The present work was submitted to the LuFG Theory of Hybrid Systems

Bachelor of Science Thesis

Comparing HyperPCTL, HyperPCTL*, and PHL

Georgios Arapatsakos

Communicated by
Prof. Dr. Erika Ábrahám

Examiners:
Prof. Dr. Erika Ábrahám
Apl. Prof. Dr. Christof Löding

Additional Advisor:
Lina Gerlach

Aachen, 28th August 2024

Abstract

Probabilistic hyperproperties are specifications that describe the form of the traces of executions of *probabilistic systems* that compare *multiple traces* simultaneously and with one another, and make statements concerning the probabilities of certain events happening across these traces. In this thesis, we explore the expressive power of three *probabilistic hyperlogics* – that is, logics that can formulate properties matching this description – in relation to one another: HyperPCTL, HyperPCTL*, and PHL, on (discrete-time) Markov Chains, and Markov Decision Processes. The focus is primarily on the relation of HyperPCTL and HyperPCTL* to PHL on Markov Chains.

Contents

Chapter 1

Introduction

Discrete-Time Markov Chains (DTMCs) are mathematical objects, not dissimilar to Kripke structures, or even some forms of automata, which can be used to model *probabilistic processes*. *Markov Decision Processes* (MDPs) combine this concept with layers of *nondeterministic choice*. The formalism for resolving the choices lies within *schedulers*, which, intuitively, replace each layer of choices with a probability distribution, while also being allowed to reference past choices during the selection of the distribution itself. Both DTMCs and MDPs have *atomically labelled* states.

The *trace* of a path on a *labelled transition system* is the ordered collection of labels that appear on it. A *trace property* is, hence, simply a specification of how a trace should look like. Alternatively, a trace property is some collection of *acceptable* traces.

We have well-established logics that can be used to express *trace properties* in labelled transition systems, with a very important example in the context of this thesis being *computational tree logic* (CTL) [CE82]. CTL makes a strict distinction between *state* and *path* formulae. The former are those that specify either branching behaviour, or make local assertions referencing labels, on certain states. The latter express *temporal modalities* along paths. That is, they express how properties change along them – or, in other words in which order they appear. CTL requires alternation of path and state formulae in nesting, and CTL* [EH86] is an expansion of it that lifts this very restriction.

PCTL, first seen in [HJ89], and PCTL*, first seen in [Azi+95], have been formulated to make assertions involving *probabilistic trace properties* in DTMCs and MDPs. These exchanged the quantifiers of their non-probabilistic counterparts for probabilistic operators of the form $\mathbb{P}_{\leq c}(\cdot)$ that can measure sets of paths fulfilling the specifications laid out in their arguments, and compare these measures with constants ("$\leq c$" in the example above).

Hyperproperties are properties that can compare *multiple* traces at once, and with one another. For the probabilistic versions of these, the logics mentioned have been reformed into HyperPCTL [ÁB18] and HyperPCTL* [Wan+21], in the former by adding quantification over states, and in the latter by making probabilistic operators draw explicit paths that can then be referenced in nested expressions. Both logics can also compare measures to one another, that is, they are not limited to comparison with

constants, as was the case with PCTL and PCTL*. HyperPCTL has further seen an extension to MDPs in [Ábr+20] by way of quantification over schedulers.

A completely distinct approach at solving the same problem is found in PHL [HJ89], which only has a formulation for MDPs, and deals with probabilistic properties completely separately from (*non*-probabilistic) hyperproperties. Essentially, it delegates the former to measures of scheduler-marked LTL [Pnu77], and the latter to HyperCTL* [Cla+14], with the extra step of binding path quantification to schedulers. A downscaled version of PHL for DTMCs will be proposed later on.

In contrast to the logics that laid their foundations, the relationship between HyperPCTL, HyperPCTL*, and PHL in terms of expressive power on DTMCs, as well as the one between HyperPCTL and PHL on MDPs, is currently largely unknown. This thesis aims to explore exactly this relationship. To this end, apart from proposing a downscaled version of PHL for DTMCs, all of the logics have been reformulated to use consistent notation and comparable mathematical abstractions in their syntax and semantics across the board. In particular, HyperPCTL* had to be changed relatively drastically due to our perceived ambiguities in its formulation in [Wan+21]. The reasoning behind this and the changes themselves, as well as a verbatim copy of the logic is provided in Appendix A.

We mostly focus on HyperPCTL vs. PHL, and HyperPCTL* vs. PHL, on DTMCs due to the following reasons.

- HyperPCTL* does not yet have a formulation for MDPs, and upscaling it is non-trivial.
- The relation between HyperPCTL and HyperPCTL* on DTMCs is explained (albeit not exhaustively proven) in [Wan+21].
- Exploring the above in detail would not have fit the time frame for this thesis.

The structure of the thesis as follows. Chapter 2 goes over the required measure-theoretical background, lays out notational conventions, and defines the models and the logics concretely. Furthermore, it proposes a downscaled version of PHL for DTMCs. In Chapter 3, definitions for relations (e.g. implication, subsumption, equivalence) between formulae and fragments across all logics are provided, and inherent differences between the logics, and how they work, are pointed out. Chapters 4 and 5 examine the relation between HyperPCTL* and PHL, and HyperPCTL and PHL, respectively. This is done by embedding fragments of one into the other. In Chapter 6, we briefly look at what relations between HyperPCTL and HyperPCTL* can be extrapolated from the results of previous chapters. Finally, Chapter 7 deals with the question of whether the results of Chapter 5 scale upwards for MDPs. Chapter 8 reiterates on the results and open questions of all chapters, and concludes the thesis.

Chapter 2

Preliminaries

Let $\wp(\cdot)$ denote the powerset operator and ω the first limit ordinal. For the purposes of this thesis, we consider ω equivalent to the set of the finite ordinals, the *natural numbers* (von-Neumann-construction). As such, the expressions $n < \omega$ and $n \in \omega$ are to be interpreted the same. Furthermore, $\mathbb{R}$ denotes the *set of the real numbers*, $\mathbb{Q}$ the *set of rational numbers*, and we set $\mathbb{R}^\infty := \mathbb{R} \cup \{-\infty, \infty\}$.

Let A be an ordered set, and $a, b \in A$. We define the *closed interval* $[a, b]$ *in* A as

$$[a, b]_A := \{c \in A \mid a \leq c \leq b\}.$$

Similarly, we also define the *open interval* (a, b) *in* A by

$$(a, b)_A := \{c \in A \mid a < c < b\}.$$

If the subscript A is omitted, it is assumed to be ω. As such $[a, b]$ denotes the *discrete interval* $\{a, a+1, \ldots, b-1, b\}$. In any case, $[a, b]_A = \varnothing$, if $a > b$, and $(a, b)_A = \varnothing$, if $a \geq b$ (in case $A = \omega$, even $a \geq b - 1$ suffices).

Let $\overline{u}$ be a sequence of length $|\overline{u}| := n \leq \omega$. The expressions u_i, and $\overline{u}(i)$, $i < n$, denote the i-th member of $\overline{u}$. Expressions of the form $u \mapsto v$ are used to represent the tuples (u, v), where it contributes to readability (e.g. assignments).

Finally, let A, B, C be sets, $\overline{u} := \big((a_i, b_i)\big)_{i<n} \in (A \times B)^n$, $n \leq \omega$ a sequence of tuples, and $\overline{v} := (c_i)_{i<n} \in C^n$ a sequence in C. $\overline{u}[\overline{v}]$ denotes the sequence $\big((a_i, c_i)\big)_{i<n}$ that results by replacing each second element of the tuples in $\overline{u}$ by the corresponding element of $\overline{v}$.

2.1 Probability Spaces

Before we start, it is essential to go over a few fundamental definitions regarding probability and measure theory. The basic building blocks we will need to use are σ-algebras, which, intuitively, represent a formal expression of measurable events in the form of collections of sets. Due to this connotation, the elements of a σ-algebra are often called events. The definitions that follow are adapted from Chapter 1 of [Bog07].

Definition 2.1 (σ-algebra). Let $\Omega \neq \varnothing$ be a space. A collection of sets $\mathfrak{A} \subset \wp(\Omega)$ is called *σ-algebra over* Ω if the following conditions are met.

(i) $\Omega \in \mathfrak{A}$.

(ii) $A \in \mathfrak{A} \Rightarrow \Omega \setminus A \in \mathfrak{A}$. (*stability under complements*)

(iii) If I is a countable set of indices, and $(A_i)_{i\in I} \subset \mathfrak{A}$ a sequence of events in $\mathfrak{A}$, then $\bigcup_{i\in I} A_i \in \mathfrak{A}$. ($\sigma$-$\cup$ *stability*)

△

The tuple $(\Omega, \mathfrak{A})$ of a space Ω and a σ-algebra $\mathfrak{A}$ over Ω is called a *sample space*. The entire collection $\wp(\Omega)$ is in itself the largest σ-algebra over Ω, and the smallest possible one is $\{\Omega, \varnothing\}$. Any other σ-algebra lies between these two.

It is often required to define measures on an arbitrary collection $\mathfrak{E}$ of subsets of Ω. To this end, the intersection of all *expansions* of $\mathfrak{E}$ with other subsets of Ω that are σ-algebras is taken. The result is a *unique* σ-algebra that contains the entire collection $\mathfrak{E}$.

Definition 2.2 (σ-algebra generated by a collection). Let $\Omega \neq \varnothing$ be a space, and $\mathfrak{E} \subset \wp(\Omega)$ a collection of sets. The smallest σ-algebra containing $\mathfrak{E}$ is called the *σ-algebra generated by* $\mathfrak{E}$ and defined by $\mathfrak{A}(\mathfrak{E}) := \bigcap_{\mathfrak{A}\in\mathfrak{F}} \mathfrak{A}$, where $\mathfrak{F} = \left\{\mathfrak{A} \subset \wp(\Omega) \;\middle|\; \mathfrak{E} \subset \mathfrak{A} \wedge \mathfrak{A} \text{ is } \sigma\text{-algebra over } \Omega\right\}$. △

The well-definedness and existence of $\mathfrak{A}(\mathfrak{E})$ from Definition 2.2 falls into the domain of measure theory [Bog07, Prop. 1.2.6]. With these definitions, we can introduce the notion of a probability measure.

Definition 2.3 (Probability measure). Let $\Omega \neq \varnothing$ be a space, $\mathfrak{A} \subset \wp(\Omega)$ a σ-algebra over Ω, and $\mu : \mathfrak{A} \to [0, \infty]_{\mathbb{R}^\infty}$. μ is called a *probability measure on* $(\Omega, \mathfrak{A})$, if

(i) $\mu(\varnothing) = 0$,

(ii) $\mu(\Omega) = 1$[i], and,

(iii) if I is a countable set of indices, and $(A_i)_{i\in I} \subset \mathfrak{A}$ a sequence of *pairwise disjunct events* in $\mathfrak{A}$, then $\mu\left(\bigcup_{i\in I} A_i\right) = \sum_{i\in I} \mu(A_i)$. ($\sigma$-*additivity*)

[i] Without this condition, μ is called simply a *measure*.

△

Putting everything together, we get

Definition 2.4 (Probability space). A *probability space* is a triple $(\Omega, \mathfrak{A}, \mu)$ where

- Ω is a nonempty space,
- $\mathfrak{A} \subset \wp(\Omega)$ is a σ-algebra over Ω, and
- $\mu : \mathfrak{A} \to [0, \infty]_{\mathbb{R}^\infty}$ is a probability measure on $(\Omega, \mathfrak{A})$. △

An adjacent term that we will need to use is the following.

Definition 2.5 (Probability distribution). Let A be a discrete set. A function $p : A \to [0,1]_{\mathbb{R}^\infty}$ is called a *(discrete) probability distribution over A*, if $\big(A, \wp(A), \mu_p\big)$ is a *probability space*, where μ_p is the *measure induced by p* and defined by

$$\mu_p(B) := \sum_{b \in B} p(b), \quad \text{for } B \subset A$$

△

We can already put the preceding definitions to use and prove a useful relation that connects the probability measure of a set with the one of its complement.

Lemma 2.6 (Dual events). Let $(\Omega, \mathfrak{A}, \mu)$ be a probability space. Then

$$\mu(\Omega \setminus A) = 1 - \mu(A), \quad \text{for all } A \in \mathfrak{A}.$$

Proof. Let $A \in \mathfrak{A}$. By Definition 2.1(ii), $\Omega \setminus A \in \mathfrak{A}$ and we get

$$\begin{aligned}
\mu(\Omega \setminus A) &= \mu(\Omega \setminus A) + \underbrace{\mu(A) - \mu(A)}_{=0} \\
&= \Big(\mu(\Omega \setminus A) + \mu(A)\Big) - \mu(A) \\
&= \mu\Big(\underbrace{(\Omega \setminus A) \cup A}_{(\Omega \setminus A) \cap A = \varnothing}\Big) - \mu(A) && \big(\text{Def. 2.3(iii)}\big) \\
&= \mu(\Omega) - \mu(A) \\
&= 1 - \mu(A). && \big(\text{Def. 2.3(ii)}\big)
\end{aligned}$$

□

2.2 Models

In this section, we shall introduce the abstract mathematical models, upon the foundation of which we will base the definitions of the logics later on.

2.2.1 Markov Chains

Discrete-Time Markov Chains, or DTMCs, are mathematical objects that can be used to model probabilistic processes. They are, intuitively, labelled transition systems, in which the transitions themselves happen probabilistically.

Definition 2.7 (Markov Chain). A *(discrete-time) Markov Chain* (DTMC) is a tuple $\mathcal{D} = (S, p, \mathsf{AP}, l)$, where

- S is a countable, nonempty *set of states*,
- $p : S \times S \to [0,1]_{\mathbb{R}}$ is a *transition probability function*, such that

$$\sum_{t \in S} p(s,t) = 1, \quad \text{for all } s \in S,$$

- AP is a *set of atomic propositions*, and
- $l : S \to \wp(\mathsf{AP})$ is a *labelling function*. △

We call a DTMC finite iff S is finite. Note that some definitions also include an *initial probability distribution* of the form $\iota : S \to [0,1]$, with $\sum_{s \in S} \iota(s) = 1$, or may specify a *unique initial state* s_ι. In the sequel, if either of these is being used, it will be explicitly noted.

Definition 2.8 (n-ary parallel composition of Markov Chains). Let $n < \omega$, and $\mathcal{D}_i = (S_i, p_i, \mathsf{AP}_i, l_i)$, for $i < n$, be a sequence of DTMCs. The *n-ary parallel composition* over this sequence is defined as the DTMC $\bigtimes_{i<n} \mathcal{D}_i = (S, p, \mathsf{AP}, l)$, with

- $S := \bigtimes_{i<n} S_i$,
- $p(\bar{s}, \bar{t}) := \prod_{i<n} p(s_i, t_i)$,
- $\mathsf{AP} := \bigtimes_{i<n} \mathsf{AP}_i$, and
- $l(\bar{s}) := \Big(l_1(s_1), \ldots, l_n(s_n)\Big)$,

with $\bar{s}, \bar{t} \in S$. △

Note that $l(\bar{s}) \in \bigtimes_{1 \le i \le n} \wp(\mathsf{AP}_i)$, instead of $l(\bar{s}) \in \wp(\mathsf{AP}) = \wp(\bigtimes_{1 \le i \le n} \mathsf{AP}_i)$, which is what one would expect after Definition 2.7. This notational choice was made for purely stylistic reasons, as it aids readability and allows us to easily define legible notation of the form "$l(\bar{s})(i)$" to get the label of the i-th component of $\bar{s}$.

Based on Definition 2.8, we call the n-ary parallel composition of a DTMC $\mathcal{D}$ with *itself* the *n-ary self-composition of $\mathcal{D}$* and denote it by $\mathcal{D}^n$.

2.2.2 Markov Decision Processes

An "expansion" of the DTMC model can be found in Markov Decision Processes, or MDPs for short. These add a layer of nondeterminism, represented by *actions*, in-between state transitions.

Definition 2.9 (Markov Decision Process). A *Markov Decision Process* is a tuple $\mathcal{M} = (S, \mathsf{Act}, p, \mathsf{AP}, l)$, where

- S is a finite, nonempty *set of states*,
- Act is a nonempty *set of actions*,
- $p : S \times \mathsf{Act} \times S \to [0,1]_{\mathbb{R}}$ is a *transition probability function*, such that
  - the partially applied function $p_{s,\alpha}$, defined by $p_{s,\alpha}(t) := p(s, \alpha, t)$, is either a probability distribution over S, or identical to 0, for all $(s, \alpha) \in S \times \mathsf{Act}$, and
  - for each $s \in S$ there is at least one $\alpha \in \mathsf{Act}$ such that $p_{s,\alpha} \not\equiv 0$.
- AP is a *set of atomic propositions*, and
- $l : S \to \wp(\mathsf{AP})$ is a *labelling function*. △

If $p_{s,\alpha} \not\equiv 0$, we call $\alpha \in \mathsf{Act}$ *enabled* at $s \in S$, and denote with $\mathsf{Act}(s)$ the set of all such actions. We will now look at a way to remove nondeterminism from MDPs, namely schedulers[i]. Intuitively, these replace each nondeterministic layer of actions with a probability distribution over these actions. Since each choice made by a scheduler can depend on its previous ones, the DTMC that results may be infinite.

Similarly to DTMCs, MDPs might have an initial state s_ι, or an initial distribution $\iota : S \to [0,1]$ with $\sum_{s \in S} \iota(s) = 1$. If either of these is being used, it will be noted explicitly.

Definition 2.10 (Scheduler). Given an MDP $\mathcal{M} = (S, \mathsf{Act}, p, \mathsf{AP}, l)$, a *scheduler for* $\mathcal{M}$ is a tuple $\sigma = (Q, \mathsf{act}, \mathsf{mode}, \mathsf{init})$, where

- Q is a countable *set of modes*,
- $\mathsf{act} : Q \times S \times \mathsf{Act} \to [0,1]_{\mathbb{R}}$ is a function such that its partial application $\mathsf{act}_{q,s}$, defined by $\mathsf{act}_{q,s}(\alpha) := \mathsf{act}(q, s, \alpha)$, is a probability distribution over $\mathsf{Act}(s)$, for all $(q, s) \in Q \times S$,
- $\mathsf{mode} : Q \times S \to Q$ is a *mode transition function*, and
- $\mathsf{init} : S \to Q$ is an *initial mode function*. △

[i] In some stochastics literature also referred to as *policies*.

We denote the set of all schedulers for an MDP $\mathcal{M}$ as $\Sigma_{\mathcal{M}}$. An alternative – but still equivalent – definition for a scheduler is found in [DFT20], in which it is formulated as a function that assigns probability distributions over Act to *histories*, that is, sequences of states and actions of the form $s_0\alpha_0 ... \alpha_{n-1}s_n \in (S \cdot \mathrm{Act})^* S$. Within Definition 2.10, the entire history space of an MDP can be encoded into different sequences of modes and transitions in the mode space and transition function respectively, as it is countable.

As it was alluded to before, a pair $\mathcal{M}, \sigma$ induces a DTMC.

Definition 2.11. Let $\mathcal{M} = (S, \mathrm{Act}, p, \mathrm{AP}, l)$ be an MDP, and $\sigma \in \Sigma_{\mathcal{M}}$. The DTMC induced by $\mathcal{M}$ with σ is $\mathcal{M}^\sigma := (S^\sigma, p^\sigma, \mathrm{AP}, l^\sigma)$, where

- $S^\sigma := Q \times S$,
- $p^\sigma\big((q,s),(r,t)\big) := \begin{cases} \sum_{\alpha \in \mathrm{Act}(s)} \mathrm{act}(q,s,\alpha) \cdot p(s,\alpha,t), & \text{if } r = \mathrm{mode}(q,s), \\ 0, & \text{otherwise, and} \end{cases}$
- $l^\sigma(q,s) := l(s)$,

with $q, r \in Q$ and $s, t \in S$. △

Definition 2.12 (n-ary parallel composition of Markov Decision Processes). Let $n < \omega$, and $\mathcal{M}_i = (S_i, \mathrm{Act}_i, p_i, \mathrm{AP}_i, l_i)$, for $i < n$, be a sequence of MDPs. The *n-ary parallel composition* over this sequence is defined as the MDP $\bigtimes_{i<n} \mathcal{M}_i = (S, \mathrm{Act}, p, \mathrm{AP}, l)$, with

- $S := \bigtimes_{i<n} S_i$,
- $\mathrm{Act} := \bigtimes_{i<n} \mathrm{Act}_i$,
- $p(\overline{s}, \overline{\alpha}, \overline{t}) := \prod_{i<n} p_i(s_i, \alpha_i, t_i)$,
- $\mathrm{AP} := \bigtimes_{i<n} \mathrm{AP}_i$, and
- $l(\overline{s}) = \Big(l_1(s_1), ..., l_n(s_n)\Big)$,

with $\overline{s}, \overline{t} \in S$, and $\overline{\alpha} \in \mathrm{Act}$. △

Again, we made the notational choice to have $l(\overline{s}) \in \bigtimes_{1 \le i \le n} \wp(\mathrm{AP}_i)$ instead of $l(\overline{s}) \in \wp(\bigtimes_{1 \le i \le n} \mathrm{AP}_i)$.

The n-ary *self*-composition of an MDP $\mathcal{M}$ is denoted by $\mathcal{M}^n$ and defined similarly to the DTMC case. For a sequence of schedulers $\overline{\sigma} \in \Sigma_{\mathcal{M}}^n$, $\mathcal{M}^{\overline{\sigma}}$ represents the DTMC $\bigtimes_{i<n} \mathcal{M}^{\sigma_i}$, that is the parallel composition of the DTMCs induced by the pairs $\mathcal{M}, \sigma_i$.

2.2.3 Paths & Reachability

Let $\mathcal{D} = (S, p, \mathrm{AP}, l)$ be a DTMC. A *path* π on $\mathcal{D}$ is an *infinite sequence* of states $\pi = (s_i)_{i<\omega} \in S^\omega$, such that $p(s_i, s_{i+1}) > 0$, for all $i < \omega$. We denote the *set of all paths* on $\mathcal{D}$ by $\mathrm{Paths}_\mathcal{D}$. For a $j < \omega$, the sequence $\pi' := (s'_i)_{i<j} \in S^j$ is called a *finite prefix* of π, written $\pi' \sqsubseteq \pi$, iff $s_i = s'_i$, for all $i < j$. We define the length of π' by $|\pi'| := j$. $\mathrm{Paths}^{<\omega}_\mathcal{D}$ represents the *set of all finite prefixes* of paths in $\mathrm{Paths}_\mathcal{D}$. Similarly, $\mathrm{Paths}_\mathcal{D}(s)$ and $\mathrm{Paths}^{<\omega}_\mathcal{D}(s)$ are defined as the sets of paths and finite path prefixes, respectively, that start at a fixed $s \in S$. A state $t \in S$ is *reachable* from $s \in S$ iff there exists a $\pi \in \mathrm{Paths}^{<\omega}_\mathcal{D}(s)$ that ends in t.

Now consider an MDP $\mathcal{M} = (S, \mathrm{Act}, p, \mathrm{AP}, l)$. A path on $\mathcal{M}$ is an infinite sequence of states $(s_i)_{i<\omega} \in S^\omega$, this time with the condition that for all $i < \omega$ there exists an action $\alpha \in \mathrm{Act}$ with $p(s_i, \alpha, s_{i+1}) > 0$. A consequence of this is, that, given $\sigma \in \Sigma_\mathcal{M}$, a path on the induced DTMC $\mathcal{M}^\sigma$ of the form $\big((q_i, s_i)\big)_{i<\omega} \in \mathrm{Paths}_{\mathcal{M}^\sigma}$ corresponds to the path $(s_i)_{i<\omega}$ on the original MDP that results by leaving out the modes q_i. The sets $\mathrm{Paths}_\mathcal{M}$, $\mathrm{Paths}^{<\omega}_\mathcal{M}$, $\mathrm{Paths}_\mathcal{M}(\cdot)$, and $\mathrm{Paths}^{<\omega}_\mathcal{M}(\cdot)$, as well as the relation $\sqsubseteq$ share the same semantics and definitions with their DTMC equivalents above.

For a DTMC or MDP $\mathcal{N}$, and one of its states s, we define

- $\mathrm{Post}_\mathcal{N}(s) := \{t \in S \mid \exists \pi \in \mathrm{Paths}_\mathcal{N}(s) :\ \pi(1) = t\}$, the set of all successors of s, and
- $\mathrm{Post}^*_\mathcal{N}(s) := \{t \in S \mid \exists \pi \in \mathrm{Paths}_\mathcal{N}(s) :\ \exists j < \omega :\ \pi(j) = t\}$, the set of all states reachable from s.

Furthermore, for $\pi \in \mathrm{Paths}_\mathcal{N}$, we denote by $\pi(i)$ the i-th element of π, and set $\pi^i := \big(\pi(i+j)\big)_{j<\omega}$ to denote the i-shift of π, that is the path that results by discarding the first i elements of π. Let $\pi(i)$ and π^i be defined the same for *finite path fragments* $\pi \in \mathrm{Paths}^{<\omega}_\mathcal{N}$, setting $\pi(i) := \bot$, and $\pi^i := \bot$ (undefined) if $i \geq |\pi|$.

2.2.4 Measurability of Events in Markov Chains

An important detail to the logics that will be discussed later on is the well-definedness of the interpretations of their probabilistic operators. To guarantee this, one needs to establish a connection between paths on DTMCs and probability spaces. This topic is covered in [BK08].

Definition 2.13 (Cylinder set). Let $\mathcal{D} = (S, p, \mathrm{AP}, l)$ be a Markov chain, and $\pi \in \mathrm{Paths}^{<\omega}_\mathcal{D}$. The *cylinder set* of π is $\mathrm{Cyl}_\mathcal{D}(\pi) := \left\{\pi' \in \mathrm{Paths}_\mathcal{D} \;\middle|\; \pi \sqsubseteq \pi'\right\}$. △

Cylinder sets are generally a concept founded in measure theory, and used as a basis to induce measures on infinite-dimensional product spaces [Bog07]. This is done by collecting the cylinder sets of all finite-dimensional subspaces; a finite path fragment

of length n can ultimately be viewed as an element of the space S^n, whereas a path is one of the space S^ω.

Definition 2.14 (σ-algebras of DTMCs). Given a pair $(\mathcal{D}, s)$ of a DTMC $\mathcal{D}$, and one of its states s, the σ-algebra associated with the pair, and denoted by $\mathfrak{A}(\mathcal{D}, s)$, is the σ-algebra generated by the set $\left\{ \mathsf{Cyl}_{\mathcal{D}}(\pi) \;\middle|\; \pi \in \mathsf{Paths}^{<\omega}_{\mathcal{D}}(s) \right\}$. △

For a DTMC $\mathcal{D} = (S, p, \mathsf{AP}, l)$, and $s \in S$ there exists a *unique probability measure* $\Pr_{\mathcal{D},s}$ on $\mathfrak{A}(\mathcal{D}, s)$, which yields the following probabilities for these cylinder sets:

$$\Pr_{\mathcal{D},s}\Big(\mathsf{Cyl}_{\mathcal{D}}(s_0 \ldots s_n)\Big) = [s = s_0] \cdot \prod_{i<n} p(s_i, s_{i+1}), \quad \text{for } s_0, \ldots, s_n \in S,$$

where $[s = s_0] := \begin{cases} 1, & \text{if } s = s_0, \\ 0, & \text{otherwise.} \end{cases}$

Finally, sets of paths of $\mathcal{D}$ starting at s can be measured as events on $\mathfrak{A}(\mathcal{D}, s)$ [BKo8]. The subscripts $\mathcal{D}$ and s in the probability measure and the cylinder set are usually omitted, given they can be inferred from context.

2.3 Logics

Let $\mathcal{L}$ be a quantified logic. We use, for reasons of brevity and clearness, the subscripts $\mathcal{L}_{\text{DTMC}}$ to refer to the formulation of $\mathcal{L}$ for DTMCs and $\mathcal{L}_{\text{MDP}}$ to refer the one for MDPs, respectively, and where applicable.

Consider a formula $\varphi \in \mathcal{L}$. We denote with $\mathsf{var}(\varphi)$ the set of all variables that appear in φ. A variable $v \in \mathsf{var}(\varphi)$ is called *free* if at least one instance of it is *not* bound by any quantifier $(\forall, \exists)$. Let $\mathsf{free}(\varphi)$ denote the set of all free variables in φ.

Similarly, v is called *bound*, if at least one instance of it is bound by a quantifier. φ is called *closed*, or a *sentence*, if it has no free variables, and *clean*, if no two quantifiers bind different instances of the same $v \in \mathsf{var}(\varphi)$ *and* no variable appears both bound and unbound in it.

Example 2.15. Consider the following first-order logic formula over a signature with a binary functional symbol f:

$$\varphi := (\exists x \exists z \forall y \; fxy = z) \wedge (\exists y \; fxy = y)$$

φ expresses that there is one x such that f becomes constant in its second argument, and that it also has a fixed point in its second argument (for a free first argument x).

We have $\mathrm{var}(\varphi) = \{x, y, z\}$. Out of these, x appears both bound and free, y is bound (twice), and z is bound. An equivalent *clean* formula is given by

$$\varphi' := (\exists v \exists z \forall y\, f v y = y) \wedge (\exists w\, f x w = w),$$

and by binding the free instance of x, for example in $\exists x \varphi$, or $\forall x \varphi'$, we get a *sentence*.△

As it was the case for CTL mentioned in Chapter 1, some of the logics we will see make clear distinctions between state formulae that make local assertions at states and specify branching behaviour, and path formluae (or path subexpressions) that express temporal modalities along paths. In some cases, this distinction is not present, and we simply have top-level formulae and certain categories of subexpressions.

In the sequel, state (or otherwise top-level) formulae will be denoted by φ and ψ, probabilistic formulae and expressions by ρ, path ones by ϑ and η, and in each case also variants such as φ', φ_i, etc. Lowercase hatted letters ($\hat{s}, \hat{\sigma}, \hat{\pi}, \ldots$) shall further be used for variables, normal letters ($s, \sigma, \pi, \ldots$) for concrete objects, and lowercase fraktur (blackletter) letters ($\mathfrak{p}, \mathfrak{r}, \mathfrak{s}, \ldots$) for variable-to-object assignments. In each case, the uppercase variants ($\hat{S}, \hat{\Sigma}, \hat{\Pi}, S, \Sigma, \Pi, \ldots$) will be used for the corresponding sets, and an overscore will be added ($\overline{s}, \overline{\sigma}, \overline{\pi}, \ldots$) for sequences. To avoid $\overline{\hat{s}}$, we use a tilde instead for sequences of variables ($\tilde{s}, \tilde{\sigma}, \tilde{\pi}, \ldots$).

In all of the logics, we will encounter the modal operators U (*until*) and $\bigcirc$ (*next*). Intuitively, for a path π and atomics a, b, $\pi \vDash a \,\mathsf{U}\, b$ holds iff we can reach a b-state on π while only crossing a-states, and $\pi \vDash \bigcirc a$ iff the state right after the current one is labelled a. Wherever the syntax allows it, we also implicitly define the following *syntactic sugar* exemplarily for atomic propositions.

(i) *Eventually* operator: $\Diamond a := \mathtt{true}\, \mathsf{U}\, a$.

(ii) *Globally* operator: $\Box a := \neg(\mathtt{true}\, \mathsf{U}\, \neg a)$.

(iii) *Implies* junctor: $a \to b := \neg a \vee b$.

The case for more complex formulae – again, where permitted by syntactical rules – is similar.

To avoid writing excessively many nested parentheses, let the binding strength of operations be as follows (strongest to weakest):

(i) Functional symbols: $c, f, g, \ldots$

(ii) Multiplication: $\cdot$

(iii) Addition: $+$

(iv) Comparisons: $<, \leq, =, \ldots$

(v) All unary operators: $\neg, \bigcirc, \Diamond, \Box, \ldots$

(vi) *Until* operator: U

(vii) *And, or* junctors: $\wedge, \vee$

(viii) *Implies* junctor: $\rightarrow$

To break (some) ties, we assume *right-associativity* for all binary operations. That is, for example

$$a \,\mathsf{U}\, b \,\mathsf{U}\, c \equiv a \,\mathsf{U}\, (b \,\mathsf{U}\, c)\,, \text{ and } \quad a \rightarrow b \rightarrow c \equiv a \rightarrow (b \rightarrow c).$$

Unary operators and functions are resolved innermost-to-outermost. For instance, if f is a binary and g a unary function, then, as a (very contrived) example, we have

$$\neg\Diamond\Box f\, g a \neg b \equiv \neg\Big(\Diamond\big(\Box\big(f(ga)(\neg b)\big)\big)\Big).$$

2.3.1 HyperPCTL

The versions of HyperPCTL that will be introduced in the following are based on [ÁB18] and [Ábr+20], but have been slightly adjusted to use variable assignments instead of syntactic replacements.

Markov Chains

Definition 2.16 ($\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ Syntax). $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ formulae are built according to the following grammar rules:

- (*state formulae*) $\varphi ::= \forall \hat{s}.\varphi \mid \exists \hat{s}.\varphi \mid \varphi \wedge \varphi \mid \neg\varphi \mid \mathtt{true} \mid a_{\hat{s}} \mid \rho < \rho$
- (*probabilistic formulae*) $\rho ::= \mathbb{P}(\vartheta) \mid \rho + \rho \mid \rho \cdot \rho \mid c$
- (*path formulae*) $\vartheta ::= \bigcirc\varphi \mid \varphi \,\mathsf{U}\, \varphi \mid \varphi \,\mathsf{U}^{[k_1,k_2]}\, \varphi$

where $k_1, k_2 < \omega$, $c \in \mathbb{Q}$, and $\hat{s}$ is a *state variable* from a countably infinite supply of state variables $\hat{S}$. △

In the following, we implicitly only consider *closed formulae*, and conflate the term $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ *formula* with *closed, clean* $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ *state formula*.

To assign state variables from $\hat{S}$ to concrete states from the state space S of a DTMC, we will use sequences of the form $\mathfrak{s} := (\hat{s}_i \mapsto s_i)_{i<n} \in \big(\hat{S} \times S\big)^n$, for $n < \omega$. $\mathfrak{s}$ will be called a *state assignment* and its length will be denoted by $|\mathfrak{s}| := n$.

We define $\mathfrak{s} \circ (\hat{s} \mapsto s)$ as the expansion of $\mathfrak{s}$ by the assignment $\hat{s} \mapsto s \in \hat{S} \times S$, that is $\mathfrak{s} \circ (\hat{s} \mapsto s) = \big(\hat{s}_0 \mapsto s_0, \ldots, \hat{s}_{n-1} \mapsto s_{n-1}, \hat{s} \mapsto s\big)$, and we denote by $\mathsf{dom}(\mathfrak{s})$ and $\mathsf{im}(\mathfrak{s})$ the sequences $(\hat{s}_i)_{i<n} \in \hat{S}^n$ and $(s_i)_{i<n} \in S^n$, respectively. The empty sequence is represented by ε.

For $\hat{s} \in \hat{S}$, we write, in function notation, $\mathfrak{s}(\hat{s})$, to recall the assignment of $\hat{s}$ in $\mathfrak{s}$. Should no such assignment exist, then we set $\mathfrak{s}(\hat{s}) := \bot$ (undefined). Since we only deal with *clean* formulae, each variable has at most one assignment in $\mathfrak{s}$, and $\mathfrak{s}$ itself takes the form of a partial function over $\hat{S} \to S \cup \{\bot\}$, justifying the notation laid out above.

