Theoretical Computer Science ••• (••••) •••–•••

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Generic weakest precondition semantics from monads enriched with order ☆

Ichiro Hasuo

Department of Computer Science, The University of Tokyo, Hongo 7-3-1, Tokyo 113-8656, Japan

ARTICLE INFO

Article history:
Received 15 September 2014
Received in revised form 21 March 2015
Accepted 30 March 2015
Available online xxxx

Keywords:
Program verification
Program logic
Precondition calculus
Nondeterminism
Game
Effect
Algebra
Coalgebra
Categorical model
Monad

ABSTRACT

We devise a generic framework where a weakest precondition semantics, in the form of indexed posets, is derived from a monad whose Kleisli category is enriched by posets. It is inspired by Jacobs' recent identification of a categorical structure that is common in various predicate transformers, but adds generality in the following aspects: (1) different notions of modality (such as "may" vs. "must") are captured by Eilenberg–Moore algebras; (2) nested alternating branching—like in games and in probabilistic systems with nondeterministic environments—is modularly modeled by a monad on the Eilenberg–Moore category of another.

1. Introduction

Among various styles of program semantics, the one by *predicate transformers* [2] is arguably the most intuitive. Its presentation is inherently logical, representing a program's behaviors by what properties (or *predicates*) hold before and after its execution. Predicate transformer semantics therefore form a basis of *program verification*, where specifications are given in the form of pre- and post-conditions [3]. It has also been used for *refinement* of specifications into programs (see e.g. [4]). Its success has driven extensions of the original nondeterministic framework, e.g. to the probabilistic one [5,6] and to the setting with both nondeterministic and probabilistic branching [7].

A categorical picture More recently, Jacobs in his series of papers [8–10] has pushed forward a categorical view on predicate transformers. It starts with a monad T that models a notion of branching. Then a program—henceforth called a *(branching) computation*—is a Kleisli arrow $X \to TY$; and the weakest precondition semantics is given as a contravariant functor $\mathbb{P}^{\mathcal{K}\ell} \colon \mathcal{K}\ell(T)^{\mathrm{op}} \to \mathbb{A}$, from the Kleisli category to the category $\mathbb{A}$ of suitable ordered algebras.

☆ An earlier version of this paper [1] has been presented at the Twelfth International Workshop on Coalgebraic Methods in Computer Science (CMCS 2014), 5–6 April 2014, Grenoble, France.

E-mail address: ichiro@is.s.u-tokyo.ac.jp.
URL: http://www-mmm.is.s.u-tokyo.ac.jp/~ichiro.

http://dx.doi.org/10.1016/j.tcs.2015.03.047

For example, in the basic nondeterministic setting, T is the powerset monad $\mathcal{P}$ on **Sets** and $\mathbb{A}$ is the category $\mathbf{CL}_\bigwedge$ of complete lattices and $\bigwedge$-preserving maps. The weakest precondition functor $\mathbb{P}^{\mathcal{K}\ell} : \mathcal{K}\ell(T)^{\mathrm{op}} \to \mathbf{CL}_\bigwedge$ then carries a function $f : X \to \mathcal{P}Y$ to

$$\mathrm{wpre}(f) \;:\; \mathcal{P}Y \longrightarrow \mathcal{P}X \;, \qquad Q \longmapsto \{x \in X \mid f(x) \subseteq Q\} \;. \tag{1}$$

Moreover it can be seen that: 1) the functor $\mathbb{P}^{\mathcal{K}\ell}$ factors through the comparison functor $K : \mathcal{K}\ell(\mathcal{P}) \to \mathcal{EM}(\mathcal{P})$ to the Eilenberg–Moore category $\mathcal{EM}(\mathcal{P})$; and 2) the extended functor $\mathbb{P}^{\mathcal{EM}}$ has a dual adjoint $\mathbb{S}$. The situation is as follows.

$$\begin{tikzcd}
\mathbf{CL}_\bigwedge \arrow[rr, bend left, "\mathbb{S}"] \arrow[rr, phantom, "\perp"] & & (\mathbf{CL}_\bigvee)^{\mathrm{op}} \cong \mathcal{EM}(\mathcal{P})^{\mathrm{op}} \arrow[ll, bend left, "\mathbb{P}^{\mathcal{EM}}"] \\
& \mathcal{K}\ell(\mathcal{P})^{\mathrm{op}} \arrow[ul, "\mathbb{P}^{\mathcal{K}\ell} = \mathbb{P}^{\mathcal{EM}} \circ K^{\mathrm{op}}"] \arrow[ur, "K^{\mathrm{op}}"'] &
\end{tikzcd} \tag{2}$$

Here the functor K carries $f : X \to \mathcal{P}Y$ to $f^\dagger : \mathcal{P}X \to \mathcal{P}Y$, $P \mapsto \bigcup_{x \in P} f(x)$. We shall call this mapping $f \mapsto f^\dagger$ a *superposed-state transformer semantics*—it can be understood as the *strongest postcondition semantics* in this specific instance of $T = \mathcal{P}$, but not necessarily in other instances. See Remark 2.11.

Therefore the picture (2)—understood as the one below—identifies a general categorical structure that underlies predicate transformer semantics. The dual adjunction here (which is in fact an isomorphism in the specific instance of (2)) indicates a "duality" between (backward) predicate transformers and (forward) superposed-state transformers.

$$\begin{tikzcd}
\left(\begin{array}{c}\text{(backward) predicate}\\ \text{transformers}\end{array}\right) \arrow[rr, bend left, "\mathbb{S}"] \arrow[rr, phantom, "\perp"] & & \left(\begin{array}{c}\text{(forward) superposed-state}\\ \text{transformers}\end{array}\right) \arrow[ll, bend left] \\
& \left(\begin{array}{c}\text{(branching)}\\ \text{computations}\end{array}\right) \arrow[ul, "\begin{array}{c}\text{weakest precondition}\\ \text{semantics,}\\ \text{predicate transformer}\\ \text{semantics}\end{array}"] \arrow[ur, "\begin{array}{c}\text{superposed-state}\\ \text{transformer semantics}\end{array}"'] &
\end{tikzcd} \tag{3}$$

Jacobs has identified other instances of (3) for: discrete probabilistic branching [8]; quantum logic [8]; and continuous probabilistic branching [9].[1] See [10] for an overview and also for additional instances. In all these instances the notion of *effect module*—originally from the study of quantum probability [11]—plays an essential role as algebras of "quantitative logics."

Towards generic weakest precondition semantics In [8–10] the picture (3) is presented through examples, and its categorical axiomatics—that encompass many different instances of the picture—have not been pursued as a main goal.[2] Finding such axiomatics is the current paper's aim. In doing so, moreover, we acquire additional generality in two aspects: *different modalities* and *nested alternating branching*.

To motivate the first aspect of generality, observe that the weakest precondition semantics in (1) is the *must* semantics. The *may* variant looks as interesting; it would carry a postcondition $Q \subseteq Y$ to $\{x \in X \mid f(x) \cap Q \neq \emptyset\}$. The difference between the two semantics is much like the one between the modal operators $\Box$ and $\Diamond$.

On the second aspect, situations are abound in computer science where a computation involves two heterogeneous layers of branching. Typically these layers correspond to two distinct *players* with conflicting interests. Examples are *games*, a two-player version of automata which are essential tools in various topics including model-checking; and *probabilistic systems* where it is common to include nondeterministic branching too for modeling the environment's choices. Further details will be discussed later in Section 4.

Predicates and modalities from monads In this paper we present two categorical setups that are inspired by [12–14]—specifically by their use of $T1$ as a domain of *truth values* or *quantities*.

The first "one-player" setup is when we have only one layer of branching. Much like in [8–10] we start from a monad T. Assuming that T is *order-enriched*—in the sense that its Kleisli category $\mathcal{K}\ell(T)$ is **Posets**-enriched—we observe that:

- a natural notion of *truth value* arises from an object $T\Omega$ (where the object Ω is typically the terminal one 1);
- and a modality (like "may" and "must") corresponds to a choice of an Eilenberg–Moore algebra $\tau : T(T\Omega) \to T\Omega$.

The required data set (T, Ω, τ) shall be called a *predicate transformer situation*. We prove that it induces a *weakest precondition semantics* functor $\mathcal{K}\ell(T)^{\mathrm{op}} \to \mathbf{Posets}$, and that it factors through $K : \mathcal{K}\ell(T) \to \mathcal{EM}(T)$, much like in (2). The general setup addresses common instances like the original nondeterministic one [2] and the probabilistic predicate transformers

[1] Different terminologies are used in [8] to describe the picture (3). See Remark 1.1.

[2] An exception is a unified treatment of branching weighted by a semiring R; see e.g. [10, §3]. This, however, does not generalize to the probabilistic branching as it is.

in [5,6]. Moreover it allows us to systematically search for different modalities, leading e.g. to a probabilistic notion of partial correctness guarantee that does not seem well-known.

The other setup is the alternating, "two-player" one. It is much like a one-player setup built on another, with two monads T and R and two "modalities" τ and ρ. A potential novelty here is that R is a monad on $\mathcal{EM}(T)$; this way we manage some known complications in nested branching, such as the difficulty of combining probability and nondeterminism. We prove that the data set $(T, \Omega, \tau, R, \rho)$ gives rise to a weakest precondition semantics, as before. Its examples include: a logic of *forced predicates* in games; and the probabilistic predicate transformers in [7].

The general categorical axiomatics in this paper does not address all the structures that are common in the known instances of (3). The most notable point that is left out is the role of effect modules that is fundamental in [8]. Indeed, on the top-left corner of (3) we always have **Posets** that is much poorer than structures like complete lattices or effect modules. We envisage a fully general framework where **Posets** is replaced by a symmetric monoidal closed category $\mathbb{V}$—that can be instantiated by the category of effect modules, for example—and we start with a monad T that is "$\mathbb{V}$-enriched" in a suitable sense. A "$\mathbb{V}$-enriched" framework like this is left as future work. Another point that is missing is that the left adjoint $\mathbb{S}$ in (3) is only obtained in case the base category is **Sets**. See Section 2.5.

Remark 1.1 *(The state-and-effect triangle).* In Jacobs' recent paper [8] the categorical picture in (3) is called a *state-and-effect triangle* and described as follows, using different terminologies.

$$\begin{tikzcd} \text{(predicate transformers)} \arrow[rr, shift left] \arrow[rr, phantom, "\perp"] & & \text{(state transformers)} \arrow[ll, shift left] \\ & \text{(base category)} \arrow[ul, "\substack{\text{predicate transformer}\\ \text{semantics}}"] \arrow[ur, "\substack{\text{state transformer}\\ \text{semantics}}"'] & \end{tikzcd} \qquad (4)$$

These terminologies have influences from quantum theory and need some explanation.

- The word "state" in (4) should not be understood in the line of *memory state* in the semantics of an imperative programming language, the latter being a function $\sigma : \mathbf{Var} \to V$ that carries a variable to a value (see e.g. [15]). The notion of state in (4) is rather like the notion of *mixed state*—a probabilistic superposition $\sum_{i \in I} c_i |\varphi_i\rangle\langle\varphi_i|$ of *pure states* φ_i—that is often simply called *state* in quantum theory.
  Indeed, in the instances of the general picture (3)–(4) presented in the current paper, what is on the top-right corner is best understood as a "T-mixture" or "T-superposition" of points—here T is a monad that models branching. See Remark 2.11.
- The word "effect" in "state-and-effect triangle" comes from the notion of *effect* in quantum theory and has little to do with *computational effect* in the theory of programming languages. In quantum theory, an effect is a convex-linear map from (quantum) states to the values in the interval $[0, 1]$; considering a value $r \in [0, 1]$ as a "likelihood," an effect plays the role of a *predicate* in quantum theory (see e.g. [16]). Therefore in the terminologies of the state-and-effect triangle (4), "effect" and "predicate" should be deemed synonymous.

In this paper we shall stick to the more programming language-oriented terminologies in (3), since quantum logic does not play a big role here.

Organization of the paper In Section 2 we introduce our first "one-player" setup, establish the triangle (3), and exhibit its examples. Many examples are based on **Sets**; but a Giry-like monad for continuous probabilistic branching (on **Meas**) is another example. We establish the left adjoint $\mathbb{S}$ in (3) only for the **Sets**-based cases, though. In Section 3 we characterize the order-enrichment requirements in the previous section in terms of algebraic operations. We rely on [17] there.

Our second "two-player" setup is first motivated in Section 4 through the examples of games and probabilistic systems, and is formally introduced in Section 5. Its examples are described in Section 6 in detail. In Section 7 we conclude.

Additions and changes in this extended version, compared to the workshop version [1], are as follows.

- In the picture (3) the top-right corner was called *strongest post-condition semantics* in [1], a name that is not suited for all the examples. This name was changed following Jacobs' recent paper [8]. See Remark 1.1.
- Section 3 is added. The additional material would hopefully aid concrete understanding of the abstract order-theoretic assumptions in the paper.
- An example for continuous probabilistic branching—based on the Giry monad—is added. It is unique in this paper in that their base category is the category **Meas** of measurable spaces, not **Sets**.
- The left adjoint $\mathbb{S}$ in (3) was totally missing in [1]. It is now present in our framework, although only when the base category is **Sets**. See Section 2.5, where we rely on a folklore result suggested by Bart Jacobs.
- Throughout the paper we added some (minor) results, explanations and examples. They include Section 2.6 that is new, and Remark 2.9.

The categorical proofs in Sections 2.5–2.6 are somewhat heavy. They are deferred to the appendix, so that they do not interrupt the main line of technical developments.

Please cite this article in press as: I. Hasuo, Generic weakest precondition semantics from monads enriched with order, Theoret. Comput. Sci. (2015), http://dx.doi.org/10.1016/j.tcs.2015.03.047

Notations and terminologies For a monad T, a T*-algebra* $TX \xrightarrow{a} X$ shall always mean an *Eilenberg–Moore algebra* for T, making the diagrams below commute. For categorical backgrounds see e.g. [18,19].

$$\begin{tikzcd} X \arrow[r, "\eta_X"] \arrow[dr, "\mathrm{id}"'] & TX \arrow[d, "a"] \\ & X \end{tikzcd} \qquad \begin{tikzcd} T(TX) \arrow[r, "Ta"] \arrow[d, "\mu_X"'] & TX \arrow[d, "a"] \\ TX \arrow[r, "a"'] & X \end{tikzcd} \tag{5}$$

Given a monad T on $\mathbb{C}$, an arrow in the Kleisli category $\mathcal{K}\ell(T)$ is denoted by $X \twoheadrightarrow Y$; an identity arrow is denoted by $\mathrm{id}_X^{\mathcal{K}\ell(T)}$; and composition of arrows is denoted by $g \odot f$. These are to be distinguished from $X \to Y$, id_X and $g \circ f$, respectively, in the base category $\mathbb{C}$.

2. Generic weakest preconditions, one-player setting

2.1. Order-enriched monad

We use monads for representing various notions of "branching." These monads are assumed to have order-enrichment ($\sqsubseteq$ for, roughly speaking, "more options"); and this will be used for an entailment relation, an important element of logic.

The category **Posets** is that of posets and monotone functions.

Definition 2.1. An *order-enriched monad* T on a category $\mathbb{C}$ is a monad together with a **Posets**-enriched structure of the Kleisli category $\mathcal{K}\ell(T)$.

The latter means specifically: 1) each homset $\mathcal{K}\ell(T)(X,Y) = \mathbb{C}(X, TY)$ carries a prescribed poset structure; and 2) composition $\odot$ in $\mathcal{K}\ell(T)$ is monotone in each argument. Such order-enrichment typically arises from the poset structure of TY in the pointwise manner. In the specific setting of $\mathbb{C} = \mathbf{Sets}$ such enrichment can be characterized by *substitutivity* and *congruence* of orders on TX. See [17], some of whose results are included in Section 3 for the record.

Remark 2.2. Note that the notion of order-enriched monad (Definition 2.1) is different from that of **Posets**-*enriched monad*, an instance of $\mathbb{V}$-*enriched monad* [20], that would be a monad on a **Posets**-enriched category. Our examples of $\mathbb{C}$ are **Sets** and **Meas**; and this constitutes a major difference from the domain-theoretic framework developed in [21–23].

Below are some examples of order-enriched monads; with each of them the order-enriched structure arises in the pointwise manner. Our intuition about an order-enriched monad T is that it represents one possible branching type, where $\eta_X : X \to TX$ represents the trivial branching with a unique option and $\mu_X : T(TX) \to TX$ represents flattening 'branching twice' into 'branching once' (see [24]). In fact each of the first three examples below ($T = \mathcal{L}, \mathcal{P}, \mathcal{D}$) has the Kleisli category $\mathcal{K}\ell(T)$ enriched by the category **Cppo** of pointed cpo's and continuous maps—not just by **Posets**—and hence is suited for generic *coalgebraic trace semantics* [24].[3]

Example 2.3.

1. The *lift monad* $\mathcal{L} = 1 + (\_)$ on **Sets**—where the element of 1 is denoted by $\perp$—has a standard monad structure induced by coproducts. For example, the multiplication $\mu^{\mathcal{L}} : 1 + 1 + X \to 1 + X$ carries $x \in X$ to itself and both $\perp$'s to $\perp$. The set $\mathcal{L}X$ is a pointed dcpo with the flat order ($\perp \sqsubseteq x$ for each $x \in X$).
The lift monad $\mathcal{L}$ models the "branching type" of potential nontermination.
2. The *powerset monad* $\mathcal{P}$ on **Sets** models (possibilistic) nondeterminism. Its action on arrows takes direct images: $(\mathcal{P}f)U = \{f(x) \mid x \in U\}$. Its unit is given by singletons: $\eta_X^{\mathcal{P}} = \{\_\} : X \to \mathcal{P}X$, and its multiplication is by unions: $\mu_X^{\mathcal{P}} = \bigcup : \mathcal{P}(\mathcal{P}X) \to \mathcal{P}X$.
3. The *subdistribution monad* $\mathcal{D}$ on **Sets** models discrete probabilistic branching. It carries a set X to the set of (probability) subdistributions over X:
$$\mathcal{D}X := \left\{ d : X \to [0,1] \;\middle|\; \textstyle\sum_{x \in X} d(x) \le 1 \right\} \;; \tag{6}$$
such d is called a *sub*distribution since the values need not add to 1.[4] Given an arrow $f : X \to Y$ in **Sets**, $\mathcal{D}f : \mathcal{D}X \to \mathcal{D}Y$ is defined by $(\mathcal{D}f)(d)(y) := \sum_{x \in f^{-1}(\{y\})} d(x)$. Its unit is the *Dirac* (or *pointmass*) *distribution*: $\eta_X^{\mathcal{D}}(x) = [x \mapsto 1;\ x' \mapsto 0 \text{ (for } x' \neq x)]$; its multiplication is $\mu_X^{\mathcal{D}}(\mathfrak{a}) = [x \mapsto \sum_{d \in \mathcal{D}X} \mathfrak{a}(d) \cdot d(x)]$ for $\mathfrak{a} \in \mathcal{D}(\mathcal{D}X)$.

[3] Trace semantics based on the fourth example $T = \mathcal{G}$ does not follow from the same order-theoretic construction. See [25] for an alternative measure-theoretic study.

[4] To be precise, the sum $\sum_{x \in X} d(x)$ in (6) over an arbitrary set X is defined by $\sup\{\sum_{x \in X'} d(x) \mid X' \text{ is a finite subset of } X\}$. It is not hard to see that such a (discrete) subdistribution d necessarily has a countable support, that is, $|\{x \in X \mid d(x) \neq 0\}| = \aleph_0$. See e.g. [26, Prop. 2.1.2].

The reason for using *sub*distributions, instead of (proper) distributions, is that otherwise the order structure becomes trivial.

4. We shall also be using the "subprobability" variant $\mathcal{G}$ of the *Giry monad* [27], on the category **Meas** of measurable spaces and measurable maps. This is for general probabilistic branching (possibly over uncountable/continuous spaces): note that the previous example $\mathcal{D}$ on **Sets** is not enough for continuous probability, since the support of d in (6) is easily shown to be at most countable (see e.g. [26]).
The monad $\mathcal{G}$ is defined is as follows (see e.g. [25,28] for further details). Given a measurable space (X, Σ_X),
$$\mathcal{G}X := \bigl\{d\colon \Sigma_X \to [0,1],\ \text{a subprobability measure}\bigr\}\ ,$$
where a *sub*probability measure is such that $d(X) \in [0,1]$ instead of $d(X) = 1$. The set $\mathcal{G}X$ is equipped with the smallest measurable structure that makes all evaluation maps measurable, that is concretely, with the σ-algebra generated by the family
$$\Bigl\{\{d \in \mathcal{G}X \mid d(S) \geq q\}\ \Big|\ S \in \Sigma_X, q \in \mathbb{Q}\Bigr\}\ .$$
The other components of $\mathcal{G}$'s monad structure is straightforward adaptation of those of $\mathcal{D}$: its unit is given by Dirac distributions; and its multiplication is given by the Lebesgue integral $\mu^{\mathcal{G}}_X(\mathfrak{a})(x) = \int_d d(x)\mathrm{d}\mathfrak{a}$. In this paper we shall call the above monad $\mathcal{G}$ the *continuous subdistribution monad*.
5. Another example is the *quantum branching monad* $\mathcal{Q}$. It is introduced in [29] for the purpose of modeling a quantum programming language that obeys the design principle of "quantum data, classical control." It comes with an order-enrichment, too, derived from the *Löwner partial order* between positive operators. The description of $\mathcal{Q}$ involves quantum theoretic constructs that are not used elsewhere in the paper, hence is deferred to [29].

2.2. PT situation and generic weakest precondition semantics

We introduce our first basic setup for our generic weakest precondition semantics. In our main examples we take **Sets** or **Meas** for the base category $\mathbb{C}$, and $\Omega = 1$ (a singleton).

