_3)) \\
&= c_l((s_2, s_3) \xrightarrow{a} \delta_{s_2} \times \mu_3),
\end{aligned}$$

as required.

Case $a \in \Sigma_2 \setminus \Sigma_3$: In this case, $\mu_3 = \delta_{s_3}$. Since $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$, it follows that there exists $s_1 \xrightarrow{a} \mu_1$ such that $\mu_2 \mathcal{L}(\mathcal{R}) \mu_1$, and $c_1(s_1 \xrightarrow{a} \mu_1) \leq c_2(s_2 \xrightarrow{a} \mu_2)$. This implies that there exists a strong combined transition $(s_1, s_3) \xrightarrow{a} \mu_1 \times \delta_{s_3}$ such that $\mu_2 \times \delta_{s_3} \mathcal{L}(\mathcal{R}_p) \mu_1 \times \delta_{s_3}$. For the cost of such transition, we have that

$$\begin{aligned}
c_r((s_1, s_3) \xrightarrow{a} \mu_1 \times \delta_{s_3}) &= \text{gen}(c_r)(c_1(s_1 \xrightarrow{a} \mu_1), c_3(s_3 \xrightarrow{\nu_a} \delta_{s_3})) \\
&= \text{gen}(c_r)(c_1(s_1 \xrightarrow{a} \mu_1), 0) \\
&\leq \text{gen}(c_r)(c_2(s_2 \xrightarrow{a} \mu_2), 0) \\
&= \text{gen}(c_l)(c_2(s_2 \xrightarrow{a} \mu_2), 0) \\
&= \text{gen}(c_l)(c_2(s_2 \xrightarrow{a} \mu_2), c_3(s_3 \xrightarrow{\nu_a} \delta_{s_3})) \\
&= c_l((s_2, s_3) \xrightarrow{a} \mu_2 \times \delta_{s_3}),
\end{aligned}$$

as required.

Case $a \in \Sigma_3 \cap \Sigma_2$: In this case, we have that $(s_2, s_3) \xrightarrow{a} \mu_2 \times \mu_3$ is generated in the parallel composition by the two transitions $s_1 \xrightarrow{a} \mu_1$ and $s_3 \xrightarrow{a} \mu_3$. Since by hypothesis we have that $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$, it follows that there exists $s_1 \xrightarrow{a} \mu_1$ such that $\mu_2 \mathcal{L}(\mathcal{R}) \mu_1$, and $c_1(s_1 \xrightarrow{a} \mu_1) \leq c_2(s_2 \xrightarrow{a} \mu_2)$. Let $\{s_1 \xrightarrow{a} \mu_{1,i}\}_{i \in I}$ and $\{p_i\}_{i \in I}$ be such that $\sum_{i \in I} p_i \cdot \mu_{1,i} = \mu_1$. Let $(s_1, s_3) \xrightarrow{a} \mu_1 \times \mu_3$ be the strong combined transition for $\mathcal{A}_1 \times \mathcal{A}_3$ obtained from $\{(s_1, s_3) \xrightarrow{a} \mu_{1,i} \times \mu_3\}_{i \in I}$ and $\{p_i\}_{i \in I}$, i.e., $\mu_1 \times \mu_3 = \sum_{i \in I} p_i \cdot \mu_{1,i} \times \mu_3$.

It is immediate to see that $\mu_2 \times \mu_3 \mathcal{L}(\mathcal{R}_p) \mu_1 \times \mu_3$; for the cost of such strong combined transition, we have that

$$\begin{aligned}
c_r((s_1, s_3) \xrightarrow{a} \mu_1 \times \mu_3) &= \sum_{i \in I} p_i \cdot c_r((s_1, s_3) \xrightarrow{a} \mu_{1,i} \times \mu_3) \\
&= \sum_{i \in I} p_i \cdot \text{gen}(c_r)(c_1(s_1 \xrightarrow{a} \mu_{1,i}), c_3(s_3 \xrightarrow{a} \mu_3)) \\
&= \text{gen}(c_r)(\sum_{i \in I} p_i \cdot c_1(s_1 \xrightarrow{a} \mu_{1,i}), c_3(s_3 \xrightarrow{a} \mu_3)) \\
&= \text{gen}(c_r)(c_1(s_1 \xrightarrow{a} \mu_1), c_3(s_3 \xrightarrow{a} \mu_3)) \\
&= \text{gen}(c_l)(c_1(s_1 \xrightarrow{a} \mu_1), c_3(s_3 \xrightarrow{a} \mu_3)) \\
&\leq \text{gen}(c_l)(c_2(s_2 \xrightarrow{a} \mu_2), c_3(s_3 \xrightarrow{a} \mu_3)) \\
&= c_l((s_2, s_3) \xrightarrow{a} \mu_2 \times \mu_3),
\end{aligned}$$

as required.

This completes the proof that \mathcal{R}_p is a minor cost strong probabilistic bisimulation, thus $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3) \lesssim_p (\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$. \square

Result A.10 (Proposition 6.7). *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , if $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$, then $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$.*

Proof. Let \mathcal{W} be the equivalence relation justifying $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$; by construction it is also a strong probabilistic bisimulation between \mathcal{A}_1 and \mathcal{A}_2 , thus it is also a weak probabilistic bisimulation between \mathcal{A}_1 and \mathcal{A}_2 . Let \mathcal{C} be $\mathcal{W} \cap S_2 \times S_1$. Obviously we have that $\mathcal{C} \subseteq \mathcal{W} \cap S_2 \times S_1$, $\bar{s}_2 \mathcal{C} \bar{s}_1$, and for each $s_2 \in S_2$ there exists $s_1 \in S_1$ such that $s_2 \mathcal{C} s_1$. (This is ensured by the fact that every state in both automata is reachable from the corresponding start state.) To complete the proof, we need to show the step condition: Let $s_2 \xrightarrow{a} \mu_2$ and $s_1 \in S_1$ be such that $s_2 \mathcal{C} s_1$. Suppose that there does not exist $\rho_2 \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_2)$ such that $\mu_2 \xrightarrow{\tau} \rho_2$, then by hypothesis, there exists $\mu_1 \in \text{Disc}(S_1)$ such that $s_1 \xrightarrow{a} \mu_1$, $\mu_2 \mathcal{L}(\mathcal{W}) \mu_1$, and $c_1(s_1 \xrightarrow{a} \mu_1) \leq c_2(s_2 \xrightarrow{a} \mu_2)$. Since $\mathcal{C} = \mathcal{W} \cap S_2 \times S_1$ and the definition of lifting only involves pairs belonging to $S_2 \times S_1$, we have also $\mu_2 \mathcal{L}(\mathcal{C}) \mu_1$, as required. Suppose that there exists $\rho_2 \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_2)$ such that $\mu_2 \xrightarrow{\tau} \rho_2$ and $\min\{c_2(\mu_2 \xrightarrow{\tau} \rho) \mid \rho \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_2)\} = c_2(\mu_2 \xrightarrow{\tau} \rho_2)$. In order to find the matching transition from s_1 , we replace each ordinary transition $t_2 \xrightarrow{b} \gamma_2$ inside $s_2 \xrightarrow{a} \rho_2$ (where $b \in \{a, \tau\}$) with the matching transitions $t_1 \xrightarrow{b} \gamma_1$ with the corresponding probabilities. It is routine to verify that the result of this replacement is indeed a weak combined transition $s_1 \xrightarrow{a} \rho_1$ for some $\rho_1 \in \text{Disc}(S_1)$ such that $\rho_2 \mathcal{L}(\mathcal{W}) \rho_1$, thus $\rho_2 \mathcal{L}(\mathcal{C}) \rho_1$ as before. Moreover, since by hypothesis each ordinary transition $t_2 \xrightarrow{b} \gamma_2$ inside $s_2 \xrightarrow{a} \rho_2$ has been matched by $t_1 \xrightarrow{b} \gamma_1$ such that $c_1(t_1 \xrightarrow{b} \gamma_1) \leq c_2(t_2 \xrightarrow{b} \gamma_2)$, it is trivial to derive that indeed $c_1(s_1 \xrightarrow{a} \rho_1) \leq c_2(s_2 \xrightarrow{a} \mu_2 \xrightarrow{\tau} \rho_2)$. The last thing we have to check is that $\rho_1 \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_1)$. This trivially holds since each $v_2 \in \text{Supp}(\rho_2)$ is a border state, thus v_2 enables a transition $v_2 \xrightarrow{b} \theta_2$ such that $b \in \mathbf{E}_2$ or $\theta_2([v_2]_{\mathcal{W}}) < 1$. Since $\rho_2 \mathcal{L}(\mathcal{C}) \rho_1$, we have that each $v_1 \in \text{Supp}(\rho_1)$ is related to some $u_2 \in \text{Supp}(\rho_2)$, thus also v_1 enables a transition $v_1 \xrightarrow{b} \theta_1$ (with $\theta_2 \mathcal{L}(\mathcal{W}) \theta_1$ by $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$) such that $b \in \mathbf{E}_1$ or $\theta_1([v_1]_{\mathcal{W}}) < 1$, respectively, i.e., v_1 is a border state, as required.