Definition 2.17 ($\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ Semantics). Let $\mathcal{D} = (S, p, \mathsf{AP}, l)$ be a DTMC, φ, ψ *state formulae*, ρ, ρ' *probabilistic formulae*, and ϑ a *path formula* of $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$. Also, let $n < \omega$, $\mathfrak{s} \in \big(\hat{S} \times S\big)^n$, $\mathfrak{Q} \in \{\forall, \exists\}$, $\star \in \{+, \cdot\}$, $c \in \mathbb{Q}$, and $a \in \mathsf{AP}$. We define

- $\mathcal{D}, \mathfrak{s} \vDash \mathfrak{Q}\hat{s}.\varphi$ iff $\mathfrak{Q}s \in S : \mathcal{D}, \mathfrak{s} \circ (\hat{s} \mapsto s) \vDash \varphi$,
- $\mathcal{D}, \mathfrak{s} \vDash \varphi \wedge \psi$ iff $\mathcal{D}, \mathfrak{s} \vDash \varphi$ and $\mathcal{D}, \mathfrak{s} \vDash \psi$,
- $\mathcal{D}, \mathfrak{s} \vDash \neg\varphi$ iff $\mathcal{D}, \mathfrak{s} \nvDash \varphi$,
- $\mathcal{D}, \mathfrak{s} \vDash \mathtt{true}$,
- $\mathcal{D}, \mathfrak{s} \vDash a_{\hat{s}}$ iff $a \in l\big(\mathfrak{s}(\hat{s})\big)$,
- $\mathcal{D}, \mathfrak{s} \vDash \rho < \rho'$ iff $[\![\rho]\!]_{\mathcal{D},\mathfrak{s}} < [\![\rho']\!]_{\mathcal{D},\mathfrak{s}}$,
- $[\![\mathbb{P}(\vartheta)]\!]_{\mathcal{D},\mathfrak{s}} = \Pr\{\pi \in \mathsf{Paths}_{\mathcal{D}^n}(\mathsf{im}(\mathfrak{s})) \mid \mathcal{D}, \mathfrak{s}, \pi \vDash \vartheta\}$,
- $[\![\rho \star \rho']\!]_{\mathcal{D},\mathfrak{s}} = [\![\rho]\!]_{\mathcal{D},\mathfrak{s}} \star [\![\rho']\!]_{\mathcal{D},\mathfrak{s}}$, and
- $[\![c]\!]_{\mathcal{D},\mathfrak{s}} = c$.

Furthermore, let $n \geq 1$, $k_1, k_2 < \omega$, and $\pi \in \mathsf{Paths}_{\mathcal{D}^n}$. We define

- $\mathcal{D}, \mathfrak{s}, \pi \vDash \bigcirc\varphi$ iff $\mathcal{D}, \mathfrak{s}[\pi(1)] \vDash \varphi$,
- $\mathcal{D}, \mathfrak{s}, \pi \vDash \varphi \,\mathsf{U}\, \psi$ iff $\exists j < \omega \,\big(\mathcal{D}, \mathfrak{s}[\pi(j)] \vDash \psi \wedge \forall i < j : \mathcal{D}, \mathfrak{s}[\pi(i)] \vDash \varphi\big)$, and
- $\mathcal{D}, \mathfrak{s}, \pi \vDash \varphi \,\mathsf{U}^{[k_1,k_2]}\, \psi$ iff $\exists j \in [k_1, k_2] \,\big(\mathcal{D}, \mathfrak{s}[\pi(j)] \vDash \psi \wedge \forall i < j : \mathcal{D}, \mathfrak{s}[\pi(i)] \vDash \varphi\big)$,

where $\mathfrak{s}[\pi(i)]$ denotes the state assignment that results by replacing $\mathsf{im}(\mathfrak{s})$ with $\pi(i)$.[i]

Lastly, if φ is a *closed, clean* $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ *formula*, let $\mathcal{D} \vDash \varphi$ iff $\mathcal{D}, \varepsilon \vDash \varphi$. △

We further allow standard syntactic sugar such as $\mathtt{false} :\equiv \neg\mathtt{true}$, $\varphi \vee \psi :\equiv \neg(\neg\varphi \wedge \neg\psi)$, $\rho_1 = \rho_2 :\equiv \neg(\rho_2 < \rho_1) \wedge \neg(\rho_1 < \rho_2)$, $\rho_1 \leq \rho_2 :\equiv (\rho_1 = \rho_2) \vee (\rho_1 < \rho_2)$, and so on.

[i] $\mathsf{im}(\mathfrak{s})$ and $\pi(i)$ always have the same type and length, since both are states of $\mathcal{D}^n$. The general definition of this notation in the context of sequences can be found on p. 9.

Markov Decision Processes

Definition 2.18 ($\mathsf{HyperPCTL}_{\mathrm{MDP}}$ Syntax). $\mathsf{HyperPCTL}_{\mathrm{MDP}}$ formulae are built by the following grammar rules:

- (*quantified formulae*) $\varphi ::= \forall \hat{\sigma}.\varphi \mid \exists \hat{\sigma}.\varphi \mid \forall \hat{s}(\hat{\sigma}).\varphi \mid \exists \hat{s}(\hat{\sigma}).\varphi \mid \psi$
- (*non-quantified formulae*) $\psi ::= \psi \wedge \psi \mid \neg\psi \mid \mathtt{true} \mid a_{\hat{s}} \mid \rho < \rho$
- (*probabilistic expressions*) $\rho ::= \mathbb{P}(\vartheta) \mid \rho + \rho \mid \rho \cdot \rho \mid c$
- (*path expressions*) $\vartheta ::= \bigcirc\psi \mid \psi \,\mathsf{U}\, \psi \mid \psi \,\mathsf{U}^{[k_1,k_2]}\, \psi$

where $\hat{\sigma}$ is a *scheduler variable* from a countably infinite supply of variables $\hat{\Sigma}$, $\hat{s}$ a *state variable* from a countably infinite supply of variables $\hat{S}$, $c \in \mathbb{Q}$, and $k_1, k_2 < \omega$. △

In the following, we only consider *clean formulae*, and refer to *closed, clean* $\mathsf{HyperPCTL}_{\mathrm{MDP}}$ *state formulae* as simply $\mathsf{HyperPCTL}_{\mathrm{MDP}}$ *formulae*.

Let $\mathcal{M}$ be an MDP. To assign schedulers from $\Sigma_{\mathcal{M}}$ to the corresponding variables from $\hat{\Sigma}$, we will use sequences of the form $\mathfrak{r} := (\hat{\sigma}_i \mapsto \sigma_i)_{i<n} \in \left(\hat{\Sigma} \times \Sigma_{\mathcal{M}}\right)^n$, for $n < \omega$. $\mathfrak{r}$ will be called a *scheduler assignment*. The expressions $\mathfrak{r}(\hat{\sigma})$, $\mathsf{im}(\mathfrak{r})$, $\mathsf{dom}(\mathfrak{r})$, as well as $|\mathfrak{r}|$ are defined similarly to state assignments as seen on p. 18.

This time, state variables will be assigned to schedulers by $\mathfrak{s}$, and the expression $\mathcal{M}^{\mathfrak{s}} := \mathcal{M}^{\mathsf{im}(\mathfrak{s})}$ will denote the DTMC induced by $\mathcal{M}$ with the tuple of schedulers $\mathsf{im}(\mathfrak{s})$, as defined in 2.11. Concrete state variable instantiations will be tracked in a sequence of states of $\mathcal{M}^{\mathfrak{s}}$.

For a state variable $\hat{s}$ such that $\mathfrak{s}(\hat{s}) \neq \bot$, and $\overline{r} \in S^{\mathfrak{s}}$, let $l^{\mathfrak{s}}(\overline{r})(\hat{s})$ be the element of the tuple $l^{\mathfrak{s}}(\overline{r})$ at the index $i < |\mathfrak{s}|$, where i corresponds to the index in $\mathfrak{s}$ in which $\hat{s}$ appears, i.e. $\mathsf{dom}(\mathfrak{s})(i) = \hat{s}$.

Definition 2.19 ($\mathsf{HyperPCTL}_{\mathrm{MDP}}$ Semantics). Let $\mathcal{M} = (S, \mathsf{Act}, p, \mathsf{AP}, l)$ be an MDP, φ a *quantified formula*, ψ, ψ' *non-quantified formulae*, ρ, ρ' *probabilistic expressions*, and ϑ a *path expression* of $\mathsf{HyperPCTL}_{\mathrm{MDP}}$. Also, let $m, n < \omega$, $\mathfrak{s} \in \left(\hat{S} \times \Sigma_{\mathcal{M}}\right)^m$, $\mathfrak{r} \in \left(\hat{\Sigma} \times \Sigma_{\mathcal{M}}\right)^n$, $\mathfrak{Q} \in \{\forall, \exists\}$, $\star \in \{+, \cdot\}$, $c \in \mathbb{Q}$, and $a \in \mathsf{AP}$, as well as $\overline{r}$ be a state of the induced DTMC $\mathcal{M}^{\mathfrak{s}}$. We define

- $\mathcal{M}, \mathfrak{r}, \mathfrak{s}, \overline{r} \vDash \mathfrak{Q}\hat{\sigma}.\varphi$ iff $\mathfrak{Q}\sigma \in \Sigma_{\mathcal{M}} : \mathcal{M}, \mathfrak{r} \circ (\hat{\sigma} \mapsto \sigma), \mathfrak{s}, \overline{r} \vDash \varphi$,
- $\mathcal{M}, \mathfrak{r}, \mathfrak{s}, \overline{r} \vDash \mathfrak{Q}\hat{s}(\hat{\sigma}).\varphi$ iff $\mathfrak{Q}s \in S : \mathcal{M}, \mathfrak{r}, \mathfrak{s} \circ (\hat{s} \mapsto \mathfrak{r}(\hat{\sigma})), \overline{r} \circ (\mathsf{init}_{\mathfrak{r}(\hat{\sigma})}(s), s) \vDash \varphi$,
- $\mathcal{M}, \mathfrak{r}, \mathfrak{s}, \overline{r} \vDash \psi$ iff $\mathcal{M}^{\mathfrak{s}}, \overline{r} \vDash \psi$,
- $\mathcal{M}^{\mathfrak{s}}, \overline{r} \vDash \psi \wedge \psi'$ iff $\mathcal{M}^{\mathfrak{s}}, \overline{r} \vDash \psi$ and $\mathcal{M}^{\mathfrak{s}}, \overline{r} \vDash \psi'$,
- $\mathcal{M}^{\mathfrak{s}}, \overline{r} \vDash \neg\psi$ iff $\mathcal{M}^{\mathfrak{s}}, \overline{r} \nvDash \psi$,
- $\mathcal{M}^{\mathfrak{s}}, \overline{r} \vDash \mathtt{true}$,

- $\mathcal{M}^{\mathfrak{s}}, \bar{r} \vDash a_{\hat{s}}$ iff $a \in l^{\mathfrak{s}}(\bar{r})(\hat{s})$,
- $\mathcal{M}^{\mathfrak{s}}, \bar{r} \vDash \rho < \rho'$ iff $[\![\rho]\!]_{\mathcal{M}^{\mathfrak{s}},\bar{r}} < [\![\rho']\!]_{\mathcal{M}^{\mathfrak{s}},\bar{r}}$,
- $[\![\mathbb{P}(\vartheta)]\!]_{\mathcal{M}^{\mathfrak{s}},\bar{r}} = \Pr\left\{\pi \in \mathsf{Paths}_{\mathcal{M}^{\mathfrak{s}}}(\bar{r}) \;\middle|\; \mathcal{M}^{\mathfrak{s}}, \pi \vDash \vartheta\right\}$
- $[\![\rho \star \rho']\!]_{\mathcal{M}^{\mathfrak{s}},\bar{r}} = [\![\rho]\!]_{\mathcal{M}^{\mathfrak{s}},\bar{r}} \star [\![\rho']\!]_{\mathcal{M}^{\mathfrak{s}},\bar{r}}$, and
- $[\![c]\!]_{\mathcal{M}^{\mathfrak{s}},\bar{r}} = c$.

Furthermore, let $n \geq 1$, $k_1, k_2 < \omega$, and $\pi \in \mathsf{Paths}_{\mathcal{M}^{\mathfrak{s}}}$. We define

- $\mathcal{M}^{\mathfrak{s}}, \pi \vDash \bigcirc\psi$ iff $\mathcal{M}^{\mathfrak{s}}, \pi(1) \vDash \psi$,
- $\mathcal{M}^{\mathfrak{s}}, \pi \vDash \psi \mathbin{\mathsf{U}} \psi'$ iff $\exists j < \omega \left(\mathcal{M}^{\mathfrak{s}}, \pi(j) \vDash \psi' \wedge \forall i < j : \mathcal{M}^{\mathfrak{s}}, \pi(i) \vDash \psi\right)$, and
- $\mathcal{M}^{\mathfrak{s}}, \pi \vDash \psi \mathbin{\mathsf{U}^{[k_1,k_2]}} \psi'$ iff $\exists j \in [k_1, k_2] \left(\mathcal{M}^{\mathfrak{s}}, \pi(j) \vDash \psi' \wedge \forall i < j : \mathcal{M}^{\mathfrak{s}}, \pi(i) \vDash \psi\right)$.

Lastly, if φ is a *closed, clean* $\mathsf{HyperPCTL}_{\mathrm{MDP}}$ *state formula*, we define $\mathcal{M} \vDash \varphi$ iff $\mathcal{M}, \varepsilon, \varepsilon, \varepsilon \vDash \varphi$, where ε denotes the empty tuple. △

2.3.2 HyperPCTL*

HyperPCTL* [Wan+21] has only been formulated for DTMCs. It is based on PCTL* and extends it by including arithmetic operations directly between probabilistic expressions, and arbitrary nesting thereof, as well as by allowing the indexing of atomics by paths, which are drawn at the level of a probabilistic operator. HyperPCTL* does not feature any explicit quantification other than this drawing of paths.

Before we start with the syntax, we are going to define an auxiliary construct that will be needed to allow the existence of well-defined clean formulae, in which nested probabilistic operators can reference paths that are drawn at the level of their parents. This is a stark deviation from the original version of HyperPCTL* presented in [Wan+21]. The reasoning behind this is explained in Appendix A.

A *path draw substitution rule over a set of variables* $\hat{\Pi}$, denoted by κ and variants, is a tuple $(\hat{\pi}, \hat{\pi}')$, for some $\hat{\pi}, \hat{\pi}' \in \hat{\Pi}$. A sequence of rules $\overline{\kappa}$ is also called a *ruleset*. For reasons that are going to become clear when we define the semantics, we write $\hat{\pi} \leftarrow \hat{\pi}'$ instead of the tuple itself and read this as $\hat{\pi}$ *draws from* $\hat{\pi}'$. Also allowed is the half-rule $\hat{\pi} \leftarrow \varepsilon$, or simply $\hat{\pi}$. We denote *the space of all path draw substitution rules over* $\hat{\Pi}$ by $\mathfrak{K}_{\hat{\Pi}}$.

Definition 2.20 (HyperPCTL* Syntax). HyperPCTL* formulae are built according to the following rules.

- (*path formulae*) $\varphi ::= \varphi \wedge \varphi \mid \neg\varphi \mid \vartheta \mid \mathtt{true} \mid a_{\hat{\pi}} \mid \rho < \rho$

- (*probabilistic expressions*) $\rho ::= f\,\overline{\rho} \mid \mathbb{P}_{\overline{\kappa}}(\varphi)$
- (*path expressions*) $\vartheta ::= \bigcirc\varphi \mid \varphi \,\mathsf{U}\, \varphi \mid \varphi \,\mathsf{U}^{\leq k}\, \varphi$

where $\hat{\pi}$ is a *path variable* from a countably infinite supply of path variables $\hat{\Pi}$, $\overline{\kappa}$ is a finite sequence of *path draw substitution rules* from $\mathfrak{K}_{\hat{\Pi}}$, a is an *atomic proposition*, $k < \omega$, $|\overline{p}| < \omega$, and f is an *elementary function*. △

In the context of HyperPCTL*, an *elementary function* is either a polynomial, a rational, trigonometric, or exponential function, or any finite composition thereof. Nullary functions represent arbitrary constants.

Closedness, in this case, refers to all variable instances being inside the scope of a $\mathbb{P}_{\overline{\kappa}}(\cdot)$ symbol and bound by a rule of $\overline{\kappa}$ (i.e. appearing in its left-hand-side). This includes the right-hand-sides of rulesets of nested probabilistic operators, where ε is considered axiomatically bound.

Cleanliness extends to path draw substitution rules as follows: If φ is cleam a HyperPCTL* formula, then for each $v \in \mathsf{var}(\varphi)$ there is *at most one rule* κ in φ with v on its left-hand-side.

In the following, we implicitly only consider *clean formulae*. Furthermore, the general term HyperPCTL* formulae shall refer only to *closed, clean* HyperPCTL* *path formulae*.

The variant of DTMCs that will be used here is the one with an explicit initial state, which we shall denote by s_ι. To define the semantics of this logic, we will need to assign path variables from $\hat{\Pi}$ to paths from the path space $\mathsf{Paths}_{\mathcal{D}}$ of a given Markov chain $\mathcal{D}$. To this end, we shall use sequences of the form $\mathfrak{p} := (\hat{\pi}_i \mapsto \pi_i)_{i<n} \in \left(\hat{\Pi} \times \mathsf{Paths}_{\mathcal{D}}\right)^n$, $n < \omega$. $\mathfrak{p}$ will be called a *path assignment*, and the expressions $\mathsf{im}(\mathfrak{p})$, $\mathsf{dom}(\mathfrak{p})$, $|\mathfrak{p}|$, and $\mathfrak{p}(\hat{\pi})$ be defined as for state (p. 18) and scheduler (p. 20) assignments.

Furthermore, let $\mathfrak{p}^k$, for $k < \omega$, be the variable assignment that results by discarding the first k elements of the paths (i.e. by shifting the paths k places) in $\mathsf{im}(\mathfrak{p})$, and, in a slight abuse of notation, we set $\mathfrak{p}(\varepsilon)(0) := s_\iota$, where ε is the empty right-hand-side of a path draw substitution rule. This makes path variables that "draw from ε" start at s_ι.

Definition 2.21 (HyperPCTL* Semantics). Let $\mathcal{D} := (S, s_\iota, p, \mathsf{AP}, l)$ be a DTMC with unique initial state $s_\iota \in S$, $n, m < \omega$, φ, φ' *path formulae*, ρ, ρ', and $\rho_1, \ldots, \rho_n$ *probabilistic expressions*, $\hat{\pi}, \hat{\pi}_1, \ldots, \hat{\pi}_n, \hat{\pi}'_1, \ldots, \hat{\pi}'_n \in \hat{\Pi}$, $\mathfrak{p} \in \left(\hat{\Pi} \times \mathsf{Paths}_{\mathcal{D}}\right)^m$, and $a \in \mathsf{AP}$. We define

- $\mathcal{D}, \mathfrak{p} \vDash \varphi \wedge \varphi'$ iff $\mathcal{D}, \mathfrak{p} \vDash \varphi$ and $\mathcal{D}, \mathfrak{p} \vDash \varphi'$,
- $\mathcal{D}, \mathfrak{p} \vDash \neg\varphi$ iff $\mathcal{D}, \mathfrak{p} \nvDash \varphi$,
- $\mathcal{D}, \mathfrak{p} \vDash \bigcirc\varphi$ iff $\mathcal{D}, \mathfrak{p}^1 \vDash \varphi$,
- $\mathcal{D}, \mathfrak{p} \vDash \varphi \,\mathsf{U}\, \varphi'$ iff $\exists j < \omega \left(\mathcal{D}, \mathfrak{p}^j \vDash \varphi' \wedge \forall i < j : \mathcal{D}, \mathfrak{p}^i \vDash \varphi\right)$,

- $\mathcal{D}, \mathfrak{p} \vDash \varphi \, \mathsf{U}^{\leq k} \, \varphi'$ iff $\exists j \leq k \;\; \big(\mathcal{D}, \mathfrak{p}^j \vDash \varphi' \;\wedge\; \forall i < j : \mathcal{D}, \mathfrak{p}^i \vDash \varphi\big)$,
- $\mathcal{D}, \mathfrak{p} \vDash \mathtt{true}$,
- $\mathcal{D}, \mathfrak{p} \vDash a_{\hat{\pi}}$ iff $a \in l\big(\mathfrak{p}(\hat{\pi})(0)\big)$,
- $\mathcal{D}, \mathfrak{p} \vDash \rho < \rho'$ iff $[\![\rho]\!]_{\mathcal{D},\mathfrak{p}} < [\![\rho']\!]_{\mathcal{D},\mathfrak{p}}$,
- $[\![f\,\rho_1 \dots \rho_n]\!]_{\mathcal{D},\mathfrak{p}} = f\,[\![\rho_1]\!]_{\mathcal{D},\mathfrak{p}} \dots [\![\rho_n]\!]_{\mathcal{D},\mathfrak{p}}$, and
- $\Big[\!\Big[\mathbb{P}_{\hat{\pi}_1 \leftarrow \hat{\pi}'_1, \dots, \hat{\pi}_n \leftarrow \hat{\pi}'_n}(\varphi)\Big]\!\Big]_{\mathcal{D},\mathfrak{p}} = \Pr_{\mathcal{D}^n}\Big\{\Big(\big(\pi_1(k), \dots, \pi_n(k)\big)\Big)_{k<\omega} \;\Big|\; \forall i \in [1,n]\; \pi_i \in \mathsf{Paths}_{\mathcal{D}}\big(\mathfrak{p}(\hat{\pi}'_i)(0)\big) \wedge \mathcal{D}, \mathfrak{p} \circ \{\hat{\pi}_i \mapsto \pi_i \;\big|\; i \in [1,n]\} \vDash \varphi\Big\}$.

Lastly, if φ is a *closed, clean* HyperPCTL* *path formula*, we set $\mathcal{D} \vDash \varphi$ iff $\mathcal{D}, \varepsilon \vDash \varphi$. $\triangle$

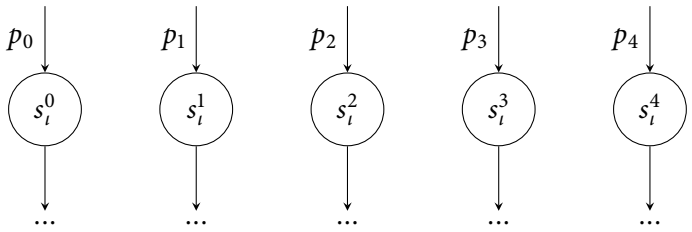
In the last item of the previous definition, path tuples $(\pi_1, \dots, \pi_n)$ are drawn from $\mathcal{D}$, where the i-th component starts where the assignment of the right-hand-side of i-th rule started. Then, for each $k < \omega$, the k-th states of the paths are collected into new tuples $(\pi_1(k), \dots, \pi_n(k))$, which are themselves states of $\mathcal{D}^n$. Finally all of these tuples are ordered together into a path of $\mathcal{D}^n$, and the probability space of $\mathcal{D}^n$ is used to take the measure. In the following, we shall write $\Pr\big\{(\pi_1, \dots, \pi_n) \;\big|\; \dots\big\}$ instead of the above for ease of notation.

To reiterate on the role of rulesets: $\mathbb{P}_{\hat{\pi} \leftarrow \hat{\pi}'}(\cdot)$ causes the assignment of the *new* variable $\hat{\pi}$ to start wherever the assignment of the *old* variable $\hat{\pi}'$ started. Hence comes the wording $\hat{\pi}$ *draws from* $\hat{\pi}'$ and the notation $\hat{\pi} \leftarrow \hat{\pi}'$. The *old* variable must be present in the left-hand-side of a parent probabilistic operator, or be ε, in which case the starting point becomes the initial state s_ι.

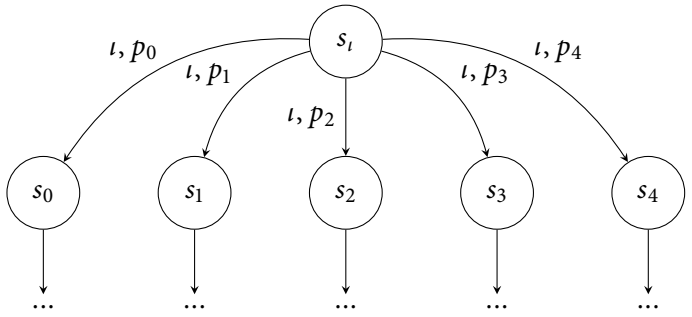
2.3.3 PHL

Probabilistic Hyper Logic (PHL) was introduced for MDPs in [DFT20]. Its fragments encompass probabilistic expressions over LTL formulae [Pnu77], in which atomics are indexed by schedulers, as well as non-probabilistic path expressions in the form of HyperCTL* [Cla+14], which are in turn indexed by paths. Later on, PHL will also be downscaled to fit DTMCs.

The original formulation from [DFT20] uses history-based schedulers, and hence it will be slightly modified to fit Definitions 2.10 and 2.11. Furthermore, we will also deviate from the original by using a unique initial state instead of an initial state distribution. This is done to keep the definitions in line with HyperPCTL and HyperPCTL*, and it doesn't restrict the expressiveness of the model. One can convert an MDP from the latter to the former by introducing a unique state s_ι, and a unique action ι, and assigning $p(s_\iota, \iota, s) = \iota(s)$, for all states s. An example of this is shown in Figure 1.



(a) MDP with initial probability distribution $\iota(s_i) = p_i$, $i \leq 4$.



(b) MDP with unique initial state s_ι, and $p(s_\iota, \iota, s_i) = p_i$, $i \leq 4$.

Figure 1: Example of MDPs with initial probability distribution and unique initial state.

Markov Decision Processes

Definition 2.22 ($\mathsf{PHL}_{\mathrm{MDP}}$ Syntax). $\mathsf{PHL}_{\mathrm{MDP}}$ formulae are built by the following grammar rules:

- (*sched.-quant. formulae*) $\varphi ::= \exists \hat{\sigma}.\varphi \mid \forall \hat{\sigma}.\varphi \mid \varphi \wedge \varphi \mid \neg\varphi \mid \vartheta \mid \rho < \rho$
- (HyperCTL* *formulae*) $\vartheta ::= a_{\hat{\pi}} \mid \mathtt{true} \mid \vartheta \wedge \vartheta \mid \neg\vartheta \mid \bigcirc\vartheta \mid \vartheta \,\mathsf{U}\, \vartheta \mid \forall \hat{\pi}(\hat{\sigma}).\vartheta$
- (*probabilistic expressions*) $\rho ::= \mathbb{P}(\eta) \mid \rho + \rho \mid c \cdot \rho \mid c$
- (*marked* LTL *formulae*) $\eta ::= a_{\hat{\sigma}} \mid \mathtt{true} \mid \eta \wedge \eta \mid \neg\eta \mid \bigcirc\eta \mid \eta \,\mathsf{U}\, \eta$

where a is an *atomic proposition*, $\hat{\sigma}$ a *scheduler variable* from a countably infinite supply of scheduler variables $\hat{\Sigma}$, $\hat{\pi}$ a *path variable* from a countably infinite supply of path variables $\hat{\Pi}$, and $c \in \mathbb{Q}$ a constant. △

In the sequel, we only consider *closed, clean formulae* and refer to *closed, clean* $\mathsf{PHL}_{\mathrm{MDP}}$ *scheduler-quantified formulae* simply as $\mathsf{PHL}_{\mathrm{MDP}}$ *formulae*.

Let $\mathcal{M} = (S, s_\iota, \mathrm{Act}, p, \mathrm{AP}, l)$ be an MDP with unique initial state $s_\iota \in S$. This time, the semantics will require a *scheduler assignment* of the form $\mathfrak{r} \in \left(\hat{\Sigma} \times \Sigma_{\mathcal{M}}\right)^n$, and a *path assignment* of the form $\mathfrak{p} \in \left((\hat{\Sigma} \times \hat{\Pi}) \times \mathrm{Paths}_{\mathcal{M}}\right)^m$, with $n, m < \omega$. $|\mathfrak{p}|$, $|\mathfrak{r}|$, $\mathrm{dom}(\cdot)$, $\mathrm{im}(\cdot)$, $\mathfrak{p}^k$, and $\mathfrak{r}(\hat{\sigma})$ for $\hat{\sigma} \in \hat{\Sigma}$ are defined as usual.

Here, multiple path variables may reference the same scheduler variable. Hence, we need a mechanism to recall path assignments both for path variables in general, as well as for *the most recent path variable that references a specific scheduler variable*.

In essence, we want $\mathfrak{p}(\hat{\pi})$ to recall the assignment of $\hat{\pi}$ (as we assume clean formulae, there will be at most one such assignment), and $\mathfrak{p}(\hat{\sigma})$ to recall the assignment of the *last* path variable associated with $\hat{\sigma}$.

As such, for $\hat{\sigma} \in \hat{\Sigma}$, let $\mathfrak{p}(\hat{\sigma}) = \mathsf{im}(\mathfrak{p})(i)$, where

$$i = \max\{k < |\mathfrak{p}| \;\big|\; \exists \hat{\pi} \in \hat{\Pi} : \mathsf{dom}(\mathfrak{p})(k) = (\hat{\sigma}, \hat{\pi})\},$$

and, for $\hat{\pi} \in \hat{\Pi}$, let $\mathfrak{p}(\hat{\pi}) = \mathsf{im}(\mathfrak{p})(j)$, with j being the *unique* index such that $\mathsf{dom}(\mathfrak{p})(j) = (\hat{\sigma}, \hat{\pi})$ for a $\hat{\sigma} \in \hat{\Sigma}$. In either case $\mathfrak{p}(\hat{\sigma}) = \bot$ or $\mathfrak{p}(\hat{\pi}) = \bot$ if no such i or j exists. Continuing the abuse of notation seen in p. 22 for HyperPCTL*, we define $\mathfrak{p}(\cdot)(0) := s_\iota$, if $\mathfrak{p}(\cdot) = \bot$.

Similarly to $\mathsf{HyperPCTL}_{\mathrm{MDP}}$, we write $\mathcal{M}^{\mathfrak{r}}$ for $\mathcal{M}^{\mathsf{im}(\mathfrak{r})}$. Since the MDP has an initial state, the induced DTMC also has one, specifically

$$s_\iota^{\mathfrak{r}} := \Big(\big(\mathsf{init}_{\sigma_1}(s_\iota), s_\iota\big), ..., \big(\mathsf{init}_{\sigma_{|\mathfrak{r}|}}(s_\iota), s_\iota\big)\Big),$$

where $\big(\sigma_1, ..., \sigma_{|\mathfrak{r}|}\big) = \mathsf{im}(\mathfrak{r})$.

Definition 2.23 ($\mathsf{PHL}_{\mathrm{MDP}}$ Semantics). Let $\mathcal{M} = (S, s_\iota, \mathsf{Act}, p, \mathsf{AP}, l)$ be an MDP with *unique initial state* $s_\iota \in S$, $\mathfrak{Q} \in \{\exists, \forall\}$, φ, φ' *scheduler-quantified formulae*, ρ, ρ' *probabilistic expressions*, ϑ, ϑ' HyperCTL* *formulae*, η, η' LTL *formulae*, $\hat{\sigma} \in \hat{\Sigma}$, $\hat{\pi} \in \hat{\Pi}$, $a \in \mathsf{AP}$, $c \in \mathbb{Q}$, $m, n < \omega$, $\mathfrak{r} \in \big(\hat{\Sigma} \times \Sigma_{\mathcal{M}}\big)^n$, and $\mathfrak{p} \in \big((\hat{\Sigma} \times \hat{\Pi}) \times \mathsf{Paths}_{\mathcal{M}}\big)^m$. We define

- $\mathcal{M}, \mathfrak{r} \vDash \mathfrak{Q}\hat{\sigma}.\varphi$ iff $\mathfrak{Q}\sigma \in \Sigma_{\mathcal{M}} : \mathcal{M}, \mathfrak{r} \circ (\hat{\sigma} \mapsto \sigma) \vDash \varphi$,
- $\mathcal{M}, \mathfrak{r} \vDash \varphi \wedge \varphi'$ iff $\mathcal{M}, \mathfrak{r} \vDash \varphi$ and $\mathcal{M}, \mathfrak{r} \vDash \varphi'$,
- $\mathcal{M}, \mathfrak{r} \vDash \neg\varphi$ iff $\mathcal{M}, \mathfrak{r} \nvDash \varphi$,
- $\mathcal{M}, \mathfrak{r} \vDash \vartheta$ iff $\mathcal{M}, \mathfrak{r}, \varepsilon \vDash \vartheta$,
- $\mathcal{M}, \mathfrak{r} \vDash \rho < \rho'$ iff $[\![\rho]\!]_{\mathcal{M},\mathfrak{r}} < [\![\rho']\!]_{\mathcal{M},\mathfrak{r}}$,
- $\mathcal{M}, \mathfrak{r}, \mathfrak{p} \vDash a_{\hat{\pi}}$ iff $a \in l\big(\mathfrak{p}(\hat{\pi})(0)\big)$,
- $\mathcal{M}, \mathfrak{r}, \mathfrak{p} \vDash \mathsf{true}$,
- $\mathcal{M}, \mathfrak{r}, \mathfrak{p} \vDash \vartheta \wedge \vartheta'$ iff $\mathcal{M}, \mathfrak{r}, \mathfrak{p} \vDash \vartheta \;\wedge\; \mathcal{M}, \mathfrak{r}, \mathfrak{p} \vDash \vartheta'$,
- $\mathcal{M}, \mathfrak{r}, \mathfrak{p} \vDash \neg\vartheta$ iff $\mathcal{M}, \mathfrak{r}, \mathfrak{p} \nvDash \vartheta$,
- $\mathcal{M}, \mathfrak{r}, \mathfrak{p} \vDash \bigcirc\vartheta$ iff $\mathcal{M}, \mathfrak{r}, \mathfrak{p}^1 \vDash \vartheta$,
- $\mathcal{M}, \mathfrak{r}, \mathfrak{p} \vDash \vartheta \,\mathsf{U}\, \vartheta'$ iff $\exists j < \omega \;\big(\mathcal{M}, \mathfrak{r}, \mathfrak{p}^j \vDash \vartheta' \;\wedge\; \forall i < j : \mathcal{M}, \mathfrak{r}, \mathfrak{p}^i \vDash \vartheta\big)$,

- $\mathcal{M}, \mathfrak{r}, \mathfrak{p} \vDash \forall \hat{\pi}(\hat{\sigma}).\vartheta$ iff $\forall \pi \in \mathsf{Paths}_{\mathcal{M}^{\mathfrak{r}(\hat{\sigma})}}(\mathfrak{p}(\hat{\sigma})(0))$ $\mathcal{M}, \mathfrak{r}, \mathfrak{p} \circ \big((\hat{\sigma}, \hat{\pi}) \mapsto \pi\big) \vDash \vartheta$,
- $[\![\mathbb{P}(\eta)]\!]_{\mathcal{M},\mathfrak{r}} = \Pr\big\{\pi \in \mathsf{Paths}_{\mathcal{M}^{\mathfrak{r}}}(s_i^{\mathfrak{r}}) \;\big|\; \mathcal{M}^{\mathfrak{r}}, \pi \vDash \eta\big\}$,
- $[\![\rho + \rho']\!]_{\mathcal{M},\mathfrak{r}} = [\![\rho]\!]_{\mathcal{M},\mathfrak{r}} + [\![\rho']\!]_{\mathcal{M},\mathfrak{r}}$,
- $[\![c \cdot \rho]\!]_{\mathcal{M},\mathfrak{r}} = [\![c]\!]_{\mathcal{M},\mathfrak{r}} \cdot [\![\rho]\!]_{\mathcal{M},\mathfrak{r}}$, and
- $[\![c]\!]_{\mathcal{M},\mathfrak{r}} = c$.

