Logical Methods in Computer Science
Vol. 13(2:6)2017, pp. 1–26
https://lmcs.episciences.org/

Submitted Dec. 20, 2015
Published May 17, 2017

A RECIPE FOR STATE-AND-EFFECT TRIANGLES

BART JACOBS

Institute for Computing and Information Sciences, Radboud University Nijmegen, The Netherlands.
URL: `www.cs.ru.nl/B.Jacobs`
e-mail address: bart@cs.ru.nl

Abstract. In the semantics of programming languages one can view programs as state transformers, or as predicate transformers. Recently the author has introduced 'state-and-effect' triangles which capture this situation categorically, involving an adjunction between state- and predicate-transformers. The current paper exploits a classical result in category theory, part of Jon Beck's monadicity theorem, to systematically construct such a state-and-effect triangle from an adjunction. The power of this construction is illustrated in many examples, covering many monads occurring in program semantics, including (probabilistic) power domains.

1. Introduction

In program semantics three approaches can be distinguished.

- Interpreting programs themselves as morphisms in certain categories. Composition in the category then corresponds to sequential composition. Parallel composition may be modeled via tensors $\otimes$. Since [41] the categories involved are often Kleisli categories $\mathcal{K}\ell(T)$ of a monad T, where the monad T captures a specific form of computation: deterministic, non-deterministic, probabilistic, *etc.*
- Interpreting programs via their actions on states, as *state transformers*. For instance, in probabilistic programming the states may be probabilistic distributions over certain valuations (mapping variables to values). Execution of a program changes the state, by adapting the probabilities of valuations. The state spaces often have algebraic structure, and take the form of Eilenberg-Moore categories $\mathcal{EM}(T)$ of a monad T.
- Interpreting programs via their actions on predicates, as *predicate transformers*. The predicates involved describe what holds at a specific point. This validity may also be quantitative (or 'fuzzy'), describing that a predicate holds with a certain probability in the unit interval $[0, 1]$. Execution of a program may then adapt the validity of predicates. A particular form of semantics of this sort is weakest precondition computation [9]. In the context of (coalgebraic) modal logic, these predicate transformers appear as modal operators.

Key words and phrases: Duality, predicate transformer, state transformer, state-and-effect triangle.

LOGICAL METHODS IN COMPUTER SCIENCE
DOI:10.23638/LMCS-13(2:6)2017

A systematic picture of these three approaches has emerged in categorical language, using triangles of the form described below, see [22], and also [20, 21, 7].

$$\begin{tikzcd}[column sep=small]
\boxed{\textbf{Heisenberg}} & & \boxed{\textbf{Schrödinger}} \\
\mathbf{Log}^{\mathrm{op}} = \left(\begin{array}{c}\text{predicate}\\ \text{transformers}\end{array}\right) \arrow[rr, bend left] \arrow[rr, phantom, "\top"] & & \left(\begin{array}{c}\text{state}\\ \text{transformers}\end{array}\right) \arrow[ll, bend left] \\
& \big(\text{computations}\big) \arrow[ul, "\mathrm{Pred}"] \arrow[ur, "\mathrm{Stat}"'] &
\end{tikzcd} \tag{1.1}$$

The three nodes in this diagram represent categories of which only the morphisms are described. The arrows between these nodes are functors, where the two arrows $\rightleftarrows$ at the top form an adjunction. The two triangles involved should commute. In the case where two up-going 'predicate' and 'state' functors Pred and Stat in (1.1) are full and faithful, we have three equivalent ways of describing computations.

On morphisms, the predicate functor Pred in (1.1) yields what is called substitution in categorical logic, but what amounts to a weakest precondition operation in program semantics, or a modal operator in programming logic. The upper category on the left is of the form $\mathbf{Log}^{\mathrm{op}}$, where $\mathbf{Log}$ is some category of logical structures. The opposite category $(-)^{\mathrm{op}}$ is needed because predicate transformers operate in the reverse direction, taking a postcondition to a precondition.

In a setting of quantum computation this translation back-and-forth $\rightleftarrows$ in (1.1) is associated with the different approaches of Heisenberg (logic-based, working backwards) and Schrödinger (state-based, working forwards), see *e.g.* [16]. In quantum foundations one speaks of the duality between states and effects (predicates). Since the above triangles first emerged in the context of semantics of quantum computation [22], they are sometimes referred to as 'state-and-effect' triangles.

In certain cases the adjunction $\rightleftarrows$ in (1.1) forms — or may be restricted to — an equivalence of categories, yielding a duality situation. It shows the importance of duality theory in program semantics and logic; this topic has a long history, going back to [1].

In [22] it is shown that in the presence of relatively weak structure in a category $\mathbf{B}$, a diagram of the form (1.1) can be formed, with $\mathbf{B}$ as base category of computations, with predicates forming effect modules (see below) and with states forming convex sets. A category with this relatively weak structure is called an *effectus*, see [7].

The main contribution of this paper is a "new" way of generating state-and-effect triangles, namely from adjunctions. We write the word 'new' between quotes, because the underlying category theory uses a famous of result of Jon Beck, and is not new at all. What the paper contributes is mainly a new perspective: it reorganises the work of Beck in such a way that an appropriate triangle appears, see Section 2. The rest of the paper is devoted to illustrations of this recipe for triangles. These include Boolean and probabilistic examples, see Sections 3 and 5 respectively. The Boolean examples are all obtained from an adjunction using "homming into $2 = \{0, 1\}$", whereas the probabilistic (quantitative) examples all arise from "homming into $[0, 1]$", where $[0, 1]$ is the unit interval of probabilities. In between we consider Plotkin-style constructions via "homming into 3", where $3 = \{0, \bowtie, 1\}$ is a three-element ordered algebra.

The series of examples in this paper involves many mathematical structures, ranging from Boolean algebras to compact Hausdorff spaces and C^*-algebras. It is impossible to explain all these notions in detail here. Hence the reader is assumed to be reasonably familiar with these structures. It does not matter so much if some of the examples involve unfamiliar mathematical notions. The structure of these sections 3, 4 and 5 is clear enough — using 2, 3 and $[0, 1]$ as dualising object, respectively — and it does not matter if some of the examples are skipped.

An exception is made for the notions of effect algebra and effect module. They are explicitly explained (briefly) in the beginning of Section 5 because they play such a prominent role in quantitative logic.

The examples involve many adjunctions that are known in the literature. Here they are displayed in triangle form. In several cases monads arise that are familiar in coalgebraic research, like the neighbourhood monad $\mathcal{N}$ in Subsection 3.1, the monotone neighbourhood monad $\mathcal{M}$ in Subsection 3.2, the Hoare power domain monad $\mathcal{H}$ in Subsection 3.8, the Smyth power domain monad $\mathcal{S}$ in Subsection 3.9, the infinite distribution monad $\mathcal{D}_\infty$ in Subsection 5.4, the Giry monad $\mathcal{G}$ in Subsection 5.5, and the valuation monad $\mathcal{V}$ in Subsection 5.6. Also we will see several examples where we have pushed the recipe to a limit, and where the monad involved is simply the identity.

This paper extends the earlier conference version [23] with several order-theoretic examples, notably using complete lattices and directed complete partial orders (for various power domains).

2. A basic result about monads

We assume that the reader is familiar with the categorical concept of a monad T, and with its double role, describing a form of computation, via the associated Kleisli category $\mathcal{K}\ell(T)$, and describing algebraic structure, via the category $\mathcal{EM}(T)$ of Eilenberg-Moore algebras.

The following result is a basic part of the theory of monads, see *e.g.* [4, Prop. 3.15 and Exercise (KEM)] or [37, Prop. 6.5 and 6.7] or [3, Thm. 20.42], and describes the initiality and finality of the Kleisli category and Eilenberg-Moore category as 'adjunction resolutions' giving rise to a monad.