Definition 2.4 *(PT situation).* A *predicate transformer situation* (a *PT situation* for short) over a category $\mathbb{C}$ is a triple (T, Ω, τ) of

- an order-enriched monad T on $\mathbb{C}$;
- an object $\Omega \in \mathbb{C}$; and
- an (Eilenberg–Moore) algebra $\tau\colon T(T\Omega) \to T\Omega$ that satisfies the following *monotonicity condition*: for each $X \in \mathbb{C}$, the correspondence
$$\begin{aligned}(\Phi_\tau)_X :\ \mathbb{C}(X, T\Omega) &\longrightarrow \mathbb{C}(TX, T\Omega)\ \text{, i.e.}\ \mathcal{K}\ell(T)(X,\Omega) \longrightarrow \mathcal{K}\ell(T)(TX,\Omega)\ ,\\ \text{given by}\quad \bigl(X \xrightarrow{p} T\Omega\bigr) &\longmapsto \bigl(TX \xrightarrow{Tp} T(T\Omega) \xrightarrow{\tau} T\Omega\bigr)\end{aligned}$$
is monotone with respect to the order-enrichment of the Kleisli category $\mathcal{K}\ell(T)$ (Definition 2.1). Note here that $\Phi_\tau : \mathbb{C}(\_, T\Omega) \Rightarrow \mathbb{C}(T\_, T\Omega)$ is nothing but the natural transformation induced by the arrow τ via the Yoneda lemma.

The data τ is called a *modality*; see the introduction (Section 1) and also Section 2.3 below.

The following lemma states that, given T, its multiplication μ gives a canonical (but not unique) modality for T.

Lemma 2.5. *If T is an order-enriched monad, (T, Ω, μ_Ω) is a PT situation. Here μ_Ω is the Ω-component of the multiplication of the monad T.*

Proof. We have only to check the monotonicity condition of μ_Ω in Definition 2.4. It is easy to see that the following commutes.
$$\begin{tikzcd} \mathbb{C}(X, T\Omega) \arrow[r, "(\Phi_{\mu_\Omega})_X = \mu_\Omega \circ T(\_)"] \arrow[d, phantom, "\shortparallel"] & \mathbb{C}(TX, T\Omega) \arrow[d, phantom, "\shortparallel"] \\ \mathcal{K}\ell(T)(X, \Omega) \arrow[r, "(\_)\odot(\mathrm{id}_{TX})^\wedge"'] & \mathcal{K}\ell(T)(TX, \Omega) \end{tikzcd}$$
Here $(\mathrm{id}_{TX})^\wedge\colon TX \to\!\!\!\!\!\!\to X$ is the arrow that corresponds to the identity id_{TX} in $\mathbb{C}$. The claim follows from the monotonicity of $\odot$ (postulated in Definition 2.1). $\square$

We shall derive a weakest precondition semantics from a given PT situation (T, Ω, τ). The goal consists of:

- a (po)set $\mathbb{P}^{\mathcal{K}\ell}(\tau)(X)$ of *predicates* for each object $X \in \mathbb{C}$, whose order $\sqsubseteq$ represents an *entailment relation* between predicates; and
- an assignment, to each *(branching) computation* $f: X \to TY$ in $\mathbb{C}$, a *predicate transformer*

$$\mathrm{wpre}(f) : \mathbb{P}^{\mathcal{K}\ell}(\tau)(Y) \longrightarrow \mathbb{P}^{\mathcal{K}\ell}(\tau)(X) \tag{7}$$

that is a monotone function.

Noting that a computation is an arrow $f: X \twoheadrightarrow Y$ in $\mathcal{K}\ell(T)$ and combining the two data above, we are aiming at a functor

$$\mathbb{P}^{\mathcal{K}\ell}(\tau) : \mathcal{K}\ell(T)^{\mathrm{op}} \longrightarrow \mathbf{Posets} \ . \tag{8}$$

Such a functor is known as an *indexed poset*, a special case of *indexed categories*. These "indexed" structures are known to correspond to "fibered" structures (*poset fibrations* and *(split) fibrations*, respectively), and all these have been used as basic constructs in categorical logic (see e.g. [30]). An indexed poset like (8) therefore puts us on the firm footing of the categorical logic tradition.

Proposition 2.6 *(The indexed poset $\mathbb{P}^{\mathcal{K}\ell}(\tau)$). Given a PT situation (T, Ω, τ), the following defines an indexed poset $\mathbb{P}^{\mathcal{K}\ell}(\tau): \mathcal{K}\ell(T)^{\mathrm{op}} \to$ **Posets**.*[5]

- *On an object $X \in \mathcal{K}\ell(T)$, $\mathbb{P}^{\mathcal{K}\ell}(\tau)(X) := \mathcal{K}\ell(T)(X, \Omega) = \mathbb{C}(X, T\Omega)$.*
- *On an arrow $f: X \twoheadrightarrow Y$, $\mathbb{P}^{\mathcal{K}\ell}(\tau)(f): \mathbb{C}(Y, T\Omega) \to \mathbb{C}(X, T\Omega)$ is defined by*

$$\left(Y \xrightarrow{q} T\Omega\right) \longmapsto \left(X \xrightarrow{f} TY \xrightarrow{Tq} T(T\Omega) \xrightarrow{\tau} T\Omega\right) \ .$$

Proof. We need to check: the monotonicity of $\mathbb{P}^{\mathcal{K}\ell}(\tau)(f)$; and that the functor $\mathbb{P}^{\mathcal{K}\ell}(\tau)$ indeed preserves identities and composition of arrows. These will be proved later, altogether in the proof of Theorem 2.14. □

A consequence of the proposition—specifically the functoriality of $\mathbb{P}^{\mathcal{K}\ell}(\tau)$—is *compositionality* of the weakest precondition semantics: given two computations $f: X \to TY$, $g: Y \to TU$ and a postcondition $r: U \to T\Omega$, Proposition 2.6 automatically ensures

$$\mathbb{P}^{\mathcal{K}\ell}(\tau)(g \odot f)(r) \;=\; \mathbb{P}^{\mathcal{K}\ell}(\tau)(f)\left(\mathbb{P}^{\mathcal{K}\ell}(\tau)(g)(r)\right) \ .$$

That is, the semantics of a sequential composition $g \odot f$ can be computed step by step.

2.3. Examples of PT situations

For each of $T = \mathcal{L}, \mathcal{P}, \mathcal{D}, \mathcal{G}$ in Example 2.3, we take $\Omega = 1$ and the set $T1$ is naturally understood as a set of "truth values" (an observation in [12–14]):

$$\mathcal{L}1 \;=\; \begin{bmatrix} (\mathtt{tt} := *) \\ \sqcup\!\! | \\ (\mathtt{ff} := \bot) \end{bmatrix} , \qquad \mathcal{P}1 \;=\; \begin{bmatrix} (\mathtt{tt} := 1) \\ \sqcup\!\! | \\ (\mathtt{ff} := \emptyset) \end{bmatrix} , \quad \text{and} \quad \mathcal{D}1 \cong \mathcal{G}1 \;\cong\; \big([0,1], \le\big) \ .$$

Here $*$ is the element of the argument 1 in $\mathcal{L}1$, and the measurable structure of $\mathcal{G}1$ is the same as the standard Borel structure of the interval $[0, 1]$. Both $\mathcal{L}1$ and $\mathcal{P}1$ represent the Boolean truth values. In the $\mathcal{D}$ and $\mathcal{G}$ cases a truth value is $r \in [0, 1]$; a predicate, being a function $X \to [0, 1]$, is hence a *random variable* that tells the certainty with which the predicate holds at each $x \in X$.

We shall introduce modalities for these monads T and $\Omega = 1$. The following observation will be used.

Lemma 2.7. *The category $\mathcal{EM}(T)$ of Eilenberg–Moore algebra is iso-closed in the category of functor T-algebras. That is, given an Eilenberg–Moore algebra $a: TX \to X$, an arrow $b: TY \to Y$, and an isomorphism $f: X \xrightarrow{\cong} Y$ such that $f \circ a = b \circ Tf$, the arrow b is also an Eilenberg–Moore algebra.*

Proof. Straightforward from diagram chasing. □

[5] For brevity we favor the notation $\mathbb{P}^{\mathcal{K}\ell}(\tau)$ over $\mathbb{P}^{\mathcal{K}\ell}(T, \Omega, \tau)$ that is more appropriate.

2.3.1. The lift monad $\mathcal{L}$: τ_{total} and τ_{partial}

We have the following two modalities. There are none other, which is easily seen from the requirement that a modality (an Eilenberg–Moore algebra) is compatible with the monad unit (5).

$$\begin{array}{rlcl} \tau_{\mathrm{total}}, \tau_{\mathrm{partial}}: & \{\bot\} + \{\mathtt{tt}, \mathtt{ff}\} = \mathcal{L}(\mathcal{L}1) & \longrightarrow & \mathcal{L}1 = \{\mathtt{tt}, \mathtt{ff}\}\ , \\ \tau_{\mathrm{total}}: & \bot \mapsto \mathtt{ff}\ ,\quad \mathtt{tt} \mapsto \mathtt{tt}\ ,\quad \mathtt{ff} \mapsto \mathtt{ff}\ , \\ \tau_{\mathrm{partial}}: & \bot \mapsto \mathtt{tt}\ ,\quad \mathtt{tt} \mapsto \mathtt{tt}\ ,\quad \mathtt{ff} \mapsto \mathtt{ff}\ . \end{array}$$

The one we obtain from multiplication $\mu_1^{\mathcal{L}}$ is τ_{total}. The other modality τ_{partial}—whose monotonicity (Definition 2.4) is easy by case distinction—is nonetheless important in program verification. Given $q: Y \to \mathcal{L}1$ and $f: X \to \mathcal{L}Y$ where f is understood as a possibly diverging computation from X to Y, the predicate

$$\mathbb{P}^{\mathcal{K}\ell}(\tau_{\mathrm{partial}})(f)(q) \;=\; \tau_{\mathrm{partial}} \circ \mathcal{L}q \circ f \quad : \quad X \longrightarrow \mathcal{L}1$$

carries $x \in X$ to $\mathtt{tt}$ in case $f(x) = \bot$, i.e., if the computation is diverging. This is therefore a *partial correctness* specification that is common in Floyd–Hoare logic (see e.g. [15]). In contrast, using τ_{total}, the logic is about *total correctness*.

2.3.2. The powerset monad $\mathcal{P}$: $\tau_\Diamond$ and $\tau_\Box$

The monad multiplication $\mu_1^{\mathcal{P}}$ yields a modality which shall be denoted by $\tau_\Diamond$. The other modality $\tau_\Box$ is given via the swapping $\sigma: \mathcal{P}1 \xrightarrow{\cong} \mathcal{P}1$:

$$\begin{tikzcd} \mathcal{P}(\mathcal{P}1) \arrow[r, "\mathcal{P}\sigma", "\cong"'] \arrow[d, "\tau_\Box"'] & \mathcal{P}(\mathcal{P}1) \arrow[d, "\tau_\Diamond"] \\ \mathcal{P}1 & \mathcal{P}1 \arrow[l, "\cong", "\sigma"'] \end{tikzcd} ; \qquad \begin{array}{l} \text{explicitly,} \\ \tau_\Diamond\{\} = \mathtt{ff},\ \tau_\Diamond\{\mathtt{tt}\} = \mathtt{tt},\ \tau_\Diamond\{\mathtt{ff}\} = \mathtt{ff},\ \tau_\Diamond\{\mathtt{tt}, \mathtt{ff}\} = \mathtt{tt}; \\ \tau_\Box\{\} = \mathtt{tt},\ \tau_\Box\{\mathtt{tt}\} = \mathtt{tt},\ \tau_\Box\{\mathtt{ff}\} = \mathtt{ff},\ \tau_\Box\{\mathtt{tt}, \mathtt{ff}\} = \mathtt{ff}. \end{array} \qquad (9)$$

In view of Lemmas 2.5 and 2.7, we have only to check that the map $\tau_\Box$ satisfies the monotonicity condition in Definition 2.4. We first observe that, for $h: X \to \mathcal{P}1$ and $U \in \mathcal{P}X$,

$$(\tau_\Box \circ \mathcal{P}h)(U) = \mathtt{ff} \quad \Longleftrightarrow \quad \mathtt{ff} \in (\mathcal{P}h)(U) \quad \Longleftrightarrow \quad \exists x \in U.\, h(x) = \mathtt{ff}\ ,$$

where the first equivalence is by (9). Now assume that $f \sqsubseteq g: X \to\!\!\!\!\to 1$ and $(\tau_\Box \circ \mathcal{P}g)(U) = \mathtt{ff}$. For showing $\tau_\Box \circ \mathcal{P}f \sqsubseteq \tau_\Box \circ \mathcal{P}g$ it suffices to show that $(\tau_\Box \circ \mathcal{P}f)(U) = \mathtt{ff}$; this follows from the above observation.

The modalities $\tau_\Diamond$ and $\tau_\Box$ capture the *may* and *must* weakest preconditions, respectively. Indeed, given $q: Y \to \mathcal{P}1$ and $f: X \to \mathcal{P}Y$, we have $\mathbb{P}^{\mathcal{K}\ell}(\tau_\Diamond)(f)(q)(x) = \mathtt{tt}$ if and only if there exists $y \in Y$ such that $y \in f(x)$ and $q(y) = \mathtt{tt}$ (for the $\tau_\Diamond$ modality); and $\mathbb{P}^{\mathcal{K}\ell}(\tau_\Box)(f)(q)(x) = \mathtt{tt}$ if and only if $y \in f(x)$ implies $q(y) = \mathtt{tt}$ (for the $\tau_\Box$ modality).

Moreover, we can show that $\tau_\Diamond$ and $\tau_\Box$ are the only modalities (in the sense of Definition 2.4) for $T = \mathcal{P}$ and $\Omega = 1$. Since the unit law in (5) forces $\tau\{\mathtt{tt}\} = \mathtt{tt}$ and $\tau\{\mathtt{ff}\} = \mathtt{ff}$, the only possible variations other than $\tau_\Diamond$ and $\tau_\Box$ are the following τ_1 and τ_2 (cf. (9)):

$$\tau_1\{\} = \mathtt{tt}\ ,\ \tau_1\{\mathtt{tt}, \mathtt{ff}\} = \mathtt{tt}\ ; \qquad \tau_2\{\} = \mathtt{ff}\ ,\ \tau_2\{\mathtt{tt}, \mathtt{ff}\} = \mathtt{ff}\ .$$

Both of these, however, fail to satisfy the multiplication law in (5).

$$\begin{tikzcd} \{\{\}, \{\mathtt{ff}\}\} \arrow[r, mapsto, "\mathcal{P}\tau_1"] \arrow[d, mapsto, "\bigcup_{\mathcal{P}1}"'] & \{\mathtt{tt}, \mathtt{ff}\} \arrow[d, mapsto, "\tau_1"] \\ \{\mathtt{ff}\} \arrow[r, mapsto, "\tau_1"'] & \mathtt{ff} \neq \mathtt{tt} \end{tikzcd} \qquad \begin{tikzcd} \{\{\}, \{\mathtt{tt}\}\} \arrow[r, mapsto, "\mathcal{P}\tau_2"] \arrow[d, mapsto, "\bigcup_{\mathcal{P}1}"'] & \{\mathtt{tt}, \mathtt{ff}\} \arrow[d, mapsto, "\tau_2"] \\ \{\mathtt{tt}\} \arrow[r, mapsto, "\tau_2"'] & \mathtt{tt} \neq \mathtt{ff} \end{tikzcd}$$

Remark 2.8 *(Predicate lifting).* We note that, in the case of $T = \mathcal{P}$, the maps $(\Phi_\tau)_X: \mathbb{C}(X, T\Omega) \to \mathbb{C}(TX, T\Omega)$ (for each X) in Definition 2.4 can be identified with a natural transformation $2^{(\_)} \Rightarrow 2^{\mathcal{P}(\_)}$. The latter is commonly called a *predicate lifting* of the functor $\mathcal{P}$ in the coalgebraic literature (see e.g. [31,32]). The name comes from the fibrational study of categorical logic: given a fibration $\begin{smallmatrix}\mathbb{P}\\ \downarrow p\\ \mathbb{C}\end{smallmatrix}$ and a functor $F: \mathbb{C} \to \mathbb{C}$ on the base category, a *predicate lifting* is a functor $\varphi: \mathbb{P} \to \mathbb{P}$ such that (φ, F) forms an endomorphism of fibration. See e.g. [33,34].

Compared to the definition of predicate lifting that is usual in coalgebraic modal logic—namely, as a natural transformation $2^{(\_)} \Rightarrow 2^{\mathcal{P}(\_)}$—what our notion of PT situation (Definition 2.4) additionally yields are the following.

- Monotonicity of $2^{(\_)} \Rightarrow 2^{\mathcal{P}(\_)}$. Monotonicity is commonly assumed in the coalgebraic literature, too, and it is in particular included in the original fibrational definition of predicate lifting.
- Compatibility with the *monad* structure of $\mathcal{P}$.

In fact the latter notion of *predicate lifting* of (not just a functor, but) a monad is introduced in [35, §4], in a canonical manner that is based on the indexed category-based formulation of categorical logic. We can readily check that the maps Φ_τ in Definition 2.4 that we obtain from a PT situation satisfy the axioms in [35, §4]; in particular we can take identities as the 2-cells θ_X and ν_X in [35].

Remark 2.9 *(Naturality of Φ_τ is in $\mathbb{C}$, not in $\mathcal{K}\ell(T)$).* The map $(\Phi_\tau)_X : \mathbb{C}(X, T\Omega) \to \mathbb{C}(TX, T\Omega)$ in Definition 2.4 is natural in $X \in \mathbb{C}$—this is a consequence of the Yoneda lemma. We note that it does not necessarily imply naturality of

$$(\Phi_\tau)_X : \mathcal{K}\ell(T)(X,\Omega) \longrightarrow \mathcal{K}\ell(T)(TX,\Omega) \tag{10}$$

in $X \in \mathcal{K}\ell(T)$, although we have $\mathbb{C}(X, T\Omega) = \mathcal{K}\ell(T)(X,\Omega)$ and $\mathbb{C}(TX, T\Omega) = \mathcal{K}\ell(T)(TX,\Omega)$. Here, in (10), the last occurrence of T is understood as that of the comonad on $\mathcal{K}\ell(T)$ that arises from the Kleisli adjunction.

For a concrete counterexample, take $\mathbb{C} = \mathbf{Sets}$, $T = \mathcal{P}$ and the modality $\tau_\Box$. We shall see the naturality of (10) fails along $f : 1 \twoheadrightarrow 1$, $* \mapsto \emptyset$, in $\mathcal{K}\ell(\mathcal{P})$. It is not hard to see that $\mathcal{P}f : \mathcal{P}1 \twoheadrightarrow \mathcal{P}1$ is given by the function $\eta_{\mathcal{P}1} \circ \mu_1 \circ \mathcal{P}f : \mathcal{P}1 \to \mathcal{P}(\mathcal{P}1)$; the latter carries $\emptyset$ to $\{\emptyset\}$ and $\{*\}$ to $\{\emptyset\}$. The naturality of (10) requires that the following diagram commute.

$$\underline{\text{in } \mathcal{K}\ell(\mathcal{P})} \qquad \begin{array}{c} 1 \\ {\scriptstyle f}\uparrow \\ 1 \end{array} \qquad \underline{\text{in } \mathbf{Sets}} \qquad \begin{tikzcd} \mathcal{K}\ell(\mathcal{P})(1,1) \arrow[r, "\Phi_\tau"] \arrow[d, "(\_)\odot f"'] & \mathcal{K}\ell(\mathcal{P})(\mathcal{P}1,1) \arrow[d, "(\_)\odot \mathcal{P}f"] \\ \mathcal{K}\ell(\mathcal{P})(1,1) \arrow[r, "\Phi_\tau"] & \mathcal{K}\ell(\mathcal{P})(\mathcal{P}1,1) \end{tikzcd}$$

However this is not the case:

$$\begin{array}{ccccc} (* \mapsto \{*\} = \mathtt{tt}) & \longmapsto & & & (\emptyset \mapsto \mathtt{tt}, \{*\} \mapsto \mathtt{tt}) \\ \downarrow & & & & \downarrow \\ (* \mapsto \emptyset = \mathtt{ff}) & \longmapsto & (\emptyset \mapsto \mathtt{tt}, \{*\} \mapsto \mathtt{ff}) & \neq & (\emptyset \mapsto \mathtt{tt}, \{*\} \mapsto \mathtt{tt}) \ . \end{array}$$

2.3.3. The subdistribution monad $\mathcal{D}$: τ_{total} and τ_{partial}

The modality $\tau_{\text{total}} : \mathcal{D}[0,1] \to [0,1]$ that arises from the multiplication $\mu^{\mathcal{D}}_1$ is such that: given $q : Y \to \mathcal{D}1$ and $f : X \to \mathcal{D}Y$, we have

$$\mathbb{P}^{\mathcal{K}\ell}(\tau_{\text{total}})(f)(q)(x) \;=\; \sum_{y \in Y} q(y) \cdot f(x)(y) \ .$$

This is precisely the expected value of the random variable q under the distribution $f(x)$; thus τ_{total} yields the *probabilistic predicate transformer* of [5,6].

In parallel to the powerset monad case, we have an isomorphism $\sigma : \mathcal{D}1 \xrightarrow{\cong} \mathcal{D}1, p \mapsto 1 - p$. Another modality $\tau_{\text{partial}} : \mathcal{D}[0,1] \to [0,1]$ then arises by $\tau_{\text{partial}} := \sigma \circ \tau_{\text{total}} \circ \mathcal{D}\sigma$, that is,

$$\begin{tikzcd} \mathcal{D}(\mathcal{D}1) \arrow[r, "\mathcal{D}\sigma", "\cong"'] \arrow[d, "\tau_{\text{partial}}"'] & \mathcal{D}(\mathcal{D}1) \arrow[d, "\tau_{\text{total}}"] \\ \mathcal{D}1 & \mathcal{D}1 \arrow[l, "\cong", "\sigma"'] \end{tikzcd} \ ,$$

which is much like in (9). To put it explicitly,

$$\begin{aligned} \tau_{\text{partial}}(d) &= \bigl(1 - \textstyle\sum_{r \in [0,1]} d(r)\bigr) + \textstyle\sum_{r \in [0,1]} r \cdot d(r) \quad \text{and} \\ \mathbb{P}^{\mathcal{K}\ell}(\tau_{\text{partial}})(f)(q)(x) &= \bigl(1 - \textstyle\sum_{y \in Y} f(x)(y)\bigr) + \textstyle\sum_{y \in Y} q(y) \cdot f(x)(y) \ . \end{aligned}$$

In the second line, the value $1 - \sum_{y \in Y} f(x)(y)$—the probability of f's divergence—is added to the τ_{total} case. Therefore the modalities τ_{partial} and τ_{total}, much like in the case of $T = \mathcal{L}$, carry the flavor of *partial* and *total* correctness guarantee. In the context of probabilistic predicate transformers (also called *expectation transformers*), those which arise from τ_{partial} are commonly called *liberal* ones. See e.g. [23].