For $\pi \in \mathsf{Paths}_{\mathcal{M}^{\mathfrak{r}}}$, let further

- $\mathcal{M}^{\mathfrak{r}}, \pi \vDash a_{\hat{\sigma}}$ iff $a \in l^{\mathfrak{r}}\big(\pi(0)\big)(\hat{\sigma})$,
- $\mathcal{M}^{\mathfrak{r}}, \pi \vDash \mathtt{true}$,
- $\mathcal{M}^{\mathfrak{r}}, \pi \vDash \eta \wedge \eta'$ iff $\mathcal{M}^{\mathfrak{r}}, \pi \vDash \eta$ and $\mathcal{M}^{\mathfrak{r}}, \pi \vDash \eta'$,
- $\mathcal{M}^{\mathfrak{r}}, \pi \vDash \neg\eta$ iff $\mathcal{M}^{\mathfrak{r}}, \pi \nvDash \eta$,
- $\mathcal{M}^{\mathfrak{r}}, \pi \vDash \bigcirc\eta$ iff $\mathcal{M}^{\mathfrak{r}}, \pi^1 \vDash \eta$, and
- $\mathcal{M}^{\mathfrak{r}}, \pi \vDash \eta \,\mathsf{U}\, \eta'$ iff $\exists j < \omega \;\big(\mathcal{M}^{\mathfrak{r}}, \pi^j \vDash \eta' \;\wedge\; \forall i < j : \mathcal{M}^{\mathfrak{r}}, \pi^i \vDash \eta\big)$,

where $l^{\mathfrak{r}}(\bar{r})(\hat{\sigma}) = l^{\mathfrak{r}}(\bar{r})(i)$, with $i = \max\{i < |\mathfrak{r}| \;\big|\; \mathrm{dom}(\mathfrak{r})(i) = \hat{\sigma}\}$, if this exists, or $\varnothing$ otherwise, and $\bar{r} \in S^{\mathfrak{r}}$.

Finally, if φ is a *closed, clean* $\mathsf{PHL}_{\mathrm{MDP}}$ *scheduler-quantified formula*, we define that $\mathcal{D} \vDash \varphi$ iff $\mathcal{D}, \varepsilon \vDash \varphi$. △

Markov Chains

Next up, $\mathsf{PHL}_{\mathrm{MDP}}$ will be downscaled for DTMCs. Since a DTMC is equivalent to an MDP with $|\mathsf{Act}(s)| = 1$ for all of its states s, we opt to carefully remove scheduler expressions from the previous definitions, as scheduler quantification collapses to the trivial scheduler.

Definition 2.24 ($\mathsf{PHL}_{\mathrm{DTMC}}$ Syntax). $\mathsf{PHL}_{\mathrm{DTMC}}$ formulae are built by the following grammar rules:

- (*top-level formulae*) $\varphi ::= \varphi \wedge \varphi \mid \neg\varphi \mid \vartheta \mid \rho < \rho$
- (HyperCTL* *formulae*) $\vartheta ::= a_{\hat{\pi}} \mid \mathtt{true} \mid \vartheta \wedge \vartheta \mid \neg\vartheta \mid \bigcirc\vartheta \mid \vartheta \,\mathsf{U}\, \vartheta \mid \forall\hat{\pi}.\vartheta$
- (*probabilistic expressions*) $\rho ::= \mathbb{P}(\eta) \mid \rho + \rho \mid c \cdot \rho \mid c$
- (LTL *formulae*) $\eta ::= a \mid \mathtt{true} \mid \eta \wedge \eta \mid \neg\eta \mid \bigcirc\eta \mid \eta \,\mathsf{U}\, \eta$

where a is an *atomic proposition*, $\hat{\pi}$ a path variable from a countably infinite supply of path variables $\hat{\Pi}$, and $c \in \mathbb{Q}$ a constant. △

In the following, we implicitly only consider *closed, clean formulae*, and call *closed, clean* $\mathsf{PHL}_{\mathrm{DTMC}}$ *top-level formulae* simply $\mathsf{PHL}_{\mathrm{DTMC}}$ *formulae*.

Definition 2.25 ($\mathsf{PHL}_{\mathrm{DTMC}}$ Semantics). Let $\mathcal{D} = (S, s_\iota, p, \mathsf{AP}, l)$ be a DTMC with *unique initial state* $s_\iota \in S$, φ, φ' *top-level formulae*, ρ, ρ' *probabilistic expressions*, ϑ, ϑ' $\mathsf{HyperCTL}^*$ *formulae*, η, η' LTL *formulae*, $\hat{\pi} \in \hat{\Pi}$, $a \in \mathsf{AP}$, $c \in \mathbb{Q}$, $n < \omega$, $\mathfrak{p} \in \left(\hat{\Pi} \times \mathrm{Paths}_{\mathcal{D}}\right)^n$. We define

- $\mathcal{D} \vDash \varphi \wedge \varphi'$ iff $\mathcal{D} \vDash \varphi$ and $\mathcal{D} \vDash \varphi'$,
- $\mathcal{D} \vDash \neg\varphi$ iff $\mathcal{D} \nvDash \varphi$,
- $\mathcal{D} \vDash \vartheta$ iff $\mathcal{D}, \varepsilon \vDash \vartheta$,
- $\mathcal{D} \vDash \rho < \rho'$ iff $[\![\rho]\!]_{\mathcal{D}} < [\![\rho']\!]_{\mathcal{D}}$,
- $\mathcal{D}, \mathfrak{p} \vDash a_{\hat{\pi}}$ iff $a \in l\big(\mathfrak{p}(\hat{\pi})(0)\big)$,
- $\mathcal{D}, \mathfrak{p} \vDash \mathtt{true}$,
- $\mathcal{D}, \mathfrak{p} \vDash \vartheta \wedge \vartheta'$ iff $\mathcal{D}, \mathfrak{p} \vDash \vartheta \ \wedge\ \mathcal{D}, \mathfrak{p} \vDash \vartheta'$,
- $\mathcal{D}, \mathfrak{p} \vDash \neg\vartheta$ iff $\mathcal{D}, \mathfrak{p} \nvDash \vartheta$,
- $\mathcal{D}, \mathfrak{p} \vDash \bigcirc\vartheta$ iff $\mathcal{D}, \mathfrak{p}^1 \vDash \vartheta$,
- $\mathcal{D}, \mathfrak{p} \vDash \vartheta \,\mathsf{U}\, \vartheta'$ iff $\exists j < \omega\ \forall i < j\ \left(\mathcal{D}, \mathfrak{p}^i \vDash \vartheta \ \wedge\ \mathcal{D}, \mathfrak{p}^j \vDash \vartheta'\right)$,
- $\mathcal{D}, \mathfrak{p} \vDash \forall\hat{\pi}.\vartheta$ iff $\forall \pi \in \mathrm{Paths}_{\mathcal{D}}(\mathsf{last}(\mathfrak{p})(0))\ \mathcal{D}, \mathfrak{p} \circ (\hat{\pi} \mapsto \pi) \vDash \vartheta$,
- $[\![\mathbb{P}(\eta)]\!]_{\mathcal{D}} = \Pr\left\{\pi \in \mathrm{Paths}_{\mathcal{D}}(s_\iota) \;\middle|\; \mathcal{D}, \pi \vDash \eta\right\}$,
- $[\![\rho + \rho']\!]_{\mathcal{D}} = [\![\rho]\!]_{\mathcal{D}} + [\![\rho']\!]_{\mathcal{D}}$,
- $[\![c \cdot \rho]\!]_{\mathcal{D}} = [\![c]\!]_{\mathcal{D}} \cdot [\![\rho]\!]_{\mathcal{D}}$, and
- $[\![c]\!]_{\mathcal{D}} = c$,

where $\mathsf{last}(\mathfrak{p})$ is the last path that was added to $\mathfrak{p}$ and $\mathsf{last}(\mathfrak{p})(0) := s_\iota$, if $\mathfrak{p} = \varepsilon$. For $\pi \in \mathrm{Paths}_{\mathcal{D}}$, let further

- $\mathcal{D}, \pi \vDash a$ iff $a \in l\big(\pi(0)\big)$
- $\mathcal{D}, \pi \vDash \mathtt{true}$,
- $\mathcal{D}, \pi \vDash \eta \wedge \eta'$ iff $\mathcal{D}, \pi \vDash \eta$ and $\mathcal{D}, \pi \vDash \eta'$,
- $\mathcal{D}, \pi \vDash \neg\eta$ iff $\mathcal{D}, \pi \nvDash \eta$,
- $\mathcal{D}, \pi \vDash \bigcirc\eta$ iff $\mathcal{D}, \pi^1 \vDash \eta$, and
- $\mathcal{D}, \pi \vDash \eta \,\mathsf{U}\, \eta'$ iff $\exists j < \omega\ \forall i < j\ \left(\mathcal{D}, \pi^i \vDash \eta \ \wedge\ \mathcal{D}, \pi^j \vDash \eta'\right)$,

where π^i, $i < \omega$ is the path that results by discarding the first i elements of π. △

This is just *one* way to downscale PHL for DTMCs, in which a feature of the MDP version is lost. Specifically, if we view a DTMC $\mathcal{D} = (S, s_\iota, p, \mathsf{AP}, l)$ as an MDP with $|\mathsf{Act}(s)| = 1$, for all $s \in S$, we can still use the probabilistic part of $\mathsf{PHL}_{\mathrm{MDP}}$ to compare probabilistic hyperproperties restricted to paths starting at s_ι, and with no specifications concerning their branching behaviour later down the line.

For example, take

$$\exists \hat{\sigma}_1. \exists \hat{\sigma}_2. \mathbb{P}(a_{\hat{\sigma}_1} \mathsf{U}\, b_{\hat{\sigma}_2}) > 0 \in \mathsf{PHL}_{\mathrm{MDP}},$$

which, when evaluated on $\mathcal{D}$, asserts that there exist a pair of paths so that we reach a b-state on the second while crossing a-states on the first. In contrast to this, our proposed downscaling only allows *unmarked* LTL in probabilistic expressions, and can express, for example

$$\mathbb{P}(a \mathsf{U}\, b) > 0, \text{ and } \quad \mathbb{P}(\Box a) > 0 \ \wedge\ \mathbb{P}(\Diamond b) > 0,$$

which both imply the original $\mathsf{PHL}_{\mathrm{MDP}}$ formula, but neither is equivalent to it.

Chapter 3

Big Picture

In the sequel, we fix a set AP of atomic propositions and consider all three logics over this set. Furthermore, we always implicitly assume all formulae are *closed* and *clean*, unless explicitly stated otherwise.

3.1 Bridging Semantics

Definition 3.1 (Semantic implication). Let $\mathcal{L}$ and $\mathcal{L}'$ each be one of $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$, $\mathsf{HyperPCTL}^*$, and $\mathsf{PHL}_{\mathrm{DTMC}}$, and $\varphi \in \mathcal{L}$, $\psi \in \mathcal{L}'$. We say that φ *implies* ψ, written $\varphi \vDash \psi$, iff *for all DTMCs* $\mathcal{D}$

$$\mathcal{D} \vDash_{\mathcal{L}} \varphi \;\Rightarrow\; \mathcal{D} \vDash_{\mathcal{L}'} \psi.$$

If the above holds only for *finite* DTMCs, we write $\varphi \vDash_{\mathsf{f}} \psi$ instead. △

Definition 3.2 (Semantic equivalence). Let $\mathcal{L}$ and $\mathcal{L}'$ each be one of $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$, $\mathsf{HyperPCTL}^*$, and $\mathsf{PHL}_{\mathrm{DTMC}}$, and $\varphi \in \mathcal{L}$, $\psi \in \mathcal{L}'$. φ is called *(semantically) equivalent to* ψ, written $\varphi \equiv \psi$, iff

$$\varphi \vDash \psi \;\text{ and }\; \psi \vDash \varphi.$$

If $\varphi \equiv \psi$ only on *finite* DTMCs, we write $\varphi \equiv_{\mathsf{f}} \psi$ instead. △

Definition 3.3 (Subsumption). Let $\mathcal{F}$ and $\mathcal{F}'$ each be a fragment of either $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$, $\mathsf{HyperPCTL}^*$, or $\mathsf{PHL}_{\mathrm{DTMC}}$. $\mathcal{F}'$ *subsumes* $\mathcal{F}$, written $\mathcal{F} \preccurlyeq \mathcal{F}'$, iff to each $\varphi \in \mathcal{F}$ there exists a $\psi \in \mathcal{F}'$ such that $\varphi \equiv \psi$.

If both $\mathcal{F} \preccurlyeq \mathcal{F}'$ and $\mathcal{F}' \preccurlyeq \mathcal{F}$, we simply write $\mathcal{F} \cong \mathcal{F}'$. △

In many cases, one of the logics does not subsume another in the mathematical sense, but a pair $(\mathcal{D}, \varphi)$ can be transformed by a polynomial-time algorithm to a pair $(\mathcal{D}', \varphi')$ such that $\mathcal{D} \vDash \varphi$ iff $\mathcal{D}' \vDash \varphi'$. That is, the model-checking problem of one of the logics is polynomially reducible to the one of the other. To cover these cases, as well as consider finite DTMCs separately, we also define some watered-down forms of subsumption.

Definition 3.4 (Weak subsumption). Let $\mathcal{F}$ and $\mathcal{F}'$ each be a fragment of either $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$, $\mathsf{HyperPCTL}^*$, or $\mathsf{PHL}_{\mathrm{DTMC}}$. We define that

(i) $\mathcal{F}'$ *weakly (algorithmically) subsumes* $\mathcal{F}$, denoted $\mathcal{F} \preccurlyeq_{\mathcal{A}} \mathcal{F}'$, if, on *finite* DTMCs, the model-checking problem of $\mathcal{F}$-sentences is Karp-reducible to the model-checking problem of $\mathcal{F}'$-sentences.

If both $\mathcal{F} \preccurlyeq_{\mathcal{A}} \mathcal{F}'$ and $\mathcal{F}' \preccurlyeq_{\mathcal{A}} \mathcal{F}$, we simply write $\mathcal{F} \cong_{\mathcal{A}} \mathcal{F}'$.

(ii) $\mathcal{F}'$ *weakly subsumes* $\mathcal{F}$ *on finite DTMCs*, denoted $\mathcal{F} \preccurlyeq_{\mathrm{f}} \mathcal{F}'$, iff to each $\varphi \in \mathcal{F}$ there exists a $\psi \in \mathcal{F}'$ such that $\varphi \equiv_{\mathrm{f}} \psi$.

If both $\mathcal{F} \preccurlyeq_{\mathrm{f}} \mathcal{F}'$ and $\mathcal{F}' \preccurlyeq_{\mathrm{f}} \mathcal{F}$, we simply write $\mathcal{F} \cong_{\mathrm{f}} \mathcal{F}'$. △

The relations of Definitions 3.1 to 3.4, are similarly defined for MDPs with $\mathsf{HyperPCTL}_{\mathrm{MDP}}$, and $\mathsf{PHL}_{\mathrm{MDP}}$. Note that in both cases $\preccurlyeq \Rightarrow \preccurlyeq_{\mathrm{f}} \Rightarrow \preccurlyeq_{\mathcal{A}}$.

3.2 Overview on DTMCs

In this section, we shall point out superficial differences between the three logics that are easy to see *a priori*.

As we have seen in Sections 2.3.1 to 2.3.3, $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ is the only one of the three logics that does not require nor use an initial state in the DTMCs on which its semantics operate, and the only one that can quantify over states arbitrarily. In contrast to this, both $\mathsf{HyperPCTL}^*$ and $\mathsf{PHL}_{\mathrm{DTMC}}$ can only quantify over paths drawn starting at a unique initial state. This simple difference alone is enough to conclude that $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ is subsumed by neither $\mathsf{HyperPCTL}^*$ nor $\mathsf{PHL}_{\mathrm{DTMC}}$, as the semantics of the first will also take unreachable states into consideration.

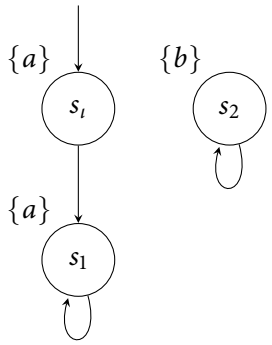


Figure 2: DTMC with a uniquely labelled unreachable state s_2.

Theorem 3.5. $\mathsf{HyperPCTL}_{\mathrm{DTMC}} \not\preccurlyeq \mathsf{HyperPCTL}^*$ and $\mathsf{HyperPCTL}_{\mathrm{DTMC}} \not\preccurlyeq \mathsf{PHL}_{\mathrm{DTMC}}$.

Proof. Consider the DTMC $\mathcal{D}$ shown in Figure 2, and the $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ formula

$$\varphi := \exists \hat{s}_1. \exists \hat{s}_2.\ a_{\hat{s}_1} \wedge b_{\hat{s}_2},$$

which asserts that there exists one state marked a and one marked b. Obviously $\mathcal{D} \vDash \varphi$, for example with the state assignment $(\hat{s}_1 \mapsto s_\iota, \hat{s}_2 \mapsto s_2)$.

In both other logics, there is no possibility to discover this pair of states laid out by Definitions 2.21 and 2.25, so any check for this, be it probabilistic or otherwise, will fail. This remains the case, no matter which state of that DTMC is chosen as initial. □

On the same note, Wang et al. [Wan+21] argue, albeit without providing an explicit proof, that $\mathsf{HyperPCTL}_{\mathrm{DTMC}} \prec_{\mathcal{A}} \mathsf{HyperPCTL}^*$.

Both $\mathsf{HyperPCTL}^*$ and $\mathsf{PHL}_{\mathrm{DTMC}}$ can express weaker formulae that imply the counterexample φ given in the preceding theorem, for example

$$\begin{aligned} &\exists \hat{\pi}_1. \exists \hat{\pi}_2.\ \Diamond a_{\hat{\pi}_1} \wedge \Diamond b_{\hat{\pi}_2} && \in \mathsf{PHL}_{\mathrm{DTMC}}, \text{ and} \\ &\mathbb{P}_{\hat{\pi}_1, \hat{\pi}_2}(\Diamond a_{\hat{\pi}_1} \wedge \Diamond b_{\hat{\pi}_2}) > 0 && \in \mathsf{HyperPCTL}^*, \end{aligned}$$

which both assert that states $s_1, s_2 \in S$ can be found, such that $a \in l(s_1)$, $b \in l(s_2)$ *and* both s_1 and s_2 are reachable from the initial state s_ι by a finite path fragment.

Chapter 4

HyperPCTL* vs. PHL on DTMCs

In this chapter, we compare HyperPCTL* with $\mathsf{PHL}_{\mathrm{DTMC}}$, and a focus is laid mostly on embedding classes of formulae of the latter into the former. First, we will look at the strictly probabilistic part of $\mathsf{PHL}_{\mathrm{DTMC}}$, and it will be shown that this part is completely – and strictly – subsumed by HyperPCTL*. Having done this, we will explore the fragment of $\mathsf{PHL}_{\mathrm{DTMC}}$ that is comprised of non-probabilistic HyperCTL* formulae, and we will identify parts of it that are also expressible in HyperPCTL*. At the very end, we will use the identified transformations to find equivalences between fragments of the two logics.

In the sequel, we consider a fixed set of atomic propositions AP. Furthermore, both of the logics use DTMCs with an explicit initial state. We assume this state to be always named s_ι.

4.1 Probabilistic Hyperproperties

As we have noted in Section 2.3.3 and Section 3.2, the probabilistic part of our proposed downscaling of PHL to DTMCs can only take measures over paths modelling *unmarked* LTL formulae — that is, it expresses probabilistic *non*-hyper properties. First up, we will prove, based on this, that HyperPCTL* is not embeddable in $\mathsf{PHL}_{\mathrm{DTMC}}$.

We proceed to fix a constant $c \in \mathbb{Q}$ and take a look at HyperPCTL* formulae that compare multiple different paths at once, such as

$$\varphi := \mathbb{P}_{\hat{\pi}_1,\hat{\pi}_2}(a_{\hat{\pi}_1} \mathsf{U}\, b_{\hat{\pi}_2}) > c.$$

This formula draws pairs of paths starting at s_ι, and asserts that the probability of continuously crossing a-labelled states on one path until reaching a b-state on the other is larger than c. Recall that the semantics (2.21) advance all drawn paths at the same time, so anything that happens on (the assignment of) $\hat{\pi}_2$ *before* reaching b on it is unimportant. If there is a way to uniquely identify the initial state by atomics, for

example if the initial state is uniquely labelled by an atomic proposition init, then this is equivalent to the following $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ formula:

$$\varphi' \coloneqq \exists \hat{s}_1.\exists \hat{s}_2.\ \mathsf{init}_{\hat{s}_1} \wedge \mathsf{init}_{\hat{s}_2} \wedge \mathbb{P}(a_{\hat{s}_1} \mathsf{U}\, b_{\hat{s}_2}) > c$$

Later, it will be shown that this argument is correct and generalises to an embedding of a considerable fragment of HyperPCTL* into $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$. However, these probabilistic comparisons are inherently incompatible with our proposed $\mathsf{PHL}_{\mathrm{DTMC}}$, which only allows unmarked LTL formulae inside its probabilistic operator. It can, nevertheless, express stricter non-probabilistic variants, such as

$$\psi \coloneqq \forall \hat{\pi}_1.\forall \hat{\pi}_2.\ a_{\hat{\pi}_1} \mathsf{U}\, b_{\hat{\pi}_2},$$

with $\psi \vDash \varphi, \varphi'$. We lay emphasis on the following: $\psi' \coloneqq \mathbb{P}(a \mathsf{U}\, b) > c \in \mathsf{PHL}_{\mathrm{DTMC}}$ is implied by neither φ, nor φ'. $a \mathsf{U}\, b$ models *single* paths that reach a b-state while only crossing a-states, while the latter two formulae take the measure over *pairs of possibly different* paths in parallel, such that only a-states are crossed on the first one until a b-state is reached on the other one.

Example 4.1. Consider the DTMC $\mathcal{D}$ in Figure 3. We compute

$$\begin{aligned}
[\![\mathbb{P}(a \mathsf{U}\, b)]\!]_{\mathcal{D}}^{\mathsf{PHL}_{\mathrm{DTMC}}} &= \Pr\left\{\pi \in \mathsf{Paths}_{\mathcal{D}}(s_\iota) \;\middle|\; \pi \vDash a \mathsf{U}\, b\right\} \\
&= \Pr\left\{s_\iota s_{12}^\omega\right\} \\
&= \Pr\left(\mathsf{Cyl}_{\mathcal{D}}(s_\iota s_{12})\right) \\
&= \frac{1}{7},
\end{aligned}$$

as well as

$$\begin{aligned}
[\![\mathbb{P}_{\hat{\pi}_1,\hat{\pi}_2}(a_{\hat{\pi}_1} \mathsf{U}\, b_{\hat{\pi}_2})]\!]_{\mathcal{D}}^{\mathsf{HyperPCTL}^*} &= \Pr\Big\{(\pi_1,\pi_2) \in \mathsf{Paths}_{\mathcal{D}}(s_\iota)^2 \;\Big|\; \\
&\qquad \mathcal{D}, (\hat{\pi}_1 \mapsto \pi_1, \hat{\pi}_2 \mapsto \pi_2) \vDash a_{\hat{\pi}_1} \mathsf{U}\, b_{\hat{\pi}_2}\Big\} \\
&= \Pr\Big\{(s_\iota s_{11} s_{21}^\omega, s_\iota s_{12}^\omega), (s_\iota s_{11} s_{21}^\omega, s_\iota s_{13} s_{23}^\omega), \\
&\qquad (s_\iota s_{12}^\omega, s_\iota s_{12}^\omega), (s_\iota s_{13} s_{23}^\omega, s_\iota s_{12}^\omega)\Big\} \\
&= \frac{3}{7}\cdot\frac{3}{7} + \frac{3}{7}\cdot\frac{1}{7} + \frac{1}{7}\cdot\frac{1}{7} + \frac{3}{7}\cdot\frac{1}{7} \\
&= \frac{16}{49}.
\end{aligned}$$

Select $c \coloneqq \frac{1}{7}$. For this c, $\mathcal{D} \nvDash \psi'$ but $\mathcal{D} \vDash \varphi$, since $c = \frac{1}{7} = \frac{7}{49} < \frac{16}{49}$. △

We will now collect and prove the preceding thoughts in the following segment. To argue about LTL formulae in probabilistic expressions of $\mathsf{PHL}_{\mathrm{DTMC}}$, we first need more auxiliary terms.

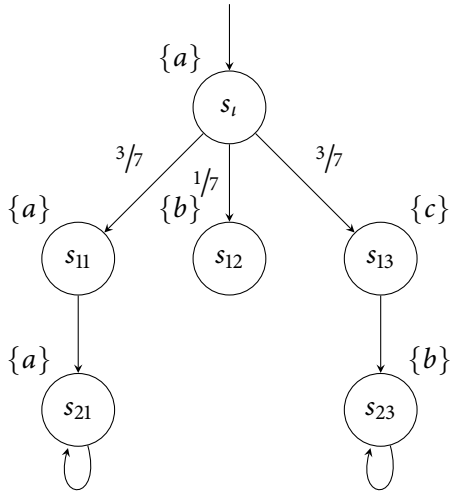


Figure 3: DTMC where b is reached at different offsets in "$\Diamond b$"-paths.

Definition 4.2 (Trace). Let $\mathcal{D}$ be a DTMC with labelling function l, and $\pi \in \mathsf{Paths}_{\mathcal{D}}$. The *trace of* π is defined as *the sequence of all labels that appear in* π:

$$\mathsf{trace}(\pi) := \Big(l\big(\pi(i)\big)\Big)_{i<\omega}$$

A *word over the trace* $\mathsf{trace}(\pi)$ is an *infinite word* that results by selecting exactly one atomic proposition from each member of the sequence. △

For example, from $(\{a\},\{b,c\},\{d\},\{d\},\ldots)$, we can extract the words abd^ω and acd^ω.

Definition 4.3 (Trace equivalence). Let π, π' be paths of one or more DTMCs. π is called *trace-equivalent* to π', denoted $\pi \sim_{\mathsf{tr}} \pi'$, iff

$$\mathsf{trace}(\pi) = \mathsf{trace}(\pi').$$ △

It can be shown [BK08] that the language of all words recognised by an LTL formula is ω-regular and that trace-equivalence implies LTL-equivalence, i.e. that two trace-equivalent paths can *not* be separated by an LTL formula.

Theorem 4.4. Let $a, b \in \mathsf{AP}$, $a \neq b$, and $\hat{\pi}_1, \hat{\pi}_2$ be path variables. $\mathsf{PHL}_{\mathrm{DTMC}}$ can *not* express a probabilistic expression which evaluates equivalently to $\mathbb{P}_{\hat{\pi}_1,\hat{\pi}_2}(a_{\hat{\pi}_1} \,\mathsf{U}\, b_{\hat{\pi}_2})$ in HyperPCTL*.

Proof. We shall construct a counterexample. Consider a family of DTMCs $(\mathcal{D}_n)_{2\leq n<\omega}$, where each $\mathcal{D}_n$, $2 \leq n < \omega$ is defined as shown in Figure 4. We have

$$\{\mathsf{trace}(\pi) \mid \pi \in \mathsf{Paths}_{\mathcal{D}_n}(s)\} = \{a^n c^\omega, a^{n+1} c^\omega, a^{n+1} b^\omega\}.$$

Importantly, no *single* LTL formula can separate $a^n c^\omega$ from $a^{n+1} c^\omega$ for all n. As such, we have the following possibilities for the evaluation of the $\mathsf{PHL}_{\mathrm{DTMC}}$ probabilistic expression $\mathbb{P}(\eta)$, $\eta \in \mathsf{LTL}$:

(i) η models only $s_\iota s_{11} \ldots s_{n1}^\omega$. Then $[\![\mathbb{P}(\eta)]\!]_{\mathcal{D}_n} = \frac{1}{n+1}$.

(ii) η models a path $s_\iota s_{1j} \ldots s_{n+1,j}^\omega$, $j \in [2, n]$. Then it models the paths of that form for all $j \in [2, n]$, since they are trace-equivalent and we compute $[\![\mathbb{P}(\eta)]\!]_{\mathcal{D}_n} = \frac{n-1}{n+1}$.

(iii) η models both the paths laid out in (i) as well as those in (ii). In this case, we get $[\![\mathbb{P}(\eta)]\!]_{\mathcal{D}_n} = \frac{n}{n+1}$.

(iv) η models no paths in $\mathcal{D}_n$. This resolves to $[\![\mathbb{P}(\eta)]\!]_{\mathcal{D}_n} = 0$.

(v) In each of the cases (i)-(iv), η additionally models $s_\iota s_1 \ldots s_{n+1}^\omega$. This adds the factor $\frac{1}{n+1}$ and results in the measures $\frac{2}{n}$, $\frac{n}{n+1}$, 1, and $\frac{1}{n+1}$, respectively.

In contrast to the preceding computations, the evaluation of the HyperPCTL* expression yields

$$\begin{aligned}
[\![\mathbb{P}(a_{\hat{\pi}_1} \mathsf{U}\, b_{\hat{\pi}_2})]\!]_{\mathcal{D}_n,\varepsilon} &= \Pr\Big\{(\pi_1, \pi_2) \;\Big|\; \Big(\pi_1 = s_\iota s_{1j} \ldots s_{n+1,j}^\omega \text{ for some } j \in [2, n] \\
&\qquad\qquad \vee\ \pi_1 = s_\iota s_1 \ldots s_{n+1}^\omega\Big) \\
&\qquad\qquad \wedge\ \pi_2 = s_\iota s_1 \ldots s_{n+1}^\omega\Big\} \\
&= \Pr(s_\iota s_1 \ldots s_{n+1}^\omega) \cdot \left(\Pr(s_\iota s_1 \ldots s_{n+1}^\omega) + \sum_{2\leq j\leq n} \Pr(s_\iota s_{1j} \ldots s_{n+1,j}^\omega)\right) \\
&= \frac{1}{n+1} \cdot \left(\frac{n}{n+1}\right) \\
&= \frac{n}{(n+1)^2},
\end{aligned}$$

which differs from all possibilities for the evaluation of $\mathbb{P}(\eta)$ in $\mathsf{PHL}_{\mathrm{DTMC}}$ by a nonconstant factor that depends on n. Hence there is no way to combine $\mathsf{PHL}_{\mathrm{DTMC}}$ probabilistic expressions by constant multiplication or finite addition such that the resulting formula evaluates to the above for all n. □

This result does not preclude finding equivalent formulae *given knowledge of a specific DTMC, or a family of DTMCs*. For example, if one selects a *constant* n, and builds a factor using this n to bridge the difference, a $\mathsf{PHL}_{\mathrm{DTMC}}$ probabilistic expression can be constructed, which is equivalent to $\mathbb{P}_{\hat{\pi}_1,\hat{\pi}_2}(a_{\hat{\pi}_1} \mathsf{U}\, b_{\hat{\pi}_2})$ *specifically on a single* $\mathcal{D}_n$.

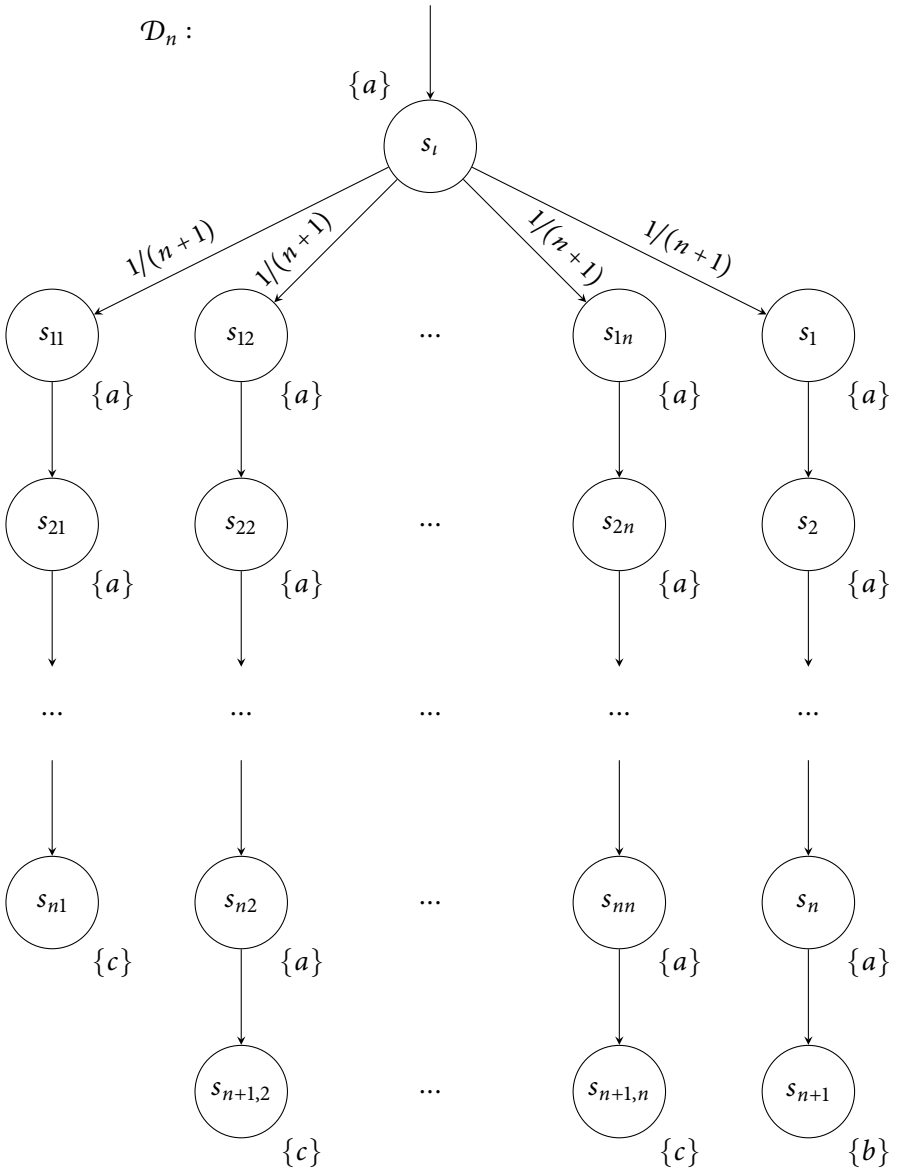


Figure 4: A DTMC, which has $n+1$ paths with equal measures, for $n \leq \omega$. The traces of the paths are (left-to-right) $a^n c^\omega$, $a^{n+1} c^\omega$ ($n-1$ times), $a^{n+1} b^\omega$.