Theorem 1. Consider an adjunction $F \dashv G$ with induced monad $T = GF$. Then there are 'comparison' functors $\mathcal{K}\ell(T) \to \mathbf{A} \to \mathcal{EM}(T)$ in a diagram:

$$
\begin{tikzcd}
\mathcal{K}\ell(T) \arrow[rr, bend left=10, "L"] \arrow[drr, bend left=10] & & \mathbf{A} \arrow[rr, bend left=10, "K"] \arrow[rr, phantom, "\top"] \arrow[d, bend left, "G"] & & \mathcal{EM}(T) \arrow[ll, dotted, "M"] \arrow[dll, bend left=10] \\
 & & \mathbf{B} \arrow[ull, bend left=10] \arrow[ull, phantom, "\top"] \arrow[u, bend left, "F"] \arrow[u, phantom, "\dashv"] \arrow[urr, bend left=10] \arrow[urr, phantom, "\perp"] \arrow[loop below, "T=GF"] & &
\end{tikzcd}
\tag{2.1}
$$

where the functor $L\colon \mathcal{K}\ell(T) \to \mathbf{A}$ is full and faithful.

In case the category $\mathbf{A}$ has coequalisers (of reflexive pairs), then K has a left adjoint M, as indicated via the dotted arrow, satisfying $MKL \cong L$.

The famous monadicity theorem of Jon Beck gives conditions that guarantee that the functor $K\colon \mathbf{A} \to \mathcal{EM}(T)$ is an equivalence of categories, so that objects of $\mathbf{A}$ are algebras.

The existence of the left adjoint M is the part of this theorem that we use in the current setting. Other (unused) parts of Beck's theorem require that the functor G preserves and reflects coequalisers of reflexive pairs. For convenience we include a proof sketch.

Proof. We write η, ε for the unit and counit of the adjunction $F \dashv G$, so that η is also the unit of the induced monad $T = GF$, with multiplication $\mu = G(\varepsilon F)$. Define $L(X) = F(X)$ and $L\big(X \xrightarrow{f} GF(Y)\big) = \varepsilon_{F(Y)} \circ F(f) \colon F(X) \to F(Y)$. This functor L is full and faithful because there is a bijective adjoint correspondence:

$$\frac{F(X) \longrightarrow F(Y)}{\overline{\overline{\quad X \longrightarrow GF(Y) = T(Y) \quad}}}$$

The functor $K \colon \mathbf{A} \to \mathcal{EM}(T)$ is defined as:

$$K(A) = \begin{pmatrix} GFG(A) \\ \downarrow G(\varepsilon_A) \\ G(A) \end{pmatrix} \qquad \text{and} \qquad K\big(A \xrightarrow{f} B\big) = G(f).$$

We leave it to the reader to see that K is well-defined. On an object $X \in \mathcal{K}\ell(T)$, that is, on $X \in \mathbf{B}$, the result $KL(X)$ is the multiplication $\mu_X = G(\varepsilon_{FX})$ of the monad $T = GF$. For a Kleisli map $f \colon X \to T(Y)$ the map $KL(f)$ is Kleisli extension:

$$KL(f) = G(\varepsilon_{F(Y)} \circ F(f)) = \mu_Y \circ T(f) : T(X) \longrightarrow T(Y).$$

Assume now that the category $\mathbf{A}$ has coequalisers. For an algebra $a \colon T(X) \to X$ let $M(X, a)$ be the (codomain of the) coequaliser in:

$$\begin{tikzcd} FGF(X) \arrow[r, shift left, "F(a)"] \arrow[r, shift right, "\varepsilon_{F(X)}"'] & F(X) \arrow[r, two heads, "c"] & M(X,a) \end{tikzcd}$$

It is not hard to see that there is a bijective correspondence:

$$\frac{M(X,a) \xrightarrow{\;f\;} A \qquad\qquad \text{in } \mathbf{A}}{\overline{\overline{\begin{pmatrix} T(X) \\ \downarrow a \\ X \end{pmatrix} \xrightarrow[g]{} \begin{pmatrix} TG(A) \\ \downarrow G(\varepsilon_A) \\ G(A) \end{pmatrix} = K(A) \qquad \text{in } \mathcal{EM}(T)}}}$$

What remains is to show $MKL \cong L$. This follows because for each $X \in \mathbf{B}$, the following diagram is a coequaliser in $\mathbf{A}$.

$$\begin{tikzcd} FGFGF(X) \arrow[r, shift left, "F(\mu_X) = FG(\varepsilon_{F(X)})"] \arrow[r, shift right, "\varepsilon_{FGF(X)}"'] & FGF(X) \arrow[r, two heads, "\varepsilon_{F(X)}"] & F(X) \end{tikzcd}$$

Hence the codomain $MKL(X)$ of the coequaliser of $FKL(X) = FG(\varepsilon_{F(X)})$ and the counit map $\varepsilon_{FGF(X)}$ is isomorphic to $F(X) = L(X)$. Proving naturality of $MKL \cong L$ (w.r.t. Kleisli maps) is a bit of work, but is essentially straightforward. □

An essential 'aha moment' underlying this paper is that the above result can be massaged into triangle form. This is what happens in the next result, to which we will refer as the 'triangle corollary'. It is the 'recipe' that occurs in the title of this paper.

Corollary 2. Consider an adjunction $F \dashv G$, where F is a functor $\mathbf{B} \to \mathbf{A}$, the category $\mathbf{A}$ has coequalisers, and the induced monad on $\mathbf{B}$ is written as $T = GF$. Diagram (2.1) then gives rise to a triangle as below, where both up-going functors are full and faithful.

$$\begin{tikzcd} \mathbf{A} \arrow[rr, bend left, "K"] \arrow[rr, phantom, "\top"] & & \mathcal{EM}(T) \arrow[ll, bend left, "M"] \\ & \mathcal{K}\ell(T) \arrow[ul, "\mathrm{Pred}=L"] \arrow[ur, "KL=\mathrm{Stat}"'] & \end{tikzcd} \tag{2.2}$$

This triangle commutes, trivially from left to right, and up-to-isomorphism from right to left, since $MKL \cong L$. In this context we refer to the functor L as the 'predicate' functor Pred, and to the functor KL as the 'states' functor Stat. □

The remainder of the paper is devoted to instances of this triangle corollary. In each of these examples the category $\mathbf{A}$ will be of the form $\mathbf{P}^{\mathrm{op}}$, where $\mathbf{P}$ is a category of predicates (with equalisers). The full and faithfulness of the functors $\mathrm{Pred}\colon \mathcal{K}\ell(T) \to \mathbf{P}^{\mathrm{op}}$ and $\mathrm{Stat}\colon \mathcal{K}\ell(T) \to \mathcal{EM}(T)$ means that there are bijective correspondences between:

$$\frac{X \xrightarrow{\text{computations}} T(Y)}{\overline{\mathrm{Pred}(Y) \xrightarrow[\text{predicate transformers}]{} \mathrm{Pred}(X)}} \qquad \frac{X \xrightarrow{\text{computations}} T(Y)}{\overline{\mathrm{Stat}(X) \xrightarrow[\text{state transformers}]{} \mathrm{Stat}(Y)}} \tag{2.3}$$

Since $\mathrm{Stat}(X) = T(X)$, the correspondence on the right is given by Kleisli extension, sending a map $f\colon X \to T(Y)$ to $\mu \circ T(f)\colon T(X) \to T(Y)$. This bijective correspondence on the right is a categorical formality. But the correspondence on the left is much more interesting, since it precisely describes to which kind of predicate transformers (preserving which structure) computations correspond. Such a correspondence is often referred to as 'healthiness' of the semantics. It is built into our triangle recipe, as will be illustrated below.