To see that τ_{partial} is indeed a modality is easy: we use Lemma 2.7; and the monotonicity can be deduced from the following explicit presentation of $\tau_{\text{partial}} \circ \mathcal{D}p$ for $p : X \to \mathcal{D}1 = [0,1]$. For each $d \in \mathcal{D}X$,

$$\begin{aligned} (\tau_{\text{partial}} \circ \mathcal{D}p)(d) &= \tau_{\text{partial}}\Bigl[r \mapsto \textstyle\sum_{x \in p^{-1}(\{r\})} d(x)\Bigr]_{r \in [0,1]} \\ &= \bigl(1 - \textstyle\sum_{r \in [0,1]} \sum_{x \in p^{-1}(\{r\})} d(x)\bigr) + \textstyle\sum_{r \in [0,1]} r \sum_{x \in p^{-1}(\{r\})} d(x) \\ &= \bigl(1 - \textstyle\sum_{x \in X} d(x)\bigr) + \textstyle\sum_{x \in X} p(x) \cdot d(x) \ . \end{aligned}$$

We do not yet know if τ_{total} and τ_{partial} are the only modalities for $\mathcal{D}$ and $\Omega = 1$.

Remark 2.10. We note the difference between a subdistribution $d \in \mathcal{D}X$ and a predicate (i.e. a random variable) $p : X \to \mathcal{D}1$. An example of the latter is $p_\top$ that is everywhere 1—this is the *truth predicate*. In contrast, the former $d \in \mathcal{D}X$ is subject to the (sub)normalization condition $\sum_x d(x) \le 1$, excluding $p_\top$ from the set $\mathcal{D}X$ (unless X is a singleton). An element $d \in \mathcal{D}X$ is therefore like one single "current state" whose whereabouts are known only probabilistically. More generally we shall think of it as a *superposed state*; see Remark 2.11 later, and also Remark 1.1.

2.3.4. The continuous subdistribution monad $\mathcal{G}$: τ_{total} and τ_{partial}

For the monad $\mathcal{G}$—a continuous analogue of $\mathcal{D}$, see Example 2.3—the situation is parallel to the one for $\mathcal{D}$. The monad multiplication $\mu_1^{\mathcal{G}}$ yields a modality by Lemma 2.5 which we denote by τ_{total}. It is such that: given $q\colon Y \to \mathcal{G}1 \cong [0,1]$ and $f\colon X \to \mathcal{G}Y$, both being measurable maps, we have

$$\mathbb{P}^{\mathcal{K}\ell}(\tau_{\mathrm{total}})(f)(q)(x) = \int_Y q \,\mathrm{d}f(x) \ ,$$

that is, the expected value of the random variable q with respect to the subprobability $f(x)$ over Y.

Since the swap isomorphism $[0,1] \overset{\cong}{\to} [0,1]$ is measurable with respect to the standard Borel structure of $[0,1]$, we have the following situation in **Meas** that yields an Eilenberg–Moore algebra τ_{partial} by Lemma 2.7.

$$\begin{tikzcd}
\mathcal{G}(\mathcal{G}1) \arrow[r, "\mathcal{G}\sigma", "\cong"'] \arrow[d, "\tau_{\mathrm{partial}}"'] & \mathcal{G}(\mathcal{G}1) \arrow[d, "\tau_{\mathrm{total}}"] \\
\mathcal{G}1 & \mathcal{G}1 \arrow[l, "\cong", "\sigma"']
\end{tikzcd}$$

Monotonicity of τ_{partial}—established much like the $\mathcal{D}$ case, relying on the monotonicity of integration—gives us another modality. The resulting weakest precondition semantics is such that

$$\mathbb{P}^{\mathcal{K}\ell}(\tau_{\mathrm{partial}})(f)(q)(x) = \Big(1 - f(x)(Y)\Big) + \int_Y q\,\mathrm{d}f(x) \ = \ \Big(1 - \int_Y \mathrm{d}f(x)\Big) + \int_Y q\,\mathrm{d}f(x) \ ,$$

with the likelihood of divergence $1 - f(x)(Y) = 1 - \int_Y \mathrm{d}f(x)$ added.

2.4. Factorization via the Eilenberg–Moore category

The indexed poset $\mathbb{P}^{\mathcal{K}\ell}(\tau)\colon \mathcal{K}\ell(T)^{\mathrm{op}} \to$ **Posets** in Proposition 2.6 is shown here to factor through the comparison functor $K\colon \mathcal{K}\ell(T) \to \mathcal{EM}(T)$, much like in (2) and (3). We shall see this K as a *superposed-state transformer semantics*—or simply a *state transformer semantics* in the terminology of [8]—for the reason we explain below.

Remark 2.11 *(Superposed-state transformer semantics).* The comparison functor $K\colon \mathcal{K}\ell(T) \to \mathcal{EM}(T)$ carries an object X to a free algebra $\begin{array}{c} T(TX) \\ \downarrow\mu \\ TX \end{array}$. An element of (the carrier object of) the latter can be naturally identified with a *mixture* or *superposition* of those of X, with respect to the branching effect specified with T—one would readily see this for the examples of $\mathcal{L}, \mathcal{P}, \mathcal{D}$ and $\mathcal{G}$. For instance, in case $T = \mathcal{D}$, an element $d \in \mathcal{D}X$ is a probability subdistribution over X and it is hence a "probabilistic superposition" of elements of X. See also Remarks 1.1 and 2.10.

The comparison functor K's action on arrows then defines a forward "(superposed-)state transformer semantics" of T-branching computations, the latter being identified with arrows $X \to TY$ in $\mathbb{C}$ (hence $X \twoheadrightarrow Y$ in $\mathcal{K}\ell(T)$). Concretely, K carries an arrow $f\colon X \twoheadrightarrow Y$ to $\mu_Y \circ Tf\colon \left(\begin{array}{c} T(TX) \\ \downarrow\mu_X \\ TX \end{array} \right) \to \left(\begin{array}{c} T(TY) \\ \downarrow\mu_Y \\ TY \end{array} \right)$ and the latter composite $\mu_Y \circ Tf$ is understood as follows. Given a superposed state $t \in TX$, the first part Tf applies f to each component state of t; and their outcomes (each with T-branching, because f is T-branching) are then "superposed" according to the original superposition present in $t \in TX$, by means of the monad multiplication μ.

In the special case of $T = \mathcal{P}$, the superposed-state transformer semantics coincides with the natural notion of *strongest postcondition semantics*—this is because a $\mathcal{P}$-superposition of states is nothing but a predicate. This is not the case in general; see Remark 2.10 for the case $T = \mathcal{D}$.

Our goal here is the diagram (11) later; it is the diagram (3)—except for the left adjoint $\mathbb{S}$—put in rigorous terms. (The left adjoint $\mathbb{S}$ will be obtained in Section 2.5.) We will be using the following result.

Lemma 2.12. *Let T be an order-enriched monad on $\mathbb{C}$, $X, Y, U \in \mathbb{C}$ and $f\colon X \to Y$ be an arrow in $\mathbb{C}$. Then $(\_) \circ f\colon \mathbb{C}(Y, TU) \to \mathbb{C}(X, TU)$ is monotone.*

Proof. Given $g\colon Y \to TU$ in $\mathbb{C}$,

$$\begin{aligned} g \circ f \ &= \mu_U \circ \eta_{TU} \circ g \circ f = \mu_U \circ Tg \circ Tf \circ \eta_X = \mu_U \circ Tg \circ Tf \circ \mu_X \circ \eta_{TX} \circ \eta_X \\ &= \mu_U \circ Tg \circ \mu_Y \circ T(Tf) \circ \eta_{TX} \circ \eta_X = \big(X \overset{J\eta_X}{\twoheadrightarrow} TX \overset{Tf}{\twoheadrightarrow} Y \overset{g}{\twoheadrightarrow} U\big) \ , \end{aligned}$$

where $J\colon \mathbb{C} \to \mathcal{K}\ell(T)$ is the Kleisli inclusion that sends the arrow $\eta_X\colon X \to TX$ to $\eta_{TX} \circ \eta_X\colon X \twoheadrightarrow TX$. In the calculation we used the monad laws as well as the naturality of η and μ. The correspondence $(\_) \odot (Tf \odot J\eta_X)$ is monotone by assumption (Definition 2.1); this proves the claim. □

Proposition 2.13 *(The indexed poset $\mathbb{P}^{\mathcal{EM}}(\tau)$). A PT situation (T, Ω, τ) induces an indexed poset $\mathbb{P}^{\mathcal{EM}}(\tau)\colon \mathcal{EM}(T)^{\mathrm{op}} \to$ **Posets** that is given by the representable functor $\mathcal{EM}(T)(\_, \tau)$. That is,*

- *On objects,*

$$\mathbb{P}^{\mathcal{EM}}(\tau)\Bigl(\begin{smallmatrix} TX \\ \downarrow a \\ X\end{smallmatrix}\Bigr) := \mathcal{EM}(T)\Bigl(\begin{smallmatrix} TX \\ \downarrow a \\ X\end{smallmatrix}, \begin{smallmatrix} T(T\Omega) \\ \downarrow \tau \\ T\Omega\end{smallmatrix}\Bigr)$$

  where the order $\sqsubseteq$ on the set $\mathcal{EM}(T)(a, \tau)$ is inherited from $\mathbb{C}(X, T\Omega)$ via the forgetful functor $U^T : \mathcal{EM}(T) \to \mathbb{C}$.
- *On an arrow $f\colon (TX \stackrel{a}{\to} X) \to (TY \stackrel{b}{\to} Y)$,*

$$\mathbb{P}^{\mathcal{EM}}(\tau)(f)\colon \mathcal{EM}(T)\Bigl(\begin{smallmatrix} TY \\ \downarrow b \\ Y\end{smallmatrix}, \begin{smallmatrix} T(T\Omega) \\ \downarrow \tau \\ T\Omega\end{smallmatrix}\Bigr) \longrightarrow \mathcal{EM}(T)\Bigl(\begin{smallmatrix} TX \\ \downarrow a \\ X\end{smallmatrix}, \begin{smallmatrix} T(T\Omega) \\ \downarrow \tau \\ T\Omega\end{smallmatrix}\Bigr) , \quad q \longmapsto q \circ f \ .$$

Proof. We only have to check the monotonicity of $\mathbb{P}^{\mathcal{EM}}(\tau)(f)$. It follows from the order-enrichment of T via Lemma 2.12. □

Theorem 2.14. *For a PT situation (T, Ω, τ), the following diagram commutes up-to a natural isomorphism. Here K is the comparison functor. e11:*

$$\begin{tikzcd} \mathbf{Posets} & & \mathcal{EM}(T)^{\mathrm{op}} \arrow[ll, "\mathbb{P}^{\mathcal{EM}}(\tau)"'] \arrow[dl, phantom, "\Psi\Uparrow\cong"] \\ & \mathcal{K}\ell(T)^{\mathrm{op}} \arrow[ul, "\mathbb{P}^{\mathcal{K}\ell}(\tau)"] \arrow[ur, "K^{\mathrm{op}}"'] & \end{tikzcd} \tag{11}$$

Proof. (Also of Proposition 2.6.) The natural isomorphism Ψ in question is of the type

$$\Psi_X\colon \quad \mathbb{P}^{\mathcal{K}\ell}(\tau)(X) = \mathbb{C}(X, T\Omega) \xrightarrow{\cong} \mathcal{EM}(T)\Bigl(\begin{smallmatrix} T(TX) \\ \downarrow \mu_X \\ TX\end{smallmatrix}, \begin{smallmatrix} T(T\Omega) \\ \downarrow \tau \\ T\Omega\end{smallmatrix}\Bigr) = \mathbb{P}^{\mathcal{EM}}(\tau)(KX)$$

and it is defined by the adjunction $\mathbb{C}(X, U^T\tau) \cong \mathcal{EM}(T)(\mu_X, \tau)$ where U^T is the forgetful functor. Explicitly: $\Psi_X\bigl(X \stackrel{p}{\to} T\Omega\bigr) = \bigl(TX \stackrel{Tp}{\to} T(T\Omega) \stackrel{\tau}{\to} T\Omega\bigr)$; and its inverse is $\Psi_X^{-1}\bigl(TX \stackrel{f}{\to} T\Omega\bigr) = \bigl(X \stackrel{\eta_X}{\to} TX \stackrel{f}{\to} T\Omega\bigr)$. The function Ψ_X is monotonic by the monotonicity of τ, see Definition 2.4; so is its inverse Ψ_X^{-1} by Lemma 2.12.

Let us turn to naturality of Ψ. Given $f\colon X \to\!\!\!\!\to Y$ in $\mathcal{K}\ell(T)$, it requires

$$\begin{tikzcd} \mathbb{C}(Y, T\Omega) \arrow[r, "\Psi_Y", "\cong"'] \arrow[d, "\mathbb{P}^{\mathcal{K}\ell}(\tau)(f) = \tau\circ T(\_)\circ f"'] & \mathcal{EM}(T)(\mu_Y, \tau) \arrow[d, "\mathbb{P}^{\mathcal{EM}}(\tau)(Kf) = (\_)\circ\mu_Y\circ Tf"] \\ \mathbb{C}(X, T\Omega) \arrow[r, "\Psi_X", "\cong"'] & \mathcal{EM}(T)(\mu_X, \tau) \ . \end{tikzcd} \tag{12}$$

Indeed, given $q\colon Y \to T\Omega$,

$$\begin{aligned} &\mathbb{P}^{\mathcal{EM}}(\tau)(Kf)(\Psi_Y q) = \mathbb{P}^{\mathcal{EM}}(\tau)(Kf)(\tau \circ Tq) = \tau \circ Tq \circ \mu_Y \circ Tf \\ &\quad = \tau \circ \mu_{T\Omega} \circ T(Tq) \circ Tf = \tau \circ T\tau \circ T(Tq) \circ Tf = \bigl(\Psi_X \circ \mathbb{P}^{\mathcal{K}\ell}(\tau)(f)\bigr)q \ , \end{aligned}$$

where the third equality is naturality of μ and the fourth is the multiplication law of τ (see (5)). By this naturality, in particular, we have that $\mathbb{P}^{\mathcal{K}\ell}(\tau)(f)$ is monotone (since the other three arrows are monotone). This is one property needed in Proposition 2.6; the other—functoriality of $\mathbb{P}^{\mathcal{K}\ell}(\tau)$—also follows from naturality of Ψ, via the functoriality of K and $\mathbb{P}^{\mathcal{EM}}(\tau)$. □

The functor $\mathbb{P}^{\mathcal{EM}}(\tau)$ turns out to be the right Kan extension of $\mathbb{P}^{\mathcal{K}\ell}(\tau)$, along the comparison functor $K\colon \mathcal{K}\ell(T) \to \mathcal{EM}(T)$. See (11). We prove this later in Section 2.6.

2.5. A dual adjunction between backward and forward semantics

Here we present the last piece missing in the triangle (3), namely the functor $\mathbb{S}$ and the dual adjunction between (backward) predicate transformer semantics and (forward superposed-)state transformer semantics. This is by an order-enriched adaptation of a folklore result (Lemma 2.15), and currently our result restricts to PT situations based on $\mathbb{C} =$ **Sets**. The result therefore covers $T = \mathcal{L}, \mathcal{P}$ and $\mathcal{D}$; but it is not clear yet if we have a similar dual adjunction for $T = \mathcal{G}$ (continuous probabilistic branching).

The following result seems to be folklore and it was brought to the author's attention by Bart Jacobs. We include its proof, since the proof of an order-enriched adaptation of the result—Lemma 2.16, a result our program logic framework relies on—builds on it.

Please cite this article in press as: I. Hasuo, Generic weakest precondition semantics from monads enriched with order, Theoret. Comput. Sci. (2015), http://dx.doi.org/10.1016/j.tcs.2015.03.047

Lemma 2.15. *Let $T\colon \mathbf{Sets} \to \mathbf{Sets}$ be a monad and $TA \xrightarrow{\alpha} A$ be an Eilenberg–Moore algebra. We have an adjunction*

$$\mathbf{Sets}^{\mathrm{op}} \underset{\mathcal{EM}(T)(\_,\alpha)}{\overset{\alpha_{(\_)}}{\underset{\longleftarrow}{\overset{\longrightarrow}{\top}}}} \mathcal{EM}(T) \quad \text{as in} \quad \dfrac{Y \longrightarrow \mathcal{EM}(T)\Bigl(\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}, \begin{smallmatrix}TA\\ \downarrow \alpha\\ A\end{smallmatrix}\Bigr) \text{ in } \mathbf{Sets}}{\Bigl(\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}\Bigr) \longrightarrow \Bigl(\begin{smallmatrix}T(A^Y)\\ \downarrow \alpha_Y\\ A^Y\end{smallmatrix}\Bigr) \text{ in } \mathcal{EM}(T)} \tag{13}$$

where the functor $\alpha_{(\_)}$ carries a set Y to an Eilenberg–Moore algebra $\alpha_Y\colon T(A^Y) \to A^Y$. Here the algebraic structure α_Y canonically arises from T's strength, namely as the adjoint transpose of

$$Y \times T(A^Y) \xrightarrow{\mathrm{str}} T(Y \times A^Y) \xrightarrow{T\mathrm{ev}} TA \xrightarrow{\alpha} A \ . \tag{14}$$

Note that any endofunctor T on **Sets** comes with a canonical strength str, essentially because **Sets** is "**Sets**-enriched." See e.g. [36].

Proof. See Appendix A. □

Below is an order-enriched adaptation of Lemma 2.15. It seems to be new. We base ourselves on the notion of PT situation; we restrict to those on **Sets**, leaving out the example $\mathcal{G}$ (on **Meas**, Section 2.3.4) for continuous probabilistic branching.

Lemma 2.16. *Let (T, Ω, τ) be a PT situation, with an additional assumption that: 1) the base category $\mathbb{C}$ is* **Sets***; and 2) the poset structure of $\mathcal{K}\ell(T)(X, Y) = \mathbf{Sets}(X, TY)$—part of the definition that T is order-enriched (Definition 2.1)—arises from the poset TY in the pointwise manner. Then we have an adjunction*

$$\mathbf{Posets}^{\mathrm{op}} \underset{\mathcal{EM}(T)(\_,\tau)}{\overset{\widetilde{\tau_{(\_)}}}{\underset{\longleftarrow}{\overset{\longrightarrow}{\top}}}} \mathcal{EM}(T) \quad \text{as in} \quad \dfrac{Y \longrightarrow \mathcal{EM}(T)\Bigl(\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}, \begin{smallmatrix}T(T\Omega)\\ \downarrow \tau\\ T\Omega\end{smallmatrix}\Bigr) \text{ in } \mathbf{Posets}}{\Bigl(\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}\Bigr) \longrightarrow \Bigl(\begin{smallmatrix}T(\mathbf{Posets}(Y,T\Omega))\\ \downarrow \widetilde{\tau_Y}\\ \mathbf{Posets}(Y,T\Omega)\end{smallmatrix}\Bigr) \text{ in } \mathcal{EM}(T)} \tag{15}$$

where the algebraic structure $\widetilde{\tau_Y}$ is given as the adjoint transpose of

$$Y \times T(\mathbf{Posets}(Y, T\Omega)) \xrightarrow{\mathrm{str}} T(Y \times \mathbf{Posets}(Y, T\Omega)) \xrightarrow{T\widetilde{\mathrm{ev}}} T(T\Omega) \xrightarrow{\tau} T\Omega \ . \tag{16}$$

Here $\widetilde{\mathrm{ev}}$ denotes evaluation of a (monotone) function, and hence is the same as the composite $Y \times \mathbf{Posets}(Y, T\Omega) \hookrightarrow Y \times \mathbf{Sets}(Y, T\Omega) \xrightarrow{\mathrm{ev}} T\Omega$.

Proof. See Appendix A. □

Combining Theorem 2.14 and Lemma 2.16 we obtain the following formalization of the informal triangle (3). The result applies to all the examples of PT situations with $T = \mathcal{L}, \mathcal{P}, \mathcal{D}$ in Section 2.3, but not to $T = \mathcal{G}$ since it is based on **Meas**.

Corollary 2.17. *Under the assumptions of Lemma 2.16 we have the following situation, with a natural isomorphism $\mathbb{P}^{\mathcal{K}\ell}(\tau) \overset{\cong}{\Rightarrow} \mathbb{P}^{\mathcal{EM}}(\tau) \circ K^{\mathrm{op}}$.*

$$\begin{tikzcd} \mathbf{Posets} \arrow[rr, bend left, "\mathbb{S}"] \arrow[rr, phantom, "\perp"] & & \mathcal{EM}(T)^{\mathrm{op}} \arrow[ll, bend left, "\mathbb{P}^{\mathcal{EM}}(\tau)"] \\ & \mathcal{K}\ell(T)^{\mathrm{op}} \arrow[ul, "\mathbb{P}^{\mathcal{K}\ell}(\tau)"] \arrow[ur, "K^{\mathrm{op}}"'] & \end{tikzcd} \qquad \square \tag{17}$$

2.6. A Kan extension

In the diagram (11) (also in (17)), it turns out that the functor $\mathbb{P}^{\mathcal{EM}}(\tau)$ is the right Kan extension of $\mathbb{P}^{\mathcal{K}\ell}(\tau)$ along the comparison functor K. We shall prove this fact here. The proof works for a general base category $\mathbb{C}$ (not just $\mathbb{C} = \mathbf{Sets}$).

We start with a prototype result that is not order-enriched yet. An algebra $\alpha\colon TA \to A$ in the following lemma will later be instantiated by $\tau\colon T(T\Omega) \to T\Omega$, in which case F will be $\mathbb{P}^{\mathcal{K}\ell}(\tau)$. Its proof relies on arguments similar to those in Beck's monadicity theorem.