Example 4.5. Select a fixed $n \geq 2$, set

$$c_{\mathcal{D}_n} := \frac{[\![\mathbb{P}_{\hat{\pi}_1,\hat{\pi}_2}(a_{\hat{\pi}_1} \mathsf{U}\, b_{\hat{\pi}_2})]\!]^{\mathsf{HyperPCTL}^*}_{\mathcal{D}_n}}{[\![\mathbb{P}(\Diamond b)]\!]^{\mathsf{PHL}_{\mathrm{DTMC}}}_{\mathcal{D}_n}} = \frac{\frac{n}{(n+1)^2}}{\frac{1}{n+1}} = \frac{n}{n+1},$$

and, with this, build the $\mathsf{PHL}_{\mathrm{DTMC}}$ expression

$$\rho_{\mathcal{D}_n} := c_{\mathcal{D}_n} \cdot \mathbb{P}(\Diamond b)\,.$$

$\rho_{\mathcal{D}_n}$ evaluates equivalently to $\mathbb{P}_{\hat{\pi}_1,\hat{\pi}_2}(a_{\hat{\pi}_1} \mathsf{U}\, b_{\hat{\pi}_2})$ *on* $\mathcal{D}_n$. △

The preceding theorem leads us directly to

Corollary 4.6. $\mathsf{HyperPCTL}^* \not\leq \mathsf{PHL}_{\mathrm{DTMC}}$.

Proof. Select a constant $c \in (0,1)_{\mathbb{Q}}$ and the formula

$$\mathbb{P}_{\hat{\pi}_1,\hat{\pi}_2}(a_{\hat{\pi}_1} \mathsf{U}\, b_{\hat{\pi}_2}) = c \in \mathsf{HyperPCTL}^*$$

The result follows directly from Theorem 4.4. □

There exist more reasons why $\mathsf{HyperPCTL}^*$ does not fit into $\mathsf{PHL}_{\mathrm{DTMC}}$. For instance, $\mathsf{HyperPCTL}^*$ allows, among others, direct multiplication of probabilistic expressions, and the usage of exponential functions, which cannot be represented exactly in a finite form to be compatible with $\mathsf{PHL}_{\mathrm{DTMC}}$.

4.2 On HyperCTL*-less PHL

Let $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}}$ be the HyperCTL*-less fragment of $\mathsf{PHL}_{\mathrm{DTMC}}$, i.e. the fragment that is generated by striking out all ϑ rules in Definition 2.24. The semantics of this fragment are fully compatible with $\mathsf{HyperPCTL}^*$.

Essentially, all of the syntactic rules can be taken over verbatim, and the only change we need to make is map $\mathbb{P}(\eta)$ to $\mathbb{P}_{\hat{\pi}}(\eta_{\hat{\pi}})$, where $\eta_{\hat{\pi}}$ is identical to η, but has its atomics marked by $\hat{\pi}$. Nevertheless, we still want to produce *clean* formulae, so we need to ensure that $\hat{\pi}$ is *fresh* in its context. To achieve this, our transformation will include a counter as a second argument. The counter will be used to index variables and will be incremented for every variable that we add.

Theorem 4.7. $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}} < \mathsf{HyperPCTL}^*$.

Proof. We know that HyperPCTL* $\not\leq \mathsf{PHL}_{\mathrm{DTMC}}$ from Corollary 4.6, so equivalence is ruled out. For the strict subsumption, we give the following transformation. Assume without loss of generality that $\hat{\Pi} = \{\hat{\pi}_0, \hat{\pi}_1, \ldots\}$, let φ, φ' be $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}}$ top-level formulae, ρ, ρ probabilistic expressions, $c \in \mathbb{Q}$, η, η' LTL formulae, $a \in \mathsf{AP}$, $i < \omega$, and set

$$
\begin{aligned}
\mathfrak{T}(\varphi) &:= \mathfrak{T}(\varphi, 0), && (1)\\
\mathfrak{T}(\varphi \wedge \varphi', i) &:= \mathfrak{T}(\varphi, i) \wedge \mathfrak{T}(\varphi', i + |\mathsf{var}(\mathfrak{T}(\varphi, i))|), && (2)\\
\mathfrak{T}(\neg\varphi, i) &:= \neg\mathfrak{T}(\varphi, i), && (3)\\
\mathfrak{T}(\rho < \rho', i) &:= \mathfrak{T}(\rho, i) < \mathfrak{T}(\rho', i + |\mathsf{var}(\mathfrak{T}(\rho, i))|), && (4)\\
\mathfrak{T}(c \cdot \rho, i) &:= \mathfrak{T}(c, i) \cdot \mathfrak{T}(\rho, i), && (5)\\
\mathfrak{T}(\rho + \rho', i) &:= \mathfrak{T}(\rho, i) + \mathfrak{T}(\rho', i + |\mathsf{var}(\mathfrak{T}(\rho, i))|), && (6)\\
\mathfrak{T}(c, i) &:= c, && (7)\\
\mathfrak{T}(\mathbb{P}(\eta), i) &:= \mathbb{P}_{\hat{\pi}_i}(\mathfrak{T}(\eta, i))\,, && (8)\\
\mathfrak{T}(\bigcirc\eta, i) &:= \bigcirc\mathfrak{T}(\eta, i), && (9)\\
\mathfrak{T}(\eta \,\mathsf{U}\, \eta', i) &:= \mathfrak{T}(\eta, i) \,\mathsf{U}\, \mathfrak{T}(\eta', i), && (10)\\
\mathfrak{T}(\neg\eta, i) &:= \neg\mathfrak{T}(\eta, i), && (11)\\
\mathfrak{T}(\eta \wedge \eta', i) &:= \mathfrak{T}(\eta, i) \wedge \mathfrak{T}(\eta', i), && (12)\\
\mathfrak{T}(a, i) &:= a_{\hat{\pi}_i}, \text{ and} && (13)\\
\mathfrak{T}(\mathtt{true}, i) &:= \mathtt{true}. && (14)
\end{aligned}
$$

- Rule 1 initialises a counter variable that is used to produce *clean* HyperPCTL* formulae.
- Rules 2, 4, and 6 recurse on the LHS, and then on the RHS after incrementing its counter by the number of variables that were used in the LHS.
- Rules 3, 5, 7, and 9 through 12 simply take over the syntactical elements of the original formula and recurse on its subformula(e), keeping the counter the same.
- Rule 8 drops into a probabilistic expression, using the variable with the same index as the current counter value.
- Rule 13 marks the original atomic proposition with the same index as the current counter value.
- Rule 14 maps the truth constant to itself.

In $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}}$, atomic propositions only appear within a probabilistic expression, and probabilistic expressions can't be nested. Hence if rule 8 is used, it is never used again recursively and all atomics that appeared in the original $\mathsf{PHL}_{\mathrm{DTMC}}$ subformula η are mapped to themselves indexed by the variable selected by that instance of rule 8. The semantics of $\wedge$, $\neg$, U, and $\bigcirc$ (within probabilistic expressions) are the same across both logics, and the syntactic rules of LTL are compatible with those of path expressions in

HyperPCTL*. Thus every probabilistic expression that is generated by rules 8 through 14 is a syntactically correct, closed HyperPCTL* probabilistic expression.

Since the top-level semantics of $\wedge$, $\neg$, $<$, $\cdot$, and $+$ are the same, it only remains to show that the resulting formula is indeed clean, and that probabilistic expressions and their mappings evaluate equivalently. The latter half is easy to see; to this end, let $\mathcal{D} = (S, s_\iota, p, \mathsf{AP}, l)$ be an arbitrary DTMC. Since $\mathsf{PHL}_{\mathrm{DTMC}}$ does not allow nesting of $\mathbb{P}$ operators, no recursive usage of rule 8 occurs, and we introduce no $\mathbb{P}$-nesting in the resulting formula. Furthermore, each path that is drawn, is drawn from s_ι. Let $i < \omega$ and note that $\mathfrak{T}(\eta, i)$ is exactly the same as η, but with its atomics marked with $\hat{\pi}_i$ by rule 13. We compute

$$\begin{aligned}
[\![\mathbb{P}(\eta)]\!]_{\mathcal{D}}^{\mathsf{PHL}_{\mathrm{DTMC}}} &= \Pr\left\{\pi \in \mathsf{Paths}_{\mathcal{D}}(s_\iota) \mid \mathcal{D}, \pi \vDash_{\mathsf{PHL}_{\mathrm{DTMC}}} \eta\right\} \\
&= \Pr\left\{\pi \in \mathsf{Paths}_{\mathcal{D}}(s_\iota) \mid \mathcal{D}, (\hat{\pi}_i \mapsto \pi) \vDash_{\mathsf{HyperPCTL}^*} \mathfrak{T}(\eta, i)\right\} \\
&= [\![\mathbb{P}_{\hat{\pi}_i}(\mathfrak{T}(\eta, i))]\!]_{\mathcal{D}}^{\mathsf{HyperPCTL}^*} \\
&= [\![\mathfrak{T}(\mathbb{P}(\eta)\,, i)]\!]_{\mathcal{D}}^{\mathsf{HyperPCTL}^*}
\end{aligned}$$

Finally, to show that the formulae that are generated are clean, we only need to consider the transformants of formulae of the form

$$\varphi \wedge \psi, \quad \text{or} \quad \rho \star \rho', \text{ for } \star \in \{<, +\}.$$

This suffices, since the counter does not change after rule 8, and rules 2, 4 and 6 are exactly the ones before rule 8, in which we recurse on both sides of the expression. We ignore rule 5, since its left-recursion cannot reach rule 8.

From rule 2, we get $\mathfrak{T}(\varphi \wedge \psi, i) = \mathfrak{T}(\varphi, i) \wedge \mathfrak{T}(\psi, i + |\mathsf{var}(\varphi)|)$. Let j be the index of a variable in ψ. We have $j \geq i + |\mathsf{var}(\varphi)|$. For a variable indexed by k in φ, we compute $i \leq k < i + |\mathsf{var}(\varphi)|$. In total, we get $j \geq i + |\mathsf{var}(\varphi)| > k$, so $j > k$.

Since j is the index of an arbitrary variable in ψ, and k the index of an arbitrary variable in φ, the maximum index in φ is strictly less than the minimum index in ψ and $\mathfrak{T}(\varphi \wedge \psi, i)$ is clean.

The same result also follows for $\rho \star \rho'$ in a similar fashion by applying rule 4, or rule 6.□

Example 4.8. Consider the $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}}$ formula

$$\mathbb{P}(\bigcirc a \mathbin{\mathsf{U}} b) < \mathbb{P}(c \mathbin{\mathsf{U}} a) \wedge \neg\left(0 < \mathbb{P}(\Diamond a)\right).$$

The most straightforward strategy to apply the transformation from above is leftmost-innermost and we can visualise that as seen in Figure 5.

By putting the branches back together using the original junctors, we finally get

$$\mathbb{P}_{\hat{\pi}_0}(\bigcirc a_{\hat{\pi}_0} \mathbin{\mathsf{U}} b_{\hat{\pi}_0}) < \mathbb{P}_{\hat{\pi}_1}(c_{\hat{\pi}_1} \mathbin{\mathsf{U}} a_{\hat{\pi}_1}) \wedge \neg\,(0 < \mathbb{P}_{\hat{\pi}_2}(\Diamond a_{\hat{\pi}_2})) \qquad \triangle$$

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\mathfrak{T}(\mathbb{P}(\bigcirc a \mathbin{\mathsf{U}} b) < \mathbb{P}(c \mathbin{\mathsf{U}} a) \wedge \neg\,(0 < \mathbb{P}(\Diamond a)))}{\mathfrak{T}(\mathbb{P}(\bigcirc a \mathbin{\mathsf{U}} b) < \mathbb{P}(c \mathbin{\mathsf{U}} a) \wedge \neg\,(0 < \mathbb{P}(\Diamond a)), 0)}\,(1)}{\mathfrak{T}(\mathbb{P}(\bigcirc a \mathbin{\mathsf{U}} b) < \mathbb{P}_{\hat{\pi}_1}(c \mathbin{\mathsf{U}} a), 0)}\,(2)}{\dfrac{\mathfrak{T}(\mathbb{P}(\bigcirc a \mathbin{\mathsf{U}} b), 0)}{\mathbb{P}_{\hat{\pi}_0}(\mathfrak{T}(\bigcirc a \mathbin{\mathsf{U}} b, 0))}\,(8)}\,(4) \qquad \dfrac{\mathfrak{T}(\mathbb{P}(c \mathbin{\mathsf{U}} a), 1)}{\mathbb{P}_{\hat{\pi}_1}(\mathfrak{T}(c \mathbin{\mathsf{U}} a, 1))}\,(8)}{\dfrac{\mathbb{P}_{\hat{\pi}_0}(\mathfrak{T}(\bigcirc a, 0) \mathbin{\mathsf{U}} \mathfrak{T}(b, 0))}{\mathbb{P}_{\hat{\pi}_0}(\bigcirc \mathfrak{T}(a, 0) \mathbin{\mathsf{U}} \mathfrak{T}(b, 0))}\,(9)}\,(10) \qquad \dfrac{\mathbb{P}_{\hat{\pi}_1}(\mathfrak{T}(c, 1) \mathbin{\mathsf{U}} \mathfrak{T}(a, 1))}{\mathbb{P}_{\hat{\pi}_1}(c_{\hat{\pi}_1} \mathbin{\mathsf{U}} \mathfrak{T}(a, 1))}\,(13)}{\dfrac{\mathbb{P}_{\hat{\pi}_0}(\bigcirc a_{\hat{\pi}_0} \mathbin{\mathsf{U}} \mathfrak{T}(b, 0))}{\mathbb{P}_{\hat{\pi}_0}(\bigcirc a_{\hat{\pi}_0} \mathbin{\mathsf{U}} b_{\hat{\pi}_0})}\,(13)}\,(10) \qquad \dfrac{\mathbb{P}_{\hat{\pi}_1}(c_{\hat{\pi}_1} \mathbin{\mathsf{U}} a_{\hat{\pi}_1})}{}\,(13)}{}\,(13)$$

$$\dfrac{\dfrac{\dfrac{\mathfrak{T}(\neg\,(0 < \mathbb{P}(\Diamond a)), 2)}{\neg \mathfrak{T}(0 < \mathbb{P}(\Diamond a), 2)}\,(3)}{\mathfrak{T}(0, 2)}\,(4) \qquad \dfrac{\mathfrak{T}(\mathbb{P}(\Diamond a), 2)}{\mathbb{P}_{\hat{\pi}_2}(\mathfrak{T}(\Diamond a, 2))}\,(8)}{0}\,(7) \qquad \dfrac{\mathbb{P}_{\hat{\pi}_2}(\Diamond \mathfrak{T}(a, 2))}{\mathbb{P}_{\hat{\pi}_2}(\Diamond a_{\hat{\pi}_2})}\,(13)}{}\,(10,14)$$

Figure 5: Visualisation of the mapping from Theorem 4.7 applied to $\mathbb{P}(\bigcirc a \mathbin{\mathsf{U}} b) < \mathbb{P}(c \mathbin{\mathsf{U}} a) \wedge \neg\,(0 < \mathbb{P}(\Diamond a))$. The numbers on the right correspond to the rules used in each step.

The preceding result naturally raises the question of how much of the rest of $\mathsf{PHL}_{\mathrm{DTMC}}$, that is HyperCTL*, fits into HyperPCTL*, which we will look into in the following.

4.3 Σ_1 and Π_1 HyperCTL* Sentences in PHL

In this segment we shall consider Σ_1/Π_1 sentences of the HyperCTL* fragment of PHL, that is sentences of the form

$$\exists \hat{\pi}.\eta, \quad \text{or} \quad \forall \hat{\pi}.\eta,$$

for a quantifier-free η. This was chosen as a starting point, since Σ_1 and Π_1 represent the only possible formulae in HyperCTL* with only one quantifier.

First up, we will show that some parts of the Σ_1/Π_1 fragments of HyperCTL* are redundant in $\mathsf{PHL}_{\mathrm{DTMC}}$, since they can also be expressed by its probabilistic expressions. For this part, we can trivially extract a mapping to HyperPCTL* by translating it to $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}}$ and then applying Theorem 4.4. This will be done by adapting a subset of the results of [BK08, Ch. 10.2.2] to fit our case, since the reasoning is very similar.

4.3.1 Redundant Rules

Let us start with an example. Consider $\forall \hat{\pi}.\Diamond a_{\hat{\pi}}$, which simply asserts that every path has at least one a-labelled state. If this is the case, then

$$\Pr\{\pi \mid \pi \vDash \Diamond a\} = 1,$$

since the measure above evaluates to the measure of $\mathsf{Cyl}(s_\iota)$. Hence

$$\forall \hat{\pi}.\Diamond a_{\hat{\pi}} \vDash \mathbb{P}(\Diamond a) = 1.$$

Using the same reasoning, a can be replaced by any propositional logic (PL) expression ζ over AP. In short, if we denote with $\zeta_{\hat{\pi}}$ the formula ζ with all of its atomics marked with $\hat{\pi}$, this generalises to

$$\forall \hat{\pi}.\Diamond \zeta_{\hat{\pi}} \vDash \mathbb{P}(\Diamond \zeta) = 1, \text{ for } \zeta \in \mathsf{PL}.$$

However, the reverse direction does not hold. For example, in the DTMC of Figure 6, we have a single "$\neg\Diamond a$"-path with measure zero, that is s_ι^ω, so the (dual) measure of $\Diamond a$ evaluates to one. Obviously, the *absolute* property $\forall \hat{\pi}.\Diamond a_{\hat{\pi}}$ does not hold here.

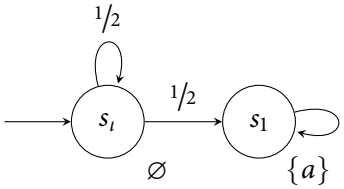


Figure 6: DTMC with $\Pr\{\pi \vDash \Diamond a\} = 1$, and a single "$\neg\Diamond a$"-path.

Lemma 4.9. Let $\zeta \in \mathsf{PL}$. The following equivalences hold:

(i) $\mathbb{P}(\bigcirc \zeta) = 1 \equiv \forall \hat{\pi}.\bigcirc \zeta_{\hat{\pi}}$.

(ii) $\mathbb{P}(\bigcirc \zeta) > 0 \equiv \exists \hat{\pi}.\bigcirc \zeta_{\hat{\pi}}$.

Proof. We will only prove the base case $\zeta = a$, for $a \in \mathsf{AP}$. Let $\mathcal{D} = (S, s_\iota, p, \mathsf{AP}, l)$ be a DTMC.

$$
\begin{aligned}
\text{(i)}\ \mathcal{D} \vDash \mathbb{P}(\bigcirc a) = 1 &\iff [\![\mathbb{P}(\bigcirc a)]\!]_{\mathcal{D}} = 1 \\
&\iff \Pr\{\pi \in \mathsf{Paths}_{\mathcal{D}}(s_\iota) \mid \mathcal{D}, \pi \vDash \bigcirc a\} = 1 \\
&\iff \Pr\{\pi \in \mathsf{Paths}_{\mathcal{D}}(s_\iota) \mid a \in l\big(\pi(1)\big)\} = 1 \\
&\iff \Pr \bigcup \{\mathsf{Cyl}_{\mathcal{D}}(s_\iota s) \mid s \in S \wedge a \in l(s)\} = 1 \\
&\iff \sum_{s \in S \wedge a \in l(s)} \Pr\big(\mathsf{Cyl}_{\mathcal{D}}(s_\iota s)\big) = 1 \\
&\iff \sum_{s \in S \wedge a \in l(s)} p(s_\iota, s) = 1 \\
&\iff \sum_{s \in S} p(s_\iota, s) = 1 \\
&\iff \forall s \in \mathsf{Post}_{\mathcal{D}}(s_\iota) : a \in l(s) \\
&\iff \forall \pi \in \mathsf{Paths}_{\mathcal{D}}(s_\iota) : a \in l\big(\pi(1)\big) \\
&\iff \mathcal{D} \vDash \forall \hat{\pi}. \bigcirc a_{\hat{\pi}}.
\end{aligned}
$$

$$
\begin{aligned}
\text{(ii)}\ \mathcal{D} \vDash \mathbb{P}(\bigcirc a) > 0 &\iff [\![\mathbb{P}(\bigcirc a)]\!]_{\mathcal{D}} > 0 \\
&\iff \Pr\{\pi \in \mathsf{Paths}_{\mathcal{D}}(s_\iota) \mid \mathcal{D}, \pi \vDash \bigcirc a\} > 0 \\
&\iff \Pr\{\pi \in \mathsf{Paths}_{\mathcal{D}}(s_\iota) \mid a \in l\big(\pi(1)\big)\} > 0 \\
&\iff \Pr \bigcup \{\mathsf{Cyl}_{\mathcal{D}}(s_\iota s) \mid s \in S \wedge a \in l(s)\} > 0 \\
&\iff \exists s \in S : a \in l(s) \wedge \Pr\big(\mathsf{Cyl}_{\mathcal{D}}(s_\iota s)\big) > 0 \\
&\iff \exists s \in \mathsf{Post}_{\mathcal{D}}(s) : a \in l(s) \\
&\iff \exists \pi \in \mathsf{Paths}_{\mathcal{D}}(s_\iota) : a \in l\big(\pi(1)\big) \\
&\iff \mathcal{D} \vDash \exists \hat{\pi}. \bigcirc a_{\hat{\pi}}.
\end{aligned}
$$

This concludes the proof. □

The argument above relies entirely on

$$t \in \mathsf{Post}_{\mathcal{D}}(s) \wedge a \in l(t) \Rightarrow \mathsf{Cyl}_{\mathcal{D}}(st) \subset \{\pi \in \mathsf{Paths}_{\mathcal{D}}(s) \mid \pi \vDash \bigcirc a\},$$

i.e. if a path fragment models $\bigcirc a$, then so do all paths that have this fragment as a prefix. This line of argumentation can also be expanded to U. The sentence $\exists \hat{\pi}. a_{\hat{\pi}} \mathsf{U}\, b_{\hat{\pi}}$ holds on paths where b can be reached by a *finite* fragment while only crossing a-labelled states. Hence, the cylinder set of this fragment is a subset of $\{\pi \in \mathsf{Paths}_{\mathcal{D}}(s) \mid \pi \vDash a \mathsf{U}\, b\}$, making the measure of the latter nonzero. Furthermore, we can also extract a special case from $\neg\Diamond a \equiv \Box\neg a$.

Lemma 4.10. Let $\zeta, \zeta' \in \mathsf{PL}$. The following equivalences hold.

(i) $\mathbb{P}(\zeta \mathsf{U}\, \zeta') > 0 \equiv \exists \hat{\pi}.\ \zeta_{\hat{\pi}} \mathsf{U}\, \zeta'_{\hat{\pi}}$.

(ii) $\mathbb{P}(\Box\zeta) = 1 \equiv \forall \hat{\pi}. \Box\zeta_{\hat{\pi}}$.

Proof. Again, we only prove the base case for $\zeta = a$ and $\zeta' = b$, $a, b \in \mathsf{AP}$. Let $\mathcal{D} = (S, s_\iota, p, \mathsf{AP}, l)$ be a DTMC.

(i)
$$\begin{aligned}
\mathcal{D} \vDash \mathbb{P}(a \mathsf{U} b) > 0 &\iff [\![\mathbb{P}(a \mathsf{U} b)]\!]_{\mathcal{D}} > 0 \\
&\iff \Pr\{\pi \in \mathsf{Paths}_{\mathcal{D}}(s_\iota) \mid \mathcal{D}, \pi \vDash a \mathsf{U} b\} > 0 \\
&\iff \Pr\{\pi \in \mathsf{Paths}_{\mathcal{D}}(s_\iota) \mid \exists j < \omega \; \forall i < j : \\
&\qquad\qquad a \in l\big(\pi(i)\big) \wedge b \in l\big(\pi(j)\big)\} > 0 \\
&\iff \exists j < \omega \; \exists s_\iota s_1 \ldots s_j \in \mathsf{Paths}^{<\omega}_{\mathcal{D}}(s_\iota) : \\
&\qquad\qquad b \in l(s_j) \wedge \forall i < j : a \in l\big(s_i\big) \\
&\qquad\qquad\qquad \wedge \Pr\big(\mathsf{Cyl}_{\mathcal{D}}(s_\iota \ldots s_j)\big) > 0 \\
&\iff \exists \pi \in \mathsf{Paths}_{\mathcal{D}}(s_\iota) \; \exists j < \omega \; \forall i < j : \\
&\qquad\qquad a \in l\big(\pi(i)\big) \wedge b \in l\big(\pi(j)\big) \\
&\iff \mathcal{D} \vDash \exists \hat{\pi}.\, a_{\hat{\pi}} \mathsf{U} b_{\hat{\pi}}.
\end{aligned}$$

(ii)
$$\begin{aligned}
\mathcal{D} \vDash \mathbb{P}(\Box a) = 1 &\iff [\![\mathbb{P}(\Box a)]\!]_{\mathcal{D}} = 1 \\
&\iff [\![1 - \mathbb{P}(\Box a)]\!]_{\mathcal{D}} = 0 \\
&\overset{2.6}{\iff} [\![\mathbb{P}(\neg\Box a)]\!]_{\mathcal{D}} = 0 \\
&\iff [\![\mathbb{P}(\Diamond\neg a)]\!]_{\mathcal{D}} = 0 \\
&\iff \neg\Big([\![\mathbb{P}(\Diamond\neg a)]\!]_{\mathcal{D}} > 0\Big) \\
&\iff \neg\Big(\mathcal{D} \vDash \mathbb{P}(\Diamond\neg a) > 0\Big) \\
&\iff \mathcal{D} \nvDash \mathbb{P}(\Diamond\neg a) > 0 \\
&\overset{\text{(i)}}{\iff} \mathcal{D} \nvDash \exists \hat{\pi}.\Diamond\neg a_{\hat{\pi}} \\
&\iff \mathcal{D} \vDash \neg\exists \hat{\pi}.\Diamond\neg a_{\hat{\pi}} \\
&\iff \mathcal{D} \vDash \forall \hat{\pi}.\Box a_{\hat{\pi}}.
\end{aligned}$$

This concludes the proof. □

The argument of item (i) in the preceding lemma can be generalised for arbitrary nesting of U operations. An exhaustive proof will be given in the next section for the more general case, however the intuition behind it is as follows. First consider right-nesting:

$$a \mathsf{U} b \mathsf{U} c$$

Any path π that models $a \mathsf{U} b \mathsf{U} c$ has a trace that has a word with an initial segment of the form $a^n b^m c$, for $n, m < \omega$. Whatever follows is irrelevant, hence every path in

$$\mathsf{Cyl}\big(\pi(0) \cdots \pi(n) \cdots \pi(n+m)\pi(n+m+1)\big)$$

models it as well. The cylinder set has a nonzero measure, whence follows the wanted result. For left-nesting, as in

$$(a \mathsf{U} b) \mathsf{U} c,$$

we would get the following situation. A path π models it iff it has an initial segment with a trace with a word of the form, for example, $a^n b^m c$, c, or similar, for $n, m < \omega$. The same argument as for right-nesting applies, since it all comes down to the existence of a finite prefix of the path that has the wanted trace.

Note, however, that we can not swap $\exists$ and $\forall$, or > 0 and $= 1$. The reason behind this is similar to the one presented in [BKo08] for $\forall \Diamond a$ and $\exists \Box b$ in the context of CTL vs. PCTL. Here, explicitly transferring this result over is beyond the point, since we are ultimately interested in embedding PHL in HyperPCTL*, and not PHL in itself; knowing that a class of formulae of HyperCTL* is not expressible in the probabilistic part of PHL does not preclude them from being expressible in HyperPCTL*.

In the same line of argumentation, one can show that $\bigcirc$ is compatible with U and $\Box$ in the above. First, note that $\bigcirc$ distributes with U:

$$\bigcirc(a \mathsf{U} b) \equiv \bigcirc a \mathsf{U} \bigcirc b$$

With this, $\exists \hat{\pi}.\bigcirc(a_{\hat{\pi}} \mathsf{U} b_{\hat{\pi}}) \equiv \mathbb{P}(\bigcirc(a \mathsf{U} b)) > 0$ follows exactly as in the preceding theorem by replacing $\pi(i)$ and $\pi(j)$ with $\pi(i+1)$ and $\pi(j+1)$, respectively. The case for $\forall \hat{\pi}.\bigcirc\Box a_{\hat{\pi}}$ follows, again, by duality. Similarly, one can show that $\exists \hat{\pi}.(\bigcirc a_{\hat{\pi}} \mathsf{U} b_{\hat{\pi}}) \equiv \mathbb{P}(\bigcirc a \mathsf{U} b) > 0$, $\exists \hat{\pi}.(a_{\hat{\pi}} \mathsf{U} \bigcirc b_{\hat{\pi}}) \equiv \mathbb{P}(a \mathsf{U} \bigcirc b) > 0$, and so on.

Chaining all of the preceding using $\wedge$, and $\vee$ is also possible. For $\wedge$, take the path fragment with the maximum length of those modelling the LHS and RHS, and for $\vee$ a path fragment that models either formula.

In essence, the Σ_1 fragment of HyperCTL*, where also no negation of U and $\bigcirc$ terms is allowed (LTL$^+$), in total denoted $[\Sigma_1|\mathsf{LTL}^+]$, and called the *1-existential LTL-positive* fragment of HyperCTL*, can be mapped to PHL probabilistic expressions, and via Theorem 4.7 to HyperPCTL* formulae. An explicit grammar for this fragment is shown in Figure 7. Note that negation of strictly propositional formulae *inside* U and $\bigcirc$ terms is still allowed. Due to this, and $\bigcirc\neg a \equiv \neg\bigcirc a$, it is implicitly allowed that $\bigcirc$ terms that contain strictly propositional formulae also be negated.

$$\begin{array}{ll} (\Sigma_1 \textit{ formulae}) & \varphi ::= \exists \hat{\pi}.\eta \\ (\mathsf{LTL}^+ \textit{ formulae}) & \eta ::= \eta \mathsf{U} \eta \mid \bigcirc\eta \mid \eta \wedge \eta \mid \eta \vee \eta \mid \zeta \\ (\mathsf{PL} \textit{ formulae}) & \zeta ::= \zeta \wedge \zeta \mid \neg\zeta \mid a_{\hat{\pi}} \end{array}$$

Figure 7: Grammar of $[\Sigma_1|\mathsf{LTL}^+]$-HyperCTL*

At last, we sum all of this up in

Theorem 4.11. $[\Sigma_1|\mathsf{LTL}^+]\text{-}\mathsf{HyperCTL}^* < \mathsf{PHL}^{\text{no}\vartheta}_{\text{DTMC}} < \mathsf{HyperPCTL}^*$.

Proof. We give the following explicit transformation from $[\Sigma_1|\mathsf{LTL}^+]$-HyperCTL* to $\mathsf{PHL}^{\text{no}\vartheta}_{\text{DTMC}}$. Let η, η' be LTL^+ formulae, and ζ, ζ' PL formulae, built as shown in the grammar of Figure 7, $a \in \mathsf{AP}$, and $\hat{\pi}$ a path variable. Set

$$
\begin{aligned}
\mathfrak{T}(\exists \hat{\pi}.\eta) &:= \mathbb{P}(\mathfrak{T}(\eta)) > 0, && (1)\\
\mathfrak{T}(\eta \,\mathsf{U}\, \eta') &:= \mathfrak{T}(\eta) \,\mathsf{U}\, \mathfrak{T}(\eta'), && (2)\\
\mathfrak{T}(\bigcirc \eta) &:= \bigcirc \mathfrak{T}(\eta), && (3)\\
\mathfrak{T}(\eta \wedge \eta') &:= \mathfrak{T}(\eta) \wedge \mathfrak{T}(\eta'), && (4)\\
\mathfrak{T}(\eta \vee \eta') &:= \mathfrak{T}(\eta) \vee \mathfrak{T}(\eta'), && (5)\\
\mathfrak{T}(\zeta \wedge \zeta') &:= \mathfrak{T}(\zeta) \wedge \mathfrak{T}(\zeta'), && (6)\\
\mathfrak{T}(\neg \zeta) &:= \neg \mathfrak{T}(\zeta), \text{ and} && (7)\\
\mathfrak{T}(a_{\hat{\pi}}) &:= a. && (8)
\end{aligned}
$$

Lemmata 4.9 and 4.10 allow us to map $\exists \hat{\pi}.\zeta \,\mathsf{U}\, \zeta'$ to $\mathbb{P}(\mathfrak{T}(\zeta) \,\mathsf{U}\, \mathfrak{T}(\zeta')) > 0$, and $\exists \hat{\pi}.\bigcirc \zeta$ to $\mathbb{P}(\bigcirc \mathfrak{T}(\zeta)) > 0$, where $\mathfrak{T}$ only strips the variable markings of ζ and ζ'. The result follows as a direct consequence of these, with the preceding argumentation concerning nesting of U and $\bigcirc$ terms, and by chaining $\mathfrak{T}_{4.7}$ of Theorem 4.7 with $\mathfrak{T}$, i.e. $\mathfrak{T}_{4.7} \circ \mathfrak{T}$.□

Let $[\Pi_1|\neg\mathsf{LTL}^+]$-HyperCTL* be the fragment that results by replacing the rule φ of Figure 7 by

$$\varphi ::= \forall \hat{\pi}.\neg \eta.$$

This will be called the *1-universal negated LTL-positive* fragment of HyperCTL*, i.e. the fragment comprised of the formulae that have exactly one universal quantifier followed by a negated top-level LTL formula, which in itself has no negations in front of modal operators. We can expand the previous theorem to map this fragment to $\mathsf{PHL}^{\text{no}\vartheta}_{\text{DTMC}}$ as follows.

Corollary 4.12. $[\Pi_1|\neg\mathsf{LTL}^+]\text{-}\mathsf{HyperCTL}^* < \mathsf{PHL}^{\text{no}\vartheta}_{\text{DTMC}} < \mathsf{HyperPCTL}^*$.

Proof. Follows from Theorem 4.11 with $\forall \hat{\pi}.\neg\eta \equiv \neg\exists \hat{\pi}.\eta$, by introducing the extra rule

$$\mathfrak{T}(\forall \hat{\pi}.\neg\eta) := \neg \mathfrak{T}(\exists \hat{\pi}.\eta) \quad (9)$$

to the transformation given in said theorem. □

Note that this corollary also includes the special case for $\forall \hat{\pi}.\Box\zeta_{\hat{\pi}}$ shown in Lemma 4.10(ii), and the one for $\forall \hat{\pi}.\bigcirc\zeta_{\hat{\pi}}$, shown in Lemma 4.9(ii).

As mentioned previously, based on [BKo8], it is reasonable to surmise that $\exists$ generally cannot be exchanged with $\forall$, and > 0 with $= 1$, in the preceding theorem and corollary. This also precludes us from systematically axiomatising HyperCTL* statements of the form $\forall\hat{\pi}.\Diamond\eta$, $\exists\hat{\pi}.\Box\eta$, and so on, in HyperPCTL*.

Conjecture 4.13. Let $a \in \mathsf{AP}$. The HyperCTL* sentences

$$\forall\hat{\pi}.\Diamond a_{\hat{\pi}}, \text{ and}$$
$$\exists\hat{\pi}.\Box a_{\hat{\pi}}$$

are not axiomatisable in HyperPCTL*. △

4.3.2 Nested LTL Negation

In the final part of this section, we will briefly make it plausible that the requirement that LTL formulae – apart from the outermost one – be not negated, which we saw in Section 4.3.1, cannot be lifted.

This will be done by looking at examples of formulae in HyperCTL* with nested negation of modal operators, which evaluate differently from their counterparts in HyperPCTL* on certain DTMCs.