This completes the proof that $(\mathcal{W}, \mathcal{C})$ is a minor cost weak probabilistic bisimulation, thus $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$. \square

Result A.11 (Proposition 6.9). *Minor cost weak probabilistic bisimilarity is a preorder on the set of CPAs.*

Proof. Reflexivity is straightforward and we omit it, so let us consider transitivity, that is, given three CPAs (\mathcal{A}_1, c_1) , (\mathcal{A}_2, c_2) , and (\mathcal{A}_3, c_3) , if $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$ and $(\mathcal{A}_2, c_2) \lesssim_p (\mathcal{A}_3, c_3)$, then $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_3, c_3)$.

Let $(\mathcal{W}_{21}, \mathcal{C}_{21})$ and $(\mathcal{W}_{32}, \mathcal{C}_{32})$ be a minor cost weak bisimulations justifying $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$ and $(\mathcal{A}_2, c_2) \lesssim_p (\mathcal{A}_3, c_3)$, respectively. Let \mathcal{W}_{31} be $\mathcal{W}_{32} \circ \mathcal{W}_{21}$. It is known [26] that \mathcal{W}_{31} is a weak probabilistic bisimulation between \mathcal{A}_1 and \mathcal{A}_3 . Let \mathcal{C}_{31} be $\mathcal{C}_{32} \circ \mathcal{C}_{21}$. We claim that $(\mathcal{W}_{31}, \mathcal{C}_{31})$ is a minor cost weak bisimulation from (\mathcal{A}_1, c_1) to (\mathcal{A}_3, c_3) .

$\bar{s}_3 \mathcal{C}_{31} \bar{s}_1$ is immediate since by hypothesis and by definition of $\mathcal{C}_{32} \circ \mathcal{C}_{21}$, $\bar{s}_3 \mathcal{C}_{32} \bar{s}_2 \mathcal{C}_{21} \bar{s}_1$.

It is immediate to see that for each $s_3 \in S_3$ there exists $s_1 \in S_1$ such that $s_3 \mathcal{C}_{31} s_1$: Let $s_3 \in S_3$; by definition of \mathcal{C}_{32} , there exists $s_2 \in S_2$ such that $s_3 \mathcal{C}_{32} s_2$ and by definition of \mathcal{C}_{21} , there exists $s_1 \in S_1$ such that $s_2 \mathcal{C}_{21} s_1$, hence $s_3 \mathcal{C}_{31} s_1$, as required. The fact that $s_3 \mathcal{W}_{31} s_1$ is immediate by the way \mathcal{W}_{31} is constructed and the fact that $\mathcal{C}_{32} \subseteq \mathcal{W}_{32}$ and $\mathcal{C}_{21} \subseteq \mathcal{W}_{21}$.

Before continuing with the proof, consider the set $\mathcal{B}(\mathcal{W}_{31})$: It is immediate to see that $\mathcal{B}(\mathcal{W}_{31}) \cap S_3 = \mathcal{B}(\mathcal{W}_{32}) \cap S_3$. In fact, by definition of $\mathcal{B}(\cdot)$, a state $t_3 \in S_3$ belongs to

$\mathcal{B}(\mathcal{W}_{31})$ since t_3 enables a transition $t_3 \xrightarrow{b} \rho_3$ such that $b \in \mathbf{E}_3$ (but this is independent from the equivalence relation, thus $t_3 \in \mathcal{B}(\mathcal{W}_{32}) \cap S_3$ or $\rho_3([t_3]_{\mathcal{W}_{31}}) < 1$, i.e., there exists $t'_3 \in \text{Supp}(\rho_3)$ such that $(t_3, t'_3) \notin \mathcal{W}_{31}$. By definition of \mathcal{W}_{31} , it follows that $(t_3, t'_3) \notin \mathcal{W}_{32}$ holds as well (otherwise $(t_3, t'_3) \in \mathcal{W}_{31}$ would hold), hence $t_3 \in \mathcal{B}(\mathcal{W}_{32}) \cap S_3$, as required. On the other hand, consider $t_3 \in S_3$ such that $t_3 \notin \mathcal{B}(\mathcal{W}_{31})$. This implies that for each $t_3 \xrightarrow{b} \rho_3$, $b \in \mathbf{H}$ and $\rho_3([t_3]_{\mathcal{W}_{31}}) = 1$, i.e., for each $t'_3 \in \text{Supp}(\rho_3)$, $t'_3 \mathcal{W}_{31} t_3$. By definition of \mathcal{W}_{31} , it follows that $t'_3 \mathcal{W}_{31} t_3$ holds as well, hence $\rho_3([t_3]_{\mathcal{W}_{32}}) = 1$, thus $t_3 \notin \mathcal{B}(\mathcal{W}_{32})$, as required.

Now, assume $s_3 \xrightarrow{a} \mu_3$ and $s_3 \mathcal{C}_{31} s_1$. Moreover, assume that $a \in \mathbf{E}$ (the case $a \in \mathbf{H}$ is just a simplification of this case). Let $s_2 \in S_2$ be a state such that $s_3 \mathcal{C}_{32} s_2 \mathcal{C}_{21} s_1$. There are two cases:

- (1) There exists $\rho_3 \in \text{Disc}(\mathcal{B}(\mathcal{W}_{31}) \cap S_3)$ such that $\mu_3 \xrightarrow{\tau} \rho_3$ and $\min\{c_3(\mu_3 \xrightarrow{\tau} \rho) \mid \rho \in \text{Disc}(\mathcal{B}(\mathcal{W}_{31}) \cap S_3)\} = c_3(\mu_3 \xrightarrow{\tau} \rho_3)$. Since $\mathcal{B}(\mathcal{W}_{31}) \cap S_3 = \mathcal{B}(\mathcal{W}_{32}) \cap S_3$ and $(\mathcal{A}_2, c_2) \lesssim_p (\mathcal{A}_3, c_3)$, it follows that there exists $\rho_2 \in \text{Disc}(\mathcal{B}(\mathcal{W}_{32}) \cap S_2)$ such that $s_2 \xrightarrow{a} \rho_2$, $\rho_3 \mathcal{L}(\mathcal{C}_{32}) \rho_2$, and $c_2(s_2 \xrightarrow{a} \rho_2) \leq c_3(s_3 \xrightarrow{a} \mu_3 \xrightarrow{\tau} \rho_3)$. Let $\gamma_2 \in \text{Disc}(S_2)$ be such that $s_2 \xrightarrow{\tau} \gamma_2 \xrightarrow{a} \rho_2$, that is, γ_2 is the probability distribution reached exactly before the a action along $s_2 \xrightarrow{a} \rho_2$. Note that $\text{Supp}(\gamma_2) \subseteq \mathcal{B}(\mathcal{W}_{32}) \cap S_2$. This implies that there exists $\gamma_1 \in \text{Disc}(\mathcal{B}(\mathcal{W}_{21}) \cap S_1)$ such that $s_1 \xrightarrow{\tau} \gamma_1$, $\gamma_2 \mathcal{L}(\mathcal{C}_{21}) \gamma_1$, and $c_1(s_1 \xrightarrow{\tau} \gamma_1) \leq c_2(s_2 \xrightarrow{\tau} \gamma_2)$. Now, from each $t_2 \in \text{Supp}(\gamma_2)$, let $t_2 \xrightarrow{a} \theta_{t_2}$ be the weak combined transition enabled by t_2 such that $\sum_{t_2 \in \text{Supp}(\gamma_2)} \gamma_2(t_2) \cdot \theta_{t_2} = \rho_2$. By construction, the external action is performed immediately, so $t_2 \xrightarrow{a} \theta_{t_2}$ is actually $t_2 \xrightarrow{a} \phi_{t_2} \xrightarrow{\tau} \theta_{t_2}$ for some distribution ϕ_{t_2} . Let $t_2 \xrightarrow{a} \varphi_{t_2} \xrightarrow{\tau} \psi_{t_2}$ be a component of such weak combined transition, i.e., $\sum_{t_2 \xrightarrow{a} \varphi_{t_2} \in D_2} \sigma(t_2)(t_2 \xrightarrow{a} \varphi_{t_2}) \cdot \varphi_{t_2} = \phi_{t_2}$ and $\sum_{t_2 \xrightarrow{a} \varphi_{t_2} \in D_2} \sigma(t_2)(t_2 \xrightarrow{a} \varphi_{t_2}) \cdot \psi_{t_2} = \theta_{t_2}$ where σ is the scheduler inducing $t_2 \xrightarrow{a} \theta_{t_2}$. Since $\rho_2 \in \text{Disc}(\mathcal{B}(\mathcal{W}_{32}) \cap S_2)$, it is immediate to see that also $\psi_{t_2} \in \text{Disc}(\mathcal{B}(\mathcal{W}_{32}) \cap S_2)$ for each $t_2 \in \text{Supp}(\gamma_2)$, thus for each $t_1 \in \text{Supp}(\gamma_1)$ such that $t_2 \mathcal{C}_{21} t_1$, there exists $\psi_{t_1} \in \text{Disc}(\mathcal{B}(\mathcal{W}_{21}) \cap S_1)$ such that $t_1 \xrightarrow{a} \psi_{t_1}$, $\psi_{t_2} \mathcal{L}(\mathcal{C}_{21}) \psi_{t_1}$, and $c_1(t_1 \xrightarrow{a} \psi_{t_1}) \leq c_2(t_2 \xrightarrow{a} \psi_{t_2})$. By combining these transitions to obtain $t_1 \xrightarrow{a} \theta_{t_1}$ as in the construction of $t_2 \xrightarrow{a} \theta_{t_2}$, we obtain that $\theta_{t_2} \mathcal{L}(\mathcal{C}_{21}) \theta_{t_1}$, and $c_1(t_1 \xrightarrow{a} \theta_{t_1}) \leq c_2(t_2 \xrightarrow{a} \theta_{t_2})$. By extending $s_1 \xrightarrow{\tau} \gamma_1$ with $\gamma_1 \xrightarrow{a} \rho_1$ with $\rho_1 = \sum_{t_1 \in \text{Supp}(\gamma_1)} \gamma_1(t_1) \cdot \theta_{t_1}$, we have that the resulting weak combined transition $s_1 \xrightarrow{a} \rho_1$ satisfies $\rho_1 \in \text{Disc}(\mathcal{B}(\mathcal{W}_{31}) \cap S_1)$, $\rho_3 \mathcal{L}(\mathcal{C}_{31}) \rho_1$, and $c_1(s_1 \xrightarrow{a} \rho_1) \leq c_3(s_3 \xrightarrow{a} \mu_3 \xrightarrow{\tau} \rho_3)$, as required.
- (2) There does not exist $\rho_3 \in \text{Disc}(\mathcal{B}(\mathcal{W}_{31}) \cap S_3)$ such that $\mu_3 \xrightarrow{\tau} \rho_3$. Since $\mathcal{B}(\mathcal{W}_{31}) \cap S_3 = \mathcal{B}(\mathcal{W}_{32}) \cap S_3$ and $(\mathcal{A}_2, c_2) \lesssim_p (\mathcal{A}_3, c_3)$, it follows that there exists $\rho_2 \in \text{Disc}(S_2)$ such that $s_2 \xrightarrow{a} \rho_2$, $\rho_3 \mathcal{L}(\mathcal{C}_{32}) \rho_2$, and $c_2(s_2 \xrightarrow{a} \rho_2) \leq c_3(s_3 \xrightarrow{a} \mu_3)$. As in the case (1), let $\gamma_2 \in \text{Disc}(S_2)$ be such that $s_2 \xrightarrow{\tau} \gamma_2 \xrightarrow{a} \rho_2$, that is, γ_2 are the probability distribution reached exactly before the a action along $s_2 \xrightarrow{a} \rho_2$. Note that $\text{Supp}(\gamma_2) \subseteq \mathcal{B}(\mathcal{W}_{32}) \cap S_2$. This implies that there exists $\gamma_1 \in \text{Disc}(\mathcal{B}(\mathcal{W}_{21}) \cap S_1)$ such that $s_1 \xrightarrow{\tau} \gamma_1$, $\gamma_2 \mathcal{L}(\mathcal{C}_{21}) \gamma_1$, and $c_1(s_1 \xrightarrow{\tau} \gamma_1) \leq c_2(s_2 \xrightarrow{\tau} \gamma_2)$. Now, from each $t_2 \in \text{Supp}(\gamma_2)$, let $t_2 \xrightarrow{a} \theta_{t_2}$ be the weak combined transition enabled by t_2 such that $\sum_{t_2 \in \text{Supp}(\gamma_2)} \gamma_2(t_2) \cdot \theta_{t_2} = \rho_2$. For each $t_2 \in \text{Supp}(\gamma_2)$, if $\text{Supp}(\theta_{t_2}) \subseteq \mathcal{B}(\mathcal{W}_{21}) \cap S_2$, then we are in the same situation as in the case (1), that is, we are able to construct $t_1 \xrightarrow{a} \theta_{t_1}$ such that $\theta_{t_2} \mathcal{L}(\mathcal{C}_{21}) \theta_{t_1}$ and $c_1(t_1 \xrightarrow{a} \theta_{t_1}) \leq c_2(t_2 \xrightarrow{a} \theta_{t_2})$. Now, suppose that $\text{Supp}(\theta_{t_2}) \not\subseteq \mathcal{B}(\mathcal{W}_{21}) \cap S_2$. Let ϕ_{t_2} be the distribution such that $t_2 \xrightarrow{a} \phi_{t_2} \xrightarrow{\tau} \theta_{t_2}$. Let $t_2 \xrightarrow{a} \varphi_{t_2} \xrightarrow{\tau} \psi_{t_2}$ be a component of such weak combined transition, i.e., $\sum_{t_2 \xrightarrow{a} \varphi_{t_2} \in D_2} \sigma(t_2)(t_2 \xrightarrow{a} \varphi_{t_2}) \cdot \varphi_{t_2} =$