Lemma 2.18. *Let T be a monad on a category $\mathbb{C}$, and $\alpha: TA \to A$ be an Eilenberg–Moore algebra. Consider*

$$\begin{tikzcd} \mathbf{Sets} & & \mathcal{EM}(T)^{\mathrm{op}} \arrow[ll, "\mathcal{EM}(T)\left(\_\,,\, \substack{TA\\ \downarrow\alpha \\ A}\right)"'] \\ & \mathcal{K}\ell(T)^{\mathrm{op}} \arrow[ul, "F"] \arrow[ur, "K^{\mathrm{op}}"'] & \end{tikzcd} \tag{18}$$

where K is the comparison functor, and the functor F is defined as follows (much like in Proposition 2.6).

$$FX := \mathbb{C}(X, A) \ ; \quad F(X \overset{f}{\twoheadrightarrow} Y)\big(Y \overset{q}{\to} A\big) := \big(X \overset{f}{\to} TY \overset{Tq}{\to} TA \overset{\alpha}{\to} A\big) \ .$$

Then the functor $\mathcal{EM}(T)\left(\_\,,\, \substack{TA\\ \downarrow\alpha \\ A}\right)$ is the right Kan extension of F along K^{op}.

Proof. See Appendix A. □

Corollary 2.19. *Let (T, Ω, τ) be a PT situation. In the diagram (11), $\mathbb{P}^{\mathcal{EM}}(\tau)$ is the right Kan extension of $\mathbb{P}^{\mathcal{K}\ell}(\tau)$ along K^{op}.*

Proof. See Appendix A. □

3. Order enrichment and PT situations in Sets, concretely

In this short section we rephrase the monotonicity requirement on PT situations into more concrete terms, in the special case of $\mathbb{C} = \mathbf{Sets}$. This is via the notions of *congruence* and *substitutivity* of an order on a monad, introduced and studied in [17] (see also [37]).

Definition 3.1 *(Congruence, substitutivity [17]).* Let T be a monad on **Sets**, and $\sqsubseteq^T = \big(\sqsubseteq^T_I \subseteq TI \times TI\big)_{I \in \mathbf{Sets}}$ be an assignment of a partial order to the set TI, for each set I.

- $\sqsubseteq^T$ is said to be *congruent* if the mapping
$$\mathbf{Sets}(J, TI) \longrightarrow \mathbf{Sets}(TJ, TI) \ , \quad t \longmapsto t^\dagger = \mu^T_I \circ Tt$$
is monotone for each J. Here the order in each homset is the pointwise extension of $\sqsubseteq^T_I$ on TI.
- $\sqsubseteq^T$ is said to be *substitutive* if, for any set I and any function $t: I \to TI$, its Kleisli extension $t^\dagger: TI \to TI$ is monotone (with respect to $\sqsubseteq^T_I$).

These properties are best understood in the correspondence (see e.g. [38]) between monads and algebraic theories. The set TI is that of (algebraic) terms with variables from I (modulo equational axioms); hence $t: J \to TI$ is a J-indexed array $(t_j)_{j\in J}$ of such terms. Its Kleisli extension $t^\dagger: TJ \to TI$ then carries a term $s \in TJ$ (with variables from $J = \{x_j \mid j \in J\}$) to $s[t_j/x_j]$. Following this line, $\sqsubseteq^T$ being congruent means that $t_j \sqsubseteq^T t'_j$ for each j implies $s[t_j/x_j] \sqsubseteq^T s[t'_j/x_j]$; its being substitutive means that $s \sqsubseteq^T s'$ (where $s, s' \in TI$, the domain of $t^\dagger$) implies $s[t_i/x_i] \sqsubseteq^T s'[t_i/x_i]$.

The following result (and the previous definition) are in [17, §2]. The proof is included for the record.

Proposition 3.2. *(See [17].) Let T be a monad on* **Sets**. *An order $\sqsubseteq^T$ on T that is congruent and substitutive is in a bijective correspondence with an order-enrichment of $\mathcal{K}\ell(T)$ that arises in the pointwise manner.*

Proof. Given $\sqsubseteq^T$ that is congruent and substitutive, we need to check that its pointwise extension—for $f, g: X \rightrightarrows TY$, we define $f \sqsubseteq g$ if and only if $f(x) \sqsubseteq^T_Y g(x)$—makes the Kleisli composition $\odot$ monotone. Let $f, f': X \to TY$, $g, g': Y \to TU$, $f \sqsubseteq f'$ and $g \sqsubseteq g'$. We have, for each $x \in X$,

$$(g \odot f)(x) \;=\; g^\dagger(f(x)) \;\sqsubseteq\; g^\dagger(f'(x)) \;\sqsubseteq\; (g')^\dagger(f'(x)) \;=\; (g' \odot f')(x) \ ,$$

where the two inequalities are due to substitutivity and congruence, respectively.

Conversely, assume a pointwise order-enrichment of $\mathcal{K}\ell(T)$. To see the order $\sqsubseteq^T$ being congruent, observe that for $t: J \to TI$, the arrow $t^\dagger$ is equal to $TJ \overset{(\mathrm{id}_{TJ})^\wedge}{\twoheadrightarrow} J \overset{t}{\twoheadrightarrow} I$. Here $(\mathrm{id}_{TJ})^\wedge$ is the function $TJ \to TJ$ in **Sets** considered to be an arrow in $\mathcal{K}\ell(T)$. The correspondence $t \mapsto t^\dagger$ is monotone since $\odot$ is. Substitutivity of $\sqsubseteq^T$ follows from the fact that for $t: I \to TI$ and $s \in TI$, the element $t^\dagger(s)$ is nothing but $1 \overset{s}{\twoheadrightarrow} I \overset{t}{\twoheadrightarrow} I$. □

We shall now adapt the above characterization of order-enrichments of a monad T, to PT situations. This is the technical contribution of the current section.

Please cite this article in press as: I. Hasuo, Generic weakest precondition semantics from monads enriched with order, Theoret. Comput. Sci. (2015), http://dx.doi.org/10.1016/j.tcs.2015.03.047

Definition 3.3. Let $\sqsubseteq^T$ be an assignment of an order on a monad T, as before.

- An algebra $T(T\Omega) \xrightarrow{\tau} T\Omega$ is *order-congruent* if, for each J, the mapping $\mathbf{Sets}(J, T\Omega) \to \mathbf{Sets}(TJ, T\Omega), t \mapsto \bigl(TJ \xrightarrow{Tt} T(T\Omega) \xrightarrow{\tau} T\Omega\bigr)$ is monotone (with respect to the pointwise order).
- τ is *order-substitutive* if, for each J and $t\colon J \to T\Omega$, the function $\tau \circ (Tt)\colon TJ \to T\Omega$ is monotone (with respect to $\sqsubseteq^T_J$ and $\sqsubseteq^T_\Omega$).

The following (obvious) characterization offers a concrete syntactic view of the monotonicity condition on a PT situation, as we will illustrate shortly by examples.

Proposition 3.4. *An algebra $T(T\Omega) \xrightarrow{\tau} T\Omega$ satisfies the monotonicity condition of Definition 2.4 if and only if it is order-congruent.* □

Example 3.5. The modality $\tau_\Box\colon \mathcal{P}(\mathcal{P}1) \to \mathcal{P}1$ in Example 2.3, while it is order-congruent, fails to be order-substitutive. Indeed, let $t\colon 1 \to \mathcal{P}1$ be defined by $t(*) = \mathtt{ff}$. For the function $\tau_\Box \circ \mathcal{P}t\colon \mathcal{P}1 \to \mathcal{P}1$ we have $\tau_\Box(\mathcal{P}t(\emptyset)) = \tau_\Box(\emptyset) = \mathtt{tt}$ and $\tau_\Box(\mathcal{P}t(\{*\})) = \tau_\Box(\{\mathtt{ff}\}) = \mathtt{ff}$, while $\emptyset \sqsubseteq \{*\}$ in $\mathcal{P}1$. Notice that $\tau_\Box$ is not even monotone[6]: $\tau_\Box\{\} = \{0\}$ and $\tau_\Box\{0\} = \{\}$.

The previous definition is again best understood in the language of algebraic terms. Let us be concrete, for illustration, by setting $T = \mathcal{P}$, $\Omega = 1$ and $J = \{x_0, x_1, x_2\}$. We write $T\Omega = \mathcal{P}1 = \bigl\{\mathtt{tt} := \{0\},\ \mathtt{ff} := \emptyset\bigr\}$, as before. An arrow $t\colon J \to T\Omega$ is a *valuation*, i.e. an assignment of truth values ($\mathtt{tt}$ or $\mathtt{ff}$) to each variable x_i, such as

$$t_0 \;=\; \bigl[\,x_0 \mapsto \mathtt{ff}, x_1 \mapsto \mathtt{tt}, x_2 \mapsto \mathtt{ff}\,\bigr] \quad\text{and}\quad t_1 \;=\; \bigl[\,x_0 \mapsto \mathtt{tt}, x_1 \mapsto \mathtt{tt}, x_2 \mapsto \mathtt{tt}\,\bigr]\ . \tag{19}$$

Look at the definition of order-congruence: the arrow $\tau \circ Tt\colon TJ \to T\Omega$ there is easily seen to carry an algebraic term s—like $s_0 = \{x_2, x_1, x_0\}$—to its *interpretation* $[\![s]\!]_{\tau,t}$ in the algebra τ under the valuation t. Order-congruence requires this interpretation to be monotone with respect to valuations, that is, $t \sqsubseteq t'$ implies $[\![s]\!]_{\tau,t} \sqsubseteq [\![s]\!]_{\tau,t'}$. To see that this holds for s_0, $t_0 \sqsubseteq t_1$ defined in (19), and $\tau = \tau_\Box$, we observe

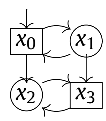
$$\begin{aligned}
[\![s_0]\!]_{\tau_\Box,t_0} \;&=\; [\![\{x_2, x_1, x_0\}]\!]_{\tau_\Box,t_0} \;=\; \tau_\Box\bigl(\{\mathtt{ff}, \mathtt{tt}, \mathtt{ff}\}\bigr) \;=\; \mathtt{ff} \\
&\sqsubseteq\; \mathtt{tt} \;=\; \tau_\Box\bigl(\{\mathtt{tt}, \mathtt{tt}, \mathtt{tt}\}\bigr) \;=\; [\![\{x_2, x_1, x_0\}]\!]_{\tau_\Box,t_1} \;=\; [\![s_0]\!]_{\tau_\Box,t_1}\ .
\end{aligned}$$

On order-substitutivity in Definition 3.3, using the same notations as above, it stipulates that $s \sqsubseteq s'$ (where s, s' are terms, i.e. elements of TJ) implies $[\![s]\!]_{\tau,t} \sqsubseteq [\![s']\!]_{\tau,t}$. This fails for $\tau = \tau_\Box$: let $s_1 = \{x_1\}$, $s_2 = \{x_0, x_1\}$; then

$$[\![s_1]\!]_{\tau_\Box,t_0} = \tau_\Box\bigl(\{\mathtt{tt}\}\bigr) = \mathtt{tt} \not\sqsubseteq \mathtt{ff} = \tau_\Box\bigl(\{\mathtt{ff}, \mathtt{tt}\}\bigr) = [\![s_2]\!]_{\tau_\Box,t_0}\ .$$

4. The two-player setting: introduction

We extend the basic framework in the previous section by adding another layer of branching. This corresponds to adding another "player" in computations or systems. The additional player typically has an interest that conflicts with the original player's: the former shall be called *Opponent* and denoted by O, while the latter (the original player) is called *Player* P.[7]



The need for two players with conflicting interests is pervasive in computer science. One example is the (nowadays heavy) use of *games* in the automata-theoretic approach to model checking (see e.g. [39]). Games here can be understood as a two-player version of automata, where it is predetermined which player makes a move in each state. An example is above on the right, where P-states are x_0, x_3 and O-states are x_1, x_2. Typical questions asked here are about what Player P *can force*: can P force that x_3 be reached? (yes); can P force that x_0 be visited infinitely often? (no). In model checking, the dualities between $\wedge$ and $\vee$, ν and μ, etc. in the modal μ-calculus are conveniently expressed as the duality between P and O; and many algorithms and proofs rely on suitably formulated games and results on them (such as the algorithm in [40] that decides the winner of a parity game). Games have also been used in the coalgebraic study of fixed-point logics [41].

Another example of nested two-player branching is found in the process-theoretic study of *probabilistic systems*; see e.g. [26,42]. There it is common to include nondeterministic branching too: while probabilistic branching models the behavior of a *system* (such as a stochastic algorithm) that flips an internal coin, nondeterministic branching models the *environment*'s behavior (such as requests from users) on which no statistical information is available. In this context, probabilistic branching is often called *angelic* while nondeterministic one is *demonic*; and a common verification goal would be to ensure a property—with a certain minimal likelihood—whatever demonic choices are to be made.

---

[6] Order-substitutivity implies monotonicity; take $\mathrm{id}_{T\Omega}\colon T\Omega \to T\Omega$ as t in Definition 3.3.

[7] Note that (capitalized) *Player* and *Opponent* are altogether called *players*.

Please cite this article in press as: I. Hasuo, Generic weakest precondition semantics from monads enriched with order, Theoret. Comput. Sci. (2015), http://dx.doi.org/10.1016/j.tcs.2015.03.047

4.1. Leading example: nondeterministic P and nondeterministic O

Let us first focus on the simple setting where: P moves first and O moves second, in each round; and both P and O make nondeterministic choices. This is a setting suited e.g. for bipartite games where P plays first. A computation with such branching is modeled by a function

$$f : X \longrightarrow \mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}} Y) \ , \tag{20}$$

where the occurrences of the powerset functor $\mathcal{P}$ are annotated to indicate which of the players it belongs to (hence $\mathcal{P}_{\mathsf{P}} = \mathcal{P}_{\mathsf{O}} = \mathcal{P}$). We are interested in what P can force; in this *logic of forced predicates*, the following notion of (pre)order seems suitable.

$$\mathfrak{a} \sqsubseteq \mathfrak{b} \quad \text{in } \mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}} Y) \quad \overset{\text{def.}}{\Longleftrightarrow} \quad \forall S \in \mathfrak{a}.\, \exists S' \in \mathfrak{b}.\, S' \subseteq_{\mathcal{P}_{\mathsf{O}} Y} S \tag{21}$$

That is: if $\mathfrak{a}$ can force Opponent to $S \subseteq Y$, then $\mathfrak{b}$—that has a greater power—can force Opponent to better (i.e. smaller) $S' \subseteq Y$.

In fact, we shall now introduce a modeling alternative to (20) which uses up-closed families of subsets, and argue for its superiority, mathematical and conceptual. It paves the way to our general setup in Section 5.

For a set Y, we define $\mathcal{UP}Y$ to be the collection of *up-closed* families of subsets of Y, that is,

$$\mathcal{UP}Y \ := \ \bigl\{\mathfrak{a} \subseteq \mathcal{P}Y \bigm| \forall S, S' \subseteq Y.\, (S \in \mathfrak{a} \wedge S \subseteq S' \Rightarrow S' \in \mathfrak{a})\bigr\} \ . \tag{22}$$

On $\mathcal{UP}Y$ we define a relation $\sqsubseteq$ by: $\mathfrak{a} \sqsubseteq \mathfrak{b}$ if $\mathfrak{a} \subseteq \mathfrak{b}$. It is obviously a partial order.

Lemma 4.1.

1. *For each set Y, the relation $\sqsubseteq$ in (21) on $\mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}} Y)$ is a preorder. It is not a partial order.*
2. *For $\mathfrak{a} \in \mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}} Y)$, let $\uparrow \mathfrak{a} := \{S \mid \exists S' \in \mathfrak{a}.\, S' \subseteq S\}$ be its* upward closure. *Then the following is an equivalence of (preorders considered to be) categories; here ι is the obvious inclusion map.*

$$\begin{tikzcd} \mathcal{UP}Y \arrow[r, bend right, "\iota"'] \arrow[r, phantom, "\simeq"] & \mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}} Y) \arrow[l, bend right, "\uparrow(\_)"'] \end{tikzcd}$$

Proof. For 1., reflexivity and transitivity of $\sqsubseteq$ is obvious. To see it is not antisymmetric consider $\{\emptyset, Y\}$ and $\{\emptyset\}$.

For 2., ι is obviously monotone. If $\mathfrak{a} \sqsubseteq \mathfrak{b}$ in $\mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}} Y)$, for any $S \in \uparrow \mathfrak{a}$ there exists $S' \in \mathfrak{b}$ such that $S' \subseteq S$, hence $S \in \uparrow \mathfrak{b}$. Therefore $\uparrow(\_)$ is monotone too. Obviously $\uparrow(\_) \circ \iota = \mathrm{id}$.

It must be checked that $\iota(\uparrow \mathfrak{a}) \simeq \mathfrak{a}$ for $\mathfrak{a} \in \mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}} Y)$, where $\simeq$ is the equivalence induced by $\sqsubseteq$. The $\sqsubseteq$ direction is immediate from the definition of $\uparrow \mathfrak{a}$; for the other direction, observe that in general $\mathfrak{a} \subseteq \mathfrak{b}$ implies $\mathfrak{a} \sqsubseteq \mathfrak{b}$ in $\mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}} Y)$. □

Proposition 4.2. *For each set Y, $(\mathcal{UP}Y, \sqsubseteq)$ is the poset induced by the preorder $\bigl(\mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}} Y), \sqsubseteq\bigr)$. Moreover $(\mathcal{UP}Y, \sqsubseteq)$ is a complete lattice.* □

Proof. The first half is immediate from Lemma 4.1. For the latter, observe that supremums are given by unions. □

The constructions $\mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}}\_)$ and $\mathcal{UP}\_$ have been studied from a coalgebraic perspective in the context of *neighborhood frames* [43,44]. There a coalgebra for the former is a model of *non-normal* modal logic (meaning that axioms like $\Box p \wedge \Box q \to \Box(p \wedge q)$ and $\Box p \to \Box(p \vee q)$ can fail); one for the latter is a model of *monotone* modal logic (meaning that validity of $\Box p \to \Box(p \vee q)$ is retained). Proposition 4.2 shows that, as long as our interests are game-theoretic and are in the logical reasoning with respect to the preorder $\sqsubseteq$ in (21), we may just as well use $\mathcal{UP}\_$ which is mathematically better-behaved.

To argue further for the mathematical convenience of $\mathcal{UP}\_$, we look at its action on arrows. For $\mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}}\_)$ there are two obvious choices ($\mathcal{PP}f$ and 2^{2^f}) of action on arrows, arising from the covariant and contravariant powerset functors, respectively. Given $f : X \to Y$ in **Sets**,

$$\begin{aligned} &\mathcal{PP}f,\ 2^{2^f} : \mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}} X) \longrightarrow \mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}} Y) \ , \\ &(\mathcal{PP}f)\mathfrak{a} := \{\amalg_f S \mid S \in \mathfrak{a}\} \ , \quad 2^{2^f}\mathfrak{a} := \{T \subseteq Y \mid f^{-1}T \in \mathfrak{a}\} \ . \end{aligned}$$

Here $\amalg_f S$ is the direct image of S by f.

These two choices are not equivalent with respect to $\sqsubseteq$ on $\mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}} Y)$. In general we have $2^{2^f}\mathfrak{a} \sqsubseteq (\mathcal{PP}f)\mathfrak{a}$. To see that, assume $U \in 2^{2^f}\mathfrak{a}$, i.e. $f^{-1}U \in \mathfrak{a}$. Then we have $\amalg_f(f^{-1}U) \subseteq U$ (a general fact, between $\amalg_f$ and f^{-1}) and $\amalg_f(f^{-1}U) \in (\mathcal{PP}f)\mathfrak{a}$ (by $f^{-1}U \in \mathfrak{a}$); hence $2^{2^f}\mathfrak{a} \sqsubseteq (\mathcal{PP}f)\mathfrak{a}$ by (21). However the converse $2^{2^f}\mathfrak{a} \sqsupseteq (\mathcal{PP}f)\mathfrak{a}$ can fail: consider $! : 2 \to 1$ (where $2 = \{0, 1\}$) and $\mathfrak{a} = \{\{0\}\}$; then $2^{2^f}\mathfrak{a} = \emptyset$ while $(\mathcal{PP}f)\mathfrak{a} = \{1\}$.

This discrepancy is absent with $\mathcal{UP}\_$. For a function $f : X \to Y$, the "covariant" action $\mathcal{UP}f$ and the "contravariant" action $\mathcal{UP}'f$ are defied as follows.

$$
\begin{tikzcd}
\mathcal{UP}X \arrow[r, "\mathcal{UP}f"] \arrow[d, hook, "\iota"'] & \mathcal{UP}Y \\
\mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}}X) \arrow[r, "\mathcal{PP}f"'] & \mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}}Y) \arrow[u, "\uparrow(\_)"']
\end{tikzcd}
\qquad
\begin{tikzcd}
\mathcal{UP}X \arrow[r, dashed, "\mathcal{UP}'f"] \arrow[d, hook] & \mathcal{UP}Y \arrow[d, hook] \\
\mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}}X) \arrow[r, "2^{2^f}"'] & \mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}}Y)
\end{tikzcd}
\tag{23}
$$

On the left, ι and $\uparrow(\_)$ are as in Lemma 4.1. On the right 2^{2^f} restricts to $\mathcal{UP}X \to \mathcal{UP}Y$ (easy by the fact that f^{-1} is monotone); on the left such is not the case (consider $f : 1 \to 2$, $0 \mapsto 0$ and $\mathfrak{a} = \{1\}$) and we need explicit use of $\uparrow(\_)$.

Lemma 4.3. $\mathcal{UP}f = \mathcal{UP}'f$.

Proof. Let $\mathfrak{a} \in \mathcal{UP}X$ (hence up-closed). In view of Lemma 4.1, it suffices to show that $2^{2^f}\mathfrak{a} \simeq (\mathcal{PP}f)\mathfrak{a}$; we have already proved the $\sqsubseteq$ direction. For the other direction, let $S \in \mathfrak{a}$; proving $\amalg_f S \in 2^{2^f}\mathfrak{a}$ will prove $(\mathcal{PP}f)\mathfrak{a} \subseteq 2^{2^f}\mathfrak{a}$, hence $(\mathcal{PP}f)\mathfrak{a} \sqsubseteq 2^{2^f}\mathfrak{a}$. That $S \subseteq f^{-1}(\amalg_f S)$ is standard; since $\mathfrak{a}$ is up-closed we have $f^{-1}(\amalg_f S) \in \mathfrak{a}$. Therefore $\amalg_f S \in (2^{2^f})\mathfrak{a}$. $\square$

We therefore define $\mathcal{UP} : \mathbf{Sets} \to \mathbf{Sets}$ by (22) on objects and either of the actions in (23) on arrows. Its functoriality is obvious from (23) on the right.