We consider the simplest case of nested-negated LTL formulae,

$$\Diamond\Box a, \quad \text{or equivalently} \quad \mathtt{true}\,\mathsf{U}\,\neg(\mathtt{true}\,\mathsf{U}\,\neg a),$$

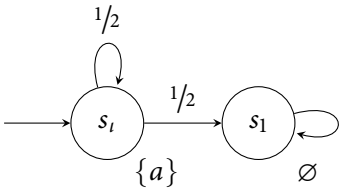
first up with existential quantification:

$$\exists\hat{\pi}.\Diamond\Box a_{\hat{\pi}} \in \text{HyperCTL}^*.$$

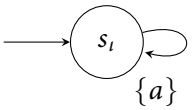
To discover the existence of such a path with a HyperPCTL* formula $\mathbb{P}_{\overline{\kappa}}(\varphi) \sim c$, with $\varphi \in$ HyperPCTL*, $c \in \mathbb{R}$, $\overline{\kappa} \in \mathfrak{K}^*_{\hat{\Pi}}$, and $\sim \in \{\leq, <, =, \neq, >, \geq\}$, we must construct φ in such a way that

$$\Pr\left\{\overline{\pi} \in (\mathsf{Paths}_{\mathcal{D}}(s_\iota))^{|\overline{\kappa}|} \;\middle|\; \mathcal{D}, \overline{\kappa}[\overline{\pi}] \vDash \varphi\right\} \sim c \iff \mathcal{D} \vDash \exists\hat{\pi}.\Diamond\Box a_{\hat{\pi}},$$

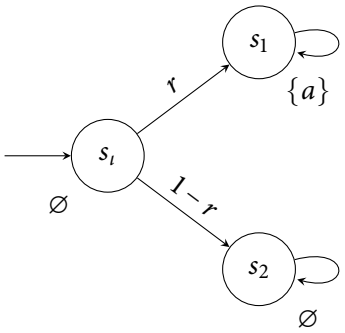
for all DTMCs $\mathcal{D}$. However, we can construct families of DTMCs $(\mathcal{D}_r)$ which contain one DTMC for each value r in the interval $[0,1]_{\mathbb{R}}$, such that the measure of "$\Diamond\Box a$"-paths on $\mathcal{D}_r$ is r, which indirectly shows that this equivalence isn't constructible via comparison to a constant. One such example is shown in Figure 8.



(a) $\mathcal{D}_0$ with $\Pr\{\pi \vDash \Diamond\Box a\} = 0$.



(b) $\mathcal{D}_1$ with $\Pr\{\pi \vDash \Diamond\Box a\} = 1$.



(c) $\mathcal{D}_r$ with $\Pr\{\pi \vDash \Diamond\Box a\} = r$, $r \in (0,1) \subset \mathbb{R}$.

Figure 8: Family of DTMCs $(\mathcal{D}_r)_{r\in[0,1]}$, where $\Pr\{\pi \vDash \Diamond\Box a\}$ spans the entire real interval $[0,1]$, while all $\mathcal{D}_r$ model $\exists\hat{\pi}.\Diamond\Box a_{\hat{\pi}}$.

Lemma 4.14. There exists no constant $c \in [0,1]_{\mathbb{R}}$ such that

$$\underbrace{\mathbb{P}_{\hat{\pi}}(\Diamond\Box a_{\hat{\pi}}) \sim c}_{\text{HyperPCTL}^*} \equiv \underbrace{\exists\hat{\pi}.\Diamond\Box a_{\hat{\pi}}}_{\text{HyperCTL}^*}$$

Proof. Consider the family $(\mathcal{D}_r)_{r\in(0,1)_{\mathbb{R}}}$ of Figure 8. All $\mathcal{D}_r$ model the HyperCTL* formula, which asserts that there exists one path π, such that an offset π^j, for some $j < \omega$, has a trace with the word a^ω. Specifically, on $\mathcal{D}_0$ and $\mathcal{D}_1$, this holds for the path s_ι^ω and e.g. $j = 0$, and on $\mathcal{D}_r$, $r \in (0,1)_{\mathbb{R}}$ for the path $s_\iota s_1^\omega$ and e.g. $j = 1$.

Nevertheless, the evaluation of the HyperPCTL* expression on $\mathcal{D}_r$ yields

$$[\![\mathbb{P}_{\hat{\pi}}(\Diamond\Box a_{\hat{\pi}})]\!]_{\mathcal{D}_r} = r, \quad \text{for } r \in [0,1]_{\mathbb{R}},$$

that is, the probability that a path models $\Diamond\Box a$ spans the entire real interval $[0,1]_{\mathbb{R}}$ on the family $(\mathcal{D}_r)$. □

Using the same family, we can draw a conclusion for

$$\forall\hat{\pi}.\Diamond\Box\neg a_{\hat{\pi}} \in \text{HyperCTL}^*.$$

Specifically, while it implies

$$\mathbb{P}_{\hat{\pi}}(\Diamond\Box\neg a_{\hat{\pi}}) = 1 \in \mathsf{HyperPCTL}^*,$$

there is no HyperPCTL* formula which is equivalent to it, simply because there exists no HyperPCTL* formula that can reliably detect $\Diamond\Box\neg a$ on *all* paths of a given DTMC. The family of Figure 8 attests this.

Lemma 4.15. There exists no constant $c \in [0,1]_{\mathbb{R}}$ such that

$$\underbrace{\mathbb{P}_{\hat{\pi}}(\Diamond\Box\neg a_{\hat{\pi}}) \sim c}_{\mathsf{HyperPCTL}^*} \equiv \underbrace{\forall\hat{\pi}.\Diamond\Box\neg a_{\hat{\pi}}}_{\mathsf{HyperCTL}^*}$$

Proof. We compute

$$[\![\mathbb{P}_{\hat{\pi}}(\Diamond\Box\neg a_{\hat{\pi}})]\!]_{\mathcal{D}_r} = 1 - r \in [0,1], \quad \text{for } r \in [0,1]_{\mathbb{R}},$$

while none of the $\mathcal{D}_r$ model the universally-quantified HyperCTL* formula. A counterexample on $\mathcal{D}_0$ and $\mathcal{D}_1$ is s_ι^ω, and one on $\mathcal{D}_r$, for $r \in (0,1)_{\mathbb{R}}$, is $s_\iota s_1^\omega$. □

Furthermore, using more (nested) quantifiers does not help, as it forces either new quantification over all paths starting as s_ι – which repeats the incompatibilities laid out in Lemmata 4.14 and 4.15 – or quantification over subtrees of a drawn path. The latter will be examined in-depth in the next section. However, for the purposes of the current argument, let it simply be noted that nested quantification in HyperPCTL* behaves similarly to its counterpart in HyperCTL*, and it should be clear, that, for example

$$\exists\hat{\pi}_1.\Diamond\Box a_{\hat{\pi}_1} \not\equiv \exists\hat{\pi}_1.\Diamond\forall\hat{\pi}_2.\Box a_{\hat{\pi}_2},$$

since $\mathcal{D}_0$ (Figure 8a) models the left, but not the right one. Replacing $\forall$ with $\exists$ in the latter furthermore cannot lead to a fitting solution; while it holds that

$$\exists\hat{\pi}_1.\Diamond\Box a_{\hat{\pi}_1} \equiv \exists\hat{\pi}_1.\Diamond\exists\hat{\pi}_2.\Box a_{\hat{\pi}_2},$$

mapping this to HyperPCTL* would require us to have a way to axiomatise $\exists\hat{\pi}.\Box a_{\hat{\pi}}$ in HyperPCTL*, which is not the case.

4.4 HyperCTL* Sentences in PHL with Multiple Quantifiers

The natural next step is to see if and how the results of the preceding section scale with multiple quantifiers. However, we now have two cases to consider, based on whether the formula in question is in *Prenex Normal Form* (PNF), that is whether it has the form

$$\mathfrak{Q}_{n-1}\hat{x}_{n-1}\,\mathfrak{Q}_{n-2}\hat{x}_{n-2}\cdots\mathfrak{Q}_1\hat{x}_1\,\mathfrak{Q}_0\hat{x}.\;\eta,$$

for a sequence of quantifiers $(\mathfrak{Q}_i)_{i<n} \subset \{\exists, \forall\}$, and a quantifier-free η. We make a distinction between

- formulae in PNF and ones that can be transformed to PNF, and
- formulae with nested quantification that cannot be transformed to PNF.

The need for this arises from the semantics of quantification in HyperCTL*. Specifically, quantifiers draw new paths from the start of the last drawn path. As such formulae in PNF draw all their paths from the initial state, whereas ones not in PNF may quantify over subtrees of paths.

In Section 4.3, we did not have to deal with this difference, since the only possible form of (closed) HyperCTL* formulae with just one quantifier is already PNF.

We introduce the following shorthand notation. For $\mathfrak{Q} \in \{\forall, \exists\}$, let

$$\mathfrak{Q}^n\tilde{x} \quad \text{stand for} \quad \mathfrak{Q}\hat{x}_0\cdots\mathfrak{Q}\hat{x}_{n-1}.$$

4.4.1 Formulae in PNF

First up, we will briefly outline that Σ_n/Π_n with alternating quantifiers is incompatible with HyperPCTL*. Σ_n/Π_n formulae are defined as having the form

$$\mathfrak{Q}_{n-1}\hat{x}_{n-1}\,\mathfrak{Q}_{n-2}\hat{x}_{n-2}\cdots\mathfrak{Q}_1\hat{x}_1\,\mathfrak{Q}_0\hat{x}.\;\eta,$$

for a sequence of quantifiers $(\mathfrak{Q}_i)_{i<n} \subset \{\exists, \forall\}$, and a quantifier-free η, where

- The quantifiers are alternating: $\mathfrak{Q}_{i+1} = \{\forall, \exists\} \setminus \mathfrak{Q}_i$, for all $i < n-1$.
- For Σ_n: The outermost quantifier is existential: $\mathfrak{Q}_{n-1} = \exists$.
- For Π_n: The outermost quantifier is universal: $\mathfrak{Q}_{n-1} = \forall$.

Lemma 4.16. Let $1 < n < \omega$. If Conjecture 4.13 holds, then neither the $[\Sigma_n|\mathsf{LTL}^+]$, nor the $[\Pi_n|\mathsf{LTL}^+]$ fragment of HyperCTL* is embeddable in HyperPCTL*.

Proof. The result follows by simple application of the well-known hierarchical inclusion chains $\Sigma_n \subset \Pi_{n+1}$, and $\Pi_n \subset \Sigma_{n+1}$. To reach a contradiction, assume that the opposite of the statement above is true, and consider

$$\varphi \coloneqq \exists \hat{\pi}_{n-1}.\, \forall \hat{\pi}_{n-2}. \cdots \exists \hat{\pi}_1.\, \forall \hat{\pi}_0.\, \big(\mathtt{true}\ \mathsf{U}\ a_{\hat{\pi}_0}\big), \text{ and}$$
$$\varphi' \coloneqq \forall \hat{\pi}_{n-1}.\, \exists \hat{\pi}_{n-2}. \cdots \forall \hat{\pi}_1.\, \exists \hat{\pi}_0. \neg\big(\mathtt{true}\ \mathsf{U}\ \neg a_{\hat{\pi}_0}\big).$$

Note that $\varphi \equiv \forall \hat{\pi}. \Diamond a_{\hat{\pi}}$, and $\varphi' \equiv \exists \hat{\pi}. \Box a_{\hat{\pi}}$. By hypothesis, there exist $\psi, \psi' \in \mathsf{HyperPCTL}^*$, with $\psi \equiv \varphi$, a $\psi' \equiv \varphi'$, however then $\psi \equiv \forall \hat{\pi}. \Diamond a_{\hat{\pi}}$, and $\psi' \equiv \exists \hat{\pi}. \Box a_{\hat{\pi}}$. In both cases, a contradiction to Conjecture 4.13! □

The result of the lemma does not prevent specific special cases of formulae with alternating quantifiers from being embeddable in HyperPCTL*, however it precludes a systematic transformation similar to the ones in the previous sections.

Here, we subsequently focus on PNF prefixed by $\exists^n$ and $\forall^n$. To continue, we need to introduce a new concept and use it to prove a key property of LTL$^+$ that was only glossed over in Section 4.3.1 — namely, when evaluating the semantics of LTL$^+$, formulae, we *always* can limit ourselves to finite path prefixes. For example, for a path π, we have

$$\pi \vDash a\ \mathsf{U}\ b \in \mathsf{LTL}^+ \iff \exists j < \omega\ \forall i < j : \pi^i \vDash a\ \wedge\ \pi^j \vDash b.$$

Unbounded expressions of the form "$j < \omega$" as above exclusively turn up with *existential* quantification, i.e. we never get $\forall i < \omega\ \ldots$, or $\exists j < \omega\ \forall k > j\ \ldots$, or anything similar that would cause an index to range over an unbounded subset of ω.

This very property allowed us to embed $[\Sigma_1|\mathsf{LTL}^+]$-HyperCTL* in HyperPCTL* – and its Π_1 counterpart by way of reduction to Σ_1 via duality. Based hereupon, we introduce the term *non-divergent* for properties that can *not* induce unbounded behaviour, similar to the example above. In the case of DTMCs, we can formalise this using prefixes and cylinder sets.

Definition 4.17 (Non-divergent properties). A trace property P is called *non-divergent*, iff for all DTMCs $\mathcal{D}$ and any path $\pi \in \mathsf{Paths}_{\mathcal{D}}$, such that $\mathcal{D}, \pi \vDash P$, we can find a *prefix* $\pi_{\text{pre}} \sqsubseteq \pi$, such that $\mathcal{D}, \pi' \vDash P$, for all $\pi' \in \mathsf{Cyl}(\pi_{\text{pre}})$. △

Let $\mathcal{D}$ be a DTMC. For a path prefix π, and a path expression η, we write $\mathsf{Cyl}_{\mathcal{D}}(\pi) \vDash \eta$ (*the cylinder set of π models φ*) as a shorthand for $\forall \pi' \in \mathsf{Cyl}_{\mathcal{D}}(\pi) : \mathcal{D}, \pi' \vDash \varphi$.

We call the path assignment $\mathfrak{p}$ *total for the sequence* $\tilde{\pi} \in \hat{\Pi}^n$, $n < \omega$, iff $\mathfrak{p}(\hat{\pi}_i) \neq \bot$, for all $i < n$, and denote the space of path assignments on $\mathcal{D}$ that are total for $\tilde{\pi}$ by $\mathsf{ta}_{\mathcal{D}}(\tilde{\pi})$.

For a tuple of path fragments $\overline{u} \coloneqq (\pi_0, \ldots, \pi_n)$, and $\tilde{\pi} = (\hat{\pi}_0, \ldots, \hat{\pi}_{n-1}) \in \hat{\Pi}^n$, we set

$$\mathfrak{P}_{\mathcal{D}}(\tilde{\pi}, \overline{u}) \coloneqq \Big\{ \mathfrak{p} \in \mathsf{ta}_{\mathcal{D}}(\tilde{\pi}) \;\Big|\; \forall i < n : \mathfrak{p}(\hat{\pi}_i) \in \mathsf{Cyl}_{\mathcal{D}}(\pi_i) \Big\},$$

that is $\mathfrak{P}_{\mathcal{D}}(\tilde{\pi}, \overline{u})$ is the set of assignments over $\tilde{\pi}$ to paths of the cylinder sets of the fragments in $\overline{u}$. Intuitively, this construct represents a translation of cylinder sets to sets of path assignments, and will now be used to expand Definition 4.17 to hyperproperties in

Definition 4.18 (Non-divergent hyperproperties). Let $\tilde{\pi}$ be a sequence of path variables, and H a hyperproperty referencing the variables of $\tilde{\pi}$. The hyperproperty H is called *non-divergent*, iff for any DTMC $\mathcal{D}$, and path assignment $\mathfrak{p} \in \langle \tilde{\pi} \to \mathsf{Paths}_{\mathcal{D}} \rangle$ with $\mathcal{D}, \mathfrak{p} \vDash H$, we can find a sequence $\overline{u}$ of *prefixes* of the paths in $\mathsf{im}(\mathfrak{p})$, such that $\mathcal{D}, \mathfrak{p} \vDash H$, for all $\mathfrak{p} \in \mathfrak{P}_{\mathcal{D}}(\tilde{\pi}, \overline{u})$. △

For a set of path assignments $\mathfrak{P}$, we write $\mathcal{D}, \mathfrak{P} \vDash \eta$, as a shorthand for $\forall \pi \in \mathfrak{P} : \mathcal{D}, \mathfrak{p} \vDash \eta$. Based on these definitions, we can now prove the following intermediary result connecting LTL^+ and non-divergence.

Lemma 4.19. (Marked) LTL^+ formulae specify *non-divergent* (hyper-)properties.

Proof. The wanted result for marked LTL^+ shall be shown via structural induction over the form of LTL^+ formulae. Let $\mathcal{D} := (S, s_\iota, p, \mathsf{AP}, l)$ be a DTMC, and $\mathfrak{p}$ a path assignment. Furthermore, let $n := |\mathfrak{p}|$, and $(\hat{\pi}_0, \ldots, \hat{\pi}_{n-1}) := \mathsf{dom}(\mathfrak{p})$.

Induction Start. Let ζ be a PL formula. We have $\mathfrak{p} \vDash \zeta$ iff the tuple of the first states of all paths in $\mathfrak{p}$ models ζ on its own, since modalities are not allowed in PL. Hence

$$\mathfrak{P}_{\mathcal{D}}\Big(\mathsf{dom}(\mathfrak{p}), \big(\mathfrak{p}(\hat{\pi}_0)(0), \ldots, \mathfrak{p}(\hat{\pi}_{n-1})(0)\big)\Big) \vDash \vartheta.$$

Induction Hypothesis. Let $\eta \in \mathsf{LTL}^+$ be non-divergent, and $1 \leq n := |\mathsf{var}(\eta)|$. Then if, for a path assignment $\mathfrak{p} \in \mathsf{ta}_{\mathcal{D}}(\mathsf{dom}(\mathfrak{p}))$, we have $\mathcal{D}, \mathfrak{p} \vDash \eta$, there exists by definition an n-tuple of finite prefixes $\overline{u} = (\pi_0, \ldots, \pi_{n-1})$ of the paths in $\mathsf{im}(\mathfrak{p})$, such that $\mathfrak{P}_{\mathcal{D}}\big(\mathsf{dom}(\mathfrak{p}), \overline{u}\big) \vDash \eta$.

Induction Step. We have the following cases.

- $\vartheta := \eta \wedge \eta'$: $\mathfrak{p}$ models ϑ iff it models both η and η' separately. From this, we extract via the hypothesis two tuples of prefixes of $\mathsf{im}(\mathfrak{p})$

  $$\overline{u}_{\mathrm{L}} = (\pi_{\mathrm{Lpre},0}, \ldots, \pi_{\mathrm{Lpre},n-1}), \text{ and}$$
  $$\overline{u}_{\mathrm{R}} = (\pi_{\mathrm{Rpre},0}, \ldots, \pi_{\mathrm{Rpre},n-1})$$

  with

  $$\mathfrak{P}_{\mathcal{D}}\big(\mathsf{dom}(\mathfrak{p}), \overline{u}_{\mathrm{L}}\big) \vDash \eta, \text{ and}$$
  $$\mathfrak{P}_{\mathcal{D}}\big(\mathsf{dom}(\mathfrak{p}), \overline{u}_{\mathrm{R}}\big) \vDash \eta'.$$

We build a sequence of prefixes $\overline{u}$ as follows. For $i < n$, set

$$u_i := \begin{cases} \pi_{\text{Lpre},i}, & \hat{\pi}_i \in \mathsf{var}(\eta) \setminus \mathsf{var}(\eta') \\ \pi_{\text{Rpre},i}, & \hat{\pi}_i \in \mathsf{var}(\eta') \setminus \mathsf{var}(\eta) \\ \arg\max_{\pi \in \{\pi_{\text{Lpre},i}, \pi_{\text{Rpre},i}\}} |\pi|, & \hat{\pi}_i \in \mathsf{var}(\eta) \cap \mathsf{var}(\eta') \end{cases}$$

In essence, for each variable $\hat{\pi} \in \mathsf{var}(\vartheta)$, we select the corresponding member of the L-sequence, if $\hat{\pi}$ only appears in η, and the corresponding member of the R-sequence, if $\hat{\pi}$ only appears in η'. If $\hat{\pi}$ appears in both formulae, we take the corresponding member with the maximum length out of the two sequences.

With this sequence, we directly get

$$\mathfrak{P}_{\mathcal{D}}\big(\mathsf{dom}(\mathfrak{p}), \overline{u}\big) \vDash \vartheta.$$

– $\vartheta := \eta \vee \eta'$: $\mathfrak{p}$ models ϑ iff it models either of the formulae. The result follows similarly to the previous case, by selecting all the members of the L-sequence, if $\mathcal{D}, \mathfrak{p} \vDash \eta$, or all the members of the R-sequence, if $\mathcal{D}, \mathfrak{p} \vDash \eta'$.

– $\vartheta := \bigcirc \eta$: $\mathfrak{p}$ models ϑ iff $\mathfrak{p}^1$ models η. From the hypothesis, we extract a sequence of prefixes $\overline{u}$ of the paths in $\mathsf{im}\big(\mathfrak{p}^1\big)$ with

$$\mathfrak{P}_{\mathcal{D}}\big(\mathsf{dom}(\mathfrak{p}), \overline{u}\big) \vDash \eta.$$

The result follows immediately prepending $\mathfrak{p}(\hat{\pi}_i)(0)$ to u_i, for all $i < |\mathfrak{p}|$.

– $\vartheta := \eta \mathbin{\mathsf{U}} \eta'$: We have, by definition

$$\mathcal{D}, \mathfrak{p} \vDash \vartheta \iff \exists j < \omega : \ \mathcal{D}, \mathfrak{p}^j \vDash \eta' \ \wedge \ \forall i < j : \mathcal{D}, \mathfrak{p}^i \vDash \eta.$$

Let $j < \omega$ be given. By hypothesis, we get tuples of prefixes

$$\begin{aligned} \overline{u}_{\text{L},i} &= (\pi_{\text{Lpre},i,0}, \dots, \pi_{\text{Lpre},i,n-1}), \text{ for each } i < j, \text{ and} \\ \overline{u}_{\text{R}} &= (\pi_{\text{Rpre},0}, \dots, \pi_{\text{Rpre},n-1}) \end{aligned}$$

of the paths in $\mathsf{im}\big(\mathfrak{p}^i\big)$, $i < j$, and $\mathsf{im}\big(\mathfrak{p}^j\big)$, respectively, with

$$\begin{aligned} \mathfrak{P}_{\mathcal{D}}\big(\mathsf{dom}(\mathfrak{p}), \overline{u}_{\text{L},i}\big) &\vDash \eta, \text{ for all } i < j, \text{ and} \\ \mathfrak{P}_{\mathcal{D}}\big(\mathsf{dom}(\mathfrak{p}), \overline{u}_{\text{R}}\big) &\vDash \eta'. \end{aligned}$$

Define for $k < n$ and $i < j$

$$\begin{aligned} \xi_{\text{R}}(k) &:= \Big(\mathfrak{p}(\hat{\pi}_k)(0)\Big) \cdots \Big(\mathfrak{p}(\hat{\pi}_k)(j-1)\Big)\, \pi_{\text{Rpre},k}, \text{ and} \\ \xi_{\text{L},i}(k) &:= \Big(\mathfrak{p}(\hat{\pi}_k)(0)\Big) \cdots \Big(\mathfrak{p}(\hat{\pi}_k)(i-1)\Big)\, \pi_{\text{Lpre},i,k}. \end{aligned}$$

The function ξ_{R} prepends the path fragments of $\overline{u}_{\text{R}}$ with initial segments starting at the corresponding initial state of $\mathsf{im}(\mathfrak{p})$. Similarly, $\xi_{\text{L},i}$ does the

same with the paths of the sequences $\overline{u}_{\text{L},i}$, $i < j$. This is done to bridge the gap between the original assignment $\mathfrak{p}$ and the shifted ones.

We set

$$\Xi_{\text{L}}(k) \coloneqq \left\{\xi_{\text{L},i}(k) \mid i < j\right\}, \text{ and}$$
$$\Xi(k) \coloneqq \Xi_{\text{L}}(k) \cup \{\xi_{\text{R}}(k)\}.$$

The set $\Xi_{\text{L}}(k)$ contains the k-th component of all updated L-sequences, and $\Xi(k)$ additionally the k-th component of the updated R-sequence.

Finally, we use these to construct the sequence $\overline{u}$ by

$$u_k \coloneqq \begin{cases} \xi_{\text{R}}(k) & \hat{\pi}_k \in \mathsf{var}(\eta') \setminus \mathsf{var}(\eta) \\ \arg\max_{\pi \in \Xi_{\text{L}}(k)} |\pi|, & \hat{\pi}_k \in \mathsf{var}(\eta) \setminus \mathsf{var}(\eta') \\ \arg\max_{\pi \in \Xi(k)} |\pi|, & \hat{\pi}_k \in \mathsf{var}(\eta) \cap \mathsf{var}(\eta'), \end{cases}$$

that is we select as the k-th element either the corresponding path of the updated R-sequence, if the related variable only appears in η', the longest path out of the k-th members of all the updated L-sequences, if it only appears in η, and the longest out of all of these if it appears in both formulae.

By construction, we have

$$\mathfrak{P}_{\mathcal{D}}\big(\mathsf{dom}(\mathfrak{p}), \overline{u}\big) \vDash \vartheta.$$

The result for unmarked formulae follows as a special case from the above, by considering unmarked $\eta \in \mathsf{LTL}^+$ equivalent to $\eta_{\hat{\pi}} \in \mathsf{LTL}^+$ marked with exactly one variable $\hat{\pi}$, and by using $\mathsf{Cyl}_{\mathcal{D}}(\pi) \vDash \eta \iff \mathfrak{P}_{\mathcal{D}}(\hat{\pi}, \pi) \vDash \eta_{\hat{\pi}}$. □

We are now equipped to prove

Theorem 4.20. Let $\eta(\tilde{x})$ be an LTL^+ formula with $1 < |\tilde{x}| =: n < \omega$ free path indices. The following equivalence holds.

$$\underbrace{\exists^n \tilde{\pi}.\eta(\tilde{\pi})}_{\mathsf{HyperCTL}^*} \equiv \underbrace{\mathbb{P}_{\tilde{\pi}}(\eta(\tilde{\pi})) > 0}_{\mathsf{HyperPCTL}^*}$$

Proof. Let ψ_{L} be the LHS and ψ_{R} the RHS of the above.

- $\dashv$ Trivial. If the measure of n-tuples of paths modelling $\eta(\tilde{\pi})$ is nonzero, then there exists at least one tuple of paths that models $\eta(\tilde{\pi})$.
- $\vDash$ The result follows similarly to Lemma 4.10. Let $\mathcal{D} \coloneqq (S, s_\iota, p, \mathsf{AP}, l)$ be a DTMC with initial state s_ι. Since LTL^+ formulae specify non-divergent properties (Lemma 4.19), $\mathcal{D} \vDash \psi_L$ holds iff there exists an n-tuple of paths $(\pi_0, ..., \pi_{n-1})$ on $\mathcal{D}$, such that the *cylinder set* of a sequence of their *prefixes*

$$\big((\pi_0(i))_{i<\mu}, ..., (\pi_{n-1}(i))_{i<\mu}\big)$$

models ψ_L, where $\mu \geq 1$ is the length of the individual paths. If $\mu = 1$, then the initial state models $\eta(\tilde{\pi})$ on its own, and the measure evaluates to 1. Otherwise, if $\mu > 1$, we get from

$$\{\overline{\pi} \in \mathsf{Paths}_{\mathcal{D}}(s_\iota)^n \mid \mathcal{D}, \tilde{\pi} \mapsto \overline{\pi} \vDash \eta(\tilde{\pi})\} \supset \{(\pi_0, \ldots, \pi_{n-1})\}$$

the following:

$$\begin{aligned}
[\![\mathbb{P}_{\tilde{\pi}}(\varphi(\tilde{\pi}))]\!]_{\mathcal{D}} &= \Pr\{\overline{\pi} \in \mathsf{Paths}_{\mathcal{D}}(s_\iota)^n \mid \mathcal{D}, \tilde{\pi} \mapsto \overline{\pi} \vDash \eta(\tilde{\pi})\} \\
&\geq \Pr\{(\pi_0, \ldots, \pi_{n-1})\} \\
&\geq \Pr\Big(\mathrm{Cyl}\big((\pi_0(i), \ldots, \pi_{n-1}(i))_{i<\mu}\big)\Big) && \text{(Lemma 4.19)} \\
&= \prod_{m<n} \Pr\Big(\mathrm{Cyl}\big((\pi_m(i))_{i<\mu}\big)\Big) \\
&\geq \prod_{m<n} \Big(\underbrace{\min_{0<i<\mu} p\big(\pi_m(i-1), \pi_m(i)\big)}_{>0, \text{ since } \pi_m \text{ is a path}}\Big)^{\mu} \\
&\qquad\qquad \underbrace{\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}_{>0 \text{ since } n<\omega} \\
&> 0.
\end{aligned}$$

The measure is nonzero in every case, and hence $\psi_{\mathrm{L}} \vDash \psi_{\mathrm{R}}$. □

In the sequel, we call the class of HyperCTL* formulae that have the form $\exists^n \tilde{\pi}.\eta(\tilde{\pi})$, for $\eta \in \mathsf{LTL}^+$, the *PNF-existential LTL-positive* fragment of HyperCTL*, denoted $[\exists^n|\mathsf{LTL}^+]$, and set

$$[\exists^*|\mathsf{LTL}^+]\text{-HyperCTL}^* := \bigcup_{n<\omega} [\exists^n|\mathsf{LTL}^+]\text{-HyperCTL}^*.$$

Corollary 4.21. $[\exists^*|\mathsf{LTL}^+]\text{-HyperCTL}^* < \text{HyperPCTL}^*$.

Proof. Direct application of Theorem 4.20. □

4.4.2 Formulae with nested quantifiers

Lastly, it will be proven that $[\exists^*|\mathsf{LTL}^+]$-HyperCTL* expanded with nesting of further $[\exists^*|\mathsf{LTL}^+]$ formulae is still compatible with HyperPCTL*. That, is we can replace LTL^+ subexpressions in $[\exists^*|\mathsf{LTL}^+]$-HyperCTL* with nested $[\exists^*|\mathsf{LTL}^+]$ formulae, and we can repeat this arbitrarily, while preserving the embeddability of the formulae in HyperPCTL*.

In short, consider formulae $\varphi(x)$ and $\psi(y)$, with placeholders x and y, such that $\varphi(\eta)$ is $[\exists^*|\mathsf{LTL}^+]$, and $\psi(\eta) \in \mathsf{HyperPCTL}^*$ and equivalent to $\varphi(\eta)$, for some $\eta \in \mathsf{LTL}^+$, then

$$\underbrace{\varphi\left(\exists^n \tilde{\pi}.\eta'(\tilde{\pi})\right)}_{\mathsf{HyperCTL}^*} \equiv \underbrace{\psi\left(\mathbb{P}_{\tilde{\pi}\leftarrow \mathsf{last}}\left(\eta'(\tilde{\pi})\right) > 0\right)}_{\mathsf{HyperPCTL}^*},$$

where last is the last quantified path variable in the context of the placeholder y – or ε if no such variable exists – and $\tilde{\pi} \leftarrow \mathsf{last}$ a shorthand for the ruleset

$$(\hat{\pi}_0 \leftarrow \mathsf{last}, \ldots, \hat{\pi}_{|\tilde{\pi}|-1} \leftarrow \mathsf{last}).$$

To subsequently formalise this, we first need a grammar (Figure 9) for the fragment of HyperCTL*, which admits this – and only this – form of nesting. This fragment will be called *recursively existential path-positive*, denoted $[\downarrow\exists^*|\pi^+]$, where $\downarrow$ stands for recursion. As before, we continue to allow negation exclusively in PL formulae. Note that this is a generalisation of $[\Sigma_1|\mathsf{LTL}^+]$, and $[\exists^*|\mathsf{LTL}^+]$, and therefore contains all of their formulae.

$$\begin{array}{ll} (\downarrow\exists^* \textit{ formulae}) & \varphi ::= \exists\hat{\pi}.\varphi \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \eta \\ (\pi^+ \textit{ formulae}) & \eta ::= \varphi \,\mathsf{U}\, \varphi \mid \bigcirc\varphi \mid \zeta \\ (\mathsf{PL} \textit{ formulae}) & \zeta ::= \zeta \wedge \zeta \mid \neg\zeta \mid a_{\hat{\pi}} \mid \mathtt{true} \end{array}$$

Figure 9: Grammar of $[\downarrow\exists^*|\pi^+]$-HyperCTL*

Lemma 4.22. π^+ expressions, built as shown in Figure 9 specify *non-divergent* hyperproperties.

Proof. The claim can be shown via structural induction over the form of π^+ formulae, similarly to Lemma 4.19. □

As a last step before moving on to the embedding itself, we need the following auxiliary term.

Definition 4.23 (Quantifier nesting depth). Let φ be a HyperCTL* formula. We denote by $\mathsf{nd}(\vartheta)$ the *quantifier nesting depth* of ϑ, which is defined recursively as follows.

$$\begin{array}{lcl} \mathsf{nd}(\vartheta \star \vartheta') & = & \max\left\{\mathsf{nd}(\vartheta), \mathsf{nd}(\vartheta')\right\} \\ \mathsf{nd}(\sim \vartheta) & = & \mathsf{nd}(\vartheta) \\ \mathsf{nd}(\mathfrak{Q}_{n-1}\hat{\pi}_{n-1}. \cdots \mathfrak{Q}_0\hat{\pi}_0.\vartheta) & = & 1 + \mathsf{nd}(\vartheta) \\ \mathsf{nd}(a_{\hat{\pi}}) & = & 0 \\ \mathsf{nd}(\mathtt{true}) & = & 0 \end{array}$$

for $\star \in \{\wedge, \mathsf{U}\}$, $\sim \in \{\neg, \bigcirc\}$, and $(\mathfrak{Q}_i)_{i<n} \subset \{\forall, \exists\}$ a sequence of quantifiers. In the 3rd rule, we assume that we always select the greatest applicable n for the formula, that is, we never split a sequence of quantifiers across two steps. △

We can now prove

Theorem 4.24. $[\downarrow\exists^*|\pi^+]$-HyperCTL* < HyperPCTL*.

Proof. We will give an explicit transformation. Let φ, φ' be $\downarrow\exists^*$ formulae, ζ be a PL formula, built as seen in the grammar of Figure 9, and $n < \omega$, and set

$$
\begin{aligned}
\mathfrak{T}(\varphi) &:= \mathfrak{T}(\varphi, \varepsilon), && (1)\\
\mathfrak{T}(\exists^n \tilde{\pi}.\varphi, \tau) &:= \mathbb{P}_{\tilde{\pi}\leftarrow\tau}(\mathfrak{T}(\varphi, \mathsf{last}(\tilde{\pi}))) > 0, && (2)\\
\mathfrak{T}(\varphi \wedge \varphi', \tau) &:= \mathfrak{T}(\varphi, \tau) \wedge \mathfrak{T}(\varphi', \tau), && (3)\\
\mathfrak{T}(\varphi \vee \varphi', \tau) &:= \mathfrak{T}(\varphi, \tau) \vee \mathfrak{T}(\varphi', \tau), && (4)\\
\mathfrak{T}(\varphi \,\mathsf{U}\, \varphi', \tau) &:= \mathfrak{T}(\varphi, \tau) \,\mathsf{U}\, \mathfrak{T}(\varphi', \tau), && (5)\\
\mathfrak{T}(\bigcirc\varphi, \tau) &:= \bigcirc\mathfrak{T}(\varphi, \tau), \text{ and} && (6)\\
\mathfrak{T}(\zeta, \tau) &:= \zeta, && (7)
\end{aligned}
$$

where $\mathsf{last}(\tilde{\pi})$ is the last element of $\tilde{\pi}$.