ϕ_{t_2} and $\sum_{t_2 \xrightarrow{a} \varphi_{t_2} \in D_2} \sigma(t_2)(t_2 \xrightarrow{a} \varphi_{t_2}) \cdot \psi_{t_2} = \theta_{t_2}$ where σ is the scheduler inducing $t_2 \xrightarrow{a} \theta_{t_2}$. If $\text{Supp}(\psi_{t_2}) \subseteq \mathcal{B}(\mathcal{W}_{21}) \cap S_2$, then we are in the same situation as in the case (1), otherwise from each $u_2 \in \text{Supp}(\varphi_2)$, let $u_2 \xrightarrow{\tau} \kappa_{u_2}$ be the weak combined transition enabled by u_2 such that $\sum_{u_2 \in \text{Supp}(\varphi_2)} \varphi_2(u_2) \cdot \kappa_{u_2} = \psi_2$. For each $u_2 \in \text{Supp}(\varphi_2)$, if $\text{Supp}(\kappa_{u_2}) \subseteq \mathcal{B}(\mathcal{W}_{21}) \cap S_2$, then we are in the same situation as in the case (1), so suppose that $\text{Supp}(\kappa_{u_2}) \not\subseteq \mathcal{B}(\mathcal{W}_{21}) \cap S_2$. Let λ_{u_2} be the distribution such that $u_2 \xrightarrow{\tau} \lambda_{u_2} \xrightarrow{\tau} \kappa_{u_2}$. (Note that here we are assuming that u_2 does not stop immediately with non-zero probability. This is not an issue since if u_2 needs to stop immediately, then this is matched by u_1 such that $u_2 \mathcal{C}_{21} u_1$ by stopping immediately with the same probability, and in both cases the cost is 0.) Let $u_2 \xrightarrow{\tau} \xi_{u_2} \xrightarrow{\tau} \chi_{u_2}$ be a component of such weak combined transition, i.e., $\sum_{u_2 \xrightarrow{\tau} \xi_{u_2} \in D_2} \sigma(u_2)(u_2 \xrightarrow{\tau} \xi_{u_2}) \cdot \xi_{u_2} = \lambda_{u_2}$ and $\sum_{u_2 \xrightarrow{\tau} \xi_{u_2} \in D_2} \sigma(u_2)(u_2 \xrightarrow{\tau} \xi_{u_2}) \cdot \chi_{u_2} = \kappa_{u_2}$ where σ_u is the scheduler inducing $u_2 \xrightarrow{\tau} \kappa_{u_2}$. If $\text{Supp}(\chi_{u_2}) \subseteq \mathcal{B}(\mathcal{W}_{21}) \cap S_2$, then we are in the same situation as in the case (1), otherwise we can apply the same technique until we obtain a strong transition $v_2 \xrightarrow{\tau} \omega_{v_2}$ such that there does not exist $\varpi_{v_2} \in \text{Disc}(\mathcal{B}(\mathcal{W}_{32}) \cap S_2)$ such that $\omega_{v_2} \xrightarrow{\tau} \varpi_{v_2}$ (actually, we usually obtain a strong combined transition, but we just focus on its components). For this transition, since $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$, we have that for each $v_1 \in S_1$ such that $v_2 \mathcal{C}_{21} v_1$, there exists $\omega_{v_1} \in \text{Disc}(S_1)$ such that $v_1 \xrightarrow{\tau} \omega_{v_1}$, $\omega_2 \mathcal{L}(\mathcal{C}_{21}) \omega_1$ and $c_1(v_1 \xrightarrow{\tau} \omega_1) \leq c_2(v_2 \xrightarrow{\tau} \omega_2)$. By combining all these weak combined transitions according to the combinations used for constructing $s_2 \xrightarrow{a} \rho_2$, we obtain a weak combined transition $s_1 \xrightarrow{a} \rho_1$ such that $\rho_1 \in \text{Disc}(S_1)$, $\rho_3 \mathcal{L}(\mathcal{C}_{31}) \rho_1$, and $c_1(s_1 \xrightarrow{a} \rho_1) \leq c_3(s_3 \xrightarrow{\tau} \mu_3)$, as required.

This completes the proof that $(\mathcal{W}_{31}, \mathcal{C}_{31})$ is a minor cost weak bisimulation from (\mathcal{A}_1, c_1) to (\mathcal{A}_3, c_3) , thus $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_3, c_3)$. \square

Result A.12 (Proposition 6.10). *Given two CPAs (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) , if $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$, then for each (\mathcal{A}_3, c_3) compatible with both (\mathcal{A}_1, c_1) and (\mathcal{A}_2, c_2) and each pair of cost-preserving functions c_l and c_r with $\text{gen}(c_l) = \text{gen}(c_r)$, $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3) \lesssim_p (\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$.*

Proof. Denoting by S_{12} the set $S_1 \uplus S_2$, let $(\mathcal{W}, \mathcal{C})$ be a minor cost weak probabilistic bisimulation justifying $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$ and $(\mathcal{W}_p, \mathcal{C}_p)$ be defined as follows:

- \mathcal{W}_p is the weak probabilistic bisimulation between $\mathcal{A}_1 \parallel \mathcal{A}_3$ and $\mathcal{A}_2 \parallel \mathcal{A}_3$. Note that $\mathcal{W}_p = \mathcal{W} \times \mathcal{I}$, where \mathcal{I} is the identity relation on S_3 . Its existence is ensured by the fact that the $\mathcal{A}_1 \approx_p \mathcal{A}_2$ and the weak probabilistic bisimulation on probabilistic automata is preserved by parallel composition;
- $\mathcal{C}_p = \{((s_2, s_3), (s_1, s_3)) \mid (s_2, s_1) \in \mathcal{C}, s_3 \in S_3\}$. Essentially, \mathcal{C}_p is the product of \mathcal{C} with the identity relation \mathcal{I} on S_3 .