4.2. Nondeterministic O*, then probabilistic* P*: search for modularity*

We have argued for the convenience of the functor $\mathcal{UP}$, over $\mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}}\_)$, for modeling alternating branching in games. A disadvantage, however, is that *modularity* is lost. Unlike $\mathcal{P}_{\mathsf{P}}(\mathcal{P}_{\mathsf{O}}\_)$, the functor $\mathcal{UP} : \mathbf{Sets} \to \mathbf{Sets}$ is not an obvious composite of two functors, each of which modeling each player's choice.

The same issue arises also in the systems with both probabilistic and nondeterministic branching (briefly discussed before). It is known (an observation by Gordon Plotkin; see e.g. [45]) that there is no distributive law $\mathcal{DP} \Rightarrow \mathcal{PD}$ of the subdistribution monad $\mathcal{D}$ over the powerset monad $\mathcal{P}$. This means we cannot compose them to obtain a new monad $\mathcal{PD}$. Two principal fixes have been proposed: one is to refine $\mathcal{D}$ into the *indexed valuation monad* [45], where $[x \mapsto 1/2, x \mapsto 1/2]$ and $[x \mapsto 1]$ are two different indexed valuations (they are the same when seen as probability distributions). The other way (see e.g. [46]) replaces $\mathcal{P}$ by the *convex powerset construction* and uses

$$
\mathcal{CD}X \;:=\; \{\mathfrak{a} \subseteq \mathcal{D}X \mid p_i \in [0,1], \textstyle\sum_i p_i = 1, d_i \in \mathfrak{a} \Rightarrow \sum_i p_i d_i \in \mathfrak{a}\}
$$

in place of $\mathcal{PD}$, an alternative we favor due to our process-theoretic interests (see Remark 6.8 later). However, much like with $\mathcal{UP}$, it is not immediate how to decompose $\mathcal{CD}$ into Player and Opponent parts.

In the rest of the paper we shall present a categorical setup that addresses this issue of separating two players. It does so by identifying one out of the two layers of branching—like up-closed powerset and convex powerset—as a monad on an Eilenberg–Moore category.

5. Generic two-player weakest precondition semantics

Definition 5.1 *(2-player PT situation).* A *2-player predicate transformer situation* over a category $\mathbb{C}$ is a quintuple $(T, \Omega, \tau, R, \rho)$ where:

- (T, Ω, τ) is a PT situation (Definition 2.4), where in particular $\tau : T(T\Omega) \to T\Omega$ is an Eilenberg–Moore algebra;
- R is a monad on the Eilenberg–Moore category $\mathcal{EM}(T)$; and
- $\rho : R\left(\begin{smallmatrix} T(T\Omega) \\ \downarrow\tau \\ T\Omega \end{smallmatrix}\right) \to \left(\begin{smallmatrix} T(T\Omega) \\ \downarrow\tau \\ T\Omega \end{smallmatrix}\right)$ is an Eilenberg–Moore R-algebra, that is also called a *modality*. It is further subject to the *monotonicity condition* that is much like in Definition 2.4: the map

$$
\mathcal{EM}(T)\left(\begin{smallmatrix} TX \\ \downarrow a \\ X \end{smallmatrix}, \begin{smallmatrix} T(T\Omega) \\ \downarrow\tau \\ T\Omega \end{smallmatrix}\right) \longrightarrow \mathcal{EM}(T)\left(R\left(\begin{smallmatrix} TX \\ \downarrow a \\ X \end{smallmatrix}\right), \begin{smallmatrix} T(T\Omega) \\ \downarrow\tau \\ T\Omega \end{smallmatrix}\right), \quad f \longmapsto \rho \circ Rf
$$

is monotone for each algebra a. Here the order of each homset is induced by the enrichment of $\mathcal{K\ell}(T)$ via $\mathcal{EM}(T)(b, \tau) \xrightarrow{U^T} \mathbb{C}(U^T b, T\Omega) = \mathcal{K\ell}(T)(U^T b, \Omega)$.

The situation is as in the following diagram.

Please cite this article in press as: I. Hasuo, Generic weakest precondition semantics from monads enriched with order, Theoret. Comput. Sci. (2015), http://dx.doi.org/10.1016/j.tcs.2015.03.047

$$
\begin{tikzcd}
\mathbb{C} \arrow[loop above, "T"] \arrow[loop left, "{U^T U^R F^R F^T = U^T R F^T}"] \arrow[r, bend right, "F^T"'] \arrow[r, phantom, "\top"] \arrow[drr, bend right, "\top"'] & \mathcal{EM}(T) \arrow[loop above, "R"] \arrow[l, bend right, "U^T"'] \arrow[r, bend right, "F^R"'] \arrow[r, phantom, "\top"] & \mathcal{EM}(R) \arrow[l, bend right, "U^R"'] \\
& & \mathcal{K}\ell(U^T R F^T) \arrow[u, "K"] \arrow[ull, bend left]
\end{tikzcd}
\tag{24}
$$

The composite adjunction yields a new monad $U^T U^R F^R F^T = U^T R F^T$ on $\mathbb{C}$; then from the Kleisli category $\mathcal{K}\ell(U^T R F^T)$ for the new monad we obtain a comparison functor to $\mathcal{EM}(R)$. It is denoted by K.[8]

We have a monad R on $\mathcal{EM}(T)$ and an algebra (modality) ρ for it. This is much like in the original notion of PT situation, where $\tau: T(T\Omega) \to T\Omega$ is a modality from which we derived a weakest precondition semantics. Indeed, the following construction is parallel to Proposition 2.13.

Proposition 5.2 *(The indexed poset $\mathbb{P}^{\mathcal{EM}}(\tau,\rho)$). A 2-player PT situation $(T, \Omega, \tau, R, \rho)$ induces an indexed poset $\mathbb{P}^{\mathcal{EM}}(\tau,\rho)$: $\mathcal{EM}(R)^{\mathrm{op}} \to$ **Posets** over $\mathcal{EM}(T)$ by:*

- *on an object $\alpha \in \mathcal{EM}(R)$,*

$$
\mathbb{P}^{\mathcal{EM}}(\tau,\rho)\Bigl(\begin{smallmatrix} R(TX \xrightarrow{a} X) \\ \downarrow\alpha \\ (TX \xrightarrow{a} X) \end{smallmatrix}\Bigr) := \mathcal{EM}(R)\Bigl(\begin{smallmatrix} R(TX \xrightarrow{a} X) \\ \downarrow\alpha \\ (TX \xrightarrow{a} X) \end{smallmatrix}, \begin{smallmatrix} R(T(T\Omega) \xrightarrow{\tau} T\Omega) \\ \downarrow\rho \\ (T(T\Omega) \xrightarrow{\tau} T\Omega) \end{smallmatrix}\Bigr)
$$

where the order $\sqsubseteq$ on the set $\mathcal{EM}(R)(\alpha,\rho)$ is inherited from $\mathbb{C}(X, T\Omega)$ via the forgetful functors $\mathcal{EM}(R) \to \mathcal{EM}(T) \to \mathbb{C}$; and

- *on an arrow $f: \bigl(\begin{smallmatrix} Ra \\ \downarrow\alpha \\ a \end{smallmatrix}\bigr) \to \bigl(\begin{smallmatrix} Rb \\ \downarrow\beta \\ b \end{smallmatrix}\bigr)$,*

$$
\mathbb{P}^{\mathcal{EM}}(\tau,\rho)(f) : \mathcal{EM}(R)\Bigl(\begin{smallmatrix} Rb \\ \downarrow\beta \\ b \end{smallmatrix}, \begin{smallmatrix} R\tau \\ \downarrow\rho \\ R\tau \end{smallmatrix}\Bigr) \longrightarrow \mathcal{EM}(R)\Bigl(\begin{smallmatrix} Ra \\ \downarrow\alpha \\ a \end{smallmatrix}, \begin{smallmatrix} R\tau \\ \downarrow\rho \\ \tau \end{smallmatrix}\Bigr) , \quad q \longmapsto q \circ f \ .
$$

Proof. The same as the proof of Proposition 2.13, relying on Lemma 2.12. □

Much like in Theorem 2.14, composition of this indexed poset $\mathbb{P}^{\mathcal{EM}}(\tau,\rho): \mathcal{EM}(R)^{\mathrm{op}} \to$ **Posets** and the comparison functor $K: \mathcal{K}\ell(U^T R F^T) \to \mathcal{EM}(R)$ will yield the weakest precondition calculus. The branching computations of our interest are therefore of the type $X \to U^T R F^T Y$. We will later see, through examples, that this is indeed what models the scenarios in Section 4.

Note that in what follows we rely heavily on the adjunction $F^T \dashv U^T$.

Proposition 5.3 *(The indexed poset $\mathbb{P}^{\mathcal{K}\ell}(\tau,\rho)$). A 2-player PT situation $(T, \Omega, \tau, R, \rho)$ induces an indexed poset $\mathbb{P}^{\mathcal{K}\ell}(\tau,\rho)$: $\mathcal{K}\ell(U^T R F^T)^{\mathrm{op}} \to$ **Posets** by:*

- *on an object $X \in \mathcal{K}\ell(U^T R F^T)$, $\mathbb{P}^{\mathcal{K}\ell}(\tau,\rho)(X) := \mathcal{K}\ell(T)(X,\Omega) = \mathbb{C}(X, T\Omega)$;*
- *given an arrow $f: X \nrightarrow Y$ in $\mathcal{K}\ell(U^T R F^T)$, it induces an arrow $f^\wedge: F^T X \to R(F^T Y)$ in $\mathcal{EM}(T)$; this is used in*

$$
\mathcal{EM}(T)(F^T Y, \tau) \longrightarrow \mathcal{EM}(T)(F^T X, \tau) , \quad q \longmapsto \bigl(F^T X \xrightarrow{f^\wedge} R(F^T Y) \xrightarrow{Rq} R\tau \xrightarrow{\rho} \tau\bigr) \ .
$$

The last map defines an arrow $\mathbb{P}^{\mathcal{K}\ell}(\tau,\rho)(f): \mathbb{P}^{\mathcal{K}\ell}(\tau,\rho)(Y) \to \mathbb{P}^{\mathcal{K}\ell}(\tau,\rho)(X)$ since we have $\mathbb{P}^{\mathcal{K}\ell}(\tau,\rho)(U) = \mathbb{C}(U, T\Omega) \cong \mathcal{EM}(T)(F^T U, \tau)$.

We have the following natural isomorphism, where K is the comparison in (24).

$$
\begin{tikzcd}
\mathbf{Posets} & & \mathcal{EM}(R)^{\mathrm{op}} \arrow[ll, bend right, "\mathbb{P}^{\mathcal{EM}}(\tau,\rho)"'] \\
& \mathcal{K}\ell(U^T R F^T)^{\mathrm{op}} \arrow[ul, "\mathbb{P}^{\mathcal{K}\ell}(\tau,\rho)"] \arrow[ur, "K^{\mathrm{op}}"'] \arrow[u, phantom, "\Xi \bullet \Psi \Uparrow \cong"]
\end{tikzcd}
\tag{25}
$$

Proof. Note here that the comparison functor K is concretely described as follows: $KX = F^R(F^T X)$ on objects, and use the correspondence

$$
\begin{aligned}
&\mathcal{K}\ell(U^T R F^T)(X,Y) = \mathbb{C}(X, U^T U^R F^R F^T Y) \cong \mathcal{EM}(T)(F^T X, U^R F^R F^T Y) \\
&\quad \cong \mathcal{EM}(R)(F^R F^T X, F^R F^T Y) = \mathcal{EM}(R)(KX, KY)
\end{aligned}
$$

[8] The existence of K exploits the universality of the Kleisli category $\mathcal{K}\ell(U^T R F^T)$ (see e.g. [18, Thm. VI.5.2]) and not that of $\mathcal{EM}(R)$. We note that monadicity is not necessarily compositional and the category $\mathcal{EM}(R)$ may not be monadic over $\mathbb{C}$.

for its action on arrows. We claim that the desired natural isomorphism $\Xi \bullet \Psi$ is the (vertical) composite

$$\mathbb{P}^{\mathcal{K}\ell}(\tau,\rho)(X) = \mathbb{C}(X, T\Omega) \xrightarrow{\Psi_X} \mathcal{EM}(T)(F^T X, \tau) \xrightarrow{\Xi_X} \mathcal{EM}(R)(F^R F^T X, \rho) = \mathbb{P}^{\mathcal{EM}}(\tau,\rho)(KX)$$

where Ψ and Ξ are isomorphisms induced by adjunctions.

We have to check that Ψ_X and Ξ_X are order isomorphisms. The map Ψ_X is monotone due to the monotonicity condition on τ (Definition 2.4); so is Ψ_X^{-1} by Lemma 2.12. Similarly, Ξ_X is monotone by the monotonicity condition on ρ (Definition 5.1); so is Ξ_X^{-1} by Lemma 2.12.

We turn to the naturality: the following diagram must be shown to commute, for each $f: X \twoheadrightarrow Y$ in $\mathcal{K}\ell(U^T R F^T)$.

$$\begin{tikzcd}
\mathbb{C}(Y, T\Omega) \arrow[r, "\Psi_Y", "\cong"'] \arrow[d, "\mathbb{P}^{\mathcal{K}\ell}(\tau,\rho)(f)"'] & \mathcal{EM}(T)(F^T Y, \tau) \arrow[r, "\Xi_Y", "\cong"'] \arrow[d, "\rho\circ R(_)\circ f^\wedge"] & \mathcal{EM}(R)(F^R(F^T Y), \rho) \arrow[d, "\mathbb{P}^{\mathcal{EM}}(\tau,\rho)(Kf) = (_)\circ Kf"] \\
\mathbb{C}(X, T\Omega) \arrow[r, "\Psi_X", "\cong"'] & \mathcal{EM}(T)(F^T X, \tau) \arrow[r, "\Xi_X", "\cong"'] & \mathcal{EM}(R)(F^R(F^T Y), \rho)
\end{tikzcd} \tag{26}$$

The square on the left commutes by the definition of $\mathbb{P}^{\mathcal{K}\ell}(\tau,\rho)(f)$ (Proposition 5.3); the one on the right is much like the one in (12) and its commutativity can be proved in the same way. Note here that $Kf = \mu^R_{F^T Y} \circ R(f^\wedge)$.

Since the diagram (26) commutes, and since Ψ and Ξ are order isomorphisms and $\mathbb{P}^{\mathcal{EM}}(\tau,\rho)(Kf)$ is monotone (Proposition 5.2), we have that $\mathbb{P}^{\mathcal{K}\ell}(\tau,\rho)f$ is monotone. The functoriality of $\mathbb{P}^{\mathcal{K}\ell}(\tau,\rho)$ is easy, too. This concludes the proof. □

6. Examples of 2-player PT situations

6.1. Nondeterministic player and then nondeterministic opponent

We continue Section 4 and locate the monad $\mathcal{UP}$—and the logic of forced predicates—in the general setup of Section 5. We identify a suitable 2-player PT situation $(\mathcal{P}, 1, \tau_\Box, \mathsf{R_G}, \rho_\mathsf{P})$, in which $T = \mathcal{P}$, $\Omega = 1$ and $\tau = \tau_\Box$ that is from Section 2.3. The choice of $\tau_\Box$ corresponds to the demonic nature of Opponent's choices: Player can force those properties which hold *whatever choices* Opponent makes.

To introduce the monad $\mathsf{R_G}$ on $\mathcal{EM}(\mathcal{P})$—corresponding to the up-closed powerset construction—we go via the following standard isomorphism.

Lemma 6.1. *Let $C: \mathcal{EM}(\mathcal{P}) \to \mathbf{CL}_\bigwedge$ be the functor such that $C\left(\begin{smallmatrix}\mathcal{P}X\\ \downarrow a\\ X\end{smallmatrix}\right) := (X, \sqsubseteq_a)$, where the order is defined by $x \sqsubseteq_a y$ if $x = a\{x, y\}$. Conversely, let $D: \mathbf{CL}_\bigwedge \to \mathcal{EM}(\mathcal{P})$ be such that $D(X, \sqsubseteq) := \left(\begin{smallmatrix}\mathcal{P}X\\ \downarrow\bigwedge\\ X\end{smallmatrix}\right)$. Both act on arrows as identities.*

Then C and D constitute an isomorphism $\mathcal{EM}(\mathcal{P}) \overset{\cong}{\to} \mathbf{CL}_\bigwedge$. □

The monad $\mathsf{R_G}$ is then defined to be the composite $\mathsf{R_G} := D \circ \mathsf{Dw} \circ C$, using the *down-closed powerset monad* Dw on $\mathbf{CL}_\bigwedge$.

$$\begin{tikzcd}
\mathcal{EM}(\mathcal{P}) \arrow[loop left, "\mathsf{R_G}"] \arrow[r, bend right, "C"'] \arrow[r, phantom, "\cong"] & \mathbf{CL}_\bigwedge \arrow[l, bend right, "D"'] \arrow[loop right, "\mathsf{Dw}"]
\end{tikzcd} \tag{27}$$

The switch between *up-closed* subsets in $\mathcal{UP}$ and *down-closed* subsets Dw may seem confusing. Later in Proposition 6.3 it is shown that everything is in harmony; and after all it is a matter of presentation since there is an isomorphism $\mathbf{CL}_\bigwedge \overset{\cong}{\to} \mathbf{CL}_\bigvee$ that reverses the order in each complete lattice. The switch here between up- and down-closed is essentially because: the bigger the set of Opponent's options is, the smaller the power of Player (to force Opponent to somewhere) is.

Concretely, the monad $\mathsf{Dw}: \mathbf{CL}_\bigwedge \to \mathbf{CL}_\bigwedge$ carries a complete lattice $(X, \sqsubseteq)$ to the set $\mathsf{Dw}(X) := \{S \subseteq X \mid x \sqsubseteq x',\ x' \in S \Rightarrow x \in S\}$. We equip $\mathsf{Dw}(X)$ with the inclusion order; this makes $\mathsf{Dw}(X)$ a complete lattice, with sups and infs given by unions and intersections, respectively. An arrow $f: X \to Y$ is carried to $\mathsf{Dw}(f): \mathsf{Dw}(X) \to \mathsf{Dw}(Y)$ defined by $S \mapsto {\downarrow}(\coprod_f S)$. Here ${\downarrow}(\_)$ denotes the downward closure and it is needed to ensure down-closedness (consider a $\bigwedge$-preserving map $f: 1 \to 2$, $0 \mapsto 1$ where $0 \sqsubseteq 1$ in 2). The monad structure of Dw is given by: $\eta^{\mathsf{Dw}}_X: X \to \mathsf{Dw}X, x \mapsto {\downarrow}\{x\}$; and $\mu^{\mathsf{Dw}}_X: \mathsf{Dw}(\mathsf{Dw}(X)) \to \mathsf{Dw}(X), \mathfrak{a} \mapsto \bigcup \mathfrak{a}$. Note in particular that η^{Dw}_X is $\bigwedge$-preserving. As in (27) we define $\mathsf{R_G} := D \circ \mathsf{Dw} \circ C$.

Finally, let us define the data $\rho_\mathsf{P}: \mathsf{R_G}(\tau_\Box) \to \tau_\Box$ in the 2-player PT situation. Via the isomorphism (27) we shall think of it as an Dw-algebra, where the $\mathcal{P}$-algebra $\tau_\Box$ is identified with the 2-element complete lattice $[\mathtt{ff} \sqsubseteq \mathtt{tt}]$ (the order is because $\tau_\Box\{\mathtt{tt}, \mathtt{ff}\} = \mathtt{ff}$). Therefore we are looking for a $\bigwedge$-preserving map

$$\mathsf{Dw}[\mathtt{ff} \sqsubseteq \mathtt{tt}] = \big[\emptyset \sqsubseteq \{\mathtt{ff}\} \sqsubseteq \{\mathtt{ff}, \mathtt{tt}\}\big] \xrightarrow{C\rho_\mathsf{P}} [\mathtt{ff} \sqsubseteq \mathtt{tt}]$$

subject to the conditions of an Eilenberg–Moore algebra in (5). In fact such $C(\rho_\mathsf{P})$ is uniquely determined: preservation of $\top$ forces $(C\rho_\mathsf{P})\{\mathtt{ff}, \mathtt{tt}\} = \mathtt{tt}$; the unit law forces $(C\rho_\mathsf{P})\{\mathtt{ff}\} = \mathtt{ff}$ and monotonicity of $C\rho_\mathsf{P}$ then forces $(C\rho_\mathsf{P})\emptyset = \mathtt{ff}$.

Lemma 6.2. $(\mathcal{P}, 1, \tau_\Box, \mathsf{R_G}, \rho_\mathsf{P})$ *thus obtained is a 2-player PT situation.*

Proof. It remains to check the monotonicity condition (Definition 5.1) for ρ_P. We shall again think in terms of complete lattices and $\bigwedge$-preserving maps; then the requirement is that the map $\big(X \xrightarrow{f} [\mathtt{ff} \sqsubseteq \mathtt{tt}]\big) \mapsto \big(\mathsf{Dw}(X) \xrightarrow{\rho_\mathsf{P}\circ\mathsf{Dw}(f)} [\mathtt{ff} \sqsubseteq \mathtt{tt}]\big)$ is monotone. Assume $g \sqsubseteq f$, $S \in \mathsf{Dw}(X)$ and $(\rho_\mathsf{P} \circ \mathsf{Dw}(f))(S) = \mathtt{ff}$. It suffices to show that $(\rho_\mathsf{P} \circ \mathsf{Dw}(g))(S) = \mathtt{ff}$; this follows from the observation that, for $h = f$ or g,

$$\big(\rho_\mathsf{P} \circ \mathsf{Dw}(h)\big)(S) = \mathtt{ff} \iff \big(\mathsf{Dw}(h)\big)S \subseteq \{\mathtt{ff}\} \iff \forall x \in S.\, hx = \mathtt{ff} \ . \qquad \Box$$

Let us check that the logic $\mathbb{P}^{\mathcal{K}\ell}(\tau_\Box, \rho_\mathsf{P})$ associated with this 2-player PT situation is indeed the logic of forced predicates in Section 4.1. For instance, we want "computations" $X \to U^\mathcal{P}\mathsf{R_G}F^\mathcal{P}Y$ to coincide with "computations" $X \to \mathcal{UP}Y$.