Before looking at triangle examples, we make the following points.

- As discussed in [22], the predicate functor $\mathrm{Pred}\colon \mathcal{K}\ell(T) \to \mathbf{A}$ is in some cases an *enriched* functor, preserving additional structure that is of semantical/logical relevance. For instance, operations on programs, like $\cup$ for non-deterministic sum, may be expressed as structure on Kleisli homsets. Preservation of this structure by the functor Pred gives the logical rules for dealing with such structure in weakest precondition computations. These enriched aspects will not be elaborated in the current context.
- The triangle picture that we use here is refined in [17]. In all our examples, the adjunction $F \dashv G$ arises by homming into a dualising object Ω. The induced monad T is then of the 'double dual' form $\Omega^{\Omega^{(-)}}$. The approach of [17] uses monads S having a map of monads $S \Rightarrow T = GF$; this monad map corresponds bijectively to an Eilenberg-Moore algebra $S(\Omega) \to \Omega$, which is understood as a logical modality.

3. Dualising with 2

We split our series of examples in three parts, determined by the dualising object: 2, 3, or $[0, 1]$. The first series of Boolean examples is obtained via adjunctions that involve 'homming into 2', where $2 = \{0, 1\}$ is the 2-element set of Booleans.

3.1. **Sets and sets.** We will present examples in the following manner, in three stages.

$$\begin{tikzcd} \mathbf{Sets}^{\mathrm{op}} \arrow[d, bend right, "\mathcal{P}=\mathrm{Hom}(-,2)"'] \arrow[d, phantom, "\dashv"] \\ \mathbf{Sets} \arrow[u, bend right, "\mathcal{P}=\mathrm{Hom}(-,2)"'] \arrow[loop below, "\mathcal{N}=\mathcal{P}\mathcal{P}"] \end{tikzcd} \qquad \begin{array}{c} \mathcal{P}(X) \xrightarrow{\mathbf{Sets}^{\mathrm{op}}} Y \\ \hline\hline Y \xrightarrow{\mathbf{Sets}} \mathcal{P}(X) \\ \hline\hline X \xrightarrow[\mathbf{Sets}]{} \mathcal{P}(Y) \end{array} \qquad \begin{tikzcd} \mathbf{Sets}^{\mathrm{op}} \arrow[rr, bend left] \arrow[rr, phantom, "\top"] & & \mathcal{E}\mathcal{M}(\mathcal{N}) = \mathbf{CABA} \arrow[ll, bend left] \\ & \mathcal{K}\ell(\mathcal{N}) \arrow[ul, "\mathrm{Pred}"] \arrow[ur, "\mathrm{Stat}"'] & \end{tikzcd}$$

On the left we describe the adjunction that forms the basis for the example at hand, together with the induced monad. In this case we have the familiar fact that the contravariant powerset functor $\mathcal{P}\colon \mathbf{Sets} \to \mathbf{Sets}^{\mathrm{op}}$ is adjoint to itself, as indicated. The induced double-powerset monad $\mathcal{P}\mathcal{P}$ on **Sets** is known in the coalgebra/modal logic community as the neighbourhood monad $\mathcal{N}$, because its coalgebras are related to neighbourhood frames in modal logic.

In the middle, the bijective correspondence is described that forms the basis of the adjunction. In this case there is the obvious correspondence between functions $Y \to \mathcal{P}(X)$ and functions $X \to \mathcal{P}(Y)$ — which are all relations on $X \times Y$.

On the right the result is shown of applying the triangle corollary 2 to the adjunction on the left. The full and faithfulness of the predicate functor $\mathrm{Pred}\colon \mathcal{K}\ell(\mathcal{N}) \to \mathbf{Sets}^{\mathrm{op}}$ plays an important role in the approach to coalgebraic dynamic logic in [13], relating coalgebras $X \to \mathcal{N}(X)$ to predicate transformer functions $\mathcal{P}(X) \to \mathcal{P}(X)$, going in the opposite direction. The category $\mathcal{E}\mathcal{M}(\mathcal{N})$ of Eilenberg-Moore algebras of the neighbourhood monad $\mathcal{N}$ is the category **CABA** of complete atomic Boolean algebras (see *e.g.* [45]). The adjunction $\mathbf{Sets}^{\mathrm{op}} \rightleftarrows \mathcal{E}\mathcal{M}(\mathcal{N})$ is thus an equivalence.

3.2. **Sets and posets.** We now restrict the adjunction in the previous subsection to posets.

$$\begin{tikzcd} \mathbf{PoSets}^{\mathrm{op}} \arrow[d, bend right, "\mathcal{P}=\mathrm{Hom}(-,2)"'] \arrow[d, phantom, "\dashv"] \\ \mathbf{Sets} \arrow[u, bend right, "\mathrm{Up}=\mathrm{Hom}(-,2)"'] \arrow[loop below, "\mathcal{M}=\mathrm{Up}\mathcal{P}"] \end{tikzcd} \qquad \begin{array}{c} Y \xrightarrow{\mathbf{PoSets}} \mathcal{P}(X) \\ \hline\hline X \xrightarrow[\mathbf{Sets}]{} \mathrm{Up}(Y) \end{array} \qquad \begin{tikzcd} \mathbf{PoSets}^{\mathrm{op}} \arrow[rr, bend left] \arrow[rr, phantom, "\top"] & & \mathcal{E}\mathcal{M}(\mathcal{M}) = \mathbf{CDL} \arrow[ll, bend left] \\ & \mathcal{K}\ell(\mathcal{M}) \arrow[ul, "\mathrm{Pred}"] \arrow[ur, "\mathrm{Stat}"'] & \end{tikzcd}$$

The functor $\mathrm{Up}\colon \mathbf{PoSets}^{\mathrm{op}} \to \mathbf{Sets}$ sends a poset Y to the collection of upsets $U \subseteq Y$, satisfying $y \geq x \in U$ implies $y \in U$. These upsets can be identified with monotone maps $p\colon Y \to 2$, namely as $p^{-1}(1)$.

Notice that this time there is a bijective correspondence between computations $X \to \mathcal{M}(Y) = \mathrm{Up}\mathcal{P}(Y)$ and *monotone* predicate transformers $\mathcal{P}(Y) \to \mathcal{P}(X)$. This fact is used in [13]. The algebras of the monad $\mathcal{M}$ are completely distributive lattices, see [39] and [30, I, Prop. 3.8].

3.3. **Sets and meet-semilattices.** We now restrict the adjunction further to meet semilattices, that is, to posets with finite meets $\wedge, \top$.

$$\begin{tikzcd} \mathbf{MSL}^{\mathrm{op}} \arrow[d, bend right, "\mathcal{P}=\mathrm{Hom}(-,2)"'] \arrow[d, phantom, "\dashv"] \\ \mathbf{Sets} \arrow[u, bend right, "\mathrm{Hom}(-,2)"'] \arrow[loop below, "\mathcal{F}=\mathbf{MSL}(\mathcal{P}(-){,}2)"] \end{tikzcd} \qquad \begin{array}{c} Y \xrightarrow{\mathbf{MSL}} \mathcal{P}(X) \\ \hline\hline X \xrightarrow[\mathbf{Sets}]{} \mathbf{MSL}(Y, 2) \end{array} \qquad \begin{tikzcd} \mathbf{MSL}^{\mathrm{op}} \arrow[rr, bend left] \arrow[rr, phantom, "\top"] & & \mathcal{E}\mathcal{M}(\mathcal{F}) = \mathbf{CCL} \arrow[ll, bend left] \\ & \mathcal{K}\ell(\mathcal{F}) \arrow[ul, "\mathrm{Pred}"] \arrow[ur, "\mathrm{Stat}"'] & \end{tikzcd}$$

Morphisms in the category **MSL** of meet semilattices preserve the meet $\wedge$ and the top element $\top$ (and hence the order too). For $Y \in \mathbf{MSL}$ one can identify a map $Y \to 2$ with a *filter* of Y, that is, with an upset $U \subseteq Y$ closed under $\wedge, \top$.