In short, the second argument of the transformation is used to keep track of the last drawn path variable, all syntactic elements apart from $\exists$ are taken over, while $\exists$ itself is mapped to a "> 0" assertion in rule 2, where it is also assumed that the largest applicable n is taken – that is, we never split a sequence of quantifiers across multiple steps of the transformation.

The most important addition in relation to Theorems 4.11 and 4.20 is that we take over the quantification context in rule 2 and use it to nest quantified formulae.

We want to prove

$$\mathcal{D}, \varepsilon \vDash \varphi \iff \mathcal{D}, \varepsilon \vDash \mathfrak{T}(\varphi, \varepsilon),$$

and start by noting that the semantics of U, $\bigcirc$, $\wedge$, and $\vee$, are the same across both logics. As such, we will only focus on rule 2. When we reach an instance of rule 2 starting from the above, and having collected a path assignment $\mathfrak{p}$ from previous steps, we map

$$\exists^n \tilde{\pi}.\varphi \quad \text{to} \quad \mathbb{P}_{\tilde{\pi}\leftarrow\tau}(\mathfrak{T}(\varphi, \mathsf{last}(\tilde{\pi}))) > 0,$$

where $\tau = \mathsf{last}(\mathfrak{p})$ is the last variable added to $\mathfrak{p}$, or ε if no such variable exists. As such, the wanted equivalence from above ultimately reduces to

$$\mathcal{D}, \mathfrak{p} \vDash \exists^n \tilde{\pi}.\varphi \iff \mathcal{D}, \mathfrak{p} \vDash \mathbb{P}_{\tilde{\pi}\leftarrow\tau}(\mathfrak{T}(\varphi, \mathsf{last}(\tilde{\pi}))) > 0.$$

Since the starting formula is closed, we have $\mathfrak{p}(\hat{\pi}) \neq \bot$, for all $\hat{\pi} \in \mathsf{free}(\exists^n \tilde{\pi}.\varphi)$. We continue via induction over the nesting depth of φ.

Induction Start. For nesting depth 0, φ is LTL^+, and $\exists^n \tilde{\pi}.\varphi$ is $[\exists^* | \mathsf{LTL}^+]$. The equivalence can be proven similarly to Theorem 4.20, by drawing paths from $\mathfrak{p}(\tau)(0)$, instead of the initial state.

Induction Hypothesis. Let $1 \leq n < \omega$ be given, such that, for all formulae (closed or not) $\varphi \in [\downarrow \exists^* | \pi^+]$-HyperCTL* with $\mathsf{nd}(\varphi) = n$,

$$\mathcal{D}, \mathfrak{p} \vDash \varphi \iff \mathcal{D}, \mathfrak{p} \vDash \mathfrak{T}(\varphi, \tau),$$

for all path assignments $\mathfrak{p}$, such that $\mathfrak{p}(\hat{\pi}) \neq \bot$, for all $\hat{\pi} \in \mathsf{free}(\exists^n \tilde{\pi}.\varphi)$, and $\tau = \mathsf{last}(\mathfrak{p})$ is the last variable that was added to $\mathfrak{p}$, or ε if no such variable exists.

Induction Step. Let $\exists^n \tilde{\pi}.\varphi$ be an $[\downarrow \exists^* | \pi^+]$ formula of depth $n+1$. We get

$$\begin{aligned}
\mathcal{D}, \mathfrak{p} \vDash \exists^n \tilde{\pi}.\varphi &\iff \exists \overline{\pi} \in \mathsf{Paths}_{\mathcal{D}}(\mathfrak{p}(\tau)(0))^n : \\
&\qquad \mathcal{D}, \mathfrak{p} \circ \{\hat{\pi}_i \mapsto \pi_i \mid i < n\} \vDash \varphi \\
&\overset{\text{IH}}{\iff} \exists \overline{\pi} \in \mathsf{Paths}_{\mathcal{D}}(\mathfrak{p}(\tau)(0))^n : \\
&\qquad \mathcal{D}, \mathfrak{p} \circ \{\hat{\pi}_i \mapsto \pi_i \mid i < n\} \vDash \mathfrak{T}(\varphi, \hat{\pi}_{n-1}) \\
&\iff \exists \overline{\pi} \in \mathsf{Paths}_{\mathcal{D}}(\mathfrak{p}(\tau)(0))^n : \\
&\qquad \mathcal{D}, \mathfrak{p} \circ \{\hat{\pi}_i \mapsto \pi_i \mid i < n\} \vDash \mathfrak{T}(\varphi, \mathsf{last}(\tilde{\pi})) \\
&\iff \Pr_{\mathcal{D}^n}\Big\{ \overline{\pi} \in \mathsf{Paths}_{\mathcal{D}}(\mathfrak{p}(\tau)(0))^n : \\
&\qquad\qquad \mathcal{D}, \mathfrak{p} \circ \{\hat{\pi}_i \mapsto \pi_i \mid i < n\} \\
&\qquad\qquad\qquad \vDash \mathfrak{T}(\varphi, \mathsf{last}(\tilde{\pi})) \Big\} > 0 \\
&\iff \mathcal{D}, \mathfrak{p} \vDash \mathbb{P}_{\tilde{\pi} \leftarrow \tau}(\mathfrak{T}(\varphi, \mathsf{last}(\tilde{\pi}))) > 0
\end{aligned}$$

□

4.5 Equivalent Fragments

In this chapter, we have mainly seen transformations that embed parts of $\mathsf{PHL}_{\mathrm{DTMC}}$ into HyperPCTL*. These transformations generate certain types of expressions in the latter, and can be inverted on these very types of expressions.

Theorem 4.7 creates HyperPCTL* expressions with in which

- no $\mathbb{P}$-nesting occurs, since $\mathsf{PHL}_{\mathrm{DTMC}}$ does not allow this at all,
- each probabilistic operator draws exactly one path starting from the initial state, and

– probability measures can be added to one another arbitrarily, but can only be multiplied with rational constants.

Let the fragment of HyperPCTL* that is comprised of these expressions be called its *draw-1 simple shallow* fragment, and be denoted by $[\mathbb{P}_1|\rho^s|\mathsf{LTL}^s]$. This fragment also excludes the usage of functions such as exponential or polynomial ones in HyperPCTL*. A grammar is given in Figure 10.

Here, *shallow* references that the content of $\mathbb{P}$ operators is reduced to singleton-marked LTL, and *simple* that functions are disallowed – with the implicit allowance of multivariate polynomials in $\mathbb{Q}$ of degree 1, that is, exactly those that can be represented by applying the rules $c \cdot \rho$ and $\rho + \rho$ finitely many times.

$$
\begin{array}{ll}
(\mathbb{P}_1 \textit{ formulae}) & \varphi ::= \varphi \wedge \varphi \mid \neg\varphi \mid \rho < \rho \\
(\rho^s \textit{ expressions}) & \rho ::= c \mid \rho + \rho \mid c \cdot \rho \mid \mathbb{P}_{\hat{\pi}}(\eta) \\
(\mathsf{LTL} \textit{ formulae}) & \eta ::= a_{\hat{\pi}} \mid \mathtt{true} \mid \eta \wedge \eta \mid \neg\eta \mid \bigcirc\eta \mid \eta \,\mathsf{U}\, \eta
\end{array}
$$

Figure 10: Grammar of $[\mathbb{P}_1|\rho^s|\mathsf{LTL}^s]$-HyperPCTL*

Theorem 4.25. $[\mathbb{P}_1|\rho^s|\mathsf{LTL}^s]$-HyperPCTL$^* \cong \mathsf{PHL}^{\mathrm{no}\,\vartheta}_{\mathrm{DTMC}}$.

Proof. "$\geqslant$" is the content of Theorem 4.7. "$\leqslant$" follows by using the same reasoning and inverting the steps of the transformation given in Theorem 4.7, specifically, simply by removing all $\hat{\pi}$-indices from the LTL formulae and $\hat{\pi}$-rules from $\mathbb{P}$ operators. □

Theorem 4.24 generates expressions with nesting, in which probability measures are only asserted to be nonzero, the only path variable that may be referenced by rules is the last one that was quantified over (last), and negation is only allowed in PL formulae. Let this fragment be named *recursively nonzero path-positive* and be denoted by $[\downarrow\mathbb{P}^{>0}_{\mathsf{last}}|\pi^+]$. A grammar is given in Figure 11.

$$
\begin{array}{ll}
(\downarrow\mathbb{P}^{>0}_{\mathsf{last}} \textit{ formulae}) & \varphi ::= \mathbb{P}_{\hat{\pi}\leftarrow\mathsf{last}}(\varphi) > 0 \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \eta \\
(\pi^+ \textit{ formulae}) & \eta ::= \varphi \,\mathsf{U}\, \varphi \mid \bigcirc\varphi \mid \zeta \\
(\mathsf{PL} \textit{ formulae}) & \zeta ::= \zeta \wedge \zeta \mid \neg\zeta \mid a_{\hat{\pi}} \mid \mathtt{true}
\end{array}
$$

Figure 11: Grammar of $[\downarrow\mathbb{P}^{>0}_{\mathsf{last}}|\pi^+]$-HyperPCTL*

Theorem 4.26. $[\downarrow\mathbb{P}^{>0}_{\mathsf{last}}|\pi^+]$-HyperPCTL$^* \cong [\downarrow\exists^*|\pi^+]$-HyperCTL*.

Proof. "$\geqslant$" is the content of Theorem 4.24. "$\leqslant$" follows by inverting the rules of the transformation given in it. □

4.6 Overview of Results

In this chapter, we first showed that $\mathsf{PHL}_{\mathrm{DTMC}}$ cannot express multivariate HyperPCTL* formulae in Section 4.1. Then, we embedded the probabilistic part of $\mathsf{PHL}_{\mathrm{DTMC}}$ in HyperPCTL* in Section 4.2, and subsequently focussed on HyperCTL*, starting with formulae with one quantifier in Section 4.3, and building up to multiple quantifiers, and quantifier nesting in Section 4.4. Finally, in Section 4.5, we identified equivalent fragments between the two logics by looking at the types of formulae that our transformations generate, and inverting them on these formulae.

In total, we saw that we can embed into HyperPCTL*

- the *entire* probabilistic part of $\mathsf{PHL}_{\mathrm{DTMC}}$: $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}}$ (Theorem 4.7), and
- the *recursively existential path-positive* fragment of HyperCTL* $[\downarrow\exists^*|\pi^+]$ (Theorem 4.24).

By looking at the forms of the formulae the given transformations generate, we found fragments of HyperPCTL* that we can embed in PHL:

- The *draw-1 simple shallow* fragment $[\mathbb{P}_1|\rho^s|\mathsf{LTL}^s]$ (Theorem 4.25)
- The *recursively nonzero path-positive* fragment $[\downarrow\mathbb{P}^{>0}_{\mathsf{last}}|\pi^+]$ (Theorem 4.26)

Moreover, in the process of examining HyperCTL*, we made it plausible in Section 4.3.2 that we cannot lift the *path-positive* modifier. That is, we cannot allow nested LTL, and subsequently nested path expressions, to be negated.

It still remains open, whether special cases of multivariate HyperPCTL* formulae are expressible in $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}}$, and whether the equivalence

$$[\downarrow\mathbb{P}^{>0}_{\mathsf{last}}|\pi^+]\text{-HyperPCTL}^* \cong [\downarrow\exists^*|\pi^+]\text{-HyperCTL}^*$$

of Theorem 4.26 represents the largest fragment equivalence between HyperPCTL* and HyperCTL* (excluding special cases). Specifically, we only saw that the most generic case of formulae with alternating quantifiers of HyperCTL* is not embeddable in HyperPCTL* in Section 4.4, basing this on Conjecture 4.13. Furthermore, we only examined the probabilistic and non-probabilistic parts of $\mathsf{PHL}_{\mathrm{DTMC}}$ *in isolation from one another*. It may be the case, that, when combined, they can express more parts of HyperPCTL* than just the fragments mentioned above.

Future work on this examination could further include proving (or disproving) the conjecture, and finding special cases of formulae with alternating quantifiers in HyperCTL* that have equivalents in HyperPCTL*.

Chapter 5

HyperPCTL vs. PHL on DTMCs

As we have noted in Chapter 3, HyperPCTL is inherently incomparable to PHL, due to the usage of initial states in the semantics of the latter, whilst the first has no intrinsic way of selecting said initial states. Moreover, both PHL's HyperCTL* fragment as well as HyperPCTL* have mechanisms to express that "a path be drawn starting at the initial state of another path," while HyperPCTL does not. This does not preclude it from *mimicking* this behaviour, however the procedure to do so – if any exists – is not obvious.

Therefore, in this chapter, we can, at most, examine algorithmic relations, which, in addition to transforming formulae also transform the DTMC itself, and more specifically ones that label specific states in a unique way.

To this end, we fix a basic marking algorithm $\mathcal{A}_{\mathfrak{m}}$ that takes a DTMC $\mathcal{D} := (S, s_\iota, p, \mathsf{AP}, l)$ with initial state s_ι and transforms it into $\mathcal{A}_{\mathfrak{m}}(\mathcal{D}) = (S, p, \mathsf{AP}_{\mathfrak{m}}, l_{\mathfrak{m}})$, with

- $\mathsf{AP}_{\mathfrak{m}} := \mathsf{AP} \cup \{\mathsf{init}\}$, where we assume that $\mathsf{init} \notin \mathsf{AP}$,
- $l_{\mathfrak{m}}(s) := \begin{cases} l(s), & \text{if } s \neq s_\iota, \\ l(s) \cup \{\mathsf{init}\}, & \text{if } s = s_\iota. \end{cases}$

This, essentially, gives HyperPCTL$_{\text{DTMC}}$ a way to select the initial state of $\mathcal{D}$ in $\mathcal{A}_{\mathfrak{m}}(\mathcal{D})$. Given a DTMC (S, p, AP, l) with a state $s \in S$ uniquely labelled $\mathsf{init} \in \mathsf{AP}$, we denote by $\mathcal{A}_{\mathfrak{m}}^{-1}$ the inverse of the algorithm $\mathcal{A}_{\mathfrak{m}}$, that generates $(S, s_\iota, p, \mathsf{AP}, l)$ with initial state $s_\iota := s$.

In the following, we will use $\mathcal{A}_{\mathfrak{m}}$ in conjunction with transformations of formulae to embed fragments of one logic into the other algorithmically. In any case, the constructs will be used orthogonally to one another; the transformation of DTMCs will always happen independently to that of formulae.

While this bridges a semantical gap between the logics, we assume that it does not expand the expressive power of HyperPCTL$_{\text{DTMC}}$ beyond letting it uniquely identify an initial state, and we explicitly do not use it for any other purpose than that.

Now, we can transfer some of the results of the previous chapter over, starting with the relation to PHL$^{\text{no}\vartheta}_{\text{DTMC}}$.

5.1 HyperCTL*-less PHL to HyperPCTL

In essence, we proceed similarly to Theorem 4.7. The idea is as follows. For an (unmarked) LTL formula η, the PHL expression $\mathbb{P}(\eta)$ evaluates equivalently (modulo $\mathcal{A}_{\mathfrak{m}}$) to the HyperPCTL expression $\mathbb{P}(\eta_{\hat{s}})$ *with* the initial assignment $\mathfrak{s} = (\hat{s} \mapsto s_\iota)$, where $\eta_{\hat{s}}$ is exactly the same as η, but with its atomics marked by $\hat{s}$. This assignment can be induced by a HyperPCTL sentence by starting it with $\exists \hat{s}.\ \mathsf{init}_{\hat{s}} \wedge \cdots$, for example

$$\exists \hat{s}.\ \mathsf{init}_{\hat{s}} \to \mathbb{P}(\eta_{\hat{s}}) > 0 \in \mathsf{HyperPCTL}_{\mathrm{DTMC}},$$

for

$$\mathbb{P}(\eta) > 0 \in \mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}}.$$

Nevertheless, we run against another problem: Purely in terms of syntax, LTL allows only direct nesting of U and $\bigcirc$ operators, while HyperPCTL requires that a probabilistic assertion be made in-between. A slight exception to this rule is the stacking of $\bigcirc$, since $\bigcirc^k \zeta \in \mathsf{LTL}$ can be expressed as $\mathtt{true}\ \mathsf{U}^{[k,k]}\ \zeta$ as a path expression of $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$. From now on, we consider $\bigcirc^k$, $k \geq 2$, to also be permissible in HyperPCTL as syntactic sugar.

Due to this, we first restrict ourselves to *shallow* LTL formulae, denoted LTL^s, which drop to a PL expression directly after U, or $\bigcirc^k$, and can *not* mix modal operators in a single expression. These make up a very small part of LTL, and thus do not contain particularly much expressive power within.

A grammar for fragment of $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}}$ restricted to shallow LTL expressions is given in Figure 12.

$$\begin{array}{ll}
(\textit{top-level formulae}) & \varphi ::= \varphi \wedge \varphi \mid \neg\varphi \mid \rho < \rho \\
(\textit{probabilistic expressions}) & \rho ::= \mathbb{P}(\eta) \mid \rho + \rho \mid c \cdot \rho \mid c \\
(\mathsf{LTL}^s\ \textit{formulae}) & \eta ::= \bigcirc^k \zeta \mid \zeta\ \mathsf{U}\ \zeta \\
(\mathsf{PL}\ \textit{formulae}) & \zeta ::= \zeta \wedge \zeta \mid \neg\zeta \mid a \mid \mathtt{true}
\end{array}$$

Figure 12: Grammar of $[\mathsf{LTL}^s]$-$\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}}$

Theorem 5.1. $[\mathsf{LTL}^s]$-$\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}} \preccurlyeq_{\mathcal{A}} \mathsf{HyperPCTL}_{\mathrm{DTMC}}$.

Proof. We give the following explicit transformation $\mathfrak{T}$. Let φ, φ' be be top-level formulae, ρ, ρ probabilistic expressions, $\eta \in \mathsf{LTL}^s$, and $\zeta, \zeta' \in \mathsf{PL}$, built as shown in Figure 12,

$c \in \mathbb{Q}$, $a \in \mathsf{AP}$, $\hat{s} \in \hat{S}$, and set

$$
\begin{aligned}
\mathfrak{T}(\varphi) &:= \exists \hat{s}.\, \mathsf{init}_{\hat{s}} \wedge \mathfrak{T}(\varphi, \hat{s}), && (1)\\
\mathfrak{T}(\varphi \wedge \varphi', \hat{s}) &:= \mathfrak{T}(\varphi, \hat{s}) \wedge \mathfrak{T}(\varphi', \hat{s}), && (2)\\
\mathfrak{T}(\neg\varphi, \hat{s}) &:= \neg\mathfrak{T}(\varphi, \hat{s}), && (3)\\
\mathfrak{T}(\rho < \rho', \hat{s}) &:= \mathfrak{T}(\rho, \hat{s}) < \mathfrak{T}(\rho', \hat{s}), && (4)\\
\mathfrak{T}(c \cdot \rho, \hat{s}) &:= \mathfrak{T}(c, \hat{s}) \cdot \mathfrak{T}(\rho, \hat{s}), && (5)\\
\mathfrak{T}(\rho + \rho', \hat{s}) &:= \mathfrak{T}(\rho, \hat{s}) + \mathfrak{T}(\rho', \hat{s}), && (6)\\
\mathfrak{T}(c, \hat{s}) &:= c, && (7)\\
\mathfrak{T}(\mathbb{P}(\eta)\,, \hat{s}) &:= \mathbb{P}(\mathfrak{T}(\eta, \hat{s}))\,, && (8)\\
\mathfrak{T}(\bigcirc^k \zeta, \hat{s}) &:= \bigcirc^k \mathfrak{T}(\zeta, \hat{s}), && (9)\\
\mathfrak{T}(\zeta \,\mathsf{U}\, \zeta', \hat{s}) &:= \mathfrak{T}(\zeta, \hat{s}) \,\mathsf{U}\, \mathfrak{T}(\zeta', \hat{s}), && (10)\\
\mathfrak{T}(\neg\zeta, \hat{s}) &:= \neg\mathfrak{T}(\zeta, \hat{s}), && (11)\\
\mathfrak{T}(\zeta \wedge \zeta', \hat{s}) &:= \mathfrak{T}(\zeta, \hat{s}) \wedge \mathfrak{T}(\zeta', \hat{s}), && (12)\\
\mathfrak{T}(a, \hat{s}) &:= a_{\hat{s}}, \text{ and} && (13)\\
\mathfrak{T}(\mathtt{true}, \hat{s}) &:= \mathtt{true}. && (14)
\end{aligned}
$$

The idea behind the transformation is laid out above, and in essence similar to Theorem 4.7.

- Rule 1 creates the wrapping expression, including the reference variable for the initial state.
- Rules 2 through 12, and rule 14 simply take over the syntactic elements of the original formula.
- Rule 13 marks atomic propositions in the original formula with the reference variable selected by rule 1.

We ultimately want to show

$$\mathcal{D} \vDash \varphi \iff \mathcal{A}_{\mathfrak{m}}(\mathcal{D}) \vDash \mathfrak{T}(\varphi),$$

and start by noting that the semantics of $\wedge$, $\neg$, $<$, $\cdot$, and $+$ are the same across both logics. Taking over the syntactic elements of $[\mathsf{LTL}^s]$-$\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}}$ creates a syntactically sound $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ formula. Furthermore, we only have one single quantification in the wrapper expression $\exists \hat{s}.\, \mathsf{init}_{\hat{s}} \wedge \cdots$, which induces the state assignment $\mathfrak{s} := (\hat{s} \mapsto s_\iota)$. With this, the above becomes

$$\mathcal{D} \vDash \varphi \iff \mathcal{A}_{\mathfrak{m}}(\mathcal{D}), \mathfrak{s} \vDash \mathfrak{T}(\varphi, \hat{s}).$$

We only need to make sure that probabilistic expressions in the original formula and their transformants evaluate equivalently (modulo $\mathcal{A}_{\mathfrak{m}}$) using the initial state

assignment above. First, we proceed to make sure LTL^{s} formulae "select" the same paths as their transformants. Let $\pi \in \mathsf{Paths}_{\mathcal{D}}$, and $\mathfrak{s} := (\hat{s} \mapsto \pi(0))$. We have the following cases.

- $\eta = \bigcirc^k \zeta$, $k \geq 1$. We get

$$\begin{aligned}
\mathcal{D}, \pi \vDash_{\mathsf{PHL}} \eta &\iff \mathcal{D}, \pi(k) \vDash_{\mathsf{PHL}} \zeta \\
&\iff \mathcal{D}, \mathfrak{s}[\pi(k)] \vDash_{\mathsf{HyperPCTL}} \mathfrak{T}(\zeta, \hat{s}) \\
&\iff \mathcal{D}, \mathfrak{s}, \pi \vDash_{\mathsf{HyperPCTL}} \bigcirc^k \mathfrak{T}(\zeta, \hat{s}) \\
&\iff \mathcal{D}, \mathfrak{s}, \pi \vDash_{\mathsf{HyperPCTL}} \mathfrak{T}(\eta, \hat{s}).
\end{aligned}$$

- $\eta = \zeta \,\mathsf{U}\, \zeta'$. We have

$$\begin{aligned}
\mathcal{D}, \pi \vDash_{\mathsf{PHL}} \eta &\iff \exists j < \omega \; \forall i < j : \mathcal{D}, \pi(i) \vDash_{\mathsf{PHL}} \zeta \\
&\qquad\qquad \wedge \mathcal{D}, \pi(j) \vDash_{\mathsf{PHL}} \zeta' \\
&\iff \exists j < \omega \; \forall i < j : \mathcal{D}, \mathfrak{s}[\pi(i)] \vDash_{\mathsf{HyperPCTL}} \mathfrak{T}(\zeta, \hat{s}) \\
&\qquad\qquad \wedge \mathcal{D}, \mathfrak{s}[\pi(j)] \vDash_{\mathsf{HyperPCTL}} \mathfrak{T}(\zeta', \hat{s}) \\
&\iff \mathcal{D}, \mathfrak{s}, \pi \vDash \mathfrak{T}(\eta, \hat{s}).
\end{aligned}$$

Using this, we compute

$$\begin{aligned}
[\![\mathbb{P}(\eta)]\!]^{\mathsf{PHL}}_{\mathcal{D}} &= \Pr\left\{\pi \in \mathsf{Paths}_{\mathcal{D}}(s_\iota) \;\middle|\; \mathcal{D}, \pi \vDash_{\mathsf{PHL}} \eta\right\} \\
&= \Pr\left\{\pi \in \mathsf{Paths}_{\mathcal{D}}(\mathsf{im}(\mathfrak{s})) \;\middle|\; \mathcal{D}, \mathfrak{s}, \pi \vDash_{\mathsf{HyperPCTL}} \mathfrak{T}(\eta, \hat{s})\right\} \\
&= [\![\mathbb{P}(\mathfrak{T}(\eta, \hat{s}))]\!]^{\mathsf{HyperPCTL}}_{\mathcal{D},\,\mathfrak{s}},
\end{aligned}$$

which concludes the proof. □

There are still other formulae of $\mathsf{PHL}_{\mathrm{DTMC}}$ that are expressible in $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$, even though they are syntactically incompatible with it at a first glance.

Lemma 5.2. Let $\zeta \in \mathsf{PL}$, and $\zeta_{\hat{s}}$ be the exact same formula with its atomics marked by $\hat{s}$. Then, for $c \in \mathbb{Q}$, and $\sim \in \{<, \leq, =, \geq, >\}$.

$$\underbrace{\mathbb{P}(\Box\Diamond\zeta) \sim c}_{\mathsf{PHL}} \quad \equiv_{\mathsf{f}} \quad \underbrace{\exists \hat{s}.\, \mathsf{init}_{\hat{s}} \wedge \mathbb{P}\Big(\Diamond\mathbb{P}\big(\Box\mathbb{P}(\Diamond\zeta_{\hat{s}}) = 1\big) = 1\Big) \sim c}_{\mathsf{HyperPCTL}} \quad (\text{modulo } \mathcal{A}_{\mathfrak{m}})$$

$$\underbrace{\mathbb{P}(\Diamond\Box\zeta) \sim c}_{\mathsf{PHL}} \quad \equiv_{\mathsf{f}} \quad \underbrace{\exists \hat{s}.\, \mathsf{init}_{\hat{s}} \wedge \mathbb{P}\Big(\Diamond\mathbb{P}\big(\Box\zeta_{\hat{s}}\big) = 1\Big) \sim c}_{\mathsf{HyperPCTL}} \quad (\text{modulo } \mathcal{A}_{\mathfrak{m}})$$

Proof. We exemplarily only show the first equivalence, and for the base case for $\zeta = a \in \mathsf{AP}$. It is assumed known that HyperPCTL subsumes PCTL modulo $\mathcal{A}_{\mathfrak{m}}$, since the

former contains the syntax and semantics of the latter. Based on this observation, it holds (mod. $\mathcal{A}_{\mathfrak{m}}$) that

$$\underbrace{\mathbb{P}\Big(\Diamond\mathbb{P}\big(\Box\mathbb{P}(\Diamond a) = 1\big) = 1\Big) \sim c}_{\text{PCTL}} \quad \equiv_{\mathfrak{f}} \quad \underbrace{\exists \hat{s}.\ \mathsf{init}_{\hat{s}} \wedge \mathbb{P}\Big(\Diamond\mathbb{P}\big(\Box\mathbb{P}(\Diamond\zeta_{\hat{s}}) = 1\big) = 1\Big) \sim c}_{\text{HyperPCTL}}$$

Furthermore, it is proven in [BK08] that the PCTL formula on the left holds on a *finite* DTMC $\mathcal{D}$ at its initial state s_ι, iff

$$\Pr\left\{\pi \in \mathsf{Paths}_{\mathcal{D}}(s_\iota) \;\middle|\; \pi \vDash \underbrace{\Box\Diamond a}_{\text{LTL}}\right\} \sim c,$$

whence we immediately get the first equivalence, since this is exactly the evaluation of the PHL formula on $\mathcal{D}$. The second equivalence can be shown in a similar fashion, again with the aid of [BK08]. □

5.2 HyperCTL* Sentences in PHL

In this part, we look again at the non-probabilistic part of $\mathsf{PHL}_{\mathrm{DTMC}}$ separately. Purely in terms of syntax, HyperCTL* allows arbitrary nesting of U and ◯ expressions, while HyperPCTL does not.

The former has, nonetheless, a fragment with strictly alternating quantification and path-expression nesting – essentially HyperCTL – but even this is not easily compatible with HyperPCTL; in HyperCTL*, nested quantifications range over subtrees of paths, whereas quantification in HyperPCTL has no effect on its own on where paths themselves start.

This leads us directly to the thought of embedding the (starkly restricted) *PNF-existential LTL-shallow* fragment of HyperCTL*, denoted $[\exists^*|\mathsf{LTL}^s]$, into HyperPCTL. A grammar is presented in Figure 13.

($\exists^*$ *formulae*)	$\varphi ::= \exists\hat{\pi}.\varphi \mid \eta$
(LTL^s *formulae*)	$\eta ::= \bigcirc^k\zeta \mid \zeta \,\mathsf{U}\, \zeta$
(PL *formulae*)	$\zeta ::= \zeta \wedge \zeta \mid \neg\zeta \mid a_{\hat{\pi}} \mid \mathtt{true}$

Figure 13: Grammar of $[\exists^*|\mathsf{LTL}^s]$-HyperCTL*

Theorem 5.3. $[\exists^*|\mathsf{LTL}^s]$-HyperCTL* $\preccurlyeq_{\mathcal{A}}$ HyperPCTL.

Proof. Let ψ, ψ' be non-quantified formulae, $\zeta, \zeta' \in \mathsf{PL}$, built as shown in Figure 13, $a \in \mathsf{AP}$, $n < \omega$. We assume without loss of generality that $\hat{\Pi} = \{\hat{\pi}_0, \hat{\pi}_1, \ldots\}$, and $\hat{S} = \{\hat{s}_0, \hat{s}_1, \ldots\}$, and consider the following transformation.

$$\mathfrak{T}(\exists \hat{\pi}_{n-1} \ldots \exists \hat{\pi}_0 . \eta) := \exists \hat{s}_{n-1} \ldots \exists \hat{s}_0 . \bigwedge_{i<n} \mathsf{init}_{\hat{s}_i} \wedge \mathbb{P}\big(\mathfrak{T}(\eta)\big) > 0, \tag{1}$$

$$\mathfrak{T}(\bigcirc^k \zeta) := \bigcirc^k \mathfrak{T}(\zeta), \tag{2}$$

$$\mathfrak{T}(\zeta \,\mathsf{U}\, \zeta') := \mathfrak{T}(\zeta) \,\mathsf{U}\, \mathfrak{T}(\zeta'), \tag{3}$$

$$\mathfrak{T}(\neg \zeta) := \neg \mathfrak{T}(\zeta), \tag{4}$$

$$\mathfrak{T}(\zeta \wedge \zeta') := \mathfrak{T}(\zeta) \wedge \mathfrak{T}(\zeta'), \tag{5}$$

$$\mathfrak{T}(a_{\hat{\pi}_i}) := a_{\hat{s}_i}, \text{ and} \tag{6}$$

$$\mathfrak{T}(\mathtt{true}) := \mathtt{true}. \tag{7}$$

In the above, it is assumed that the n in rule 1 is the maximum applicable n for the formula, that is, we always convert *all* quantifiers at the start of the formula in one step.

- Rule 1 maps sequences of quantified path variables to sequences of quantified state variables, asserts that the computation trees bound to the new variables are all rooted at the initial state, and wraps the rest of the formula in a nonzero assertion.
- Rules 2 though 5, and rule 7, take over the syntactic elements of the original formula.
- Rule 6 replaces the path variable markings $\hat{\pi}_i$ with state variable markings $\hat{s}_i$.

The equivalence can be shown similarly to Corollary 4.21. □

At this point we surmise that a version of the above that also allows recursion of $[\exists^*|\mathsf{LTL}^s]$ formulae, the *recursively* PNF*-existential LTL-shallow* fragment of HyperCTL*, denoted $[\downarrow\exists^*|\mathsf{LTL}^s]$, is still embeddable in HyperPCTL. A grammar is given in Figure 14.

($\downarrow\exists^*$ *formulae*)	$\varphi ::= \exists \hat{\pi} . \varphi \mid \eta$
(LTL^s *formulae*)	$\eta ::= \bigcirc^k \psi \mid \psi \,\mathsf{U}\, \psi \mid \zeta$
(PL *formulae*)	$\zeta ::= \zeta \wedge \zeta \mid \neg \zeta \mid a_{\hat{\pi}} \mid \mathtt{true}$

Figure 14: Grammar of $[\downarrow\exists^*|\mathsf{LTL}^s]$-HyperCTL*

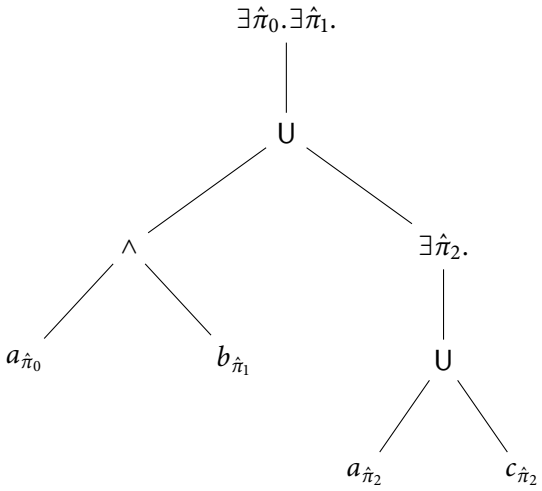
Conjecture 5.4. $[\downarrow\exists^*|\mathsf{LTL}^s]$-HyperCTL$^* \preccurlyeq_{\mathcal{A}}$ HyperPCTL. △

A complete transformation will not be given here, but rather just a sketch of how one could work, specifically for the case that each nested $\exists$ formula is closed.

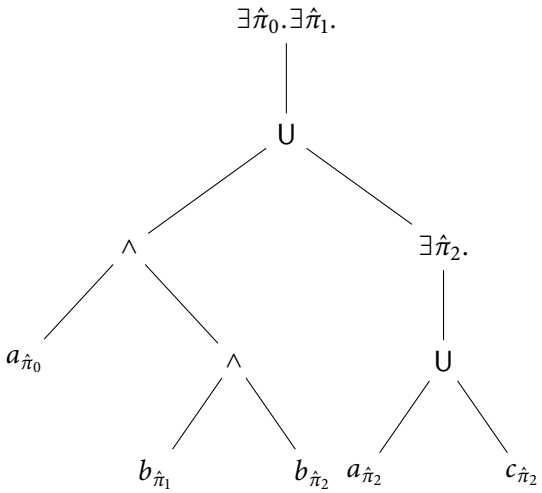
Example 5.5. Let $\varphi := \exists \hat{\pi}_0.\exists \hat{\pi}_1.\left(a_{\hat{\pi}_0} \wedge b_{\hat{\pi}_1}\right) \mathsf{U} \left(\exists \hat{\pi}_2.a_{\hat{\pi}_2} \mathsf{U}\, c_{\hat{\pi}_2}\right)$.