Proposition 6.3. *For any set X we have $U^\mathcal{P}\mathsf{R_G}F^\mathcal{P}X = \mathcal{UP}X$. In fact they are equal as complete lattices, that is,* $\mathsf{Dw} \circ C \circ F^\mathcal{P} = \mathcal{UP}\colon \mathbf{Sets} \to \mathbf{CL}_\bigwedge$ *where the functor $\mathcal{UP}$ is equipped with the inclusion order.*

Proof. Given $X \in \mathbf{Sets}$, the definition of C dictates that $C(F^\mathcal{P}X) = (\mathcal{P}X, \supseteq)$ and its order be given by the reverse inclusion order. Hence $\mathsf{Dw}(C(F^\mathcal{P}X))$ is the collection of families $\mathfrak{a} \subseteq \mathcal{P}X$ that are $\supseteq$-down-closed, i.e. $\subseteq$-up-closed. It is easily checked that the two functors coincide on arrows, too, using the characterization on the left in (23). $\Box$

Next we describe the logic $\mathbb{P}^{\mathcal{K}\ell}(\tau_\Box, \rho_\mathsf{P})$ (Proposition 5.3) in concrete terms. We base ourselves again in $\mathbf{CL}_\bigwedge$ via the isomorphism $\mathcal{EM}(\mathcal{P}) \cong \mathbf{CL}_\bigwedge$ in (27). Consider a postcondition $q\colon Y \to \mathcal{P}1$ and a branching computation $f\colon X \to \mathcal{UP}Y$. These are in one-to-one correspondences with the following arrows in $\mathbf{CL}_\bigwedge$:

$$\begin{aligned}
q^\wedge\colon\ & C(F^\mathcal{P}Y) = (\mathcal{P}Y, \supseteq) \longrightarrow [\mathtt{ff} \sqsubseteq \mathtt{tt}] = C(\tau_\Box) \ , \\
f^\wedge\colon\ & C(F^\mathcal{P}X) = (\mathcal{P}X, \supseteq) \longrightarrow \mathsf{Dw}(\mathcal{P}Y, \supseteq) = C(\mathsf{R_G}(F^\mathcal{P}Y)) \ ,
\end{aligned}$$

where we used Proposition 6.3. Since $q^\wedge$ are $f^\wedge$ are $\bigwedge$-preserving, we have

$$q^\wedge W = q^\wedge(\textstyle\bigcup_{y\in W}\{y\}) = q^\wedge(\bigwedge_{y\in W}\{y\}) = \bigwedge_{y\in W} q^\wedge\{y\} = \bigwedge_{y\in W} qy \ ;$$

and similarly $f^\wedge S = \bigcap_{x\in S} fx$. Recall that $\mathsf{Dw}(\mathcal{P}Y, \supseteq)$ has the inclusion order.

Now Proposition 5.3 states that the weakest precondition $\mathbb{P}^{\mathcal{K}\ell}(\tau_\Box, \rho_\mathsf{P})(f)(q)$ is the arrow $X \to \mathcal{P}1$ that corresponds, via the adjunction $C \circ F^\mathcal{P} \dashv U^\mathcal{P} \circ D$, to

$$(\mathcal{P}X, \supseteq) \xrightarrow{f^\wedge} \mathsf{Dw}(\mathcal{P}Y, \supseteq) \xrightarrow{\mathsf{Dw}(q^\wedge)} \mathsf{Dw}[\mathtt{ff} \sqsubseteq \mathtt{tt}] \xrightarrow{\rho_\mathsf{P}} [\mathtt{ff} \sqsubseteq \mathtt{tt}] \quad \text{in } \mathbf{CL}_\bigwedge .$$

Unweaving definitions it is straightforward to see that, for $S \subseteq X$,

$$\begin{aligned}
&\big(\rho_\mathsf{P} \circ \mathsf{Dw}(q^\wedge) \circ f^\wedge\big)S = \mathtt{tt} \iff \exists W \subseteq Y.\, \big(\forall x \in S.\, W \in fx \wedge \forall y \in W.\, qy = \mathtt{tt}\big) \ ; \\
&\text{therefore } \mathbb{P}^{\mathcal{K}\ell}(\tau_\Box, \rho_\mathsf{P})(f)(q)(x) = \mathtt{tt} \iff \exists W \subseteq Y.\, \big(W \in fx \wedge \forall y \in W.\, qy = \mathtt{tt}\big) \ .
\end{aligned} \tag{28}$$

The last condition reads: among the set fx of possible moves of Player, there exists a move W, from which q holds no matter what Opponent's move y is. Therefore $\mathbb{P}^{\mathcal{K}\ell}(\tau_\Box, \rho_\mathsf{P})(f)(q)(x) = \mathtt{tt}$ if Player can *force* the predicate q from x after the (two-layer branching) computation f.

6.2. Nondeterministic opponent and then nondeterministic player

We change the order of Player and Opponent: O moves first and then P moves. The general setup in Section 5 successfully models this situation too, with a choice of a 2-player PT situation $(\mathcal{P}, 1, \tau_\Diamond, \mathsf{R_G}, \rho_\mathsf{O})$ that is dual to the previous one.

The modality $\tau_\Diamond$ is from Section 2.3. Although the monad $\mathsf{R_G}$ is the same as in Section 6.1, we now prefer to present it in terms of $\mathcal{EM}(\mathcal{P}) \cong \mathbf{CL}_\bigvee$ instead of $\mathbf{CL}_\bigwedge$. The reason is that this way the algebra $\tau_\Diamond$ gets identified with $[\mathtt{ff} \sqsubseteq \mathtt{tt}]$, which is intuitive. The situation is as follows.

$$\begin{tikzcd}
\mathcal{EM}(\mathcal{P}) \arrow[loop left, "\mathsf{R_G}"] \arrow[r, bend right, "C"'] \arrow[r, phantom, "\cong"] \arrow[rr, bend right=50, "C' := C''\circ C"'] & \mathbf{CL}_\bigwedge \arrow[l, bend right, "D"'] \arrow[loop above, "\mathsf{Dw}"] \arrow[r, bend right, "C''"'] \arrow[r, phantom, "\cong"] & \mathbf{CL}_\bigvee \arrow[l, bend right, "D''"'] \arrow[loop right, "\mathsf{Up}"] \arrow[ll, bend right=50, "D' := D\circ D''"']
\end{tikzcd} \tag{29}$$

The functors C'' and D'' carries a complete lattice $(X, \sqsubseteq)$ to $(X, \sqsupseteq)$, reversing the order. The monad Up is defined by $\mathsf{Up} := C'' \circ \mathsf{Dw} \circ D''$; concretely it carries $(X, \sqsubseteq)$ to the set of its up-closed subsets, equipped with the *reverse* inclusion order $\supseteq$. That is,

$$\mathsf{Up}(X, \sqsubseteq) \ := \ \big(\{S \subseteq X \mid S \ni x \sqsubseteq x' \Rightarrow x' \in S\}, \ \supseteq\big) \ .$$

We have $\mathsf{R_G} = D \circ \mathsf{Dw} \circ C = D' \circ \mathsf{Up} \circ C'$.

The modality $\rho_{\mathsf{O}} \colon \mathsf{R_G}(\tau_\Diamond) \to \tau_\Diamond$ is identified, via the isomorphism C' in (29), with an Up-algebra on $[\mathtt{ff} \sqsubseteq \mathtt{tt}]$. The latter is a $\bigvee$-preserving map

$$\mathsf{Up}[\mathtt{ff} \sqsubseteq \mathtt{tt}] = \big[\{\mathtt{ff}, \mathtt{tt}\} \sqsubseteq \{\mathtt{tt}\} \sqsubseteq \emptyset\big] \xrightarrow{C'\rho_{\mathsf{O}}} [\mathtt{ff} \sqsubseteq \mathtt{tt}] \ ;$$

note here that the order in $\mathsf{Up}(X, \sqsubseteq)$ is the reverse inclusion $\supseteq$. Such $C'\rho_{\mathsf{O}}$ is uniquely determined (as before): the unit law forces $(C'\rho_{\mathsf{O}})\{\mathtt{tt}\} = \mathtt{tt}$; preservation of $\bot$ forces $(C'\rho_{\mathsf{O}})\{\mathtt{tt}, \mathtt{ff}\} = \mathtt{ff}$; and then by monotonicity $(C'\rho_{\mathsf{O}})\emptyset = \mathtt{tt}$.

It is straightforward to see that $(\mathcal{P}, 1, \tau_\Diamond, \mathsf{R_G}, \rho_{\mathsf{O}})$ is indeed a 2-player PT situation; the proof is symmetric to the one in Section 6.1. Also symmetrically, the weakest precondition semantics $\mathbb{P}^{\mathcal{K}\ell}(\tau_\Diamond, \rho_{\mathsf{O}})$ is concretely described as follows: given a postcondition $q \colon Y \to \mathcal{P}1$ and a branching computation $f \colon X \to \mathcal{UP}Y$,

$$\mathbb{P}^{\mathcal{K}\ell}(\tau_\Diamond, \rho_{\mathsf{O}})(f)(q)(x) = \mathtt{tt} \iff \forall W \subseteq Y. \big(W \in fx \Rightarrow \exists y \in W. qy = \mathtt{tt}\big) \ .$$

This is dual to (28) and reads: whatever move W Opponent takes, there exists Player's move $y \in W$ so that q holds afterwards.

We note that the analogue of Proposition 6.3 becomes: $\mathsf{Up} \circ C' \circ F^{\mathcal{P}} = \mathcal{UP} \colon \mathbf{Sets} \to \mathbf{CL}_{\bigvee}$, where each $\mathcal{UP}X$ is equipped with the *reverse* inclusion order. This order ($\mathfrak{a} \sqsubseteq \mathfrak{b}$ in $\mathcal{UP}X$ if $\mathfrak{a} \supseteq \mathfrak{b}$) is intuitive if we think of $\sqsubseteq$ as the power of Player.

Remark 6.4. The constructions have been described in concrete terms; this is for intuition. An abstract view is possible too: the modality $\tau_\Diamond$ is the dual of $\tau_\Box$ via the swapping σ (see (9)); and the other modality ρ_{O} is also the dual of ρ_{P} by $\rho_{\mathsf{O}} = \big(\mathsf{R_G}(\tau_\Diamond) \stackrel{\mathsf{R_G}\sigma}{\to} \mathsf{R_G}(\tau_\Box) \stackrel{\rho_{\mathsf{P}}}{\to} \tau_\Box \stackrel{\sigma}{\to} \tau_\Diamond\big)$.

6.3. Nondeterministic opponent and then probabilistic player

In our last example Opponent O moves nondeterministically first, and then Player P moves probabilistically. Such nested alternating branching occurs in many process-theoretic models of probabilistic systems (see Section 4, in particular Section 4.2), most notably in Segala's *probabilistic automata* [47]. We identify a 2-player PT situation $(\mathcal{D}, 1, \tau_{\mathrm{total}}, \mathcal{C}v, \rho_{\mathrm{inf}})$ for this situation; then the associated logic $\mathbb{P}^{\mathcal{K}\ell}(\tau_{\mathrm{total}}, \rho_{\mathrm{inf}})$ is that of the probabilistic predicate transformers in [7] (see also [21–23]. The modality τ_{total} is from Section 2.3. The other components $(\mathcal{C}v, \rho_{\mathrm{inf}})$ are to be described in terms of *convex cones* and their *convex subsets*.

In what follows a $\mathcal{D}$-algebra is referred to as a *convex cone*, adopting the notation $\sum_{i \in I} w_i x_i$ to denote an element $a\big([x_i \mapsto w_i]_{i \in I}\big) \in X$ in a convex cone $a \colon \mathcal{D}X \to X$. Here I is a countable index set,[9] $w_i \in [0,1]$, and $\sum_{i \in I} w_i \le 1$. Note that, since $\mathcal{D}$ is the *sub*distribution monad, the zero distribution $\mathbf{0}$ is allowed in $\mathcal{D}X$ and therefore a convex cone $a \colon \mathcal{D}X \to X$ has its *apex* $a(\mathbf{0}) \in X$.

One can picture a convex cone as a shape that is convex (the line segment $\{\lambda x + (1-\lambda)y \mid \lambda \in [0,1]\}$ that connects two points x, y is contained in the shape—in fact its generalization to countably many points is also true), has an apex $a(\mathbf{0})$, and allows "scaling" with respect to the apex (meaning $a([x \mapsto \lambda]) \in X$ for $\lambda \in [0,1]$).

Likewise, a morphism of $\mathcal{D}$-algebras is referred to as a *convex linear map*.

Definition 6.5 *(Convex subset).* A subset $S \subseteq X$ of a convex cone $a \colon \mathcal{D}X \to X$ is said to be *convex* if, for any $p_i \in [0,1]$ such that $\sum_{i \in I} p_i = 1$ and any $x_i \in S$, the convex combination $\sum_{i \in I} p_i x_i$ belongs to S.

We emphasize that in the last definition $\sum_i p_i$ is required to be $= 1$. This is unlike $\sum_i w_i \le 1$ in the definition of convex cone. Therefore a convex subset S need not include the apex $a(\mathbf{0})$; one can think of the base of a 3-dimensional cone as an example. This variation in the definitions is also found in [46, §2.1.2]; one reason is technical: if we allow $\sum_i p_i \le 1$ then it is hard to find the monad unit of $\mathcal{C}v$ (see below). Another process-theoretic reason is described later in Remark 6.8.

Definition 6.6 *(The monad $\mathcal{C}v$).* The functor $\mathcal{C}v \colon \mathcal{EM}(\mathcal{D}) \to \mathcal{EM}(\mathcal{D})$ carries a convex cone $a \colon \mathcal{D}X \to X$ to $\mathcal{C}vX := \{S \subseteq X \mid S \text{ is convex}\}$; the latter is a convex cone by

$$\textstyle\sum_i w_i S_i \ := \ \{\sum_i w_i x_i \mid x_i \in S_i\} \ .$$

[9] The countability requirement is superfluous since, if $\sum_{i \in I} p_i = 1$, then only countably many p_i's are nonzero.

It is easy to see that $\sum_i w_i S_i$ is indeed a convex subset of X. Given a convex linear map $f: X \to Y$, $\mathcal{C}v f: \mathcal{C}v X \to \mathcal{C}v Y$ is defined by $(\mathcal{C}v f) S := \amalg_f S$, which is obviously convex in Y, too.

The monad structure of $\mathcal{C}v$ is as follows. Its unit is $\eta^{\mathcal{C}v}_X := \{\_\}: X \to \mathcal{C}v X$; note that a singleton $\{x\}$ is a convex subset of X (Definition 6.5). The monad multiplication is $\mu^{\mathcal{C}v}_X := \bigcup: \mathcal{C}v(\mathcal{C}v X) \to \mathcal{C}v X$. It is easy to see that $\eta^{\mathcal{C}v}_X$ and $\mu^{\mathcal{C}v}_X$ are convex linear maps, and that they satisfy the monad axioms.

We introduce the last component, namely the modality $\rho_{\inf}: \mathcal{C}v(\tau_{\mathrm{total}}) \to \tau_{\mathrm{total}}$. A convex subset S of the carrier $\mathcal{D}1 = [0,1]$ of τ_{total} is nothing but an interval (its endpoints may or may not be included); $\rho_{\inf}$ then carries such S to its infimum $\inf S \in [0,1]$. It is easy to see that $\rho_{\inf}$ is convex linear, and that it satisfies the Eilenberg–Moore axioms: a direct proof is straightforward.

Lemma 6.7. $(\mathcal{D}, 1, \tau_{\mathrm{total}}, \mathcal{C}v, \rho_{\inf})$ *thus obtained is a 2-player PT situation.* □

Proof. It remains to check the monotonicity condition (Definition 5.1) for $\rho_{\inf}$. Assume $f, g: X \to [0,1]$ and $g \sqsubseteq f$, and $S \subseteq X$ is a convex subset. We have, for both of $h \in \{f, g\}$,

$$\big(\rho_{\inf} \circ \mathcal{C}v(h)\big)(S) \;=\; \inf\{h(x) \mid x \in S\} \;\;;$$

from which $\big(\rho_{\inf} \circ \mathcal{C}v(g)\big)(S) \sqsubseteq \big(\rho_{\inf} \circ \mathcal{C}v(f)\big)(S)$ is obvious. □

The resulting logic $\mathbb{P}^{\mathcal{K}\ell}(\tau_{\mathrm{total}}, \rho_{\inf})$ is as follows. Given a postcondition $q: Y \to \mathcal{D}1$ and a computation $f: X \to U^{\mathcal{D}} \mathcal{C}v F^{\mathcal{D}} Y$, the weakest precondition is

$$\mathbb{P}^{\mathcal{K}\ell}(\tau_{\mathrm{total}}, \rho_{\inf})(f)(q)(x) \;=\; \inf\Big\{ \textstyle\sum_{y \in Y} d(y) \cdot q(y) \,\Big|\, d \in f(x) \Big\} \;\; . \tag{30}$$

Here d is a subdistribution chosen by Opponent; and the value $\sum_{y \in Y} d(y) \cdot q(y)$ is the expected value of the random variable q under the distribution d. Therefore the weakest precondition computed above is the least expected value of q when Opponent picks a distribution in harm's way. This is the same as in [7].

Remark 6.8. The use of the convex powerset construction, instead of (plain) powersets, was motivated in Section 4.2 through the technical difficulty in getting a monad. Convex powersets are commonly used in the process-theoretic study of probabilistic systems, also because they model a *probabilistic scheduler*: Opponent (called a *scheduler* in this context) can not only pick one distribution but also use randomization in doing so. See e.g. [48].

The definition of convex subset (Definition 6.5)—where we insist on $\sum_i p_i = 1$ instead of ≤ 1—is natural in view of the logic $\mathbb{P}^{\mathcal{K}\ell}(\tau_{\mathrm{total}}, \rho_{\inf})$ described above. Relaxing this definition entails that the zero distribution $\mathbf{0}$ is always included in a "convex subset," and hence always in Opponent's options. This way, however, the weakest precondition in (30) can always be forced to 0 and the logic gets trivial.

We can also model the situation where the roles of Player and Opponent are swapped: we can follow the same path as in Remark 6.4 and obtain a 2-player PT situation $(\mathcal{D}, 1, \tau_{\mathrm{partial}}, \mathcal{C}v, \rho_{\sup})$; the resulting modality $\rho_{\sup}$ carries an interval to its supremum.

7. Conclusions and future work

Inspired by Jacobs' recent work [9,8] we pursued a foundation of predicate transformers (more specifically weakest precondition semantics) based on an order-enriched monad. We saw that a simple notion of PT situation yields a triangle (11)—and (17), a situation that is called the state-and-effect triangle in [8], in the **Sets**-based case. Compositionality of the semantics is one of the properties that follow.

Our monad-based foundation accommodates different notions of modality (such as "may" vs. "must") as different Eilenberg–Moore algebras. Nested alternating branching with two conflicting players can be modeled in a modular way, too, by a monad R on an Eilenberg–Moore category $\mathcal{EM}(T)$. Instances of this generic framework include probabilistic weakest preconditions, those augmented with nondeterminism, and the logic of forced predicates in games.

As future work we wish to address the components in the picture (2)–(3) that are missing in the current framework, most notably richer order structures than posets, like effect modules in [8]. A generic weakest precondition calculus presented in a *syntactic* form is another direction, on which there are some classic works (including [49]) and also a recent work that is also based on monads and orders [13]. Most probably relationships between monads and algebraic theories (see e.g. [50]) will be exploited there. So-called *healthiness conditions*—i.e. characterization of the image of $\mathbb{P}^{\mathcal{K}\ell}(\tau)$ in (11), to be precise its action on arrows—are yet another topic, generalizing [2,7].

The current work is hopefully a step forward towards a coalgebraic theory of games, and hence towards *coalgebraic model checking* where automata (on infinite trees), games and fixed-point logics interplay. For example, we suspect that our categorical formulation of the logic of forced predicates should be useful in putting *game (bi)simulation* (studied e.g. in [51, 52]) in coalgebraic terms. Possibly related, we plan to work on the relationship to the coalgebraic theory of traces and simulations formulated in a Kleisli category [24,53] since most of the monads in Example 2.3 fit in this trace framework.

In this paper we relied on an order-enrichment of a monad to obtain the entailment order. We are nevertheless interested in what our current framework brings for other monads, like the ones that model computational effects [54] (global state, I/O, continuation, etc.). Also interesting is a higher-order extension of the current work, where the logic will probably take the form of dependent types. Related work in this direction is [55]. Finally, we shall investigate the relationship between the 2-player framework (Section 5) and other known ways of composing computational effects, including *monad transformers* (see e.g. [56]), *sum* and *tensor* of Lawvere theories [57], and *handlers* of algebraic effects [58].

Acknowledgements

Thanks are due to Kazuyuki Asada, Marcello Bonsangue, Kenta Cho, Corina Cîrstea, Sergey Goncharov, H. Peter Gumm, Bart Jacobs, Tetsuri Moriya, the participants of CMCS 2014, and the anonymous referees for useful discussions, comments and encouragements. The author is supported by Grants-in-Aid for Young Scientists (A) No. 24680001, JSPS, and by Aihara Innovative Mathematical Modelling Project, FIRST Program, JSPS/CSTP.

Appendix A. Omitted proofs

A.1. Proof of Lemma 2.15

Proof. In what follows we denote the "currying" and "uncurrying" correspondences in the adjunction $Y \times (\_) \dashv (\_)^Y$ by $(\_)^\wedge$ and $(\_)^\vee$, respectively.