By the way \mathcal{C}_p is defined, it is immediate to see that $\mathcal{C}_p \subseteq \mathcal{W}_p \cap (S_2 \times S_3) \times (S_1 \times S_3)$ such that for each $(s_2, s_3) \in S_2 \times S_3$ there exists $(s_1, s_3) \in S_1 \times S_3$ such that $(s_2, s_3) \mathcal{C}_p (s_1, s_3)$.

We now show that $(\mathcal{W}_p, \mathcal{C}_p)$ is actually a minor cost weak probabilistic bisimulation from $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3)$ to $(\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$.

The fact that \mathcal{W}_p is a weak probabilistic bisimulation between $\mathcal{A}_1 \parallel \mathcal{A}_3$ and $\mathcal{A}_2 \parallel \mathcal{A}_3$ is immediate by definition. Since by hypothesis, $\bar{s}_2 \mathcal{C} \bar{s}_1$, it is immediate to see that $(\bar{s}_2, \bar{s}_3) \mathcal{C}_p (\bar{s}_1, \bar{s}_3)$. So, consider a pair of states (s_2, s_3) and (s_1, s_3) such that $(s_2, s_3) \mathcal{C}_p (s_1, s_3)$ and a transition $(s_2, s_3) \xrightarrow{a} \mu_2 \times \mu_3$. Now, there are three cases:

Case $a \in \Sigma_3 \setminus \Sigma_2$: In this case, $\mu_2 = \delta_{s_2}$. Suppose that there exists the distribution $\rho_2 \times \rho_3 \in \text{Disc}(\mathcal{B}(\mathcal{W}_p) \cap (S_2 \times S_3))$ such that $\mu_2 \times \mu_3 \xrightarrow{\tau}_C \rho_2 \times \rho_3$. Since $\mu_2 \times \mu_3 \xrightarrow{\tau}_C \rho_2 \times \rho_3$ is an internal weak combined transition (we remark that τ is used as symbol for any internal action, it is not a specific action), each transition chosen by the scheduler inducing $\mu_2 \times \mu_3 \xrightarrow{\tau}_C \rho_2 \times \rho_3$ either corresponds to a transition from \mathcal{A}_2 or from \mathcal{A}_3 , but none of them is the result of the synchronisation between transitions of the two automata. Moreover, when a transition from \mathcal{A}_2 is performed from the product state (t_2, t_3) , the reached states are of the form (t'_2, t_3) , and similarly for transitions from \mathcal{A}_3 . Since each state (t_2, t_3) is a border state either because t_2 or t_3 is a border state, the minimum cost is obtained only by choosing transitions from either \mathcal{A}_2 or \mathcal{A}_3 , but not from both. The only exception is when transitions have cost 0, since they do not affect the resulting cost but they also do not affect whether it is cheaper to reach the border with only transitions from either \mathcal{A}_2 or \mathcal{A}_3 .

Now, suppose that only transitions from \mathcal{A}_2 are used, hence we have $\rho_3 = \mu_3$: Let $\{s_2 \xrightarrow{\tau} \theta_i\}_{i \in I}$ be the set of transitions chosen by the scheduler σ for the execution fragment $\alpha = s_2$ during the construction of $\mu_2 \xrightarrow{\tau}_C \rho_2$. Since $s_2 \mathcal{C} s_1$ and for each θ_i there exists $\phi_i \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_2)$ such that $\theta_i \xrightarrow{\tau}_C \phi_i$ and $\sum_{i \in I} \sigma(s_2)(s_2 \xrightarrow{\tau} \theta_i) \cdot \phi_i = \rho_2$, by $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$ it follows that there exists $\chi_i \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_1)$ such that $s_1 \xrightarrow{\tau}_C \chi_i$, $\phi_i \mathcal{L}(\mathcal{C}) \chi_i$, and $c_1(s_1 \xrightarrow{\tau}_C \chi_i) \leq c_2(s_2 \xrightarrow{\tau} \mu_2 \xrightarrow{\tau}_C \phi_i)$. It is easy to see that the convex combination $\sum_{i \in I} \sigma(s_2)(s_2 \xrightarrow{\tau} \theta_i) \cdot s_1 \xrightarrow{\tau}_C \chi_i$ results in a transition $s_1 \xrightarrow{\tau}_C \chi$ such that $\chi \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_1)$, $\rho_2 \mathcal{L}(\mathcal{C}) \chi$, and $c_1(s_1 \xrightarrow{\tau}_C \chi) \leq c_2(s_2 \xrightarrow{\tau} \mu_2 \xrightarrow{\tau}_C \rho_2)$. Since $\rho_2 \mathcal{L}(\mathcal{C}) \chi$, we have $\rho_2 \times \rho_3 \mathcal{L}(\mathcal{C}_p) \chi \times \rho_3$ and $\chi \times \rho_3 \in \text{Disc}(\mathcal{B}(\mathcal{W}_p) \cap S_1 \times S_3)$. To complete the proof for this case, we need to show that $c_l((s_1, s_3) \xrightarrow{a}_C \chi \times \mu_3) \leq c_r((s_2, s_3) \xrightarrow{a}_C \rho_2 \times \mu_3)$:

$$\begin{aligned} c_l((s_1, s_3) \xrightarrow{a}_C \chi \times \mu_3) &= \text{gen}(c_l)(c_1(s_1 \xrightarrow{\tau}_C \chi), c_3(s_3 \xrightarrow{a}_C \mu_3)) \\ &\leq \text{gen}(c_l)(c_2(s_2 \xrightarrow{\tau} \mu_2 \xrightarrow{\tau}_C \rho_2), c_3(s_3 \xrightarrow{a}_C \mu_3)) \\ &= \text{gen}(c_r)(c_2(s_2 \xrightarrow{\tau} \mu_2 \xrightarrow{\tau}_C \rho_2), c_3(s_3 \xrightarrow{a}_C \mu_3)) \\ &= c_r((s_2, s_3) \xrightarrow{a}_C \rho_2 \times \mu_3), \end{aligned}$$

hence $c_l((s_1, s_3) \xrightarrow{a}_C \chi \times \mu_3) \leq c_r((s_2, s_3) \xrightarrow{a}_C \rho_2 \times \mu_3)$ as required.

Instead, if only transitions from \mathcal{A}_3 are used, hence $\rho_2 = \delta_{s_2}$, then we simply consider the weak combined transition $(s_1, s_3) \xrightarrow{a} \delta_{s_1} \times \mu_3 \xrightarrow{\tau}_C \delta_{s_1} \times \rho_3$ obtained by performing only transitions from \mathcal{A}_3 that fulfils the required properties: $\delta_{s_2} \times \rho_3 \mathcal{L}(\mathcal{C}_p) \delta_{s_1} \times \rho_3$,

$$\begin{aligned} c_l((s_1, s_3) \xrightarrow{a}_C \delta_{s_1} \times \rho_3) &= \text{gen}(c_l)(c_1(s_1 \xrightarrow{\nu_a} \delta_{s_1}), c_3(s_3 \xrightarrow{a} \mu_3 \xrightarrow{\tau}_C \rho_3)) \\ &= \text{gen}(c_l)(0, c_3(s_3 \xrightarrow{a} \mu_3 \xrightarrow{\tau}_C \rho_3)) \\ &= \text{gen}(c_r)(0, c_3(s_3 \xrightarrow{a} \mu_3 \xrightarrow{\tau}_C \rho_3)) \\ &= \text{gen}(c_r)(c_2(s_2 \xrightarrow{\nu_a} \delta_{s_2}), c_3(s_3 \xrightarrow{a} \mu_3 \xrightarrow{\tau}_C \rho_3)) \\ &= c_r((s_2, s_3) \xrightarrow{a}_C \delta_{s_2} \times \rho_3), \end{aligned}$$

thus we obtain that $c_l((s_1, s_3) \xrightarrow{a}_C \delta_{s_1} \times \rho_3) \leq c_r((s_2, s_3) \xrightarrow{a}_C \delta_{s_2} \times \rho_3)$, as expected.