This formula asserts that we can find paths π_0, π_1 starting at the initial state, such that π_0 has an initial segment $\pi_{0\text{pre}}$ marked a, and π_1 has in parallel an initial segment at least as long as $\pi_{0\text{pre}}$ marked b, such that we can branch off of π_1 at some point $i < \omega$, and take a path π_2 staring at $\pi_1(i)$, on which $a \mathsf{U}\, c$ holds.

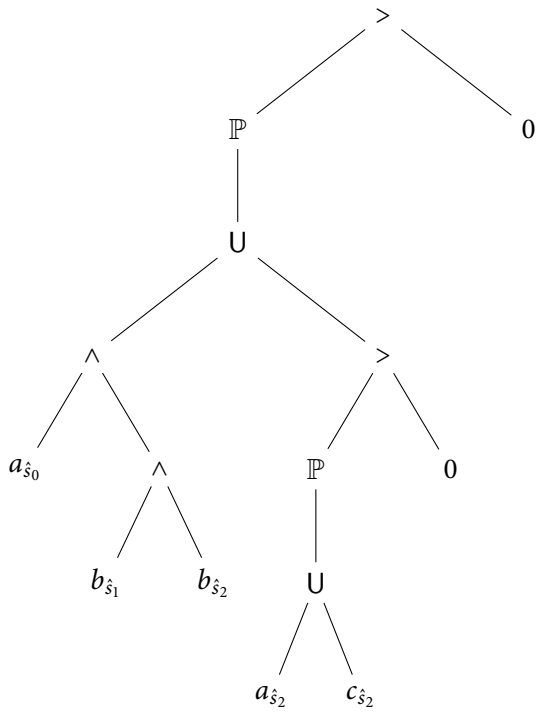
The syntax tree of φ looks as follows.



Here, we kept $\exists \hat{\pi}_0.\exists \hat{\pi}_1.$ together as one node to simplify the diagram. Now, we look at the leaves descending from the U-node *before* the nested formula. We want $\hat{\pi}_2$ to mimic the behaviour of the path that comes before it – $\hat{\pi}_1$. Hence, we replace the leaf $b_{\hat{\pi}_1}$ with the syntax tree of $b_{\hat{\pi}_1} \wedge b_{\hat{\pi}_2}$.



Next, we replace the quantifier sequences with $\mathbb{P}(\cdot) > 0$, and path variables $\hat{\pi}_i$ with state variables $\hat{s}_i$ in one step, and get



Collapsing this tree yields the expression

$$\mathbb{P}\Big(\big(a_{\hat{s}_0} \wedge b_{\hat{s}_1} \wedge b_{\hat{s}_2}\big) \,\mathsf{U}\, \big(\mathbb{P}(a_{\hat{s}_2} \,\mathsf{U}\, c_{\hat{s}_2}) > 0\big)\Big) > 0,$$

and with this, we build the HyperPCTL formula

$$\exists \hat{s}_0.\exists \hat{s}_1.\exists \hat{s}_2.\ \mathsf{init}_{\hat{s}_0} \wedge \mathsf{init}_{\hat{s}_1} \wedge \mathsf{init}_{\hat{s}_2} \wedge \mathbb{P}\Big(\big(a_{\hat{s}_0} \wedge b_{\hat{s}_1} \wedge b_{\hat{s}_2}\big) \,\mathsf{U}\, \big(\mathbb{P}(a_{\hat{s}_2} \,\mathsf{U}\, c_{\hat{s}_2}) > 0\big)\Big) > 0.$$

In essence, this makes computation tree that is bound to $\hat{s}_2$ behave the same as the one bound to $\hat{s}_1$ before the nested formula. This *imitates* the behaviour of nested quantification in existential formulae of HyperPCTL*; instead of explicitly drawing $\hat{\pi}_2$ at the current position of $\hat{\pi}_1$ at some point later down the line, we implicitly "bind" the behaviour of $\hat{\pi}_2$ to that of $\hat{\pi}_1$ and draw them both at the start.

Indeed, the generated HyperPCTL formula draws 3 paths at the initial state, finds one π_0 with an initial segment $\pi_{0\text{pre}}$ labelled a, and a pair (π_1, π_2) with initial segments at least as long as $\pi_{0\text{pre}}$ labelled b, until at some point $i < \omega$, π_2^i models $a \,\mathsf{U}\, c$.

In the context of the original HyperCTL* formula φ, selecting $\hat{\pi}_0 \mapsto \pi_0$, $\hat{\pi}_1 \mapsto \pi_2$, and $\hat{\pi}_2 \mapsto \pi_2^i$ creates a satisfying assignment.

Conversely, if paths π_0, π_1, π_2 exist that form a satisfying path assignment for φ, with π_0, π_1 starting at s_ι, and π_2 starting at $\pi_1(i)$, for some $i < \omega$, then π_2 is measured nonzero by the nested $\mathbb{P}$ operator, since $a \mathsf{U} c$ is nondivergent (compare Lemma 4.19), and subsequently, the triple (π_0, π_1, π_1) is measured nonzero by the outer $\mathbb{P}$ operator.△

The procedure we can extrapolate from this example is the following. Let φ be a $[\downarrow\exists^*|\mathsf{LTL}^s]$ formula of HyperCTL*, such that each nested $[\downarrow\exists^*|\mathsf{LTL}^s]$ subformula is closed, and a supply of state variables $\hat{S} = \{\hat{s}_0, \hat{s}_1, \dots\}$ be given. Assume without loss of generality that $\mathrm{var}(\varphi) = \{\hat{\pi}_0, \dots, \hat{\pi}_n\}$, $n < \omega$, and that variable quantification in φ follows the order of the subscripts.

(i) Build the syntax tree of φ.

(ii) For each nested $[\downarrow\exists^*|\mathsf{LTL}^s]$ formula ψ, let $\hat{\pi}_j$ be the first variable quantified in ψ, $j \geq 1$, and do the following.

  (a) Iterate over the nodes on the path from the root to the start of ψ, excluding the root and ψ themselves.

  (b) Replace each leaf that descends from these nodes and that is marked with $\hat{\pi}_{j-1}$ – i.e. marked atomic proposition $a_{\hat{\pi}_{j-1}}$, with $a \in \mathsf{AP}$ – with the syntax tree of

$$a_{\hat{\pi}_{j-1}} \wedge \bigwedge_{\hat{\pi} \in \mathrm{var}(\psi)} a_{\hat{\pi}},$$

  This step implicitly excludes the leaves of ψ itself, since we required it to be closed, and has the effect of making all variables in ψ mimic the behaviour of the path assigned to $\hat{\pi}_{j-1}$ before the context of ψ.

(iii) Replace sequences of quantifiers on the tree with $\mathbb{P}(\cdot) > 0$.

(iv) Replace each $\hat{\pi}_i$ with a unique $\hat{s}_i$, $i \leq n$.

(v) Collapse the tree into an expression φ'.

(vi) Build

$$\exists \hat{s}_0 \cdots \exists \hat{s}_n. \bigwedge_{i \leq n} \mathsf{init}_{\hat{s}_i} \wedge \varphi' \in \mathsf{HyperPCTL}.$$

We have already examined universal formulae, and formulae with alternating quantifiers, in Chapter 4 when comparing HyperCTL* with HyperPCTL* and seen that most, apart from a few special cases, are not embeddable in the latter. Since $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ is weakly subsumed by HyperPCTL* [Wan+21], we also cannot embed these classes into $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ without considering more complex algorithms than our simple marking one $\mathcal{A}_{\mathfrak{m}}$. This procedure would, however, escape the purpose of this thesis.

As a closing note on this examination, the aforementioned exceptions for universal quantification seen in Lemmata 4.9(i) and 4.10(ii) can be transferred over to $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$.

To execute this, we first declare the following syntactic sugar: If φ is a $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ state formula (possibly not closed), set

$$\mathbb{P}(\Box\varphi) := 1 - \mathbb{P}(\Diamond\neg\varphi)\,.^{\mathrm{i}}$$

Lemma 5.6. Let $\zeta \in \mathsf{PL}$, and let $\zeta_{\hat{s}}, \zeta_{\hat{\pi}}$ represent the exact same same formula with its atomics marked by $\hat{s}$ and $\hat{\pi}$ respectively. The following equivalences hold.

$$\underbrace{\forall\hat{\pi}.\,\Box\zeta_{\hat{\pi}}}_{\mathsf{HyperCTL}^*} \quad \equiv_{\mathsf{f}} \quad \underbrace{\exists\hat{s}.\,\mathsf{init}_{\hat{s}} \wedge \mathbb{P}(\Box\zeta_{\hat{s}}) = 1}_{\mathsf{HyperPCTL}_{\mathrm{DTMC}}} \quad (\text{modulo } \mathcal{A}_{\mathfrak{m}})$$

$$\underbrace{\forall\hat{\pi}.\,\bigcirc\zeta_{\hat{\pi}}}_{\mathsf{HyperCTL}^*} \quad \equiv_{\mathsf{f}} \quad \underbrace{\exists\hat{s}.\,\mathsf{init}_{\hat{s}} \wedge \mathbb{P}(\bigcirc\zeta_{\hat{\pi}}) = 1}_{\mathsf{HyperPCTL}_{\mathrm{DTMC}}} \quad (\text{modulo } \mathcal{A}_{\mathfrak{m}})$$

Proof. Similar to Lemmata 4.9(i) and 4.10(ii). □

5.3 Equivalent Fragments

Similarly to Section 4.5, we will now look at what kind of formulae our transformations generate, in order to identify equivalent fragments between the logics.

Theorem 5.1 generates HyperPCTL formulae with the following constraints.

- They begin with the wrapping expression

  $$\exists\hat{s}.\,\mathsf{init}_{\hat{s}} \wedge \cdots,$$

- Probabilistic expressions may *not* be nested, and may *only* be multiplied with rational constants.
- The only usage of the syntactic rule $\varphi\ \mathsf{U}^{[k_1,k_2]}\ \varphi$ that occurs is $\mathtt{true}\ \mathsf{U}^{[k,k]}\ \varphi$, which stems from stacking $\bigcirc$ operators inside shallow LTL expressions in PHL formulae.

We call probabilistic expressions with only constant multiplication *simple*. Since we added stacking $\bigcirc$ operators as syntactic sugar to HyperPCTL, we ignore the rule in the last item altogether here. We name the part of HyperPCTL induced by these constraints its *1-initial-path simple LTL-shallow* fragment, and denote it by $[\mathsf{init}^1|\rho^s|\mathsf{LTL}^s]$. A grammar is given in Figure 15.

Theorem 5.7. $[\mathsf{init}^1|\rho^s|\mathsf{LTL}^s]\text{-}\mathsf{HyperPCTL} \cong_{\mathcal{A}} [\mathsf{LTL}^s]\text{-}\mathsf{PHL}^{\mathrm{no}\,\vartheta}_{\mathrm{DTMC}}$.

[i] The "naïve" definition $\mathbb{P}(\Box\varphi) = \mathbb{P}(\neg(\mathtt{true}\ \mathsf{U}\ \neg\varphi))$ is incompatible with HyperPCTL syntax.

(init^1 *formulae*)	$\varphi ::= \exists \hat{s}.\ \mathsf{init}_{\hat{s}} \wedge \psi$
(*non-quantified formulae*)	$\psi ::= \psi \wedge \psi \mid \neg\psi \mid \rho < \rho$
(ρ^s *expressions*)	$\rho ::= \mathbb{P}(\eta) \mid \rho + \rho \mid c \cdot \rho \mid c$
(LTL^s *formulae*)	$\eta ::= \bigcirc^k \zeta \mid \zeta \,\mathsf{U}\, \zeta$
(PL *formulae*)	$\zeta ::= \zeta \wedge \zeta \mid \neg\zeta \mid a_{\hat{s}} \mid \mathtt{true}$

Figure 15: Grammar of $[\mathsf{init}^1|\rho^s|\mathsf{LTL}^s]$-HyperPCTL

Proof. The subsumption "$\succcurlyeq_{\mathcal{A}}$" is the content of Theorem 5.1. The reverse direction follows by a similar argument to that of the referenced theorem. A transformation is given by discarding the wrapper "$\exists \hat{s}.\ \mathsf{init}_{\hat{s}} \wedge \cdots$", taking over every other syntactic element, and finally removing the state markings on atomic propositions. □

Theorem 5.3 creates existential formulae that

- have the form $\exists \hat{s}_{n-1} \cdots \exists \hat{s}_0.\ \bigwedge_{i<n} \mathsf{init}_{\hat{s}_i} \wedge \psi$, for a quantifier-free ψ,
- have *no arithmetic* between probabilistic expressions,
- can only assert probabilities to be *nonzero*, and
- can only use *shallow* LTL expressions.

Let this fragment of HyperPCTL be called *initial-path nonzero LTL-shallow* fragment, and be denoted $[\mathsf{init}^*|\mathbb{P}^{>0}|\mathsf{LTL}^s]$. A grammar is presented in Figure 16.

(init^* *formulae*)	$\varphi ::= \exists \hat{s}_{n-1} \cdots \exists \hat{s}_0.\ \bigwedge_{i<n} \mathsf{init}_{\hat{s}_i} \wedge \mathbb{P}(\eta) > 0$
(LTL^s *formulae*)	$\eta ::= \bigcirc^k \zeta \mid \zeta \,\mathsf{U}\, \zeta$
(PL *formulae*)	$\zeta ::= \zeta \wedge \zeta \mid \neg\zeta \mid a_{\hat{s}} \mid \mathtt{true}$

Figure 16: Grammar of $[\mathsf{init}^*|\mathbb{P}^{>0}|\mathsf{LTL}^s]$-HyperPCTL

Theorem 5.8. $[\mathsf{init}^*|\mathbb{P}^{>0}|\mathsf{LTL}^s]$-HyperPCTL $\cong_{\mathcal{A}}$ $[\exists^*|\mathsf{LTL}^s]$-HyperCTL*

Proof. The subsumption "$\succcurlyeq_{\mathcal{A}}$" is the subject of Theorem 5.3. The reverse direction follows by inverting the rules of the transformation given in it. □

5.4 Overview of Results

In this chapter, we started by comparing the syntactically compatible fragments of both logics. We established that we can embed

- the *LTL-shallow* fragment of $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}}$: $[\mathsf{LTL}^s]$ (Theorem 5.1), and
- the *PNF-existential LTL-shallow* fragment of HyperCTL*: $[\exists^*|\mathsf{LTL}^s]$ (Theorem 5.3)

into HyperPCTL, where we relied on the marking algorithm $\mathcal{A}_{\mathrm{m}}$ to give us access to the initial state in the latter. By identifying the type of HyperPCTL formulae the transformation in the aforementioned theorems generate, we extrapolated the following fragments of HyperPCTL that we can embed into PHL:

- The *1-initial path simple LTL-shallow* fragment $[\mathsf{init}^1|\rho^s|\mathsf{LTL}^s]$ (Theorem 5.7)
- The *initial-path nonzero LTL-shallow* fragment $[\mathsf{init}^*|\mathbb{P}^{>0}|\mathsf{LTL}^s]$ (Theorem 5.8)

Furthermore, we found special cases of formulae in PHL that are syntactically incompatible with HyperPCTL at a first glance, but still translatable from the former to the latter. Specifically, we have proven the following 2 equivalences (modulo $\mathcal{A}_{\mathrm{m}}$) in Lemma 5.2.

$$\underbrace{\mathbb{P}(\Box\Diamond\zeta) \sim c}_{\mathsf{PHL}} \quad \equiv_{\mathsf{f}} \quad \underbrace{\exists \hat{s}.\ \mathsf{init}_{\hat{s}} \wedge \mathbb{P}\Big(\Diamond\mathbb{P}\big(\Box\mathbb{P}(\Diamond\zeta_{\hat{s}}) = 1\big) = 1\Big) \sim c}_{\mathsf{HyperPCTL}}$$

$$\underbrace{\mathbb{P}(\Diamond\Box\zeta) \sim c}_{\mathsf{PHL}} \quad \equiv_{\mathsf{f}} \quad \underbrace{\exists \hat{s}.\ \mathsf{init}_{\hat{s}} \wedge \mathbb{P}\Big(\Diamond\mathbb{P}\big(\Box\zeta_{\hat{s}}\big) = 1\Big) \sim c}_{\mathsf{HyperPCTL}}$$

In Lemma 5.6, we have also proven the following special cases.

$$\underbrace{\forall \hat{\pi}.\ \Box\zeta_{\hat{\pi}}}_{\mathsf{HyperCTL}^*} \quad \equiv_{\mathsf{f}} \quad \underbrace{\exists \hat{s}.\ \mathsf{init}_{\hat{s}} \wedge \mathbb{P}(\Box\zeta_{\hat{s}}) = 1}_{\mathsf{HyperPCTL}_{\mathrm{DTMC}}}$$

$$\underbrace{\forall \hat{\pi}.\ \bigcirc\zeta_{\hat{\pi}}}_{\mathsf{HyperCTL}^*} \quad \equiv_{\mathsf{f}} \quad \underbrace{\exists \hat{s}.\ \mathsf{init}_{\hat{s}} \wedge \mathbb{P}(\bigcirc\zeta_{\hat{\pi}}) = 1}_{\mathsf{HyperPCTL}_{\mathrm{DTMC}}}$$

Lastly, in Conjecture 5.4, we postulated that we can expand the equivalence of Theorem 5.8 to $[\downarrow\exists^*|\mathsf{LTL}^s]$, which allows arbitrary nesting of $[\exists^*|\mathsf{LTL}^s]$ formulae, and gave an example of a scheme to translate certain types of $[\downarrow\exists^*|\mathsf{LTL}^s]$ formulae to HyperPCTL.

We left the question of whether there are more formulae of PHL that are syntactically incompatible with, but still translatable to HyperPCTL, open. Furthermore, we did not examine whether we can expand the special cases of Lemmata 5.2 and 5.6 with nesting.

Future work on this part could include answering these questions, and examining Conjecture 5.4, possibly finding an algorithm that can translate all $[\downarrow\exists^*|\mathsf{LTL}^s]$-HyperCTL* formulae to HyperPCTL.

Chapter 6

HyperPCTL vs. HyperPCTL* on DTMCs

While the thesis primarily focuses on the relation of HyperPCTL and HyperPCTL* to PHL on DTMCs, we will briefly go over the relations between fragments of the first two that are either obvious, or ones we can extrapolate from our arguments in Chapters 4 and 5. It is assumed known from [Wan+21] that HyperPCTL $<_{\mathcal{A}}$ HyperPCTL*.

The naming HyperPCTL* might create the false impression that it has the same relation to HyperPCTL that CTL* has to CTL, or the one that PCTL* has to PCTL. That is, it might create the impression that it is an *expansion* of HyperPCTL, with the requirement for alternation between state and path formulae lifted.

Expanding on Theorem 3.5, here, it will be made plausible that the fragment of HyperPCTL* with this very restriction artificially introduced to it can be weakly mapped to a strict subset of HyperPCTL.

First, consider the restricted fragment of HyperPCTL* that has the strict PCTL-style alternation between path formulae and path expressions, can only draw paths from the initial state, and has no functions. Let $[\mathbb{P}_\varepsilon \uparrow\downarrow \pi]$ denote the *draw-ε state-path alternating* fragment of HyperPCTL*, where $\uparrow\downarrow$ stands for alternation. A grammar is given in Figure 17.

$$
\begin{aligned}
&(\textit{path formulae}) && \varphi ::= \varphi \wedge \varphi \mid \neg\varphi \mid \rho < \rho \mid a_{\hat{\pi}} \mid \texttt{true} \\
&(\mathbb{P}_\varepsilon \textit{ expressions}) && \rho ::= \rho + \rho \mid \rho \cdot \rho \mid c \mid \mathbb{P}_{\hat{\pi} \leftarrow \varepsilon}(\eta) \\
&(\textit{path expressions}) && \eta ::= \varphi \,\mathsf{U}\, \varphi \mid \bigcirc\varphi \mid \varphi \,\mathsf{U}^{[k_1,k_2]}\, \varphi
\end{aligned}
$$

Figure 17: Grammar of $[\mathbb{P}_\varepsilon \uparrow\downarrow \pi]$-HyperPCTL*

The operator $\mathsf{U}^{[k_1,k_2]}$ can be constructed recursively in HyperPCTL* as follows.

$$
\varphi \,\mathsf{U}^{[k_1,k_2]}\, \varphi' := \begin{cases} \texttt{false}, & \text{if } k_1 > k_2 \\ \varphi', & \text{if } k_1 = k_2 = 0 \\ \varphi' \vee \Big(\varphi \wedge \bigcirc\big(\varphi \,\mathsf{U}^{[0,k_2-1]}\, \varphi'\big)\Big), & \text{if } k_1 = 0 \wedge k_2 > 0 \\ \varphi \wedge \bigcirc\big(\varphi \,\mathsf{U}^{[k_1-1,k_2-1]}\, \varphi'\big), & \text{if } k_1 > 0 \wedge k_2 > 0 \end{cases}
$$

Let $[\mathsf{init}^*]$-HyperPCTL be the fragment of HyperPCTL that is comprised of formulae of the form

$$\exists \hat{s}_{n-1} \cdots \exists \hat{s}_0. \bigwedge_{i<n} \mathsf{init}_{\hat{s}_i} \wedge \psi,$$

for $n < \omega$, and a *quantifier-free* ψ. We call this fragment simply the *initial-paths* fragment of HyperPCTL.

Theorem 6.1. $[\mathbb{P}_\varepsilon \uparrow\downarrow \pi]$-HyperPCTL$^* \cong_{\mathcal{A}} [\mathsf{init}^*]$-HyperPCTL

Proof (Sketch). A transformation from the former to the latter is given by assigning a unique $\hat{s}_i$ to each variable $\hat{\pi}_i$ of a given $[\mathbb{P}_\varepsilon \uparrow\downarrow \pi]$-HyperPCTL* formula φ, replacing all occurences of $\hat{\pi}_i$ with $\hat{s}_i$, for all i, to get an expression ψ, and building the $[\mathsf{init}^*]$-HyperPCTL formula

$$\varphi' := \exists \hat{s}_{n-1} \cdots \exists \hat{s}_0. \bigwedge_{i<n} \mathsf{init}_{\hat{s}_i} \wedge \psi,$$

with $\varphi \equiv \varphi'$. For the reverse direction, revert the steps of this transformation. □

Lifting the requirement to only draw paths starting at the initial state in the $[\mathbb{P}_\varepsilon \uparrow\downarrow \pi]$ fragment of HyperPCTL* while preserving its embeddability in HyperPCTL seems to also be possible via a procedure similar to the one given for Conjecture 5.4. Let $[\mathbb{P} \uparrow\downarrow \pi]$-HyperPCTL* denote the fragment generated by the grammar of Figure 17, with the amended rule

$$(\textit{probabilistic expressions})\ \rho ::= \rho + \rho \mid \rho \cdot \rho \mid c \mid \mathbb{P}_{\overline{\kappa}}(\eta)\,,$$

where $\overline{\kappa}$ is an *arbitrary ruleset*. In conclusion of this short excursus, we surmise

Conjecture 6.2. $[\mathbb{P} \uparrow\downarrow \pi]$-HyperPCTL$^* \cong_{\mathcal{A}} [\mathsf{init}^*]$-HyperPCTL. △

We note that the procedure given for Conjecture 5.4 is likely to change the value of the probability measures, and hence, the algorithmic relation for Conjecture 6.2 would likely need a more complex algorithm than $\mathcal{A}_{\mathfrak{m}}$ that also takes the DTMC into consideration while transforming a formula.

This consideration was not needed in Section 5.2, since we only asserted measures to be nonzero, and the procedure preserves this behaviour.

If Conjecture 6.2 holds, it would mean that HyperPCTL* with strict alternation introduced to it, is in weak bijection to HyperPCTL restricted to reachable states.

Chapter 7

HyperPCTL vs. PHL on MDPs

In this final chapter before the thesis reaches its conclusion, we shall revisit the results of Chapter 5, and look at whether they scale up for MDPs. As it was the case for DTMCs, one of the logics requires and uses an initial state while the other does not. As such, we first expand our marking algorithm $\mathcal{A}_{\mathfrak{m}}$ to MDPs. If $\mathcal{M} = (S, s_\iota, \mathsf{Act}, p, \mathsf{AP}, l)$ is an MDP with initial state s_ι, let $\mathcal{A}_{\mathfrak{m}}(\mathcal{M})$ be the MDP $(S, \mathsf{Act}, p, \mathsf{AP}_{\mathfrak{m}}, l_{\mathfrak{m}})$ with

- $\mathsf{AP}_{\mathfrak{m}} := \mathsf{AP} \cup \{\mathsf{init}\}$, where we assume that $\mathsf{init} \notin \mathsf{AP}$, and
- $l_{\mathfrak{m}}(s) := \begin{cases} l(s), & \text{if } s \neq s_\iota, \\ l(s) \cup \{\mathsf{init}\}, & \text{if } s = s_\iota. \end{cases}$

Thus, we give $\mathsf{HyperPCTL}_{\mathrm{MDP}}$ a way to select the initial state of $\mathcal{M}$ in $\mathcal{A}_{\mathfrak{m}}(\mathcal{M})$. In the following, we will use $\mathcal{A}_{\mathfrak{m}}$ in conjunction with transformations of formulae to embed fragments of one logic into the other. In any case, the transformation of formulae will happen independently of the usage of $\mathcal{A}_{\mathfrak{m}}$.

We will explicitly only use this new construct to identify the initial state, and assume that it does not expand the expressivity of $\mathsf{HyperPCTL}_{\mathrm{MDP}}$ beyond letting it select that state.

As usual, we start with the comparison to the probabilistic part of $\mathsf{PHL}_{\mathrm{MDP}}$ in isolation in the following section.

7.1 HyperCTL*-less PHL to HyperPCTL

Let $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{MDP}}$ denote the *probabilistic part* of $\mathsf{PHL}_{\mathrm{MDP}}$, that results by striking out all ϑ-rules in Definition 2.22. As we have noted while introducing our downscaling to DTMCs in Section 2.3.3, the MDP version can express a restricted version of probabilistic hyperproperties by quantifying over schedulers.

We start by looking at the fragment of $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{MDP}}$ that is syntactically compatible with $\mathsf{HyperPCTL}_{\mathrm{MDP}}$. To achieve this, we restrict $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{MDP}}$ to use only *shallow* LTL formulae inside its probabilistic expressions, which drop to a PL expression directly after U or $\bigcirc^k$,

for $k \geq 1$. Let this fragment be denoted $[\mathsf{LTL}^s]\text{-}\mathsf{PHL}^{\text{no}\vartheta}_{\text{MDP}}$, and called the *LTL-shallow* fragment of $\mathsf{PHL}^{\text{no}\vartheta}_{\text{MDP}}$. As we did for $\mathsf{HyperPCTL}_{\text{DTMC}}$, we allow $\bigcirc^k \varphi$ in $\mathsf{HyperPCTL}_{\text{MDP}}$ as syntactic sugar for $\mathtt{true}\ \mathsf{U}^{[k,k]}\ \varphi$, $k \geq 2$.

Theorem 7.1. $[\mathsf{LTL}^s]\text{-}\mathsf{PHL}^{\text{no}\vartheta}_{\text{MDP}} \preccurlyeq_{\mathcal{A}} \mathsf{HyperPCTL}_{\text{MDP}}$.

Proof (Sketch). Since scheduler quantification can not be nested in probabilistic expressions in PHL and HyperPCTL, we can assume that all formulae in either logic can be written in a scheduler-PNF form, in which all scheduler quantifications are on the very front of the formula.

A transformation from the former to the latter is given by mapping sequences of scheduler quantifiers, and quantifier-free ψ

$$\underbrace{\mathfrak{Q}_{n-1}\hat{\sigma}_{n-1} \cdots \mathfrak{Q}_0\hat{\sigma}_0.\ \psi}_{[\mathsf{LTL}^s]\text{-}\mathsf{PHL}^{\text{no}\vartheta}_{\text{MDP}}}$$

to

$$\underbrace{\mathfrak{Q}_{n-1}\hat{\sigma}_{n-1} \cdots \mathfrak{Q}_0\hat{\sigma}_0.\ \exists \hat{s}_{n-1}(\hat{\sigma}_{n-1}) \cdots \exists \hat{s}_0(\hat{\sigma}_0).\ \bigwedge_{i<n} \mathsf{init}_{\hat{s}_i} \wedge \mathfrak{T}(\psi),}_{\mathsf{HyperPCTL}_{\text{MDP}}}$$

where, $(\mathfrak{Q}_i)_{i<n} \subset \{\forall, \exists\}$ is a sequence of quantifiers, and $\mathfrak{T}(\psi)$ represents the formula that results by replacing the markings $\hat{\sigma}_i$ in ψ with $\hat{s}_i$, for all $i < n$. Since, after the scheduler-quantification step, we only use the DTMC induced by the drawn schedulers in the semantics of both logics, the equivalence can be proven similarly to Theorem 5.1.□

By the same reasoning, the equivalences of Lemma 5.2 *should* also scale upwards with scheduler quantification, and a similar mapping to the one presented in the preceding theorem. However, since the lemma relies on finiteness for DTMCs, this is only the case if the DTMC induced by the MDP with the drawn schedulers is finite. This would require an artificial restriction of the scheduler space to e.g. *finite-memory* schedulers that can reference a restricted amount of past choices. This investigation exceeds the frame of this thesis, and hence we chose to omit an explicit proof here.

We now move on to the *non*-probabilistic part of $\mathsf{PHL}_{\text{MDP}}$.

7.2 HyperCTL* Sentences in PHL

In this section, we will expand Theorem 5.3 to MDPs. We start by defining the *PNF-existential LTL-shallow* fragment of HyperCTL* for MDPs as the set of formulae generated by the grammar of Figure 18.

$$
\begin{array}{lll}
\text{(\textit{sched.-quantified formulae})} & \varphi^{\sigma} ::= \exists \hat{\sigma}.\varphi^{\sigma} \mid \forall \hat{\sigma}.\varphi^{\sigma} \mid \varphi^{s} \\
\text{($\exists^{*}_{\pi}$ \textit{formulae})} & \varphi^{s} ::= \exists \hat{\pi}(\hat{\sigma}).\varphi^{s} \mid \eta \\
\text{(LTL^{s} \textit{formulae})} & \eta ::= \bigcirc^{k} \zeta \mid \zeta \,\mathsf{U}\, \zeta \\
\text{(PL \textit{formulae})} & \zeta ::= \zeta \wedge \zeta \mid \neg\zeta \mid a_{\hat{\pi}} \mid \mathtt{true}
\end{array}
$$

Figure 18: Grammar of $[\exists^{*}|\mathsf{LTL}^{s}]$-HyperCTL* for MDPs

Theorem 7.2. $[\exists^{*}|\mathsf{LTL}^{s}]$-HyperCTL$^{*} \preccurlyeq_{\mathcal{A}} \mathsf{HyperPCTL}_{\mathrm{MDP}}$.

Proof (Sketch). A transformation from the former to the latter is given by mapping formulae of the form

$$
\underbrace{\mathfrak{Q}_{n-1}\hat{\sigma}_{n-1} \cdots \mathfrak{Q}_{0}\hat{\sigma}_{0}.\ \exists \hat{\pi}_{m-1}(\hat{\sigma}_{i_{m-1}}) \cdots \exists \hat{\pi}_{0}(\hat{\sigma}_{i_{0}}).\ \eta}_{[\exists^{*}|\mathsf{LTL}^{s}]\text{-HyperCTL}^{*}},
$$

for a quantifier-free η, to

$$
\underbrace{\mathfrak{Q}_{n-1}\hat{\sigma}_{n-1} \cdots \mathfrak{Q}_{0}\hat{\sigma}_{0}.\ \exists \hat{s}_{m-1}(\hat{\sigma}_{i_{m-1}}) \cdots \exists \hat{s}_{0}(\hat{\sigma}_{i_{0}}).\ \bigwedge_{i<m} \mathsf{init}_{\hat{s}_{i}} \wedge \mathbb{P}\big(\mathfrak{T}(\eta)\big) > 0}_{\mathsf{HyperPCTL}},
$$

where $n, m < \omega$, $(i_j)_{j<m} \subset [0, n-1]$ a sequence of indices in $[0, n-1]$, $(\mathfrak{Q}_j)_{j<n} \subset \{\forall, \exists\}$ a sequence of quantifiers, and $\mathfrak{T}(\eta)$ represents the formula η with all of its $\hat{\pi}_{i_j}$ markings replaced by the corresponding $\hat{s}_{i_j}$ markings, for $j < m$.

The equivalence can be proven similarly to the case for DTMCs in Theorem 5.3, after resolving the schedulers, and using the DTMC induced by them. □

7.3 Equivalent Fragments

Theorem 7.1 generates HyperPCTL formulae of the following form.

- They have the form $\mathfrak{Q}_{n-1}\hat{\sigma}_{n-1} \cdots \mathfrak{Q}_{0}\hat{\sigma}_{0}.\exists \hat{s}_{n-1}(\hat{\sigma}_{n-1}) \cdots \exists \hat{s}_{0}(\hat{\sigma}_{0}).\ \bigwedge_{i<n} \mathsf{init}_{\hat{s}_{i}} \wedge \psi$, for a quantifier-free ψ.
- Inside the probabilistic expressions of ψ, only shallow LTL is used.
- Probabilistic expressions may be added to one another, but only multiplied by rational constants.[i]

Let this fragment be called the *initial-path simple LTL-shallow* fragment of $\mathsf{HyperPCTL}_{\mathrm{MDP}}$, and be denoted by $[\mathsf{init}^{*}|\rho^{s}|\mathsf{LTL}^{s}]$. A grammar is given in Figure 19.

[i] Recall that these probabilistic expressions of this form are named *simple*.

$$
\begin{array}{ll}
(\mathsf{init}^* \textit{ formulae}) & \varphi ::= \mathfrak{Q}_{n-1}\hat{\sigma}_{n-1} \cdots \mathfrak{Q}_0\hat{\sigma}_0.\ \exists \hat{s}_{n-1}(\hat{\sigma}_{n-1}) \cdots \exists \hat{s}_0(\hat{\sigma}_0).\ \bigwedge_{i<n} \mathsf{init}_{\hat{s}_i} \wedge \psi \\
(\textit{non-quant. formulae}) & \psi ::= \varphi \wedge \varphi \mid \neg\varphi \mid \rho < \rho \\
(\rho^s \textit{ expressions}) & \rho ::= \mathbb{P}(\eta) \mid \rho + \rho \mid c \cdot \rho \mid c \\
(\mathsf{LTL}^s \textit{ formulae}) & \eta ::= \bigcirc^k \zeta \mid \zeta \,\mathsf{U}\, \zeta \\
(\mathsf{PL} \textit{ formulae}) & \zeta ::= \zeta \wedge \zeta \mid \neg\zeta \mid a_{\hat{s}} \mid \mathtt{true}
\end{array}
$$

Figure 19: Grammar of $[\mathsf{init}^*|\rho^s|\mathsf{LTL}^s]$-$\mathsf{HyperPCTL}_{\mathrm{MDP}}$

Theorem 7.3. $[\mathsf{init}^*|\rho^s|\mathsf{LTL}^s]\text{-}\mathsf{HyperPCTL}_{\mathrm{MDP}} \cong_{\mathcal{A}} [\mathsf{LTL}^s]\text{-}\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{MDP}}$.