We first show that the image of the upper functor $\alpha_{(\_)}$ is indeed an Eilenberg–Moore algebra. For compatibility with multiplication, we have to check that $\alpha_Y \circ T(\alpha_Y) = \alpha_Y \circ \mu_{A^Y}$:

$$
\begin{aligned}
\big(\alpha_Y \circ T(\alpha_Y)\big)^\vee &= (\alpha_Y)^\vee \circ \big(Y \times T(\alpha_Y)\big) && \text{naturality of } (\_)^\vee \\
&= \alpha \circ T\mathrm{ev} \circ \mathrm{str} \circ \big(Y \times T(\alpha_Y)\big) && \text{definition of } \alpha_Y \\
&= \alpha \circ T\mathrm{ev} \circ T(Y \times \alpha_Y) \circ \mathrm{str} && \text{naturality of str} \\
&= \alpha \circ T\big((\alpha_Y)^\vee\big) \circ \mathrm{str} && \text{by } \mathrm{ev} \circ (Y \times \alpha_Y) = (\alpha_Y)^\vee \\
&= \alpha \circ T\alpha \circ T(T\mathrm{ev}) \circ T\mathrm{str} \circ \mathrm{str} && \text{definition of } \alpha_Y \\
&= \alpha \circ \mu_A \circ T(T\mathrm{ev}) \circ T\mathrm{str} \circ \mathrm{str} && \alpha \text{ is an Eilenberg–Moore algebra} \\
&= \alpha \circ T\mathrm{ev} \circ \mu_{Y \times A^Y} \circ T\mathrm{str} \circ \mathrm{str} && \text{naturality of } \mu \\
&= \alpha \circ T\mathrm{ev} \circ \mathrm{str} \circ (Y \times \mu_{A^Y}) && \text{pentagon coherence of str} \\
&= (\alpha_Y)^\vee \circ (Y \times \mu_{A^Y}) && \text{definition of } \alpha_Y \\
&= (\alpha_Y \circ \mu_{A^Y})^\vee && \text{naturality of } (\_)^\vee.
\end{aligned}
$$

For compatibility with unit:

$$
\begin{aligned}
\big(\alpha_Y \circ \eta_{A^Y}\big)^\vee &= (\alpha_Y)^\vee \circ (Y \times \eta_{A^Y}) && \text{naturality of } (\_)^\vee \\
&= \alpha \circ T\mathrm{ev} \circ \mathrm{str} \circ (Y \times \eta_{A^Y}) && \text{definition of } \alpha_Y \\
&= \alpha \circ T\mathrm{ev} \circ \eta_{Y \times A^Y} && \text{str's compatibility with } \eta \\
&= \alpha \circ \eta_A \circ \mathrm{ev} && \text{naturality of } \eta \\
&= \mathrm{ev} = (\mathrm{id})^\vee && \alpha \text{ is an Eilenberg–Moore algebra.}
\end{aligned}
$$

The functor $\alpha_{(\_)}$ carries an arrow $f : Y \to Z$ in **Sets** to a function

$$
\big(A^Z \xrightarrow{A^f} A^Y\big) := \big(Y \times A^Z \xrightarrow{f \times A^Z} Z \times A^Z \xrightarrow{\mathrm{ev}} A\big)^\wedge \ .
$$

We have to check that A^f is an algebra homomorphism from $T(A^Z) \stackrel{\alpha_Z}{\to} A^Z$ to $T(A^Y) \stackrel{\alpha_Y}{\to} A^Y$, that is, $A^f \circ \alpha_Z = \alpha_Y \circ T(A^f)$.

$$
\begin{aligned}
\bigl(A^f \circ \alpha_Z\bigr)^\vee &= (A^f)^\vee \circ (Y \times \alpha_Z) && \text{naturality of } (\_)^\vee \\
&= \mathrm{ev} \circ (f \times A^Z) \circ (Y \times \alpha_Z) && \text{definition of } A^f \\
&= \mathrm{ev} \circ (Z \times \alpha_Z) \circ (f \times T(A^Z)) && \\
&= (\alpha_Z)^\vee \circ (f \times T(A^Z)) && \text{by } \mathrm{ev} \circ (Z \times \alpha_Z) = (\alpha_Z)^\vee \\
&= \alpha \circ T\mathrm{ev} \circ \mathrm{str} \circ (f \times T(A^Z)) && \text{definition of } \alpha_Z \\
&= \alpha \circ T\mathrm{ev} \circ T(f \times A^Z) \circ \mathrm{str} && \text{naturality of str} \\
&= \alpha \circ T\mathrm{ev} \circ T(Y \times A^f) \circ \mathrm{str} && (*) \\
&= \alpha \circ T\mathrm{ev} \circ \mathrm{str} \circ (Y \times T(A^f)) && \text{naturality of str} \\
&= (\alpha_Y)^\vee \circ (Y \times T(A^f)) && \text{definition of } \alpha_Y \\
&= (\alpha_Y \circ T(A^f))^\vee && \text{naturality of } (\_)^\vee,
\end{aligned}
$$

where, for the step $(*)$, we used the following commutativity:

$$
\begin{tikzcd}
Y \times A^Z \arrow[r, "Y \times A^f"] \arrow[d, "f \times A^Z"'] & Y \times A^Y \arrow[d, "\mathrm{ev}_Y"] \\
Z \times A^Z \arrow[r, "\mathrm{ev}_Z"'] & A
\end{tikzcd}
$$

that holds since both paths are equal to $(A^f)^\vee$.

It is straightforward that $\alpha_{(\_)}$ preserves identity arrows and composition, establishing that $\alpha_{(\_)}$ is indeed a functor.

Finally, to see that we have an adjunction, let $f: X \times Y \to A$ be a function with $f': Y \to A^X$ and $f'': X \to A^Y$ being its two curryings. The correspondences between these three are bijective and natural. Therefore it suffices to show that the following two are equivalent: 1) f' factors through $\mathcal{EM}(T)\bigl(\begin{smallmatrix} TX \\ \downarrow a \\ X \end{smallmatrix}, \begin{smallmatrix} TA \\ \downarrow \alpha \\ A \end{smallmatrix}\bigr) \hookrightarrow \mathbf{Sets}(X, A) = A^X$; and 2) f'' is an algebra homomorphism from a to α. We shall do so via the third equivalent condition: 3) $\alpha \circ Tf \circ \mathrm{str} = f \circ (a \times Y)$, as in

$$
\begin{tikzcd}
TX \times Y \arrow[r, "\mathrm{str}"] \arrow[d, "a \times Y"'] & T(X \times Y) \arrow[r, "Tf"] & TA \arrow[d, "\alpha"] \\
X \times Y \arrow[rr, "f"'] & & A
\end{tikzcd} .
$$

(Here and henceforth we use symmetry isomorphisms implicitly.)

To show that the conditions 1) and 3) are equivalent, note that the condition 1) amounts to $A^a \circ f' = \alpha^{TX} \circ T_{X,A} \circ f'$, as in

$$
\begin{tikzcd}
Y \arrow[r, "f'"] & A^X \arrow[r, "A^a"] \arrow[d, "T_{X,A}"'] & A^{TX} \\
& (TA)^{TX} \arrow[ur, "\alpha^{TX}"'] &
\end{tikzcd} ,
$$

where $T_{X,A}: A^X \to (TA)^{TX}$ is the functor T's action on arrows from X to A. Now

$$
\begin{aligned}
(A^a \circ f')^\vee &= (A^a)^\vee \circ (TX \times f') && \text{naturality of } (\_)^\vee \\
&= \mathrm{ev} \circ (a \times A^X) \circ (TX \times f') && \text{definition of } A^a \\
&= \mathrm{ev} \circ (X \times f') \circ (a \times Y) && \\
&= f \circ (a \times Y) \ ; && \\
(\alpha^{TX} \circ T_{X,A} \circ f')^\vee &= \mathrm{ev} \circ (TX \times \alpha^{TX}) \circ (TX \times T_{X,A}) \circ (TX \times f') && \\
&= \alpha \circ \mathrm{ev} \circ (TX \times T_{X,A}) \circ (TX \times f') && \\
&= \alpha \circ T\mathrm{ev} \circ \mathrm{str} \circ (TX \times f') && (*) \\
&= \alpha \circ T\mathrm{ev} \circ T(X \times f') \circ \mathrm{str} && \text{naturality of str} \\
&= \alpha \circ Tf \circ \mathrm{str} \ , &&
\end{aligned}
$$

where the step $(*)$ holds because

$$\begin{aligned}
\big(\mathrm{ev}\circ(TX\times T_{X,A})\big)^{\wedge} &= T_{X,A} \\
&= T_{X,A}\circ\big((X\times A^X)^X \xrightarrow{\mathrm{ev}^X} A^X\big)\circ\big(A^X \xrightarrow{\eta} (X\times A^X)^X\big) && \text{triangular equality} \\
&= (T\mathrm{ev})^{TX}\circ T_{X,X\times A^X}\circ\eta && \text{functoriality, } T\big(\mathrm{ev}\circ(_)\big)=T\mathrm{ev}\circ T(_) \\
&= (T\mathrm{ev})^{TX}\circ \mathrm{str}^{\wedge} && (\dagger) \\
&= (T\mathrm{ev}\circ\mathrm{str})^{\wedge} && \text{naturality of } (_)^{\wedge}.
\end{aligned}$$

In the above step (†), note that in **Sets** an endofunctor is equipped with a unique strength that arises from its action on arrows. This proves equivalence of 1) and 3).

To see that 2) and 3) are equivalent, note first that 2) means $f''\circ a=\alpha_Y\circ Tf''$. We have

$$\begin{aligned}
(f''\circ a)^{\vee} &= f\circ(a\times Y)\ ; \\
(\alpha_Y\circ Tf'')^{\vee} &= \alpha\circ T\mathrm{ev}\circ\mathrm{str}\circ(Tf''\times Y) && \text{definition of } \alpha_Y \\
&= \alpha\circ T\mathrm{ev}\circ T(f''\times Y)\circ\mathrm{str} && \text{definition of } \alpha_Y \\
&= \alpha\circ Tf\circ\mathrm{str}\ .
\end{aligned}$$

This concludes the proof. □

A.2. Proof of Lemma 2.16

Proof. Firstly we check that the composite function (16) is monotone in Y, so that its transpose $\widetilde{\tau_Y}$ factors through $\mathbf{Posets}(Y,T\Omega)\hookrightarrow\mathbf{Sets}(Y,T\Omega)$. Assume $y\sqsubseteq y'$ in Y; identifying elements of Y with functions $1\to Y$, it suffices to show that the following inequality holds.

$$\begin{tikzcd}
T(\mathbf{Posets}(Y,T\Omega)) \arrow[r, "\lambda^{-1}", "\cong"'] \arrow[d, "\lambda^{-1}"', "\cong"] & 1\times T(\mathbf{Posets}(Y,T\Omega)) \arrow[r, "y'\times\mathrm{id}"] \arrow[d, phantom, "\sqsubseteq"] & Y\times T(\mathbf{Posets}(Y,T\Omega)) \arrow[d, "\tau\circ T\widetilde{\mathrm{ev}}\circ\mathrm{str}"] \\
1\times T(\mathbf{Posets}(Y,T\Omega)) \arrow[r, "y\times\mathrm{id}"'] & Y\times T(\mathbf{Posets}(Y,T\Omega)) \arrow[r, "\tau\circ T\widetilde{\mathrm{ev}}\circ\mathrm{str}"'] & T\Omega
\end{tikzcd} \tag{A.1}$$

Here λ is the canonical isomorphism $1\times Z\overset{\cong}{\to}Z$, and the order between the arrows arises from that of $T\Omega$ in the pointwise manner. In proving (A.1) we shall exploit the monotonicity condition of a PT situation. There we use the following sublemma.

Sublemma A.1. *The composite $\tau\circ T\widetilde{\mathrm{ev}}\circ\mathrm{str}\circ(y\times\mathrm{id})\circ\lambda^{-1}$ in the diagram (A.1) coincides with*

$$\Phi_\tau\Big(\mathbf{Posets}(Y,T\Omega)\xrightarrow[\cong]{\lambda^{-1}}1\times\mathbf{Posets}(Y,T\Omega)\xrightarrow{y\times\mathrm{id}}Y\times\mathbf{Posets}(Y,T\Omega)\xrightarrow{\widetilde{\mathrm{ev}}}T\Omega\Big)\ ,$$

where $\Phi_\tau:\mathbf{Sets}(Z,T\Omega)\to\mathbf{Sets}(TZ,T\Omega)$ is from Definition 2.4.

Proof of Sublemma A.1.

$$\begin{aligned}
&\Phi_\tau(\widetilde{\mathrm{ev}}\circ(y\times\mathrm{id})\circ\lambda^{-1}) \\
&= \tau\circ T\widetilde{\mathrm{ev}}\circ T(y\times\mathrm{id})\circ T\lambda^{-1} && \text{definition of } \Phi_\tau \\
&= \tau\circ T\widetilde{\mathrm{ev}}\circ T(y\times\mathrm{id})\circ\mathrm{str}\circ\lambda^{-1} && (*) \\
&= \tau\circ T\widetilde{\mathrm{ev}}\circ\mathrm{str}\circ(y\times\mathrm{id})\circ\lambda^{-1}\ ,
\end{aligned}$$

where $(*)$ is because of the strength's compatibility with the monoidal unit:

$$\begin{tikzcd}
1\times TZ \arrow[rr, "\mathrm{str}_{1,Z}"] \arrow[dr, "\lambda"'] & & T(1\times Z) \arrow[dl, "T\lambda"] \\
& TZ &
\end{tikzcd}\ . \qquad \square$$

We turn back to the proof of Lemma 2.16. In view of the last sublemma and the monotonicity of Φ_τ (that is assumed—see Definition 2.4), it suffices to show that

$$\widetilde{\mathrm{ev}}\circ(y\times\mathrm{id})\circ\lambda^{-1}\sqsubseteq\widetilde{\mathrm{ev}}\circ(y'\times\mathrm{id})\circ\lambda^{-1}\ ,$$

that is, the evaluation function $\widetilde{\mathrm{ev}}\colon Y \times \mathbf{Posets}(Y, T\Omega) \to T\Omega$ is monotone in the first argument Y. This is obvious since the second argument $f \in \mathbf{Posets}(Y, T\Omega)$ is a monotone function. This establishes that the function $\widetilde{\tau}_Y$ is indeed of the type $T(\mathbf{Posets}(Y, T\Omega)) \to \mathbf{Posets}(Y, T\Omega)$.

We now prove that $\widetilde{\tau}_Y$ is indeed an Eilenberg–Moore algebra. We rely on the following result.

Sublemma A.2. *Between the Eilenberg–Moore algebra $\tau_Y\colon T((T\Omega)^Y) \to (T\Omega)^Y$ defined in Lemma 2.15 and the arrow $\widetilde{\tau}_Y$, we have the following commute.*

$$
\begin{tikzcd}
T(\mathbf{Posets}(Y, T\Omega)) \arrow[r, hook, "T\iota"] \arrow[d, "\widetilde{\tau}_Y"'] & T((T\Omega)^Y) \arrow[d, "\tau_Y"] \\
\mathbf{Posets}(Y, T\Omega) \arrow[r, hook, "\iota"'] & (T\Omega)^Y
\end{tikzcd}
\;.
$$

Here ι is a canonical inclusion.

Proof of Sublemma A.2.

$$
\begin{aligned}
(\tau_Y \circ T\iota)^\vee &= \tau \circ T\mathrm{ev} \circ \mathrm{str} \circ (Y \times T\iota) && \text{definition of } \tau_Y \\
&= \tau \circ T\mathrm{ev} \circ T(Y \times \iota) \circ \mathrm{str} && \text{naturality of str} \\
&= \tau \circ T\widetilde{\mathrm{ev}} \circ \mathrm{str} = (\widetilde{\tau}_Y)^\vee && \text{definition of } \widetilde{\mathrm{ev}} \text{ and } \widetilde{\tau}_Y. \qquad \square
\end{aligned}
$$

Compatibility of $\widetilde{\tau}_Y$ with unit and multiplication follows immediately from Sublemma A.2. See below; note also that $\mathbf{Posets}(Y, T\Omega) \overset{\iota}{\hookrightarrow} (T\Omega)^Y$ is a mono.

$$
\begin{tikzcd}
\mathbf{Posets}(Y, T\Omega) \arrow[rr, "\eta"] \arrow[dr, hook] \arrow[ddrr, equal] & & T(\mathbf{Posets}(Y, T\Omega)) \arrow[dr, hook] \arrow[dd, "\widetilde{\tau}_Y"' near start] & \\
& (T\Omega)^Y \arrow[rr, "\eta" near start] \arrow[ddrr, equal] & & T((T\Omega)^Y) \arrow[dd, "\tau_Y"] \\
& & \mathbf{Posets}(Y, T\Omega) \arrow[dr, hook] & \\
& & & (T\Omega)^Y
\end{tikzcd}
$$

$$
\begin{tikzcd}
T(T(\mathbf{Posets}(Y, T\Omega))) \arrow[rr, "\mu"] \arrow[dr, hook] \arrow[dd, "T\widetilde{\tau}_Y"'] & & T(\mathbf{Posets}(Y, T\Omega)) \arrow[dr, hook] \arrow[dd, "\widetilde{\tau}_Y"' near start] & \\
& T(T((T\Omega)^Y)) \arrow[rr, "\mu" near start] \arrow[dd, "T\tau_Y" near start] & & T((T\Omega)^Y) \arrow[dd, "\tau_Y"] \\
T(\mathbf{Posets}(Y, T\Omega)) \arrow[rr, "\widetilde{\tau}_Y"' near end] \arrow[dr, hook] & & \mathbf{Posets}(Y, T\Omega) \arrow[dr, hook] & \\
& T((T\Omega)^Y) \arrow[rr, "\tau_Y"'] & & (T\Omega)^Y
\end{tikzcd}
$$

Finally we are to show that the scheme in (15) is indeed an adjunction. We begin with an observation that the correspondence (13) preserves monotonicity, that is, it restricts as in the diagram below. Note here that X is a set and Y is a poset; and that $(T\Omega)^Y$ is a poset in the pointwise manner, as assumed.

$$
\begin{tikzcd}
\mathbf{Sets}(Y, (T\Omega)^X) \arrow[r, "\cong"] & \mathbf{Sets}(X, (T\Omega)^Y) \\
\mathbf{Posets}(Y, (T\Omega)^X) \arrow[r, "\cong"] \arrow[u, hook] & \mathbf{Sets}\big(X, \mathbf{Posets}(Y, T\Omega)\big) \arrow[u, hook]
\end{tikzcd}
\tag{A.2}
$$

The above restriction is possible because the following conditions are mutually equivalent: 1) $f\colon X \times Y \to T\Omega$ is monotone in Y; 2) its currying $f'\colon Y \to (T\Omega)^X$ is monotone; and 3) f's other currying $f''\colon X \to (T\Omega)^Y$ carries any $x \in X$ to a monotone function $f''(x)\colon Y \to T\Omega$.

In view of the prototype result (Lemma 2.15) and the last observation (A.2), what remains to be shown is that the top-to-bottom correspondence in (13) indeed yields an algebra homomorphism. Given a function $f'\colon Y \to (T\Omega)^X$ such that $f'(x)$ is an algebra homomorphism (from a to τ) for each $x \in X$, let $f\colon X \times Y \to T\Omega$ be its uncurrying and $f''\colon X \to (T\Omega)^Y$ be the other currying of f. By Lemma 2.15 we know that f'' is a homomorphism from a to τ_Y; and by (A.2) we know that f'' factors through $\mathbf{Posets}(Y, T\Omega) \hookrightarrow (T\Omega)^Y$. Our goal then immediately follows from the diagram below. Note that the rightmost square commutes by Sublemma A.2.

$$
\begin{array}{c}
\text{in } \left(K\downarrow \begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}\right)\\ \hline
\Big\downarrow{\scriptstyle F\circ P}\\
\text{in } \underline{\mathbf{Sets}}
\end{array}
\qquad
\begin{tikzcd}
\Big(Y,\ \big(\begin{smallmatrix}T^2Y\\ \downarrow\mu\\ TY\end{smallmatrix}\big) \xrightarrow{c} \big(\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}\big)\Big) \arrow[r, "g"] \arrow[d, "\eta_X\circ c\circ\eta_Y"] & \Big(Y',\ \big(\begin{smallmatrix}T^2Y'\\ \downarrow\mu\\ TY'\end{smallmatrix}\big) \xrightarrow{c'} \big(\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}\big)\Big) \\
\Big(X,\ \big(\begin{smallmatrix}T^2X\\ \downarrow\mu\\ TX\end{smallmatrix}\big) \xrightarrow{a} \big(\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}\big)\Big) & \Big(TX,\ \big(\begin{smallmatrix}T^3X\\ \downarrow\mu\\ T^2X\end{smallmatrix}\big) \xrightarrow{a\circ\mu_X=a\circ Ta} \big(\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}\big)\Big) \arrow[l, shift right, "\mathrm{id}_{TX}"'] \arrow[l, shift left, "\eta_X\circ a"] \\
\mathcal{EM}(T)\big(\begin{smallmatrix}T^2Y\\ \downarrow\mu\\ TY\end{smallmatrix},\ \begin{smallmatrix}TA\\ \downarrow\alpha\\ A\end{smallmatrix}\big) & \mathcal{EM}(T)\big(\begin{smallmatrix}T^2Y'\\ \downarrow\mu\\ TY'\end{smallmatrix},\ \begin{smallmatrix}TA\\ \downarrow\alpha\\ A\end{smallmatrix}\big) \arrow[l, "(\_)\circ\mu_{Y'}\circ Tg"'] \\
\mathcal{EM}(T)\big(\begin{smallmatrix}T^2X\\ \downarrow\mu\\ TX\end{smallmatrix},\ \begin{smallmatrix}TA\\ \downarrow\alpha\\ A\end{smallmatrix}\big) \arrow[u, "(\_)\circ Tc\circ T\eta_Y"] \arrow[r, shift left, "(\_)\circ\mu_X"] \arrow[r, shift right, "(\_)\circ Ta"'] & \mathcal{EM}(T)\big(\begin{smallmatrix}T^3X\\ \downarrow\mu\\ T^2X\end{smallmatrix},\ \begin{smallmatrix}TA\\ \downarrow\alpha\\ A\end{smallmatrix}\big)
\end{tikzcd}
$$

Fig. 1. The diagram of the limit (A.3).

$$
\begin{tikzcd}
TX \arrow[r, dashed] \arrow[d, "a"'] \arrow[rr, bend right=15] & T(\mathbf{Posets}(Y,T\Omega)) \arrow[r, hook] \arrow[d, "\widetilde{\tau_Y}"] & T((T\Omega)^Y) \arrow[d, "\tau_Y"] \\
X \arrow[r, dashed] \arrow[rr, bend right=15, "f''"'] & \mathbf{Posets}(Y,T\Omega) \arrow[r, hook] & (T\Omega)^Y
\end{tikzcd}
$$

This concludes the proof of Lemma 2.16. □

A.3. Proof of Lemma 2.18

Proof. We shall use a concrete presentation of right Kan extensions by pointwise limits; see [18, §X.3]. In the current setting, the desired right Kan extension $\mathrm{Ran}_{K^{\mathrm{op}}} F$ is given by the following limit:

$$
(\mathrm{Ran}_{K^{\mathrm{op}}} F)\big(\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}\big) \;=\; \mathrm{Lim}\Big(\big(\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}\downarrow K^{\mathrm{op}}\big) \xrightarrow{P} \mathcal{K}\ell(T)^{\mathrm{op}} \xrightarrow{F} \mathbf{Sets}\Big)\ , \tag{A.3}
$$

where P is the canonical projection functor from a comma category. We aim to show that the limit on the right-hand side is given by $\mathcal{EM}(T)\big(\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix},\ \begin{smallmatrix}TA\\ \downarrow\alpha\\ A\end{smallmatrix}\big)$. The diagram for the limit is illustrated in Fig. 1.