The resulting monad $\mathcal{F}(X) = \mathbf{MSL}(\mathcal{P}(X), 2)$ gives the filters in $\mathcal{P}(X)$. This monad is thus called the *filter monad*. In [47] it is shown that its category of algebras $\mathcal{EM}(\mathcal{F})$ is the category **CCL** of continuous complete lattices, that is, of complete lattices in which each element x is the (directed) join $x = \bigvee\{y \mid y \ll x\}$ of the elements way below it.

3.4. **Sets and complete lattices.** A poset is called a complete lattice if each subset has a join, or equivalently, if each subset has a meet. Since these complete lattices will be used in several examples, we elaborate some basic properties first. We shall consider two categories with complete lattices as objects, namely:

- $\mathbf{CL}_\bigvee$ whose morphisms preserve all joins $\bigvee$;
- $\mathbf{CL}_\bigwedge$ whose morphisms preserve all meets $\bigwedge$.

We write L^{op} for the complete lattice obtained from L by reversing the order. Thus, $f\colon L \to K$ in $\mathbf{CL}_\bigvee$ gives a map $f\colon L^{\mathrm{op}} \to K^{\mathrm{op}}$ in $\mathbf{CL}_\bigwedge$. Hence we have an isomorphism $\mathbf{CL}_\bigvee \cong \mathbf{CL}_\bigwedge$. Notice that we have:

$$\mathbf{CL}_\bigwedge(L, K) \cong \mathbf{CL}_\bigvee(L^{\mathrm{op}}, K^{\mathrm{op}}) \quad \text{as sets}$$

But:

$$\mathbf{CL}_\bigwedge(L, K)^{\mathrm{op}} \cong \mathbf{CL}_\bigvee(L^{\mathrm{op}}, K^{\mathrm{op}}) \quad \text{as posets}$$

There is another isomorphism between these two categories of complete lattices. A basic fact in order theory is that each map $f\colon L \to K$ in $\mathbf{CL}_\bigvee$ has a right adjoint $f^{\#}\colon K \to L$ in $\mathbf{CL}_\bigwedge$, given by:

$$f^{\#}(b) = \bigvee\{x \in L \mid f(x) \le b\}. \tag{3.1}$$

Clearly, $f(a) \le b$ implies $a \le f^{\#}(b)$. For the reverse direction we apply f to an inequality $a \le f^{\#}(b)$ and obtain:

$$f(a) \;\le\; f\big(f^{\#}(b)\big) \;=\; f\big(\bigvee\{x \mid f(x) \le b\}\big) \;=\; \bigvee\{f(x) \mid f(x) \le b\} \;\le\; b.$$

This gives an isomorphism of categories $\mathbf{CL}_\bigvee \cong \big(\mathbf{CL}_\bigwedge\big)^{\mathrm{op}}$. Via a combination with the above isomorphism $\mathbf{CL}_\bigvee \cong \mathbf{CL}_\bigwedge$ we see that the two categories $\mathbf{CL}_\bigvee$ and $\mathbf{CL}_\bigwedge$ are self-dual.

Lemma 3. For a complete lattice L there are isomomorphisms of posets:

$$\mathbf{CL}_\bigvee\big(L, 2\big) \xrightarrow{\;\cong\;} L^{\mathrm{op}} \xleftarrow{\;\cong\;} \mathbf{CL}_\bigvee\big(L, 2^{\mathrm{op}}\big) \tag{3.2}$$

Similarly there are isomorphisms:

$$\mathbf{CL}_\bigwedge\big(L, 2\big) \xrightarrow{\;\cong\;} L^{\mathrm{op}} \xleftarrow{\;\cong\;} \mathbf{CL}_\bigwedge\big(L, 2^{\mathrm{op}}\big) \tag{3.3}$$

Proof. We restrict ourselves to describing the four isomorphisms. The isomorphism on the left in (3.2) sends a join-preserving map $\varphi\colon L \to 2$ and an element $a \in L$ to:

$$\widehat{\varphi} = \bigvee\{x \in L \mid \varphi(x) = 0\} \qquad \text{and} \qquad \widehat{a}(x) = \begin{cases} 0 & \text{if } x \le a \\ 1 & \text{otherwise.} \end{cases}$$

The isomorphism on the right in (3.2) maps a $\varphi\colon L \to 2^{\mathrm{op}}$ and $a \in L$ to:

$$\widetilde{\varphi} = \bigvee\{x \mid \varphi(x) = 1\} \qquad \text{and} \qquad \widetilde{a}(x) = 1 \iff x \leq a.$$

We turn to the isomorphisms in (3.3). They are a consequence of (3.2) since:

$$\mathbf{CL}_{\bigwedge}(L, 2) \cong \mathbf{CL}_{\bigvee}(L^{\mathrm{op}}, 2^{\mathrm{op}})^{\mathrm{op}} \cong ((L^{\mathrm{op}})^{\mathrm{op}})^{\mathrm{op}} \cong L^{\mathrm{op}}.$$

And similarly:

$$\mathbf{CL}_{\bigwedge}(L, 2^{\mathrm{op}}) \cong \mathbf{CL}_{\bigvee}(L^{\mathrm{op}}, 2)^{\mathrm{op}} \cong ((L^{\mathrm{op}})^{\mathrm{op}})^{\mathrm{op}} \cong L^{\mathrm{op}}.$$

The isomorphism on the left in (3.3) is described explicitly by:

$$\widehat{\varphi} = \bigwedge\{x \in L \mid \varphi(x) = 1\} \qquad \text{and} \qquad \widehat{a}(x) = 1 \iff a \leq x.$$

The isomorphism on the right in (3.3) is described explicitly by:

$$\widetilde{\varphi} = \bigwedge\{x \in L \mid \varphi(x) = 0\} \qquad \text{and} \qquad \widetilde{a}(x) = 0 \iff a \leq x. \qquad \square$$

We note that the composite isomorphisms $\mathbf{CL}_{\bigvee}(L, 2) \cong \mathbf{CL}_{\bigvee}(L, 2^{\mathrm{op}})$ in (3.2) and $\mathbf{CL}_{\bigwedge}(L, 2) \cong \mathbf{CL}_{\bigwedge}(L, 2^{\mathrm{op}})$ in (3.3) are given by $\varphi \mapsto \neg\varphi$, where $\neg\varphi(x) = 1$ iff $\varphi(x) = 0$.