Suppose that there does not exist $\rho_2 \times \rho_3 \in \text{Disc}(\mathcal{B}(\mathcal{W}_p) \cap (S_2 \times S_3))$ such that $\mu_2 \times \mu_3 \xrightarrow{\tau}_C \rho_2 \times \rho_3$: The step condition is trivially satisfied by taking $\mu_1 = \delta_{s_1}$ and the weak combined transition $(s_1, s_3) \xrightarrow{a}_C \delta_{s_1} \times \mu_3 = (s_1, s_3) \xrightarrow{a} \delta_{s_1} \times \mu_3$. The condition $\delta_{s_2} \times \mu_3 \mathcal{L}(\mathcal{C}_p) \delta_{s_1} \times \mu_3$ trivially holds since $s_2 \mathcal{C} s_1$ and thus, for each $t_3 \in \text{Supp}(\mu_3)$,

$(s_2, t_3) \mathcal{C}_p (s_1, t_3);$

$$\begin{aligned}
c_l((s_1, s_3) \xrightarrow{a}_C \delta_{s_1} \times \mu_3) &= c_l((s_1, s_3) \xrightarrow{a} \delta_{s_1} \times \mu_3) \\
&= \text{gen}(c_l)(c_1(s_1 \xrightarrow{\nu_a} \delta_{s_1}), c_3(s_3 \xrightarrow{a} \mu_3)) \\
&= \text{gen}(c_l)(0, c_3(s_3 \xrightarrow{a} \mu_3)) \\
&= \text{gen}(c_r)(0, c_3(s_3 \xrightarrow{a} \mu_3)) \\
&= \text{gen}(c_r)(c_2(s_2 \xrightarrow{\nu_a} \delta_{s_2}), c_3(s_3 \xrightarrow{a} \mu_3)) \\
&= c_r((s_2, s_3) \xrightarrow{a} \delta_{s_2} \times \mu_3),
\end{aligned}$$

thus $c_l((s_1, s_3) \xrightarrow{a}_C \delta_{s_1} \times \mu_3) \leq c_r((s_2, s_3) \xrightarrow{a} \delta_{s_2} \times \mu_3)$, as required.

Case $a \in \Sigma_2 \setminus \Sigma_3$: It is essentially the same as the previous case, where the roles of $\mathcal{A}_2/\mathcal{A}_1$ and \mathcal{A}_3 are exchanged. For instance, we have $\mu_3 = \delta_{s_3}$ and consider the case that there exists the distribution $\rho_2 \times \rho_3 \in \text{Disc}(\mathcal{B}(\mathcal{W}_p) \cap (S_2 \times S_3))$ such that $\mu_2 \times \mu_3 \xrightarrow{\tau}_C \rho_2 \times \rho_3$ and suppose that only transitions from \mathcal{A}_2 are used in such transition. Let $\{s_2 \xrightarrow{\tau} \theta_i\}_{i \in I}$ be the set of transitions chosen by the scheduler σ for the execution fragment $\alpha = s_2$ during the construction of $\mu_2 \xrightarrow{\tau}_C \rho_2$. Since $(\mathcal{A}_c, 1_c) \lesssim_p (\mathcal{A}_2, c_2)$, $s_2 \mathcal{C} s_1, s_2 \xrightarrow{a} \mu_2$, and for each θ_i there exists $\phi_i \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_2)$ such that $\theta_i \xrightarrow{\tau}_C \phi_i$ and $\sum_{i \in I} \sigma(s_2)(s_2 \xrightarrow{\tau} \theta_i) \cdot \phi_i = \rho_2$, by $(\mathcal{A}_1, c_1) \lesssim_p (\mathcal{A}_2, c_2)$ it follows that there exists $\chi_i \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_1)$ such that $s_1 \xrightarrow{a}_C \chi_i$, $\phi_i \mathcal{L}(\mathcal{C}) \chi_i$, and $c_1(s_1 \xrightarrow{a}_C \chi_i) \leq c_2(s_2 \xrightarrow{\tau} \mu_2 \xrightarrow{\tau}_C \phi_i)$. It is easy to see that the convex combination $\sum_{i \in I} \sigma(s_2)(s_2 \xrightarrow{\tau} \theta_i) \cdot s_1 \xrightarrow{a}_C \chi_i$ results in a transition $s_1 \xrightarrow{a}_C \chi$ such that $\chi \in \text{Disc}(\mathcal{B}(\mathcal{W}) \cap S_1)$, $\rho_2 \mathcal{L}(\mathcal{C}) \chi$, and $c_1(s_1 \xrightarrow{a}_C \chi) \leq c_2(s_2 \xrightarrow{a} \mu_2 \xrightarrow{\tau}_C \rho_2)$. Since $\rho_2 \mathcal{L}(\mathcal{C}) \chi$ and trivially $\rho_3 \mathcal{L}(\mathcal{C}) \rho_3$, we have $\rho_2 \times \rho_3 \mathcal{L}(\mathcal{C}_p) \chi \times \rho_3$ and $\chi \times \rho_3 \in \text{Disc}(\mathcal{B}(\mathcal{W}_p) \cap S_1 \times \rho_3)$. To complete the proof for this case, we need to show that $c_l((s_1, s_3) \xrightarrow{a}_C \chi \times \mu_3) \leq c_r((s_2, s_3) \xrightarrow{a}_C \rho_2 \times \mu_3)$:

$$\begin{aligned}
c_l((s_1, s_3) \xrightarrow{a}_C \chi \times \mu_3) &= \text{gen}(c_r)(c_1(s_1 \xrightarrow{a}_C \chi), c_3(s_3 \xrightarrow{\nu_a} \delta_{s_3})) \\
&= \text{gen}(c_r)(c_1(s_1 \xrightarrow{a}_C \chi), 0) \\
&\leq \text{gen}(c_r)(c_2(s_2 \xrightarrow{\tau} \mu_2 \xrightarrow{\tau}_C \rho_2), 0) \\
&= \text{gen}(c_l)(c_2(s_2 \xrightarrow{\tau} \mu_2 \xrightarrow{\tau}_C \rho_2), 0) \\
&= \text{gen}(c_l)(c_2(s_2 \xrightarrow{\tau} \mu_2 \xrightarrow{\tau}_C \rho_2), c_3(s_3 \xrightarrow{\nu_a} \delta_{s_3})) \\
&= c_r((s_2, s_3) \xrightarrow{a}_C \rho_2 \times \mu_3),
\end{aligned}$$

hence $c_l((s_1, s_3) \xrightarrow{a}_C \chi \times \mu_3) \leq c_r((s_2, s_3) \xrightarrow{a}_C \rho_2 \times \mu_3)$ as required.

Case $a \in \Sigma_3 \cap \Sigma_2$: the definition of parallel composition implies that one obtains the transition $(s_2, s_3) \xrightarrow{a} \mu_2 \times \mu_3$ by combining the transitions $s_2 \xrightarrow{a} \mu_2 \in D_2$ and $s_3 \xrightarrow{a} \mu_3 \in D_3$.

The remainder of the proof for this case is just the expected combination of the above two cases.

This completes the proof that $(\mathcal{W}_p, \mathcal{C}_p)$ is a minor cost weak probabilistic bisimulation, thus $(\mathcal{A}_1, c_1) \parallel_{c_l} (\mathcal{A}_3, c_3) \lesssim_p (\mathcal{A}_2, c_2) \parallel_{c_r} (\mathcal{A}_3, c_3)$. \square

Result A.13 (Theorem 7.7). *Given a CPA (\mathcal{A}, c) , $\mathcal{R} \subseteq S \times S$, $a \in \Sigma$, $\mu \in \text{Disc}(S)$, and $t \in S$, consider the $\min_c t \xrightarrow{a}_C \mathcal{L}(\mathcal{R})\mu$ LP problem. The following implications hold:*

- (1) *If there exists a scheduler σ for \mathcal{A} that induces $t \xrightarrow{a}_C \mu_t$ such that $\mu \mathcal{L}(\mathcal{R}) \mu_t$, then $\min_c t \xrightarrow{a}_C \mathcal{L}(\mathcal{R})\mu$ has an optimal solution f^o such that $\mathfrak{C} \leq c(t \xrightarrow{a}_C \mu_t)$.*

- (2) If $\min_c t \xrightarrow{a}_C \mathcal{L}(\mathcal{R})\mu$ has an optimal solution f^o , then there exists a scheduler σ for \mathcal{A} that induces $t \xrightarrow{a}_C \mu_t$ such that $\mu \mathcal{L}(\mathcal{R}) \mu_t$ and $c(t \xrightarrow{a}_C \mu_t) = \mathfrak{C}$.

Proof. The proof is mainly based on the proof of [15, Theorem 8]. We recall that the min cost \mathfrak{C} is defined as $\mathfrak{C} = \sum_{(x,y) \in E} c_f((x,y)) \cdot f_{x,y}^o$.

Let $F(b)$ be the set $\{\alpha \in \text{frags}^*(\mathcal{A}) \mid \text{trace}(\alpha) = \text{trace}(b)\}$ and $F(b,q)$ be the set $\{\alpha \in F(b) \mid \text{last}(\alpha) = q\}$; denote by E_τ , E^a , and E_τ^a the sets $\{(q, q^{\tau}) \mid tr = q \xrightarrow{\tau} \mu \in D\}$, $\{(q, q_a^{\tau}) \mid tr = q \xrightarrow{a} \mu \in D\}$, and $\{(q_a, q_a^{\tau}) \mid tr = q \xrightarrow{\tau} \mu \in D\}$, respectively.

We prove the theorem for $a \in \mathbf{E}$; the case $a \in \mathbf{H}$ is similar.