We make some remarks on Fig. 1.

1. In the upper half of Fig. 1, we illustrate (the opposite of) the index category in (A.3): note that $\big(K\downarrow \begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}\big)^{\mathrm{op}} \cong \big(\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}\downarrow K^{\mathrm{op}}\big)$.
2. In the first row, objects $(Y,c),(Y',c')$ and an arrow g between them designate general objects and arrows in the category.
3. In the second row, we present two special objects $(X,a),(TX,a\circ\mu_X)$ and two special arrows $\mathrm{id}_{TX},\eta_X\circ a$ in the index category $\big(K\downarrow \begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}\big)$. They play a special role in the rest of the proof.
   To see that id_{TX} and $\eta_X\circ a$ are indeed arrows in the comma category is easy, noting that $K\mathrm{id}_{TX}=\mu_X$ and $K(\eta_X\circ a)=Ta$.
4. On the vertical arrow $\eta_X\circ c\circ\eta_Y\colon Y\to TX$ in the upper half of Fig. 1: we note that, given any object (Y,c), we have such an arrow to a special object (X,a).
   To see that, notice first that $K(\eta_X\circ c\circ\eta_Y)=Tc\circ T\eta_Y$; we have to check that it is indeed an arrow in the comma category, i.e. $a\circ(Tc\circ T\eta_Y)=c$. This is proved by:
   $$
   a\circ(Tc\circ T\eta_Y) \;=\; c\circ\mu_Y\circ T\eta_Y \;=\; c\ , \tag{A.4}
   $$
   where the first equality is because c is an algebra homomorphism from μ_Y to a (the top row of Fig. 1).
5. The lower half of Fig. 1 is the image of the upper half, under the diagram functor $F\circ P$ in (A.3). We note that the diagram (18) commutes up-to a natural isomorphism (as is shown much like in Theorem 2.14); therefore

$$F(P(Y,c)) \;=\; FY \;\cong\; \mathcal{EM}(T)\big(KY,\, \begin{smallmatrix}TA\\ \downarrow\alpha\\ A\end{smallmatrix}\big) \;\cong\; \mathcal{EM}(T)\big(\begin{smallmatrix}T^2Y\\ \downarrow\mu\\ TY\end{smallmatrix},\, \begin{smallmatrix}TA\\ \downarrow\alpha\\ A\end{smallmatrix}\big)\;.$$

We use the last presentation for objects, under which the action of $F\circ P$ on arrows is presented by: an arrow $g\colon(Y,c)\to(Y',c')$ is carried to $(\_)\circ Kg = (\_)\circ\mu_{Y'}\circ Tg$ in the opposite direction. For the special arrows id_{TX}, $\eta_X\circ a$ and $\eta_X\circ c\circ\eta_Y$ in Fig. 1, we have already shown that their images under K are indeed as shown in Fig. 1—namely μ_X, Ta and $Tc\circ T_{\eta_Y}$, respectively.

We shall now introduce a cone γ whose vertex is $\mathcal{EM}(T)(\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix},\, \begin{smallmatrix}TA\\ \downarrow\alpha\\ A\end{smallmatrix})$. For an object $\Big(Y,\, (\begin{smallmatrix}T^2Y\\ \downarrow\mu\\ TY\end{smallmatrix}) \xrightarrow{c} (\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix})\Big)$ in $(K{\downarrow}\, \begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix})$, we assign the following function as the corresponding component.

$$\gamma_{(Y,c)} \;:=\; (\_)\circ c \quad : \mathcal{EM}(T)(\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix},\, \begin{smallmatrix}TA\\ \downarrow\alpha\\ A\end{smallmatrix}) \longrightarrow \mathcal{EM}(T)(\begin{smallmatrix}T^2Y\\ \downarrow\mu\\ TY\end{smallmatrix},\, \begin{smallmatrix}TA\\ \downarrow\alpha\\ A\end{smallmatrix})$$

The naturality of γ thus defined is easy: in the diagram below, we have $c'\circ Kg = c$ since g is an arrow in the comma category.

$$\underline{\text{in } (K\downarrow \begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix})} \qquad \begin{tikzcd}(Y,c) \arrow[rr, "g"] & & (Y',c')\end{tikzcd}$$

$$\underline{\text{in } \mathbf{Sets}} \qquad \begin{tikzcd}[column sep=small] \mathcal{EM}(T)(\mu_Y,\alpha) & & \mathcal{EM}(T)(\mu_{Y'},\alpha) \arrow[ll, "(\_)\circ Kg"'] \\ & \mathcal{EM}(T)(a,\alpha) \arrow[ul, "\gamma_{(Y,c)}=(\_)\circ c"] \arrow[ur, "\gamma_{(Y',c')}=(\_)\circ c'"'] & \end{tikzcd}$$

Universality of the cone γ thus defined remains to be shown. Let δ be another cone over the same diagram; since we are in **Sets**, we can assume that its vertex is a singleton 1. The cone δ, in particular, singles out an algebra homomorphism

$$\delta_{(X,a)} : \big(\begin{smallmatrix}T^2X\\ \downarrow\mu\\ TX\end{smallmatrix}\big) \longrightarrow \big(\begin{smallmatrix}TA\\ \downarrow\alpha\\ A\end{smallmatrix}\big) \tag{A.5}$$

as its (X,a)-component $\delta_{(X,a)}\colon 1\to\mathcal{EM}(T)(\mu,\alpha)$. See Fig. 1.

We claim that the arrow $\delta_{(X,a)}$ in (A.5) satisfies $\delta_{(X,a)}\circ\mu_X = \delta_{(X,a)}\circ Ta$ as in the diagram (A.6) below. Indeed, by the naturality of the cone δ we have $\delta_{(X,a)}\circ\mu_X = \delta_{(TX,a\circ\mu_X)} = \delta_{(TX,a\circ Ta)} = \delta_{(X,a)}\circ Ta$ (see the second and fourth rows of Fig. 1).

$$\begin{tikzcd} \big(\begin{smallmatrix}T^3X\\ \downarrow\mu\\ T^2X\end{smallmatrix}\big) \arrow[r, shift left, "\mu_X"] \arrow[r, shift right, "Ta"'] & \big(\begin{smallmatrix}T^2X\\ \downarrow\mu\\ TX\end{smallmatrix}\big) \arrow[r, "a"] \arrow[dr, "\delta_{(X,a)}"'] & \big(\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}\big) \arrow[d, dashed, bend left, "m"] \\ & & \big(\begin{smallmatrix}TA\\ \downarrow\alpha\\ A\end{smallmatrix}\big) \end{tikzcd} \tag{A.6}$$

The top row of the last diagram (A.6) is a coequalizer known as the canonical presentation of $\begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix}$ [18, §VI.7]; it splits in **Sets**, after applying the forgetful functor. Therefore we obtain an algebra homomorphism m as a mediating arrow, as in (A.6).

We identify this m with a function $1\to\mathcal{EM}(T)(a,\alpha)$ and claim that it is a mediating arrow from the cone δ to γ. See below.

$$\underline{\text{in } (K\downarrow \begin{smallmatrix}TX\\ \downarrow a\\ X\end{smallmatrix})} \qquad \begin{tikzcd}(Y,c) \arrow[rr, "g"] & & (Y',c')\end{tikzcd}$$

$$\underline{\text{in } \mathbf{Sets}} \qquad \begin{tikzcd}[column sep=small] & 1 \arrow[dl, "\delta_{(Y,c)}"'] \arrow[dr, "\delta_{(Y',c')}"] \arrow[dd, bend right, "m"'] & \\ \mathcal{EM}(T)(\mu_Y,\alpha) & & \mathcal{EM}(T)(\mu_{Y'},\alpha) \arrow[ll, "(\_)\circ Kg"'] \\ & \mathcal{EM}(T)(a,\alpha) \arrow[ul, "\gamma_{(Y,c)}=(\_)\circ c"] \arrow[ur, "\gamma_{(Y',c')}=(\_)\circ c'"'] & \end{tikzcd} \tag{A.7}$$

We have to show that $m\circ c = \delta_{(Y,c)}$.

$$
\begin{aligned}
m \circ c &= m \circ a \circ Tc \circ T\eta_Y && \text{since } c \text{ is a homomorphism, see (A.4)} \\
&= \delta_{(X,a)} \circ Tc \circ T\eta_Y && \text{definition of } m, \text{ see (A.6)} \\
&= \delta_{(Y,c)} && \text{naturality of } \delta, \text{ see the vertical arrows in Fig. 1.}
\end{aligned}
$$

This proves that m is a mediating arrow from δ to γ, as in (A.7).

Finally we prove uniqueness of such a mediating map m'. In particular it has to be compatible with the (X, a)-component of the cones.

$$
\underline{\text{in } \mathbf{Sets}} \qquad
\begin{tikzcd}
 & 1 \arrow[dl, "\delta_{(X,a)}"'] \arrow[dd, bend left, "m'"] \\
\mathcal{EM}(T)(\mu_X, \alpha) & \\
 & \mathcal{EM}(T)(a, \alpha) \arrow[ul, "\gamma_{(X,a)}=(\_)\circ a"]
\end{tikzcd}
$$

This is precisely the requirement that the triangle on the right in (A.6) commute. Hence uniqueness of m' follows from the universality of a coequalizer. This concludes the proof. □

A.4. Proof of Corollary 2.19

Proof. The proof goes much like that of Lemma 2.18. Only the last bit requires modification: let the vertex of a cone δ be, instead of a singleton 1, a two-point set $\mathbf{2} = (0 \sqsubseteq 1)$. Then each component of δ is a monotone function $\delta_{(Y,c)} \colon \mathbf{2} \to \mathcal{EM}(T)(\mu_Y, \tau)$.

Now each element $i \in \mathbf{2}$ determines a mediating arrow $m^{(i)} \colon \left(\begin{smallmatrix} TX \\ \downarrow a \\ X \end{smallmatrix}\right) \to \left(\begin{smallmatrix} TA \\ \downarrow \alpha \\ A \end{smallmatrix}\right)$ such that $m^{(i)} \circ a = \delta_{(X,a)}(i)$, as in (A.6). (Here note that $A = T\Omega$ and $\alpha = \tau$.) It suffices to show that $m^{(0)} \sqsubseteq m^{(1)}$.

$$
\begin{aligned}
m^{(0)} &= m^{(0)} \circ a \circ \eta_X && a \text{ is an Eilenberg–Moore algebra} \\
&= \delta_{(X,a)}(0) \circ \eta_X && \text{definition of } m^{(0)} \\
&\sqsubseteq \delta_{(X,a)}(1) \circ \eta_X && \delta_{(X,a)} \text{ is monotone, and Lemma 2.12} \\
&= \cdots = m^{(1)} . && \square
\end{aligned}
$$

References

[1] I. Hasuo, Generic weakest precondition semantics from monads enriched with order, in: Bonsangue [59], pp. 10–32, http://dx.doi.org/10.1007/978-3-662-44124-4.
[2] E.W. Dijkstra, A Discipline of Programming, Prentice Hall, 1976.
[3] C.A.R. Hoare, An axiomatic basis for computer programming, Commun. ACM 12 (1969) 576–580, 583.
[4] C. Morgan, Programming from Specifications, Prentice-Hall, 1990.
[5] D. Kozen, Semantics of probabilistic programs, J. Comput. System Sci. 22 (3) (1981) 328–350.
[6] C. Jones, Probabilistic non-determinism, Ph.D. thesis, Univ. Edinburgh, 1990.
[7] C. Morgan, A. McIver, K. Seidel, Probabilistic predicate transformers, ACM Trans. Program. Lang. Syst. 18 (3) (1996) 325–353.
[8] B. Jacobs, New directions in categorical logic, for classical, probabilistic and quantum logic, Log. Methods Comput. Sci. (2015), forthcoming, arXiv:1205.3940.
[9] B. Jacobs, Measurable spaces and their effect logic, in: LICS, IEEE Computer Society, 2013, pp. 83–92.
[10] B. Jacobs, Dijkstra monads in monadic computation, in: Bonsangue [59], pp. 135–150, http://dx.doi.org/10.1007/978-3-662-44124-4.
[11] D.J. Foulis, M.K. Bennett, Effect algebras and unsharp quantum logics, Found. Phys. 24 (10) (1994) 1331–1352.
[12] A. Kock, Monads and extensive quantities, arXiv:1103.6009, 2011.
[13] S. Goncharov, L. Schröder, A relatively complete generic Hoare logic for order-enriched effects, in: Proc. 28th Annual Symposium on Logic in Computer Science, LICS 2013, IEEE, 2013, pp. 273–282.
[14] C. Cîrstea, A coalgebraic approach to linear-time logics, in: A. Muscholl (Ed.), Foundations of Software Science and Computation Structures—17th International Conference, Proceedings, FOSSACS 2014, Held as Part of the European Joint Conferences on Theory and Practice of Software, ETAPS 2014, Grenoble, France, April 5–13, 2014, in: Lecture Notes in Computer Science, vol. 8412, Springer, 2014, pp. 426–440.
[15] G. Winskel, The Formal Semantics of Programming Languages, MIT Press, 1993.
[16] T. Heinosaari, M. Ziman, The Mathematical Language of Quantum Theory: From Uncertainty to Entanglement, Cambridge Univ. Press, 2012.
[17] S. Katsumata, T. Sato, Preorders on monads and coalgebraic simulations, in: F. Pfenning (Ed.), FoSSaCS, in: Lecture Notes in Computer Science, vol. 7794, Springer, 2013, pp. 145–160.
[18] S. Mac Lane, Categories for the Working Mathematician, 2nd edition, Springer, Berlin, 1998.
[19] M. Barr, C. Wells, Toposes, Triples and Theories, Springer, Berlin, 1985, available online.
[20] G.M. Kelly, Basic Concepts of Enriched Category Theory, LMS, vol. 64, Cambridge Univ. Press, 1982, available online http://www.tac.mta.ca/tac/reprints/articles/10/tr10abs.html.
[21] K. Keimel, G.D. Plotkin, Predicate transformers for extended probability and non-determinism, Math. Structures Comput. Sci. 19 (3) (2009) 501–539, http://dx.doi.org/10.1017/S0960129509007555.
[22] K. Keimel, A. Rosenbusch, T. Streicher, A Minkowski type duality mediating between state and predicate transformer semantics for a probabilistic nondeterministic language, Ann. Pure Appl. Logic 159 (3) (2009) 307–317, http://dx.doi.org/10.1016/j.apal.2008.06.020.

[23] K. Keimel, A. Rosenbusch, T. Streicher, Relating direct and predicate transformer partial correctness semantics for an imperative probabilistic-nondeterministic language, Theoret. Comput. Sci. 412 (25) (2011) 2701–2713, http://dx.doi.org/10.1016/j.tcs.2010.12.029.
[24] I. Hasuo, B. Jacobs, A. Sokolova, Generic trace semantics via coinduction, Log. Methods Comput. Sci. 3 (2007) 1–36.
[25] H. Kerstan, B. König, Coalgebraic trace semantics for continuous probabilistic transition systems, Log. Methods Comput. Sci. 9 (4) (2013), http://dx.doi.org/10.2168/LMCS-9(4:16)2013.
[26] A. Sokolova, Coalgebraic analysis of probabilistic systems, Ph.D. thesis, Techn. Univ. Eindhoven, 2005.
[27] M. Giry, A categorical approach to probability theory, in: Proc. Categorical Aspects of Topology and Analysis, in: Lecture Notes in Math., vol. 915, 1982, pp. 68–85.
[28] P. Panangaden, Labelled Markov Processes, Imperial College Press, 2009.
[29] I. Hasuo, N. Hoshino, Semantics of higher-order quantum computation via geometry of interaction, in: LICS, IEEE Computer Society, 2011, pp. 237–246.
[30] B. Jacobs, Categorical Logic and Type Theory, North Holland, Amsterdam, 1999.
[31] D. Pattinson, Coalgebraic modal logic: soundness, completeness and decidability of local consequence, Theoret. Comput. Sci. 309 (1–3) (2003) 177–193, http://dx.doi.org/10.1016/S0304-3975(03)00201-9.
[32] C. Kupke, D. Pattinson, Coalgebraic semantics of modal logics: an overview, Theoret. Comput. Sci. 412 (38) (2011) 5070–5094.
[33] C. Hermida, B. Jacobs, Structural induction and coinduction in a fibrational setting, Inform. and Comput. 145 (1998) 107–152.
[34] I. Hasuo, K. Cho, T. Kataoka, B. Jacobs, Coinductive predicates and final sequences in a fibration, in: Mathematical Foundations of Programming Semantics, MFPS XXIX, in: Electron. Notes Theor. Comput. Sci., vol. 298, 2013, pp. 197–214.
[35] B. Jacobs, Predicate logic for functors and monads, preprint, available at the author's webpage (March 2010).
[36] A. Kock, Strong functors and monoidal monads, Arch. Math. XXIII (1972) 113–120.
[37] T. Sato, Identifying all preorders on the subdistribution monad, Electron. Notes Theor. Comput. Sci. 308 (2014) 309–327.
[38] M. Hyland, J. Power, The category theoretic understanding of universal algebra: Lawvere theories and monads, Electron. Notes Theor. Comput. Sci. 172 (2007) 437–458.
[39] E. Grädel, W. Thomas, T. Wilke (Eds.), Automata, Logics, and Infinite Games: A Guide to Current Research 2001, Lecture Notes in Computer Science, vol. 2500, Springer, 2002.
[40] M. Jurdzinski, Small progress measures for solving parity games, in: H. Reichel, S. Tison (Eds.), STACS, in: Lecture Notes in Computer Science, vol. 1770, Springer, 2000, pp. 290–301.
[41] C. Kupke, Y. Venema, Coalgebraic automata theory: basic results, Log. Methods Comput. Sci. 4 (4) (2008) 1–43.
[42] L. Cheung, Reconciling nondeterministic and probabilistic choices, Ph.D. thesis, Radboud Univ., Nijmegen, 2006.
[43] H.H. Hansen, C. Kupke, A coalgebraic perspective on monotone modal logic, Electron. Notes Theor. Comput. Sci. 106 (2004) 121–143.
[44] H.H. Hansen, C. Kupke, E. Pacuit, Neighbourhood structures: bisimilarity and basic model theory, Log. Methods Comput. Sci. 5 (2) (2009).
[45] D. Varacca, G. Winskel, Distributing probabililty over nondeterminism, Math. Structures Comput. Sci. 16 (1) (2006) 87–113.
[46] R. Tix, K. Keimel, G.D. Plotkin, Semantic domains for combining probability and non-determinism, Electron. Notes Theor. Comput. Sci. 129 (2005) 1–104.
[47] N.A. Lynch, R. Segala, F.W. Vaandrager, Compositionality for probabilistic automata, in: R.M. Amadio, D. Lugiez (Eds.), CONCUR 2003, in: Lecture Notes in Comput. Sci., vol. 2761, Springer, 2003, pp. 204–222.
[48] S. Cattani, R. Segala, Decision algorithms for probabilistic bisimulation, in: L. Brim, P. Jancar, M. Kretínský, A. Kucera (Eds.), CONCUR, in: Lecture Notes in Computer Science, vol. 2421, Springer, 2002, pp. 371–385.
[49] M.M. Bonsangue, J.N. Kok, The weakest precondition calculus: recursion and duality, Form. Asp. Comput. 6 (6A) (1994) 788–800.
[50] G.D. Plotkin, J. Power, Adequacy for algebraic effects, in: F. Honsell, M. Miculan (Eds.), FoSSaCS, in: Lecture Notes in Computer Science, vol. 2030, Springer, 2001, pp. 1–24.
[51] C. Kissig, Y. Venema, Complementation of coalgebra automata, in: A. Kurz, M. Lenisa, A. Tarlecki (Eds.), CALCO, in: Lecture Notes in Comput. Sci., vol. 5728, Springer, 2009, pp. 81–96.
[52] Y. Venema, C. Kissig, Game bisimulations between basic positions, Talk at Highlights of Logic, Games and Automata, Paris, 2013.
[53] I. Hasuo, Generic forward and backward simulations II: probabilistic simulation, in: P. Gastin, F. Laroussinie (Eds.), CONCUR, in: Lecture Notes in Comput. Sci., vol. 6269, Springer, 2010, pp. 447–461.
[54] E. Moggi, Notions of computation and monads, Inform. and Comput. 93 (1) (1991) 55–92.
[55] S. Katsumata, Relating computational effects by $\top\top$-lifting, Inform. and Comput. 222 (2013) 228–246.
[56] N. Benton, J. Hughes, E. Moggi, Monads and effects, in: G. Barthe, P. Dybjer, L. Pinto, J. Saraiva (Eds.), Applied Semantics, International Summer School, APPSEM 2000, Caminha, Portugal, September 9–15, 2000, Advanced Lectures, in: Lecture Notes in Computer Science, vol. 2395, Springer, 2000, pp. 42–122.
[57] M. Hyland, G.D. Plotkin, J. Power, Combining effects: sum and tensor, Theoret. Comput. Sci. 357 (1–3) (2006) 70–99.
[58] G.D. Plotkin, M. Pretnar, Handling algebraic effects, Log. Methods Comput. Sci. 9 (4) (2013) 1–36.
[59] M.M. Bonsangue (Ed.), Coalgebraic Methods in Computer Science – 12th IFIP WG 1.3 International Workshop, CMCS 2014, Colocated with ETAPS 2014, Grenoble, France, April 5–6, 2014, Lecture Notes in Computer Science, vol. 8446, Springer, April 2014.