Proof. The subsumption "$\succcurlyeq_{\mathcal{A}}$" is the subject of Theorem 7.1. A reverse transformation is given by removing all state quantifications and $\bigwedge_{i<n} \mathsf{init}_{\hat{s}_i}$ from a given $[\mathsf{init}^*|\rho^s|\mathsf{LTL}^s]$ formula, and replacing all of its $\hat{s}_i$ markings with $\hat{\sigma}_i$ markings. □

The transformation of Theorem 7.2 creates formulae that

- start with arbitrary scheduler quantifications, followed by an arbitrary number of existential state quantifications,
- can only assert *one* measure to be nonzero,
- have no arithmetic between probabilistic expressions, and
- are limited to shallow LTL.

Let this the fragment of HyperPCTL defined by these constraints be called *quantified initial-path nonzero shallow* and be denoted by $[\mathfrak{Q}^*_\sigma|\mathsf{init}^*|\mathbb{P}^{>0}|\mathsf{LTL}^s]$. A grammar is given in Figure 20.

$$
\begin{array}{ll}
(\textit{sched.-quant. formulae}) & \varphi^\sigma ::= \exists\hat{\sigma}.\ \varphi^\sigma \mid \forall\hat{\sigma}.\ \varphi^\sigma \mid \varphi^s \\
(\exists^*_s \textit{ formulae}) & \varphi^s ::= \exists\hat{s}(\hat{\sigma}).\ \varphi^s \mid \psi \\
(\mathsf{init}^* \textit{ expressions}) & \psi ::= \mathsf{init}_{\hat{s}} \wedge \psi \mid \mathsf{init}_{\hat{s}} \wedge \mathbb{P}(\eta) > 0 \\
(\mathsf{LTL}^s \textit{ formulae}) & \eta ::= \bigcirc^k \zeta \mid \zeta \,\mathsf{U}\, \zeta \\
(\mathsf{PL} \textit{ formulae}) & \zeta ::= \zeta \wedge \zeta \mid \neg\zeta \mid a_{\hat{s}} \mid \mathtt{true}
\end{array}
$$

Figure 20: Grammar of $[\mathfrak{Q}^*_\sigma|\mathsf{init}^*|\mathbb{P}^{>0}|\mathsf{LTL}^s]$-$\mathsf{HyperPCTL}_{\mathrm{MDP}}$

Theorem 7.4. $[\mathfrak{Q}^*_\sigma|\mathsf{init}^*|\mathbb{P}^{>0}|\mathsf{LTL}^s]\text{-}\mathsf{HyperPCTL}_{\mathrm{MDP}} \cong_{\mathcal{A}} [\exists^*|\mathsf{LTL}^s]\text{-}\mathsf{HyperCTL}^*$

Proof. The subsumption "$\succcurlyeq_{\mathcal{A}}$" is shown in Theorem 7.2. A reverse transformation is given by inverting the steps of the one given in the referenced theorem. □

This chapter is now brought to a conclusion in

7.4 Overview of Results

Upscaling the results of Chapter 5, in this segment, we embedded the following fragments of $\mathsf{PHL}_{\mathrm{MDP}}$ into $\mathsf{HyperPCTL}_{\mathrm{MDP}}$.

- The *LTL-shallow* fragment of $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{MDP}}$: $[\mathsf{LTL}^s]$ (Theorem 7.1)
- The *PNF-existential LTL-shallow* fragment of $\mathsf{HyperCTL}^*$: $[\exists^*|\mathsf{LTL}^s]$ (Theorem 7.2)

By identifying the types of the transformed formulae, we inferred that the following fragments of $\mathsf{HyperPCTL}_{\mathrm{MDP}}$ can be embedded back into $\mathsf{PHL}_{\mathrm{MDP}}$.

- The *initial-path simple LTL-shallow* fragment $[\mathsf{init}^*|\rho^s|\mathsf{LTL}^s]$ (Theorem 7.3)
- The *initial-path nonzero LTL-shallow* fragment $[\mathfrak{Q}^*_\sigma|\mathsf{init}^*|\mathbb{P}^{>0}|\mathsf{LTL}^s]$ (Theorem 7.4)

The questions for this part are mostly the ones for DTMC case that were enumerated in Section 5.4. In addition to those, it is unclear how much expressive power $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{MDP}}$ has — as we noted in Section 2.3.3 when we proposed our downscaling of PHL for DTMCs, the original formulation for MDPs includes a restricted version of probabilistic hyperproperties, which we lost in the process of downscaling, as we saw in more detail in Section 4.1.

Chapter 8

Conclusion

In conclusion of the thesis, we shall revisit and summarise the results and open questions of Chapters 4 to 7.

8.1 Summary

Markov Chains

In Chapter 4, we first showed that $\mathsf{PHL}_{\mathrm{DTMC}}$ cannot express multivariate HyperPCTL* formulae in Section 4.1. Then, we embedded the probabilistic part of $\mathsf{PHL}_{\mathrm{DTMC}}$ in HyperPCTL* in Section 4.2, and subsequently focussed on HyperCTL*, starting with formulae with one quantifier in Section 4.3, and building up to multiple quantifiers, and quantifier nesting in Section 4.4. Finally, in Section 4.5, we identified equivalent fragments between the two logics by looking at the types of formulae that our transformations generate, and inverting them on these formulae.

In total, we saw that we can embed into HyperPCTL*

- the *entire* probabilistic part of $\mathsf{PHL}_{\mathrm{DTMC}}$: $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}}$ (Theorem 4.7), and
- the *recursively existential path-positive* fragment of HyperCTL* $[\downarrow\exists^*|\pi^+]$ (Theorem 4.24).

By looking at the forms of the formulae the given transformations generate, we found fragments of HyperPCTL* that we can embed in PHL:

- The *draw-1 simple shallow* fragment $[\mathbb{P}_1|\rho^s|\mathsf{LTL}^s]$ (Theorem 4.25)
- The *recursively nonzero path-positive* fragment $[\downarrow\mathbb{P}^{>0}_{\mathsf{last}}|\pi^+]$ (Theorem 4.26)

Moreover, in the process of examining HyperCTL*, we made it plausible in Section 4.3.2 that we cannot lift the *path-positive* modifier. That is, we cannot allow nested LTL, and subsequently nested path expressions, to be negated.

In Chapter 5, we relied on our marking algorithm $\mathcal{A}_{\mathfrak{m}}$ to give us access to the initial state in HyperPCTL, and started by comparing the syntactically compatible fragments of both $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ and $\mathsf{PHL}_{\mathrm{DTMC}}$. We established that we can embed

- the *LTL-shallow* fragment of $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}}$: $[\mathsf{LTL}^{s}]$ (Theorem 5.1), and
- the *PNF-existential LTL-shallow* fragment of $\mathsf{HyperCTL}^{*}$: $[\exists^{*}|\mathsf{LTL}^{s}]$ (Theorem 5.3)

into HyperPCTL. By identifying the type of HyperPCTL formulae the transformation in the aforementioned theorems generate, we extrapolated the following fragments of HyperPCTL that we can embed into PHL:

- The *1-initial path simple LTL-shallow* fragment $[\mathsf{init}^{1}|\rho^{s}|\mathsf{LTL}^{s}]$ (Theorem 5.7)
- The *initial-path nonzero LTL-shallow* fragment $[\mathsf{init}^{*}|\mathbb{P}^{>0}|\mathsf{LTL}^{s}]$ (Theorem 5.8)

Furthermore, we found special cases of formulae in PHL that are syntactically incompatible with HyperPCTL at a first glance, but still translatable from the former to the latter. Specifically, we have proven the following 2 equivalences (modulo $\mathcal{A}_{\mathfrak{m}}$) in Lemma 5.2.

$$\underbrace{\mathbb{P}(\Box\Diamond\zeta) \sim c}_{\mathsf{PHL}} \quad \equiv_{\mathsf{f}} \quad \underbrace{\exists \hat{s}.\ \mathsf{init}_{\hat{s}} \wedge \mathbb{P}\Big(\Diamond\mathbb{P}\big(\Box\mathbb{P}(\Diamond\zeta_{\hat{s}}) = 1\big) = 1\Big) \sim c}_{\mathsf{HyperPCTL}}$$

$$\underbrace{\mathbb{P}(\Diamond\Box\zeta) \sim c}_{\mathsf{PHL}} \quad \equiv_{\mathsf{f}} \quad \underbrace{\exists \hat{s}.\ \mathsf{init}_{\hat{s}} \wedge \mathbb{P}\Big(\Diamond\mathbb{P}\big(\Box\zeta_{\hat{s}}\big) = 1\Big) \sim c}_{\mathsf{HyperPCTL}}$$

In Lemma 5.6, we have also proven the following special cases.

$$\underbrace{\forall \hat{\pi}.\ \Box\zeta_{\hat{\pi}}}_{\mathsf{HyperCTL}^{*}} \quad \equiv_{\mathsf{f}} \quad \underbrace{\exists \hat{s}.\ \mathsf{init}_{\hat{s}} \wedge \mathbb{P}(\Box\zeta_{\hat{s}}) = 1}_{\mathsf{HyperPCTL}_{\mathrm{DTMC}}}$$

$$\underbrace{\forall \hat{\pi}.\ \bigcirc\zeta_{\hat{\pi}}}_{\mathsf{HyperCTL}^{*}} \quad \equiv_{\mathsf{f}} \quad \underbrace{\exists \hat{s}.\ \mathsf{init}_{\hat{s}} \wedge \mathbb{P}(\bigcirc\zeta_{\hat{\pi}}) = 1}_{\mathsf{HyperPCTL}_{\mathrm{DTMC}}}$$

In Conjecture 5.4, we postulated that we can expand the equivalence of Theorem 5.8 to $[\downarrow\exists^{*}|\mathsf{LTL}^{s}]$-HyperPCTL*, which allows arbitrary nesting of $[\exists^{*}|\mathsf{LTL}^{s}]$ formulae, and gave an example of a scheme to translate certain types of $[\downarrow\exists^{*}|\mathsf{LTL}^{s}]$ formulae to HyperPCTL.

In the short excursus of Chapter 6, we gave a scheme to embed the fragment of HyperPCTL* with all paths drawn from an intial state and strict alternation between path formulae and path expressions into HyperPCTL (Theorem 6.1). Moreover, we hypothesised that this can be expanded to nested path quantification via a more complex algorithm in Conjecture 6.2.

Markov Decision Processes

Upscaling the results of Chapter 5, in Chapter 7, we embedded the following fragments of $\mathsf{PHL}_{\mathrm{MDP}}$ into $\mathsf{HyperPCTL}_{\mathrm{MDP}}$.

- The *LTL-shallow* fragment of $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{MDP}}$: $[\mathsf{LTL}^s]$ (Theorem 7.1)
- The *PNF-existential LTL-shallow* fragment of $\mathsf{HyperCTL}^*$: $[\exists^*|\mathsf{LTL}^s]$ (Theorem 7.2)

By identifying the types of the transformed formulae, we inferred that the following fragments of $\mathsf{HyperPCTL}_{\mathrm{MDP}}$ can be embedded back into $\mathsf{PHL}_{\mathrm{MDP}}$.

- The *initial-path simple LTL-shallow* fragment $[\mathsf{init}^*|\rho^s|\mathsf{LTL}^s]$ (Theorem 7.3)
- The *initial-path nonzero LTL-shallow* fragment $[\mathfrak{Q}^*_\sigma|\mathsf{init}^*|\mathbb{P}^{>0}|\mathsf{LTL}^s]$ (Theorem 7.4)

8.2 Future work

It still remains open, whether special cases of multivariate $\mathsf{HyperPCTL}^*$ formulae are expressible in $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{DTMC}}$, and whether the equivalence

$$[\downarrow\mathbb{P}^{>0}_{\mathrm{last}}|\pi^+]\text{-}\mathsf{HyperPCTL}^* \cong [\downarrow\exists^*|\pi^+]\text{-}\mathsf{HyperCTL}^*$$

of Theorem 4.26 represents the largest fragment equivalence between $\mathsf{HyperPCTL}^*$ and $\mathsf{HyperCTL}^*$ (excluding special cases). Specifically, we only saw that the most generic case of formulae with alternating quantifiers of $\mathsf{HyperCTL}^*$ is not embeddable in $\mathsf{HyperPCTL}^*$ in Section 4.4, basing this on Conjecture 4.13. Furthermore, we only examined the probabilistic and non-probabilistic parts of $\mathsf{PHL}_{\mathrm{DTMC}}$ *in isolation from one another*. It may be the case, that, when combined, they can express more parts of $\mathsf{HyperPCTL}^*$ than just the fragments mentioned above.

Future work on examining the relation between $\mathsf{HyperPCTL}^*$ and $\mathsf{PHL}_{\mathrm{DTMC}}$ could further include proving (or disproving) Conjecture 4.13, and finding special cases of formulae with alternating quantifiers in $\mathsf{HyperCTL}^*$ that have equivalents in $\mathsf{HyperPCTL}^*$.

In Chapter 5, we left the question of whether there are more formulae of $\mathsf{PHL}_{\mathrm{DTMC}}$ that are syntactically incompatible with, but still translatable to $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$, open. Furthermore, we did not examine whether we can expand the special cases of Lemmata 5.2 and 5.6 with nesting. Further research on this part may further encompass examining Conjecture 5.4, possibly finding a scheme that can translate all $[\downarrow\exists^*|\mathsf{LTL}^s]$-$\mathsf{HyperCTL}^*$ formulae to $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$.

The questions that were raised through the comparison of $\mathsf{HyperPCTL}_{\mathrm{MDP}}$ to $\mathsf{PHL}_{\mathrm{MDP}}$ in Chapter 7 are mostly the ones for DTMC case that were enumerated in Section 5.4. In addition to those, it is unclear how much expressive power $\mathsf{PHL}^{\mathrm{no}\vartheta}_{\mathrm{MDP}}$ has — as we noted

in Section 2.3.3 when we proposed our downscaling of PHL for DTMCs, the original formulation for MDPs includes a restricted version of probabilistic hyperproperties, which we lost in the process of downscaling, as we saw in more detail in Section 4.1. One potential direction for future exploration is to find an alternative way of downscaling PHL, in which this expressiveness is preserved, and revisiting the comparisons to $\mathsf{HyperPCTL}_{\mathrm{DTMC}}$ and $\mathsf{HyperPCTL}^*$ with it.

Das Ende der Melodie ist nicht deren Ziel; aber trotzdem: Hat die Melodie ihr Ende nicht erreicht, so hat sie auch ihr Ziel nicht erreicht.

— Friedrich Wilhelm Nietzsche
Der Wanderer und sein Schatten

Appendix A

On the Topic of HyperPCTL*

As we noted before, the version of HyperPCTL* that was presented in Section 2.3.2 was changed drastically compared to the original of [Wan+21]. In this part, we will go over the changes and justify them. First, a verbatim copy of the original HyperPCTL* will be given – with minimal changes to its notation to conform to the notational scheme of the thesis.

Definition A.1 (Original HyperPCTL* Syntax). HyperPCTL* formulae are defined by the following grammar.

- $\varphi ::= a^{\hat{\pi}} \mid \varphi^{\hat{\pi}} \mid \neg\varphi \mid \varphi \wedge \varphi \mid \bigcirc\varphi \mid \varphi \,\mathsf{U}^{\leq k}\, \varphi \mid \varphi \,\mathsf{U}\, \varphi \mid \rho \sim \rho$
- $\rho ::= f\,\overline{\rho} \mid \mathbb{P}^{\tilde{\pi}}(\varphi) \mid \mathbb{P}^{\tilde{\pi}}(\rho)$

where

- $a \in \mathsf{AP}$ is an atomic proposition,
- $\hat{\pi} \in \hat{\Pi}$ is a path variable from a countably infinite set of variables $\hat{\Pi}$,
- $\tilde{\pi} \in \hat{\Pi}^m$ a sequence of path variables from $\hat{\Pi}$, for $m < \omega$,
- $k < \omega$ a natural number,
- $\sim \in \{<, \leq, =, \geq, >\}$ a comparison,
- $\overline{\rho}$ a sequence of ρ-formulae, of length $|\overline{\rho}| < \omega$, and
- $f : \mathbb{R}^n \to \mathbb{R}$ is an n-ary function, for $n := |\overline{\rho}|$, that is either polynomial, exponential, rational, or trigonometric, or any finite sum, product, or composition thereof, or the inverse of any of these. △

Given a DTMC $\mathcal{D}$, the original formulation uses path assignments in the form of functions $\mathfrak{p} : \hat{\Pi} \to \mathsf{Paths}_{\mathcal{D}}$. By default, (unset) path variables start at s_ι. In our notation, that is $\mathfrak{p}(\hat{\pi})(0) = s_\iota$, if $\mathfrak{p}(\hat{\pi}) = \bot$. The semantics are defined as follows.

Definition A.2 (Original HyperPCTL* Semantics). Let $\mathcal{D} = (S, s_\iota, p, \mathsf{AP}, l)$ be a DTMC, $\hat{\pi}, \hat{\pi}_1, \ldots, \hat{\pi}_n \in \hat{\Pi}$, $a \in \mathsf{AP}$, φ, φ' φ-formulae, ρ, ρ' ρ-formulae, $\mathfrak{p}$ a path assignment, $k < \omega$, and $\sim \in \{<, \leq, =, \geq, >\}$.

- $\mathcal{D}, \mathfrak{p} \vDash a^{\hat{\pi}}$ iff $a \in l\big(\mathfrak{p}(\hat{\pi})(0)\big)$
- $\mathcal{D}, \mathfrak{p} \vDash \varphi^{\hat{\pi}}$ iff $\mathcal{D}, \mathfrak{p}' \vDash \varphi$ where $\mathfrak{p}'$ is the assignment

$$\mathfrak{p}'\big(\hat{\pi}'\big) := \begin{cases} \mathfrak{p}(\hat{\pi}), & \text{if } \hat{\pi}' \in \mathsf{free}(\varphi) \\ \mathfrak{p}(\hat{\pi}'), & \text{otherwise} \end{cases}$$

- $\mathcal{D}, \mathfrak{p} \vDash \neg\varphi$ iff $\mathcal{D}, \mathfrak{p} \nvDash \varphi$
- $\mathcal{D}, \mathfrak{p} \vDash \varphi \wedge \varphi'$ iff $\mathcal{D}, \mathfrak{p} \vDash \varphi \ \wedge \ \mathcal{D}, \mathfrak{p} \vDash \varphi'$,
- $\mathcal{D}, \mathfrak{p} \vDash \bigcirc\varphi$ iff $\mathcal{D}, \mathfrak{p}^1 \vDash \varphi$
- $\mathcal{D}, \mathfrak{p} \vDash \varphi \,\mathsf{U}^{\leq k}\, \varphi'$ iff $\exists j \leq k \left(\mathcal{D}, \mathfrak{p}^j \vDash \varphi' \ \wedge \ \forall i < j : \mathcal{D}, \mathfrak{p}^i \vDash \varphi\right)$
- $\mathcal{D}, \mathfrak{p} \vDash \varphi \,\mathsf{U}\, \varphi'$ iff $\exists j < \omega \left(\mathcal{D}, \mathfrak{p}^j \vDash \varphi' \ \wedge \ \forall i < j : \mathcal{D}, \mathfrak{p}^i \vDash \varphi\right)$
- $\mathcal{D}, \mathfrak{p} \vDash \rho \sim \rho'$ iff $[\![\rho]\!]_{\mathcal{D},\mathfrak{p}} \sim [\![\rho']\!]_{\mathcal{D},\mathfrak{p}}$,
- $\left[\!\left[\mathbb{P}^{(\hat{\pi}_1,\ldots,\hat{\pi}_n)}(\varphi)\right]\!\right]_{\mathcal{D},\mathfrak{p}} = \Pr\Big\{(\pi_i)_{i<n} \;\Big|\; \forall i < n : \pi_i \in \mathsf{Paths}_{\mathcal{D}}\big(\mathfrak{p}(\hat{\pi}_i)\big) \wedge \mathfrak{p} \circ \{\hat{\pi}_i \mapsto \pi_i \mid i < n\} \vDash \varphi\Big\}$
- $\left[\!\left[\mathbb{P}^{(\hat{\pi}_1,\ldots,\hat{\pi}_n)}(\rho)\right]\!\right]_{\mathcal{D},\mathfrak{p}} = \Pr\Big\{(\pi_i)_{i<n} \;\Big|\; \forall i < n : \pi_i \in \mathsf{Paths}_{\mathcal{D}}\big(\mathfrak{p}(\hat{\pi}_i)\big) \wedge \mathfrak{p} \circ \{\hat{\pi}_i \mapsto \pi_i \mid i < n\} \vDash \rho\Big\}$

△

This original formulation has the following incompatibilities to our version.

- It includes the $\mathbb{P}^{\tilde{\pi}}(\rho)$ rule in syntax and semantics, which allows nesting of probabilistic expressions *without* comparisons.
- Instead of drawing nested paths at $\mathbb{P}$ operators (via rulesets), variables are overwritten later on via the $\varphi^{\hat{\pi}}$ syntax, that is, this overwrites all free variables in φ with the assignment of the superscripted variable.

Hence, the first significant change is the removal of the rule $\mathbb{P}^{\tilde{\pi}}(\rho)$. The reasoning behind this is that it was unclear how nested formulae generated by it are to be evaluated. Consider, for example, the formula

$$\varphi := \mathbb{P}^{\hat{\pi}_1}\Big(\mathbb{P}^{\hat{\pi}_2}\big(a^{\hat{\pi}_2}\big)\Big) > 0.$$

By applying the semantics starting with the empty assignment $\mathfrak{p} = \varepsilon$, we immediately get

$$[\![\varphi]\!]_{\mathcal{D},\mathfrak{p}} = \Pr\left\{\pi_1 \in \mathsf{Paths}_{\mathcal{D}}(s_\iota) \;\middle|\; \underbrace{\mathcal{D}, (\hat{\pi}_1 \mapsto \pi_1) \vDash \mathbb{P}^{\hat{\pi}_2}\left(a^{\hat{\pi}_2}\right)}_{\text{not well-defined}}\right\} > 0.$$

That is, we arrive at an expression of the form $\mathcal{D}, \mathfrak{p} \vDash \mathbb{P}^{\hat{\pi}}(\varphi')$, for which none of the rules of the semantics are applicable.

The second change was the replacement of variable overwriting with rulesets. This was done to align the formal semantics of the logic with the textual descriptions and pictures given in [Wan+21].

Consider as an example the formula marked (7) in [Wan+21].

$$\psi := \mathbb{P}^{\hat{\pi}_1}\left(\Diamond\left(\mathbb{P}^{(\hat{\pi}_2,\hat{\pi}_3)}\left(a^{\hat{\pi}_2} \mathbin{\mathsf{U}} (a^{\hat{\pi}_3})^{\hat{\pi}_1}\right) > c_2\right)\right) > c_1$$

The semantics of ψ are textually described as follows.

> The formula (7) states that with probability greater than c_1, we can find a path π_1, such that finally from some state s on π_1, with probability greater than c_2, we can find a pair of paths (π_2, π_3) from the pair of states (s_{init}, s) to satisfy "$[a^{\hat{\pi}_2}]$ until $[a^{\hat{\pi}_3}]$". That is, the computation tree of π_3 is a subtree of the computation tree of π_1 (rooted at s_{init}), since $[\hat{\pi}_3]$ in $[(a^{\hat{\pi}_3})^{\hat{\pi}_1}]$ is in the scope of $[\hat{\pi}_1]$. [...]

However, by applying the semantics, we arrive at an inherently different result:

$$
\begin{aligned}
& \mathcal{D}, \mathfrak{p} \vDash \mathbb{P}^{\hat{\pi}_1}\left(\Diamond\left(\mathbb{P}^{\hat{\pi}_2,\hat{\pi}_3}\left(a^{\hat{\pi}_2} \,\mathsf{U}\, (a^{\hat{\pi}_3})^{\hat{\pi}_1}\right) > c_2\right)\right) > c_1 \\
\iff & \left[\!\!\left[\mathbb{P}^{\hat{\pi}_1}\left(\Diamond\left(\mathbb{P}^{\hat{\pi}_2,\hat{\pi}_3}\left(a^{\hat{\pi}_2} \,\mathsf{U}\, (a^{\hat{\pi}_3})^{\hat{\pi}_1}\right) > c_2\right)\right)\right]\!\!\right]_{\mathcal{D},\mathfrak{p}} > c_1 \\
\iff & \Pr\Bigg\{\pi_1 \in \mathsf{Paths}\big(\mathfrak{p}(\hat{\pi}_1)(0)\big) : \\
& \qquad \mathcal{D}, \mathfrak{p}\big[\hat{\pi}_1 \mapsto \pi_1\big] \vDash \Diamond\left(\mathbb{P}^{\hat{\pi}_2,\hat{\pi}_3}\left(a^{\hat{\pi}_2} \,\mathsf{U}\, (a^{\hat{\pi}_3})^{\hat{\pi}_1}\right) > c_2\right)\Bigg\} > c_1 \\
\iff & \Pr\Bigg\{\pi_1 \in \mathsf{Paths}\big(s_\iota\big) : \\
& \qquad \mathcal{D}, \mathfrak{p}\big[\hat{\pi}_1 \mapsto \pi_1\big] \vDash \Diamond\left(\mathbb{P}^{\hat{\pi}_2,\hat{\pi}_3}\left(a^{\hat{\pi}_2} \,\mathsf{U}\, (a^{\hat{\pi}_3})^{\hat{\pi}_1}\right) > c_2\right)\Bigg\} > c_1 \\
\iff & \Pr\Bigg\{\pi_1 \in \mathsf{Paths}\big(s_\iota\big) : \ (\exists i < \omega) \\
& \qquad \mathcal{D}, \mathfrak{p}\big[\hat{\pi}_1 \mapsto \pi_1^i\big] \\
& \qquad\quad \vDash \mathbb{P}^{\hat{\pi}_2,\hat{\pi}_3}\left(a^{\hat{\pi}_2} \,\mathsf{U}\, (a^{\hat{\pi}_3})^{\hat{\pi}_1}\right) > c_2\Bigg\} > c_1 \\
\iff & \Pr\Bigg\{\pi_1 \in \mathsf{Paths}\big(s_\iota\big) : \ (\exists i < \omega) \\
& \qquad \mathcal{D}, \mathfrak{p}\big[\hat{\pi}_1 \mapsto \pi_1^i\big] \\
& \qquad\quad \vDash \Pr\left\{(\pi_2, \pi_3) \in \mathsf{Paths}^2(s_\iota) : a^{\hat{\pi}_2} \,\mathsf{U}\, (a^{\hat{\pi}_3})^{\hat{\pi}_1}\right\} > c_2\Bigg\} > c_1 \\
\iff & \Pr\Bigg\{\pi_1 \in \mathsf{Paths}\big(s_\iota\big) : \ (\exists i < \omega) \\
& \qquad \mathcal{D}, \mathfrak{p}\big[\hat{\pi}_1 \mapsto \pi_1^i\big] \\
& \qquad\quad \vDash \Pr\Big\{(\pi_2, \pi_3) \in \mathsf{Paths}^2(s_\iota) : (\exists j < \omega \ \forall k < j) \\
& \qquad\qquad \mathcal{D}, \mathfrak{p}^j\big[\hat{\pi}_1 \mapsto \pi_1^i, \hat{\pi}_3 \mapsto \pi_3\big] \vDash (a^{\hat{\pi}_3})^{\hat{\pi}_1} \\
& \qquad\qquad \wedge\, \mathcal{D}, \mathfrak{p}^k\big[\hat{\pi}_1 \mapsto \pi_1^i, \hat{\pi}_2 \mapsto \pi_2\big] \vDash a^{\hat{\pi}_2}\Big\} > c_2\Bigg\} > c_1
\end{aligned}
$$

$$
\begin{aligned}
\iff \Pr\Bigg\{ & \pi_1 \in \mathsf{Paths}(s_\iota) : \ (\exists i < \omega) \\
& \mathcal{D}, \mathfrak{p}\big[\hat{\pi}_1 \mapsto \pi_1^i\big] \\
& \quad \vDash \Pr\Big\{(\pi_2, \pi_3) \in \mathsf{Paths}^2(s_\iota) : (\exists j < \omega \ \forall k < j) \\
& \qquad \mathcal{D}, \mathfrak{p}\big[\hat{\pi}_1 \mapsto \pi_1^{i+j}, \hat{\pi}_3 \mapsto \pi_3^j\big] \vDash (a^{\hat{\pi}_3})^{\hat{\pi}_1} \\
& \qquad \wedge \mathcal{D}, \mathfrak{p}\big[\hat{\pi}_1 \mapsto \pi_1^{i+k}, \hat{\pi}_2 \mapsto \pi_2^k\big] \vDash a^{\hat{\pi}_2}\Big\} > c_2\Bigg\} > c_1 \\
\iff \Pr\Bigg\{ & \pi_1 \in \mathsf{Paths}(s_\iota) : \ (\exists i < \omega) \\
& \mathcal{D}, \mathfrak{p}\big[\hat{\pi}_1 \mapsto \pi_1^i\big] \\
& \quad \vDash \Pr\Big\{(\pi_2, \pi_3) \in \mathsf{Paths}^2(s_\iota) : (\exists j < \omega \ \forall k < j) \\
& \qquad \mathcal{D}, \mathfrak{p}\big[\hat{\pi}_1 \mapsto \pi_1^{i+j}, \hat{\pi}_3 \mapsto \pi_1^{i+j}\big] \vDash a^{\hat{\pi}_3} \\
& \qquad \wedge \mathcal{D}, \mathfrak{p}\big[\hat{\pi}_1 \mapsto \pi_1^{i+k}, \hat{\pi}_2 \mapsto \pi_2^k\big] \vDash a^{\hat{\pi}_2}\Big\} > c_2\Bigg\} > c_1 \\
\iff \Pr\Bigg\{ & \pi_1 \in \mathsf{Paths}(s_\iota) : \ (\exists i < \omega) \\
& \Pr\Big\{\pi_2 \in \mathsf{Paths}(s_\iota) : (\exists j < \omega \ \forall k < j) \\
& \qquad a \in l\Big(\pi_1(i+j)\Big) \wedge a \in l\Big(\pi_2(k)\Big)\Big\} > c_2\Bigg\} > c_1
\end{aligned}
$$

Specifically, from the context of the inner $\mathbb{P}$ operator, the assignment of $\hat{\pi}_1$ is "set in stone", as it was drawn by the outer $\mathbb{P}$ operator. After that, it is only shifted around by the superscripts. With this $\hat{\pi}_3$ gets ultimately assigned to a shift π_1^{i+j} of the original path π_1, and cannot branch away from it, as it is *not* redrawn from it, but rather only reuses the existing assignment.

As such, $\hat{\pi}_3$ is not ranging over subtrees of $\hat{\pi}_1$, but rather, each time only over suffixes of a fixed π_1. In the meanwhile π_3 is completely ignored by the measure, hence the original formula is equivalent to the following:

$$
\mathbb{P}^{\hat{\pi}_1}\Bigg(\Diamond\Big(\mathbb{P}^{\hat{\pi}_2}\big(a^{\hat{\pi}_2} \mathbin{\mathsf{U}} a^{\hat{\pi}_1}\big) > c_2\Big)\Bigg) > c_1,
$$

which asserts we can find a π_1 with probability at least c_1, such that, there exists a shift π_1^i, from the starting point of which we can find a *single* π_2 with probability at least c_2, where π_2 has an a-labelled initial segment/prefix that is at least as long as the longest $(\neg a)$-labelled initial segment of π_1^i.

Our proposed change from the above to rulesets that cause paths to be redrawn at the level of $\mathbb{P}$ operators is meant to make the textual description given on p. 91 expressible in the logic. As an example, the formula ψ with the *intended* semantics according to the description given above is now expressible as

$$\psi' := \mathbb{P}_{\hat{\pi}_1}\left(\Diamond\left(\mathbb{P}_{\hat{\pi}_2,\hat{\pi}_3\leftarrow\hat{\pi}_1}\left(a_{\hat{\pi}_2}\ \mathsf{U}\ a_{\hat{\pi}_3}\right) > c_2\right)\right) > c_1$$

in our version.

Bibliography

[ÁB18] Erika Ábrahám and Borzoo Bonakdarpour. "HyperPCTL: A Temporal Logic for Probabilistic Hyperproperties". In: *Quantitative Evaluation of Systems*. Springer International Publishing, 2018, pp. 20–35. ISBN: 978-3-319-99154-2. DOI: doi.org/10.1007/978-3-319-99154-2_2.

[Ábr+20] Erika Ábrahám, Ezio Bartocci, Borzoo Bonakdarpour and Oyendrila Dobe. "Probabilistic Hyperproperties with Nondeterminism". In: *Automated Technology for Verification and Analysis*. Springer International Publishing, 2020, pp. 518–534. ISBN: 978-3-030-59152-6. DOI: 10.1007/978-3-030-59152-6_29.

[Azi+95] Adnan Aziz, Vigyan Singhal, Felice Balarin, Robert K. Brayton and Alberto L. Sangiovanni-Vincentelli. "It usually works: The temporal logic of stochastic systems". In: *Computer Aided Verification*. Springer Berlin Heidelberg, 1995, pp. 155–165. ISBN: 978-3-540-49413-3. DOI: 0.1007/3-540-60045-0_48.

[BK08] Christel Baier and Joost-Pieter Katoen. *Principles of Model Checking*. The MIT Press, 2008. ISBN: 978-0-262-02649-9.

[Bog07] Vladimir I. Bogachev. *Measure Theory*. Springer-Verlag Berlin Heidelberg, 2007. ISBN: 978-3-540-34514-5. DOI: 10.1007/978-3-540-34514-5.

[CE82] Edmund M. Clarke and E. Allen Emerson. "Design and synthesis of synchronization skeletons using branching time temporal logic". In: *Logics of Programs*. Springer Berlin Heidelberg, 1982, pp. 52–71. ISBN: 978-3-540-39047-3. DOI: 10.1007/BFb0025774.

[Cla+14] Michael R. Clarkson, Bernd Finkbeiner, Masoud Koleini, Kristopher K. Micinski, Markus N. Rabe and Cé sar Sánchez. "Temporal Logics for Hyperproperties". In: *Principles of Security and Trust*. Springer Berlin Heidelberg, 2014, pp. 265–284. ISBN: 978-3-642-54792-8. DOI: 10.1007/978-3-642-54792-8_15.

[DFT20] Rayna Dimitrova, Bernd Finkbeiner and Hazem Torfah. "Probabilistic Hyperproperties of Markov Decision Processes". In: *Automated Technology for Verification and Analysis*. Springer International Publishing, 2020, pp. 484–500. ISBN: 978-3-030-59152-6. DOI: 10.1007/978-3-030-59152-6_27.

[EH86] E. Allen Emerson and Joseph Y. Halpern. "'Sometimes' and 'not never' revisited: on branching versus linear time temporal logic". In: *J. ACM* 33.1 (1986), pp. 151–178. ISSN: 0004-5411. DOI: 10.1145/4904.4999.

[HJ89] H. Hansson and B. Jonsson. "A framework for reasoning about time and reliability". In: *1989 Real-Time Systems Symposium*. IEEE Computer Society, 1989, pp. 102–111. DOI: 10.1109/REAL.1989.63561.

[Pnu77] Amir Pnueli. "The temporal logic of programs". In: *18th Annual Symposium on Foundations of Computer Science (sfcs 1977)*. 1977, pp. 46–57. DOI: 10.1109/SFCS.1977.32.

[Wan+21] Yu Wang, Siddhartha Nalluri, Borzoo Bonakdarpour and Miroslav Pajic. "Statistical model checking for hyperproperties". In: *2021 IEEE 34th Computer Security Foundations Symposium*. IEEE. 2021, pp. 1–16. DOI: 10.1109/CSF51468.2021.00009.

Symbol Index

Logics and Fragments Index