- (1) Suppose that there exists a scheduler σ for \mathcal{A} that induces $t \xrightarrow{a}_C \mu_t$ such that $\mu \mathcal{L}(\mathcal{R}) \mu_t$. This implies, by [15, Theorem 8], that $t \xrightarrow{a}_C \mathcal{L}(\mathcal{R})\mu$ has a solution f^* such that for each transition $tr = q \xrightarrow{b} \mu$,

- (a) $f_{q,q^{\tau}}^* = \sum_{\alpha \in F(\tau,q)} \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr)$ if $b = \tau$,
- (b) $f_{q_a,q_a^{\tau}}^* = \sum_{\alpha \in F(a,q)} \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr)$ if $b = \tau$, and
- (c) $f_{q,q_a^{\tau}}^* = \sum_{\alpha \in F(\tau,q)} \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr)$ if $b \neq \tau$,

This implies that $\sum_{(x,y) \in E} c_f((x,y)) f_{x,y}^* = c_\sigma(t \xrightarrow{a}_C \mu_t)$. In fact, it holds that

$$\begin{aligned}
& \sum_{(x,y) \in E} c_f((x,y)) \cdot f_{x,y}^* \\
&= \sum_{(x,y) \in E_\tau} c_f((x,y)) \cdot f_{x,y}^* + \sum_{(x,y) \in E_\tau^a} c_f((x,y)) \cdot f_{x,y}^* \\
& \quad + \sum_{(x,y) \in E^a} c_f((x,y)) \cdot f_{x,y}^* + \sum_{(x,y) \in E \setminus (E_\tau \cup E^a \cup E_\tau^a)} c_f((x,y)) \cdot f_{x,y}^* \\
&= \sum_{tr=q \xrightarrow{\tau} \mu \in D} c(tr) \cdot f_{q,q^{\tau}}^* + \sum_{tr=q \xrightarrow{\tau} \mu \in D} c(tr) \cdot f_{q_a,q_a^{\tau}}^* + \sum_{tr=q \xrightarrow{a} \mu \in D} c(tr) \cdot f_{q,q_a^{\tau}}^* \\
&= \sum_{tr=q \xrightarrow{\tau} \mu \in D} c(tr) \cdot \sum_{\alpha \in F(\tau,q)} \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr) \\
& \quad + \sum_{tr=q \xrightarrow{\tau} \mu \in D} c(tr) \cdot \sum_{\alpha \in F(a,q)} \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr) \\
& \quad + \sum_{tr=q \xrightarrow{a} \mu \in D} c(tr) \cdot \sum_{\alpha \in F(\tau,q)} \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr) \\
&= \sum_{tr=q \xrightarrow{\tau} \mu \in D} \sum_{\alpha \in F(\tau,q)} c(tr) \cdot \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr) \\
& \quad + \sum_{tr=q \xrightarrow{\tau} \mu \in D} \sum_{\alpha \in F(a,q)} c(tr) \cdot \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr) \\
& \quad + \sum_{tr=q \xrightarrow{a} \mu \in D} \sum_{\alpha \in F(\tau,q)} c(tr) \cdot \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr) \\
&= \sum_{\alpha \in F(\tau)} \sum_{tr \in D(\tau)} c(tr) \cdot \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr)
\end{aligned}$$

$$\begin{aligned}
& + \sum_{\alpha \in F(a)} \sum_{tr \in D(\tau)} c(tr) \cdot \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr) \\
& + \sum_{\alpha \in F(\tau)} \sum_{tr \in D(a)} c(tr) \cdot \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr)
\end{aligned}$$

since $\sigma(\alpha)(tr) = 0$ when $src(tr) \neq last(\alpha)$

$$\begin{aligned}
& = \sum_{\alpha \in F(\tau)} \sum_{tr \in D(\tau)} c(tr) \cdot \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr) \sum_{r \in S} \mu_{tr}(r) \\
& + \sum_{\alpha \in F(a)} \sum_{tr \in D(\tau)} c(tr) \cdot \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr) \sum_{r \in S} \mu_{tr}(r) \\
& + \sum_{\alpha \in F(\tau)} \sum_{tr \in D(a)} c(tr) \cdot \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr) \sum_{r \in S} \mu_{tr}(r) \\
& = \sum_{\alpha \in F(\tau)} \sum_{r \in S} \sum_{tr \in D(\tau)} c(tr) \cdot \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr) \cdot \mu_{tr}(r) \\
& + \sum_{\alpha \in F(a)} \sum_{r \in S} \sum_{tr \in D(\tau)} c(tr) \cdot \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr) \cdot \mu_{tr}(r) \\
& + \sum_{\alpha \in F(\tau)} \sum_{r \in S} \sum_{tr \in D(a)} c(tr) \cdot \mu_{\sigma,t}(C_\alpha) \cdot \sigma(\alpha)(tr) \cdot \mu_{tr}(r) \\
& = \sum_{\alpha \in F(\tau)} \sum_{r \in S} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(\tau)} c(tr) \cdot \sigma(\alpha)(tr) \cdot \mu_{tr}(r) \\
& + \sum_{\alpha \in F(a)} \sum_{r \in S} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(\tau)} c(tr) \cdot \sigma(\alpha)(tr) \cdot \mu_{tr}(r) \\
& + \sum_{\alpha \in F(\tau)} \sum_{r \in S} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(a)} c(tr) \cdot \sigma(\alpha)(tr) \cdot \mu_{tr}(r) \\
& = \sum_{\alpha \tau r \in \{\beta \tau r \mid \beta \in F(\tau)\}} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(\tau)} c(tr) \cdot \sigma(\alpha)(tr) \cdot \mu_{tr}(r) \\
& + \sum_{\alpha \tau r \in \{\beta \tau r \mid \beta \in F(a)\}} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(\tau)} c(tr) \cdot \sigma(\alpha)(tr) \cdot \mu_{tr}(r) \\
& + \sum_{\alpha a r \in \{\beta a r \mid \beta \in F(\tau)\}} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(a)} c(tr) \cdot \sigma(\alpha)(tr) \cdot \mu_{tr}(r) \\
& = \sum_{abr \in \{\beta br \mid \beta \in F(\tau) \cup F(a)\}} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(b)} c(tr) \cdot \sigma(\alpha)(tr) \cdot \mu_{tr}(r)
\end{aligned}$$

since $\sigma(\alpha)(tr) = 0$ when $trace(\alpha br) \notin \{\varepsilon, trace(a)\}$

$$\begin{aligned}
& = \sum_{abr \in \{\beta br \mid \beta \in frags^*(A)\}} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(b)} c(tr) \cdot \sigma(\alpha)(tr) \cdot \mu_{tr}(r) \\
& = 0 + \sum_{abr \in \{\beta br \mid \beta \in frags^*(A)\}} c_{\sigma,t}(\alpha br)
\end{aligned}$$

$$\begin{aligned}
&= \sum_{\alpha \in \{ \beta \in \text{frags}^*(\mathcal{A}) \mid \beta = q \in S \}} c_{\sigma,t}(\alpha) + \sum_{\alpha \in \{ \beta br \mid \beta \in \text{frags}^*(\mathcal{A}) \}} c_{\sigma,t}(\alpha) \\
&= \sum_{\alpha \in \text{frags}^*(\mathcal{A})} c_{\sigma,t}(\alpha) \\
&= c_{\sigma}(t \xrightarrow{a} \mathcal{C} \mu_t).
\end{aligned}$$

Since f^* is a feasible solution of $t \xrightarrow{a} \mathcal{C} \mathcal{L}(\mathcal{R})\mu$, Proposition 7.5 implies that f^* is also a feasible solution of $\min_c t \xrightarrow{a} \mathcal{C} \mathcal{L}(\mathcal{R})\mu$. This implies that there exists a (possibly different) optimal solution f^o such that $\mathfrak{C} = \sum_{(x,y) \in E} c_f((x,y)) \cdot f_{x,y}^o \leq \sum_{(x,y) \in E} c_f((x,y)) \cdot f_{x,y}^* = c_{\sigma}(t \xrightarrow{a} \mathcal{C} \mu_t)$, as required.

- (2) Suppose that $\min_c t \xrightarrow{a} \mathcal{C} \mathcal{L}(\mathcal{R})\mu$ has an optimal solution f^o ; Proposition 7.5 implies that f^o is also a feasible solution of $t \xrightarrow{a} \mathcal{C} \mathcal{L}(\mathcal{R})\mu$ and thus, by [15, Theorem 8], there exists a scheduler σ for \mathcal{A} that induces $t \xrightarrow{a} \mathcal{C} \mu_t$ such that $\mu \mathcal{L}(\mathcal{R}) \mu_t$. In particular, from the proof of [15, Theorem 8] we know that such scheduler is defined as follows: For each execution fragment $\phi \in \text{frags}^*(\mathcal{A})$,

$$\sigma(\phi)(x) = \begin{cases} f_{v,v^{tr}}^o / \vec{f}_v^o & \text{if } \vec{f}_v^o \neq 0, \text{ trace}(\phi) = \varepsilon, \text{ and } x = tr = v \xrightarrow{\tau} \rho \in D; \\ f_{v,v^a}^o / \vec{f}_v^o & \text{if } \vec{f}_v^o \neq 0, \text{ trace}(\phi) = \varepsilon, a \neq \tau, \text{ and } x = tr = v \xrightarrow{a} \rho \in D; \\ f_{v_a,v_a^{tr}}^o / \vec{f}_{v_a}^o & \text{if } \vec{f}_{v_a}^o \neq 0, \text{ trace}(\phi) = a \neq \tau, \text{ and } x = tr = v \xrightarrow{\tau} \rho \in D; \\ f_{v,\mathcal{R}(v)}^o / \vec{f}_v^o & \text{if } \vec{f}_v^o \neq 0, \text{ trace}(\phi) = \varepsilon, a = \tau, \text{ and } x = \perp; \\ f_{v_a,\mathcal{R}(v)}^o / \vec{f}_{v_a}^o & \text{if } \vec{f}_{v_a}^o \neq 0, \text{ trace}(\phi) = a \neq \tau, \text{ and } x = \perp; \\ 1 & \text{if } \text{trace}(\phi) \notin \{ \varepsilon, \text{trace}(a) \} \text{ and } x = \perp; \\ 1 & \text{if } \vec{f}_v^o = 0, \text{ trace}(\phi) = \varepsilon \text{ and } x = \perp; \\ 1 & \text{if } \vec{f}_{v_a}^o = 0, \text{ trace}(\phi) = a \neq \tau \text{ and } x = \perp; \\ 0 & \text{otherwise} \end{cases}$$