The state-and-effect triangle of this subsection is given by the following situation.

$$\begin{tikzcd}
(\mathbf{CL}_{\bigwedge})^{\mathrm{op}} \arrow[d, bend left, "\mathrm{Hom}(-,2^{\mathrm{op}})\cong(-)^{\mathrm{op}}"] \arrow[d, phantom, "\dashv"] \\
\mathbf{Sets} \arrow[u, bend left, "\mathcal{P}=\mathrm{Hom}(-,2^{\mathrm{op}})"] \arrow[loop below, "\mathcal{P}(-)"]
\end{tikzcd}
\qquad
\begin{array}{c}
L \xrightarrow{\mathbf{CL}_{\bigwedge}} \mathcal{P}(X) \\
\hline\hline
X \xrightarrow[\mathbf{Sets}]{} L
\end{array}
\qquad
\begin{tikzcd}
(\mathbf{CL}_{\bigwedge})^{\mathrm{op}} \arrow[rr, bend left] \arrow[rr, phantom, "\top"] & & \mathcal{EM}(\mathcal{P}) = \mathbf{CL}_{\bigvee} \arrow[ll, bend left] \\
& \mathcal{K}\ell(\mathcal{P}) \arrow[ul, "\mathrm{Pred}"] \arrow[ur, "\mathrm{Stat}"'] &
\end{tikzcd}$$

The upgoing functor on the left $\mathcal{P} = \mathrm{Hom}(-, 2^{\mathrm{op}})$ is the contravariant powerset functor. In the other direction, the functor $L \mapsto \mathrm{Hom}(L, 2^{\mathrm{op}}) \cong L^{\mathrm{op}}$, by (3.3), maps a complete lattice L to its underlying set. It sends a $\bigwedge$-preserving map $L \to K$ to the associated ($\bigvee$-preserving) map $K \to L$.

The adjoint correspondence in the middle sends a meet-preserving map $f\colon L \to \mathcal{P}(X)$ and a function $g\colon X \to L$ to the transposes:

$$\overline{f}(x) = \bigwedge\{a \in L \mid x \in f(a)\} \qquad \text{and} \qquad \overline{g}(a) = \{x \mid g(x) \leq a\}.$$

By taking $L = \mathcal{P}(Y)$ we get the classical healthiness of the $\Box$-predicate transformer semantics for non-deterministic computation [9], with a bijective correspondence between Kleisli maps $X \to \mathcal{P}(Y)$ and meet-preserving maps $\mathcal{P}(Y) \to \mathcal{P}(X)$.

The adjunction $\rightleftarrows$ in the state-and-effect triangle on the right is an isomorphism of categories, as discussed before Lemma 3. This triangle captures the essence of non-deterministic program semantics from [9], involving computations, predicate transformation and state transformation.

There is also an adjunction that gives rise to $\Diamond$-predicate transformer semantics, as join preserving maps. In order to describe it properly, with opposite orders, we need to use posets instead of sets, see Subsection 3.7 below.

3.5. **Sets and Boolean algebras.** We further restrict the adjunction $\mathbf{MSL}^{\mathrm{op}} \leftrightarrows \mathbf{Sets}$ from Subsection 3.3 to the category $\mathbf{BA}$ of Boolean algebras.

$$
\begin{tikzcd}
\mathbf{BA}^{\mathrm{op}} \arrow[d, bend left, "\mathrm{Hom}(-,2)"] \arrow[d, phantom, "\dashv"] \\
\mathbf{Sets} \arrow[u, bend left, "\mathcal{P}=\mathrm{Hom}(-,2)"] \arrow[loop below, "\mathcal{U}=\mathbf{BA}(\mathcal{P}(-),2)"]
\end{tikzcd}
\qquad
\frac{Y \xrightarrow[]{\mathbf{BA}} \mathcal{P}(X)}{X \xrightarrow[\mathbf{Sets}]{} \mathbf{BA}(Y,2)}
\qquad
\begin{tikzcd}
\mathbf{BA}^{\mathrm{op}} \arrow[rr, bend left] \arrow[rr, phantom, "\top"] & & \mathcal{EM}(\mathcal{U}) = \mathbf{CH} \arrow[ll, bend left] \\
& \mathcal{K}\ell(\mathcal{U}) \arrow[ul, "\mathrm{Pred}"] \arrow[ur, "\mathrm{Stat}"'] &
\end{tikzcd}
$$

The functor $\mathrm{Hom}(-,2)\colon \mathbf{BA}^{\mathrm{op}} \to \mathbf{Sets}$ sends a Boolean algebra Y to the set $\mathbf{BA}(Y,2)$ of Boolean algebra maps $Y \to 2$. They can be identified with *ultrafilters* of Y. The resulting monad $\mathcal{U} = \mathbf{BA}(\mathcal{P}(-),2)$ is the *ultrafilter monad*, sending a set X to the BA-maps $\mathcal{P}(X) \to 2$, or equivalently, the ultrafilters of $\mathcal{P}(X)$.

An important result of Manes (see [38], and also [30, III, 2.4]) says that the category of Eilenberg-Moore algebras of the ultrafilter monad $\mathcal{U}$ is the category $\mathbf{CH}$ of compact Hausdorff spaces. This adjunction $\mathbf{BA}^{\mathrm{op}} \rightleftarrows \mathbf{CH}$ restricts to an equivalence $\mathbf{BA}^{\mathrm{op}} \simeq \mathbf{Stone}$ called Stone duality, where $\mathbf{Stone} \hookrightarrow \mathbf{CH}$ is the full subcategory of Stone spaces — in which each open subset is the union of the clopens contained in it.

3.6. **Sets and complete Boolean algebras.** We can restrict the adjunction $\mathbf{BA}^{\mathrm{op}} \rightleftarrows \mathbf{Sets}$ from the previous subsection to an adjunction $\mathbf{CBA}^{\mathrm{op}} \rightleftarrows \mathbf{Sets}$ between *complete* Boolean algebras and sets. The resulting monad on $\mathbf{Sets}$ is of the form $X \mapsto \mathbf{CBA}(\mathcal{P}(X),2)$. But here we hit a wall, since this monad is the identity.

Lemma 4. For each set X the unit map $\eta\colon X \to \mathbf{CBA}(\mathcal{P}(X),2)$, given by $\eta(x)(U) = 1$ iff $x \in U$, is an isomorphism.

Proof. Let $h\colon \mathcal{P}(X) \to 2$ be a map of complete Boolean algebras, preserving the BA-structure and all joins (unions). Since each $U \in \mathcal{P}(X)$ can be described as union of singletons, the function h is determined by its values $h(\{x\})$ for $x \in X$. We have $1 = h(X) = \bigcup_{x \in X} h(\{x\})$. Hence $h(\{x\}) = 1$ for some $x \in X$. But then $h(X - \{x\}) = h(\neg\{x\}) = \neg h(\{x\}) = \neg 1 = 0$. This implies $h(\{x'\}) = 0$ for each $x' \neq x$. Hence $h = \eta(x)$. $\square$

3.7. **Posets and complete lattices.** We return to complete lattices, from Subsection 3.4, but now consider them with join-preserving maps:

$$
\begin{tikzcd}
(\mathbf{CL}_{\bigvee})^{\mathrm{op}} \arrow[d, bend left, "\mathrm{Hom}(-,2)\cong(-)^{\mathrm{op}}"] \arrow[d, phantom, "\dashv"] \\
\mathbf{PoSets} \arrow[u, bend left, "\mathrm{Up}=\mathrm{Hom}(-,2)"] \arrow[loop below, "\mathrm{Dwn}"]
\end{tikzcd}
\qquad
\frac{L \xrightarrow{\mathbf{CL}_{\bigvee}} \mathrm{Up}(X)}{X \xrightarrow[\mathbf{PoSets}]{} L^{\mathrm{op}}}
\qquad
\begin{tikzcd}
(\mathbf{CL}_{\bigvee})^{\mathrm{op}} \arrow[rr, bend left] \arrow[rr, phantom, "\top"] & & \mathcal{EM}(\mathrm{Dwn}) = \mathbf{CL}_{\bigvee} \arrow[ll, bend left] \\
& \mathcal{K}\ell(\mathrm{Dwn}) \arrow[ul, "\mathrm{Pred}"] \arrow[ur, "\mathrm{Stat}"'] &
\end{tikzcd}
$$

Recall from Subsection 3.2 that we write $\mathrm{Up}(X)$ for the poset of upsets in a poset X, ordered by inclusion. This poset is a complete lattice via unions. For a monotone function $f\colon X \to Y$ between posets, the inverse image map f^{-1} restricts to $\mathrm{Up}(Y) \to \mathrm{Up}(X)$ and preserves unions. This gives the functor $\mathrm{Up}\colon \mathbf{PoSets} \to (\mathbf{CL}_{\bigvee})^{\mathrm{op}}$, which is isomorphic to $\mathrm{Hom}(-,2)$, as already noted in Subsection 3.2.