where $v = \text{last}(\phi)$, $f_{v,\mathcal{R}(v)}^o$ is the total flow from the vertex v to the vertices $u_{\mathcal{R}}$ such that $v \mathcal{R} u$, and \vec{f}_v^o is the total incoming flow in the vertex v . As pointed out in the proof of [15, Theorem 8], σ is a determinate scheduler, i.e., for each pair of $\phi, \phi' \in \text{frags}^*(\mathcal{A})$ such that $\text{last}(\phi) = \text{last}(\phi')$ and $\text{trace}(\phi) = \text{trace}(\phi')$, we have that $\sigma(\phi) = \sigma(\phi')$.

$$\begin{aligned}
&c_{\sigma}(t \xrightarrow{a} \mathcal{C} \mu_t) \\
&= \sum_{\alpha \in \text{frags}^*(\mathcal{A})} c_{\sigma,t}(\alpha) \\
&= \sum_{\alpha \in \{ \beta \in \text{frags}^*(\mathcal{A}) \mid \beta = q \in S \}} c_{\sigma,t}(\alpha) + \sum_{\alpha \in \{ \beta br \mid \beta \in \text{frags}^*(\mathcal{A}) \}} c_{\sigma,t}(\alpha) \\
&= 0 + \sum_{abr \in \{ \beta br \mid \beta \in \text{frags}^*(\mathcal{A}) \}} c_{\sigma,t}(abr) \\
&= \sum_{abr \in \{ \beta br \mid \beta \in \text{frags}^*(\mathcal{A}) \}} \mu_{\sigma,t}(C_{\alpha}) \cdot \sum_{tr \in D(b)} c(tr) \cdot \sigma(\alpha)(tr) \cdot \mu_{tr}(r) \\
&= \sum_{\alpha \in \text{frags}^*(\mathcal{A})} \sum_{b \in \Sigma} \sum_{r \in S} \mu_{\sigma,t}(C_{\alpha}) \cdot \sum_{tr \in D(b)} c(tr) \cdot \sigma(\alpha)(tr) \cdot \mu_{tr}(r)
\end{aligned}$$

$$\begin{aligned}
&= \sum_{\alpha \in \text{frags}^*(\mathcal{A})} \sum_{b \in \Sigma} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(b)} c(tr) \cdot \sigma(\alpha)(tr) \cdot \sum_{r \in S} \mu_{tr}(r) \\
&= \sum_{\alpha \in \text{frags}^*(\mathcal{A})} \sum_{b \in \Sigma} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(b)} c(tr) \cdot \sigma(\alpha)(tr) \\
&= \sum_{\alpha \in F(\tau)} \sum_{b \in \{a,\tau\}} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(b)} c(tr) \cdot \sigma(\alpha)(tr) \\
&\quad + \sum_{\alpha \in F(a)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(\tau)} c(tr) \cdot \sigma(\alpha)(tr) \\
&= \sum_{\alpha \in F(\tau)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(\tau)} c(tr) \cdot \sigma(\alpha)(tr) \\
&\quad + \sum_{\alpha \in F(\tau)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(a)} c(tr) \cdot \sigma(\alpha)(tr) \\
&\quad + \sum_{\alpha \in F(a)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(\tau)} c(tr) \cdot \sigma(\alpha)(tr) \\
&= \sum_{v \in S} \sum_{\alpha \in F(\tau,v)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(\tau)} c(tr) \cdot \sigma(\alpha)(tr) \\
&\quad + \sum_{v \in S} \sum_{\alpha \in F(\tau,v)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(a)} c(tr) \cdot \sigma(\alpha)(tr) \\
&\quad + \sum_{v \in S} \sum_{\alpha \in F(a,v)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr \in D(\tau)} c(tr) \cdot \sigma(\alpha)(tr) \\
&= \sum_{v \in S} \sum_{\alpha \in F(\tau,v)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr=v \xrightarrow{\tau} \mu \in D(\tau)} c(tr) \cdot \sigma(\alpha)(tr) \\
&\quad + \sum_{v \in S} \sum_{\alpha \in F(\tau,v)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr=v \xrightarrow{a} \mu \in D(a)} c(tr) \cdot \sigma(\alpha)(tr) \\
&\quad + \sum_{v \in S} \sum_{\alpha \in F(a,v)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr=v \xrightarrow{\tau} \mu \in D(\tau)} c(tr) \cdot \sigma(\alpha)(tr) \\
&= \sum_{v \in S} \sum_{\alpha \in F(\tau,v)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr=v \xrightarrow{\tau} \mu \in D} c(tr) \cdot \sigma(\alpha)(tr) \\
&\quad + \sum_{v \in S} \sum_{\alpha \in F(\tau,v)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr=v \xrightarrow{a} \mu \in D} c(tr) \cdot \sigma(\alpha)(tr) \\
&\quad + \sum_{v \in S} \sum_{\alpha \in F(a,v)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr=v \xrightarrow{\tau} \mu \in D} c(tr) \cdot \sigma(\alpha)(tr) \\
&= \sum_{v \in S} \sum_{\alpha \in F(\tau,v)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr=v \xrightarrow{\tau} \mu \in D} c(tr) \cdot \frac{f_{v,v}^{\sigma}{}^{tr}}{f_v^{\sigma}} \\
&\quad + \sum_{v \in S} \sum_{\alpha \in F(\tau,v)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr=v \xrightarrow{a} \mu \in D} c(tr) \cdot \frac{f_{v,v}^{\sigma}{}^{tr}}{f_v^{\sigma}}
\end{aligned}$$

$$\begin{aligned}
& + \sum_{v \in S} \sum_{\alpha \in F(a,v)} \mu_{\sigma,t}(C_\alpha) \cdot \sum_{tr=v \xrightarrow{\tau} \mu \in D} c(tr) \cdot \frac{f_{v_a, v_a^{tr}}^o}{\vec{f}_{v_a}^o} \\
= & \sum_{v \in S} \sum_{\alpha \in F(\tau,v)} \sum_{tr=v \xrightarrow{\tau} \mu \in D} c(tr) \cdot \frac{f_{v, v^{tr}}^o}{\vec{f}_v^o} \cdot \mu_{\sigma,t}(C_\alpha) \\
& + \sum_{v \in S} \sum_{\alpha \in F(\tau,v)} \sum_{tr=v \xrightarrow{a} \mu \in D} c(tr) \cdot \frac{f_{v, v_a^{tr}}^o}{\vec{f}_v^o} \cdot \mu_{\sigma,t}(C_\alpha) \\
& + \sum_{v \in S} \sum_{\alpha \in F(a,v)} \sum_{tr=v \xrightarrow{\tau} \mu \in D} c(tr) \cdot \frac{f_{v_a, v_a^{tr}}^o}{\vec{f}_{v_a}^o} \cdot \mu_{\sigma,t}(C_\alpha) \\
= & \sum_{v \in S} \sum_{tr=v \xrightarrow{\tau} \mu \in D} c(tr) \cdot \frac{f_{v, v^{tr}}^o}{\vec{f}_v^o} \cdot \sum_{\alpha \in F(\tau,v)} \mu_{\sigma,t}(C_\alpha) \\
& + \sum_{v \in S} \sum_{tr=v \xrightarrow{a} \mu \in D} c(tr) \cdot \frac{f_{v, v_a^{tr}}^o}{\vec{f}_v^o} \cdot \sum_{\alpha \in F(\tau,v)} \mu_{\sigma,t}(C_\alpha) \\
& + \sum_{v \in S} \sum_{tr=v \xrightarrow{\tau} \mu \in D} c(tr) \cdot \frac{f_{v_a, v_a^{tr}}^o}{\vec{f}_{v_a}^o} \cdot \sum_{\alpha \in F(a,v)} \mu_{\sigma,t}(C_\alpha) \\
= & \sum_{v \in S} \sum_{tr=v \xrightarrow{\tau} \mu \in D} c(tr) \cdot \frac{f_{v, v^{tr}}^o}{\vec{f}_v^o} \cdot \vec{f}_v^o + \sum_{v \in S} \sum_{tr=v \xrightarrow{a} \mu \in D} c(tr) \cdot \frac{f_{v, v_a^{tr}}^o}{\vec{f}_v^o} \cdot \vec{f}_v^o \\
& + \sum_{v \in S} \sum_{tr=v \xrightarrow{\tau} \mu \in D} c(tr) \cdot \frac{f_{v_a, v_a^{tr}}^o}{\vec{f}_{v_a}^o} \cdot \vec{f}_{v_a}^o
\end{aligned}$$

by [15, Long version, Corollary 2]

$$\begin{aligned}
= & \sum_{v \in S} \sum_{tr=v \xrightarrow{\tau} \mu \in D} c(tr) \cdot f_{v, v^{tr}}^o + \sum_{v \in S} \sum_{tr=v \xrightarrow{a} \mu \in D} c(tr) \cdot f_{v, v_a^{tr}}^o \\
& + \sum_{v \in S} \sum_{tr=v \xrightarrow{\tau} \mu \in D} c(tr) \cdot f_{v_a, v_a^{tr}}^o \\
= & \sum_{tr=v \xrightarrow{\tau} \mu \in D} c(tr) \cdot f_{v, v^{tr}}^o + \sum_{tr=v \xrightarrow{a} \mu \in D} c(tr) \cdot f_{v, v_a^{tr}}^o + \sum_{tr=v \xrightarrow{\tau} \mu \in D} c(tr) \cdot f_{v_a, v_a^{tr}}^o \\
= & \sum_{tr=v \xrightarrow{\tau} \mu \in D} c_f((v, v^{tr})) \cdot f_{v, v^{tr}}^o + \sum_{tr=v \xrightarrow{a} \mu \in D} c_f((v, v_a^{tr})) \cdot f_{v, v_a^{tr}}^o \\
& + \sum_{tr=v \xrightarrow{\tau} \mu \in D} c_f((v_a, v_a^{tr})) \cdot f_{v_a, v_a^{tr}}^o + 0 \\
= & \sum_{(v, v^{tr}) \in E_\tau} c_f((v, v^{tr})) \cdot f_{v, v^{tr}}^o + \sum_{(v, v_a^{tr}) \in E^a} c_f((v, v_a^{tr})) \cdot f_{v, v_a^{tr}}^o
\end{aligned}$$

$$\begin{aligned}
& + \sum_{(v_a, v_a^{tr}) \in E_\tau^a} c_f((v_a, v_a^{tr})) \cdot f_{v_a, v_a^{tr}}^o + \sum_{(x, y) \in E \setminus (E_\tau \cup E^a \cup E_\tau^a)} 0 \cdot f_{x, y}^o \\
= & \sum_{(v, v^{tr}) \in E_\tau} c_f((v, v^{tr})) \cdot f_{v, v^{tr}}^o + \sum_{(v, v_a^{tr}) \in E^a} c_f((v, v_a^{tr})) \cdot f_{v, v_a^{tr}}^o \\
& + \sum_{(v_a, v_a^{tr}) \in E_\tau^a} c_f((v_a, v_a^{tr})) \cdot f_{v_a, v_a^{tr}}^o + \sum_{(x, y) \in E \setminus (E_\tau \cup E^a \cup E_\tau^a)} c_f((x, y)) \cdot f_{x, y}^o \\
= & \sum_{(x, y) \in E} c_f((x, y)) \cdot f_{x, y}^o \\
= & \mathfrak{C}.
\end{aligned}$$

This completes the proof that if $\min_c t \xrightarrow{a}_C \mathcal{L}(\mathcal{R})\mu$ has an optimal solution f^o , then there exists a scheduler σ for \mathcal{A} that induces $t \xrightarrow{a}_C \mu_t$ such that $\mu \mathcal{L}(\mathcal{R}) \mu_t$ and $c(t \xrightarrow{a}_C \mu_t) = \mathfrak{C}$, and the proof of the theorem. \square

REFERENCES

- [1] Guy Avni and Orna Kupferman. Making weighted containment feasible: A heuristic based on simulation and abstraction. In *CONCUR*, volume 7454 of *LNCS*, pages 84–99, 2012.
- [2] Christel Baier, Bettina Engelen, and Mila Majster-Cederbaum. Deciding bisimilarity and similarity for probabilistic processes. *J. Computer and Systems Science*, 60(1):187–231, 2000.
- [3] Richard Bellman. A Markovian decision process. *Indiana University Mathematics Journal*, 6:679–684, 1957.
- [4] Patricia Bouyer, Uli Fahrenberg, Kim Guldstrand Larsen, Nicolas Markey, and Jiří Srba. Infinite runs in weighted timed automata with energy constraints. In *FORMATS*, volume 5215 of *LNCS*, pages 33–47, 2008.
- [5] Stefano Cattani and Roberto Segala. Decision algorithms for probabilistic bisimulation. In *CONCUR*, volume 2421 of *LNCS*, pages 371–385, 2002.
- [6] Krishnendu Chatterjee, Rupak Majumdar, and Thomas A. Henzinger. Markov decision processes with multiple objectives. In *STACS*, volume 3884 of *LNCS*, pages 325–336, 2006.
- [7] Ghassan Chehaibar, Hubert Garavel, Laurent Mounier, Nadia Tawbi, and Ferruccio Zulian. Specification and verification of the PowerScale[®] bus arbitration protocol: An industrial experiment with LOTOS. In *FORTE*, pages 435–450, 1996.
- [8] Nicolas Coste, Hubert Garavel, Holger Hermanns, Frédéric Lang, Radu Mateescu, and Wendelin Serwe. Ten years of performance evaluation for concurrent systems using CADP. In *ISoLa*, volume 6416 of *LNCS*, pages 128–142, 2010.
- [9] Yuxin Deng. *Axiomatisations and Types for Probabilistic and Mobile Processes*. PhD thesis, École des Mines de Paris, 2005.
- [10] Christian Eisentraut, Holger Hermanns, Johann Schuster, Andrea Turrini, and Lijun Zhang. The quest for minimal quotients for probabilistic automata. In *TACAS*, volume 7795 of *LNCS*, pages 16–31, 2013.
- [11] Kousha Etesami, Marta Kwiatkowska, Moshe Y. Vardi, and Mihalis Yannakakis. Multi-objective model checking of Markov decision processes. *Logical Methods in Computer Science*, 4(8):1–21, 2008.
- [12] Wan Fokkink, Jun Pang, and Jaco van de Pol. Cones and foci: A mechanical framework for protocol verification. *Formal Methods in System Design*, 29(1):1–31, 2006.
- [13] Robert Givan, Thomas Dean, and Matthew Greig. Equivalence notions and model minimization in Markov decision processes. *Artificial Intelligence*, 147(1-2):163–223, 2003.
- [14] Holger Hermanns and Joost-Pieter Katoen. Automated compositional Markov chain generation for a plain-old telephone system. *Science of Computer Programming*, 36(1):97–127, 2000.
- [15] Holger Hermanns and Andrea Turrini. Deciding probabilistic automata weak bisimulation in polynomial time. In *FSTTCS*, pages 435–447, 2012.
- [16] Holger Hermanns and Andrea Turrini. Cost preserving bisimulations for probabilistic automata. In *CONCUR*, volume 8052 of *LNCS*, pages 349–363, 2013.

- [17] Andrew Hinton, Marta Kwiatkowska, Gethin Norman, and David Parker. PRISM: A tool for automatic verification of probabilistic systems. In *TACAS*, volume 3920 of *LNCS*, pages 441–444, 2006.
- [18] Ronald A. Howard. *Dynamic Probabilistic Systems, Volume II: Semi-Markov and Decision Processes*. Dover Publications, 2007.
- [19] Bengt Jonsson and Kim Guldstrand Larsen. Specification and refinement of probabilistic processes. In *LICS*, pages 266–277, 1991.
- [20] Joost-Pieter Katoen, Tim Kemna, Ivan S. Zapreev, and David N. Jansen. Bisimulation minimisation mostly speeds up probabilistic model checking. In *TACAS*, volume 4424 of *LNCS*, pages 76–92, 2007.
- [21] Joost-Pieter Katoen, Ivan S. Zapreev, Ernst Moritz Hahn, Holger Hermanns, and David N. Jansen. The ins and outs of the probabilistic model checker MRMC. *Performance Evaluation*, 68(2):90–104, 2011.
- [22] Nancy A. Lynch, Roberto Segala, and Frits W. Vaandrager. Observing branching structure through probabilistic contexts. *SIAM J. on Computing*, 37(4):977–1013, 2007.
- [23] Robin Milner. *Communication and Concurrency*. Prentice-Hall International, Englewood Cliffs, 1989.
- [24] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Number 594 in Wiley Series in Probability and Statistics. John Wiley & Sons, Inc., 2005.
- [25] Karin Quaas. Wighted timed MSO logics. In *Developments in Language Theory*, volume 5583 of *LNCS*, pages 419–430, 2009.
- [26] Roberto Segala. *Modeling and Verification of Randomized Distributed Real-Time Systems*. PhD thesis, MIT, 1995.
- [27] Roberto Segala. Probability and nondeterminism in operational models of concurrency. In *CONCUR*, volume 4137 of *LNCS*, pages 64–78, 2006.
- [28] Ron J. van Glabbeek and W. Peter Weijland. Branching time and abstraction in bisimulation semantics. *J. ACM*, 43(3):555–600, 1996.
- [29] Pavol Černý, Thomas A. Henzinger, and Arjun Radhakrishna. Simulation distances. *TCS*, 413(1):21–35, 2012.
- [30] Ralf Wimmer, Marc Herbstritt, Holger Hermanns, Kelley Strampp, and Bernd Becker. Sigref - a symbolic bisimulation tool box. In *ATVA*, volume 4218 of *LNCS*, pages 477–492, 2006.
- [31] Lijun Zhang. *Decision Algorithm for Probabilistic Simulations*. PhD thesis, Saarland University, 2008.