The downgoing functor $\mathrm{Hom}(-,2)\colon (\mathbf{CL}_{\bigvee})^{\mathrm{op}} \to \mathbf{PoSets}$ is isomorphic to taking the opposite order $(-)^{\mathrm{op}}$, see Lemma 3. A map $f\colon L \to K$ in $\mathbf{CL}_{\bigvee}$ is mapped to the monotone adjoint function $f^{\#}\colon K^{\mathrm{op}} \to L^{\mathrm{op}}$, as in (3.1), given by $f^{\#}(a) = \bigvee\{b \mid f(b) \leq a\}$.

We elaborate the bijective correspondence in the middle in detail.

- Given a join preserving map $f\colon L \to \mathrm{Up}(X)$ we define $\overline{f}\colon X \to L^{\mathrm{op}}$ in $\mathbf{PoSets}$ as $\overline{f}(x) = \bigvee\{a \in L \mid x \notin f(a)\}$. It is easy to see that $\overline{f}$ is monotone.
- In the other direction, given a monotone function $g\colon X \to L^{\mathrm{op}}$ we take $\overline{g}\colon L \to \mathrm{Up}(X)$ to be $\overline{g}(a) = \{x \in X \mid a \not\leq g(x)\}$. This yields an upset: if $x' \geq x \in \overline{g}(a)$, then $a \not\leq g(x')$. If $a \leq g(x')$ then $a \leq g(x)$ since $g(x') \leq g(x)$ because g reverses the order. This map $\overline{g}$ preserves joins since:
$$\begin{aligned} x \notin \overline{g}(\textstyle\bigvee_i a_i) \iff \textstyle\bigvee_i a_i \leq g(x) &\iff \forall i.\, a_i \leq g(x) \\ &\iff \forall i.\, x \notin \overline{g}(a_i) \iff x \notin \textstyle\bigcup_i \overline{g}(a_i). \end{aligned}$$

The transformations are each other's inverse:
$$\overline{\overline{g}}(x) = \bigvee\{a \mid x \notin \overline{g}(a)\} = \bigvee\{a \mid a \leq g(x)\} = g(x).$$
And:
$$x \notin \overline{\overline{f}}(a) \iff a \leq \overline{f}(x) = \bigvee\{b \mid x \notin f(b)\} \overset{(*)}{\iff} x \notin f(a).$$
The direction $(\Leftarrow)$ of the marked equivalence is obvious, and for $(\Rightarrow)$ we reason as follows. Let $a \leq \overline{f}(x) = \bigvee\{b \mid x \notin f(b)\}$. Then, using that f preserves joins:
$$f(a) \subseteq f\big(\bigvee\{b \mid x \notin f(b)\}\big) = \bigcup\{f(b) \mid x \notin f(b)\}.$$
Hence if $x \in f(a)$, then $x \in f(b)$ for some $b \in L$ with $x \notin f(b)$. Clearly, this is impossible.

We notice that the induced monad on $\mathbf{PoSets}$ is given by taking downsets $\mathrm{Dwn}(-)$, since the reversed poset $\mathrm{Up}(X)^{\mathrm{op}}$ is the poset $\mathrm{Dwn}(X)$ of downsets of X, ordered by inclusion. The isomorphism $\mathrm{Up}(X)^{\mathrm{op}} \cong \mathrm{Dwn}(X)$ is given by complements. For a monotone map $f\colon X \to Y$ the function $\mathrm{Dwn}(f)\colon \mathrm{Dwn}(X) \to \mathrm{Dwn}(Y)$ sends a downset $U \subseteq X$ to the downclosure of the image: ${\downarrow} f(U) = \{y \in Y \mid \exists x \in U.\, y \leq f(x)\}$. This function $\mathrm{Dwn}(f)$ is clearly monotone.

If we incorporate this isomorphism $\mathrm{Up}(X)^{\mathrm{op}} \cong \mathrm{Dwn}(X)$, then the adjoint correspondence specialises to:
$$\frac{\mathrm{Up}(Y) \xrightarrow{\;f\;} \mathrm{Up}(X)}{X \xrightarrow[\;g\;]{} \mathrm{Dwn}(Y)} \qquad \text{given by} \qquad \begin{cases} \overline{f}(x) = \bigcap\{U \in \mathrm{Dwn}(X) \mid x \notin f(\neg U)\} \\ \overline{g}(V) = \{x \in X \mid g(x) \cap V \neq \emptyset\} \end{cases} \tag{3.4}$$
We see that in this adjunction $(\mathbf{CL}_{\bigvee})^{\mathrm{op}} \leftrightarrows \mathbf{PoSets}$ gives rise to the $\Diamond$-predicate transformer. Again, healthiness is built into the construction.

This correspondence gives a handle on the downsets monad Dwn on $\mathbf{PoSets}$. The unit $\eta\colon X \to \mathrm{Dwn}(X)$ is obtained by transposing the identity on $\mathrm{Up}(X)$, so that:
$$\eta(x) = \overline{\mathrm{id}}(x) = \bigcap\{U \in \mathrm{Dwn}(X) \mid x \notin \neg U\} = \bigcap\{U \in \mathrm{Dwn}(X) \mid x \in U\} = {\downarrow} x.$$
The multiplication $\mu\colon \mathrm{Dwn}^2(X) \to \mathrm{Dwn}(X)$ is given by union. To see this, we first transpose the identity map on $\mathrm{Dwn}(X)$ upwards, giving a map $\varepsilon\colon \mathrm{Up}(X) \to \mathrm{Up}(\mathrm{Dwn}(X))$ described by:
$$\varepsilon(V) = \{U \in \mathrm{Dwn}(X) \mid U \cap V \neq \emptyset\}.$$

We then obtain the multiplication map μ of the downset monad by applying the $(-)^{\mathrm{op}}$ functor to ε, and using complement on both sides:

$$\begin{aligned}\mu(B) \;=\; \neg\varepsilon^{\mathrm{op}}(\neg B) &= \neg\bigcup\{V\in\mathrm{Up}(X)\mid \varepsilon(V)\subseteq\neg B\}\\ &= \bigcap\{V\in\mathrm{Dwn}(X)\mid B\subseteq\neg\varepsilon(\neg V)\}\\ &= \bigcap\{V\in\mathrm{Dwn}(X)\mid \forall U\in B.\, U\cap\neg V=\emptyset\}\\ &= \bigcap\{V\in\mathrm{Dwn}(X)\mid \bigcup B\subseteq V\}\\ &= \bigcup B.\end{aligned}\tag{3.5}$$

This last equation holds because the union of downclosed sets is downclosed.

The category $\mathcal{EM}(\mathrm{Dwn})$ of Eilenberg-Moore algebras of this downset monad Dwn is the category $\mathbf{CL}_\bigvee$ of complete lattices and join-preserving maps. Hence the adjunction $\rightleftarrows$ above on the right is an isomorphism of categories.

3.8. **Dcpo's and complete lattices.** We write **Dcpo** for the category with directed complete partial orders (dcpos) as objects, and (Scott) continuous functions (preserving directed joins) as morphisms between them. A subset $U\subseteq X$ of a dcpo X is (Scott) open if U is an upset satisfying for each directed collection (x_i), if $\bigvee_i x_i\in U$, then $x_i\in U$ for some index i. The (Scott) closed sets are then the downsets that are closed under directed joins. We write $\mathcal{O}(X)$ and $\mathcal{C}\ell(X)$ for the sets of open and closed subsets of X.

Lemma 5. For each dcpo X there are isomorphisms:

$$\mathcal{O}(X)\cong\mathbf{Dcpo}(X,2)\qquad\text{and}\qquad\mathcal{C}\ell(X)\cong\mathbf{Dcpo}(X,2^{\mathrm{op}}).\tag{3.6}$$

Moreover, via complements we have an isomorphism of complete lattices $\mathcal{O}(X)^{\mathrm{op}}\cong\mathcal{C}\ell(X)$. In combination with (3.2) we get $\mathcal{C}\ell(X)\cong\mathbf{CL}_\bigvee(\mathcal{O}(X),2)$.

Proof. The first isomorphism in (3.6) sends an open subset $U\subseteq X$ to the function $\widehat{U}\colon X\to 2$ given by $\widehat{U}(x)=1$ iff $x\in U$. In the other direction, for a continuous function $\varphi\colon X\to 2$ we take the open subset $\widehat{\varphi}=\{x\in L\mid\varphi(x)=1\}$. Similarly, the second isomorphism sends a closed subset V to the function $\widetilde{V}\colon X\to 2^{\mathrm{op}}$ with $\widetilde{V}(x)=1$ iff $x\notin V$, and conversely sends $\psi\colon X\to 2^{\mathrm{op}}$ to $\widetilde{\psi}=\{x\in X\mid\psi(x)=0\}$. $\square$

We shall be using a subcategory $\mathbf{CL}_{\bigvee,1}\hookrightarrow\mathbf{CL}_\bigvee$ of complete lattices where maps are not only join-preserving but also preserve the top element 1. The following is then an easy adaptation of Lemma 3 and Lemma 5.

Lemma 6. For a complete lattice L and a dcpo X there are isomorphisms:

$$\mathbf{CL}_{\bigvee,1}(L,2)\cong\big(L\backslash 1\big)^{\mathrm{op}}\qquad\text{and thus}\qquad\mathbf{CL}_{\bigvee,1}(\mathcal{O}(X),2)\cong\mathcal{C}\ell(X)\backslash\emptyset.$$

Proof. Following the proof of Lemma 3 one easily shows that $\varphi\colon L\to 2$ in $\mathbf{CL}_\bigvee$ preserves 1 iff the corresponding element $\widehat{\varphi}=\bigvee\{x\mid\varphi(x)=0\}\in L$ is not 1. This gives the first isomorphism. The second one then easily follows, see Lemma 5. $\square$

We now restrict the adjunction $(\mathbf{CL}_\bigvee)^{\mathrm{op}} \leftrightarrows \mathbf{PoSets}$ from Subsection 3.7 to dcpos.

$$\begin{tikzcd}
(\mathbf{CL}_{\bigvee,1})^{\mathrm{op}} \arrow[d, bend left, "\mathrm{Hom}(-,2)"] \arrow[d, phantom, "\dashv"] \\
\mathbf{Dcpo} \arrow[u, bend left, "\mathcal{O}=\mathrm{Hom}(-,2)"] \arrow[loop below, "\mathcal{H}"]
\end{tikzcd}
\qquad
\frac{L \xrightarrow{\mathbf{CL}_{\bigvee,1}} \mathcal{O}(X)}{X \xrightarrow[\mathbf{Dcpo}]{} (L\backslash 1)^{\mathrm{op}}}
\qquad
\begin{tikzcd}
(\mathbf{CL}_{\bigvee,1})^{\mathrm{op}} \arrow[rr, bend left] \arrow[rr, phantom, "\top"] & & \mathcal{EM}(\mathcal{H}) \arrow[ll, bend left] \\
& \mathcal{K}\ell(\mathcal{H}) \arrow[ul, "\mathrm{Pred}"] \arrow[ur, "\mathrm{Stat}"'] &
\end{tikzcd}$$

In this situation we encounter Smyths [43] topological view on predicate transformers, as maps between complete lattices of open subsets $\mathcal{O}(X) \cong \mathrm{Hom}(X, 2)$, see Lemma 5. Notice that the poset $(L\backslash 1)^{\mathrm{op}}$ is a dcpo, with directed joins given by meets in L.

The adjoint transposes for the above adjunction are defined precisely as in Subsection 3.7. We only have to prove some additional properties.

- For $f\colon L \to \mathcal{O}(X)$ in $\mathbf{CL}_{\bigvee,1}$ we have $\overline{f}(x) = \bigvee\{a \mid x \notin f(a)\}$. We check:
  - $\overline{f}(x) \neq 1$ for each $x \in X$. Towards a contradiction, let $\overline{f}(x) = 1$. Then, using that f preserves 1 and $\bigvee$ we get:
  $$x \in X = f(1) = f(\overline{f}(x)) = \bigcup\{f(a) \mid x \notin f(a)\}.$$
  We get $x \in \bigcup\{f(a) \mid x \notin f(a)\}$, which is impossible.
  - The function $\overline{f}\colon X \to (L\backslash 1)^{\mathrm{op}}$ sends directed joins $\bigvee_i x_i$ to meets. By monotonicity of $\overline{f}\colon X \to L^{\mathrm{op}}$ we have $\overline{f}(\bigvee_i x_i) \leq \overline{f}(x_j)$, for each j, and thus $\overline{f}(\bigvee_i x_i) \leq \bigwedge_i \overline{f}(x_i)$. For the reverse inequality we reason as follows.
    - We have $x_j \notin f(\bigwedge_i \overline{f}(x_i))$, for each j; otherwise, because $f\colon L \to \mathcal{O}(X)$ is monontone and preserves joins, we get a contradiction:
    $$\begin{aligned} x_j \in f\big(\textstyle\bigwedge_i \overline{f}(x_i)\big) \leq f\big(\overline{f}(x_j)\big) &= f\big(\textstyle\bigvee\{y \mid x_j \notin f(y)\}\big) \\ &= \textstyle\bigcup\{f(y) \mid x_j \notin f(y)\}. \end{aligned}$$
    - Since $f(\bigwedge_i \overline{f}(x_i))$ is open, we get $\bigvee_i x_i \notin f(\bigwedge_i \overline{f}(x_i))$.
    - But then $\bigwedge_i \overline{f}(x_i) \leq \bigvee\{y \mid \bigvee_i x_i \notin f(y)\} = \overline{f}(\bigvee_i x_i)$.
- We also check that $\overline{g}(a) = \{x \mid a \not\leq g(x)\}$ is open. We already know from Subsection 3.7 that it is an upset. So let $\bigvee_i x_i \in \overline{g}(a)$. Then $a \not\leq g(\bigvee_i x_i)$. Let $a \leq g(x_i)$ for all i. Then $a \leq \bigwedge_i g(x_i) = g(\bigvee_i x_i)$, which is impossible. Hence $a \not\leq g(x_i)$ for some index i. But then $x_i \in \overline{g}(a)$.

  We need to add that $\overline{g}$ preserves the top element 1, *i.e.* that $\overline{g}(1) = X$. We thus have to show that $x \in \overline{g}(1)$ holds for each x. But this is clear, since $1 \not\leq g(x)$ *i.e.* $g(x) \neq 1$. The latter holds because g has type $X \to (L\backslash 1)^{\mathrm{op}}$.

The induced monad on $\mathbf{Dcpo}$ is $X \mapsto \big(\mathcal{O}(X)\backslash X\big)^{\mathrm{op}} \cong \mathcal{C}\ell(X)\backslash\emptyset$. This is what is called the Hoare power monad [2], written as $\mathcal{H}$, which sends a dcpo to its non-empty closed subsets. For a continuous map $f\colon X \to Y$ we have $\mathcal{H}(f)\colon \mathcal{H}(X) \to \mathcal{H}(Y)$ given by $\mathcal{H}(f)(U) = \overline{f(U)}$, that is, by the (topological) closure of the image. The unit $\eta\colon X \to \mathcal{H}(X)$ of the Hoare monad is determined as $\eta(x) = \downarrow x$, and the multiplication $\mu\colon \mathcal{H}^2(X) \to \mathcal{H}(X)$ as $\mu(A) = \overline{\bigcup A}$. This closure arises in the last step of (3.5).

The predicate transformer $\mathcal{O}(Y) \to \mathcal{O}(X)$ that is bijectively associated with a Kleisli map $g\colon X \to \mathcal{H}(Y)$ is the $\Diamond$-version, given by $g^\Diamond(V) = \{x \mid V \cap g(x) \neq \emptyset\}$. Like in (3.4) the bijective correspondence has to take the isomorphism $\mathcal{O}(X)^{\mathrm{op}} \cong \mathcal{C}\ell(X)$ via complement $\neg$ into account.

The Eilenberg-Moore algebra of the Hoare monad are the dcpos with a binary join operation. They are also called affine complete lattices, see *e.g.* [19].

3.9. **Dcpo's and Preframes.** A *preframe* is a dcpo with finite meets, in which the binary meet operation $\wedge$ is continuous in both variables. We write $\mathbf{PreFrm}$ for the category of preframes, where maps are both (Scott) continuous and preserve finite meets $(\wedge, \top)$. The two-element set $2 = \{0, 1\} = \{\bot, \top\}$ is a preframe, with obvious joins and meets. Each set of opens of a topological space is also a preframe.

In fact we shall use a subcategory $\mathbf{PreFrm}_0 \hookrightarrow \mathbf{PreFrm}$ of preframes with a bottom element 0, which is preserved by (preframe) homomorphisms. We shall use this category as codomain of the functor $\mathcal{O} = \mathrm{Hom}(-, 2) : \mathbf{Dcpo} \to (\mathbf{PreFrm}_0)^{\mathrm{op}}$.

We obtain a functor in the opposite direction also by homming into 2. We note that for a preframe L the preframe-homomorphisms $f : L \to 2$ correspond to Scott open filters $f^{-1}(1) \subseteq L$, that is, to filters which are at the same time open subsets in the Scott topology. If we require that f is a map in $\mathbf{PreFrm}_0$, additionally preserving 0, then the Scott open filter $f^{-1}(1)$ is proper, that is, not the whole of L.

We shall write the resulting functor as $\mathcal{OF} = \mathrm{Hom}(-, 2) : (\mathbf{PreFrm}_0)^{\mathrm{op}} \to \mathbf{Dcpo}$. Here we use that these proper Scott open filters, ordered by inclusion, form a dcpo.

If we put things together we obtain:

$$\begin{tikzcd}
(\mathbf{PreFrm}_0)^{\mathrm{op}} \arrow[d, bend left, "\mathrm{Hom}(-,2)\cong\mathcal{OF}"] \arrow[d, phantom, "\dashv"] \\
\mathbf{Dcpo} \arrow[u, bend left, "\mathcal{O}=\mathrm{Hom}(-,2)"] \arrow[loop below, "\mathcal{S}"]
\end{tikzcd}
\qquad
\begin{array}{c}
L \xrightarrow{\mathbf{PreFrm}_0} \mathcal{O}(X) \\
\hline\hline
X \xrightarrow[\mathbf{Dcpo}]{} \mathcal{OF}(L)
\end{array}
\qquad
\begin{tikzcd}
(\mathbf{PreFrm}_0)^{\mathrm{op}} \arrow[rr, bend left] \arrow[rr, phantom, "\top"] & & \mathcal{EM}(\mathcal{S}) \arrow[ll, bend left] \\
& \mathcal{K}\ell(\mathcal{S}) \arrow[ul, "\mathrm{Pred}"] \arrow[ur, "\mathrm{Stat}"'] &
\end{tikzcd}$$

The induced monad $\mathcal{S}(X) = \mathcal{OF}(\mathcal{O}(X))$ takes the proper Scott open filters in the preframe $\mathcal{O}(X)$ of Scott open subsets of a dcpo X. This is the Smyth power domain, see [34]. We recall the Hofmann-Mislove theorem [18, 35]: in a sober topological space Y, the Scott open filters in $\mathcal{O}(Y)$ correspond to compact saturated subsets of Y. This subset is non-empty if and only if the corresponding filter is proper. We also recall that if X is a continuous dcpo, where each element is the directed join of elements way below it, then its Scott topology is sober, see *e.g.* [30, VII, Lemma 2.6]. This explains why the Smyth power domain is often defined on continuous dcpos. We shall not follow this route here, and will continue to work with functions instead of with subsets.

The induced functor $\mathrm{Pred} : \mathcal{K}\ell(\mathcal{S}) \to (\mathbf{PreFrm}_0)^{\mathrm{op}}$ is full and faithful, corresponding to healthiness of the predicate transformer semantics. Specifically, for a Kleisli map $g : X \to \mathcal{S}(Y) = \mathcal{OF}(\mathcal{O}(Y))$ we have $\mathrm{Pred}(g) : \mathcal{O}(Y) \to \mathcal{O}(X)$ given by the preframe homomorphism:

$$\mathrm{Pred}(g)(V) = \{x \in X \mid V \in g(x)\}.$$

The Eilenberg-Moore algebras of the Smyth power domain monad $\mathcal{S}$ are dcpos with an additional binary meet operation.

4. Dualising with 3

Using a three-element set 3 as dualising object is unusual. We will elaborate one example only, leading to a description of the Plotkin power domain on the category of dcpos. We start from the following notion, which seems new, but is used implicitly in the theory of Plotkin power domains, notably in [15], see also [5].

Definition 7. A *Plotkin algebra* is a poset X with least and greatest elements $0, 1 \in X$, and with a binary operation $\amalg$ and a special element $\bowtie \in X$ such that:

- $\amalg$ is idempotent, commutative, associative, and monotone;
- $\bowtie$ is an absorbing element for $\amalg$, so that $x \amalg \bowtie \, = \, \bowtie \, = \, \bowtie \amalg x$.

A Plotkin algebra is called directed complete if the poset X is a dcpo and the operation $\amalg$ is continuous. We write **DcPA** for the category of directed complete Plotkin algebras. A morphism in this category is a continuous function that preserves $\amalg$ and $\bowtie, 0, 1$.

Each meet semilattice $(X, \wedge, 1)$ with a least element 0 is a Plotkin algebra with $\bowtie = 0$. Similarly, each join semilattice $(X, \vee, 0)$ with a greatest element 1 is a Plotkin algebra with $\bowtie = 1$. These observations can be extended to the directed complete case via functors:

$$\mathbf{CL}_{\bigvee,1} \longrightarrow \mathbf{DcPA} \longleftarrow \mathbf{PreFrm}_0$$

They give a connection with the categories that we have seen in Subsections 3.8 and 3.9 for the Hoare and Smyth power domain.

A *frame* is complete lattice whose binary meet operation $\wedge$ preserves all joins on both sides. The morphisms in the category **Frm** of frames preserve both joins $\bigvee$ and finite meets $(\wedge, 1)$. Hence there are forgetful functors.

$$\mathbf{CL}_{\bigvee,1} \longleftarrow \mathbf{Frm} \longrightarrow \mathbf{PreFrm}_0$$

But there is also another construction to obtain a Plotkin algebra from a frame.

Definition 8. Each frame X gives rise to a directed complete Plotkin algebra, written as $X \ltimes X$, via:

$$X \ltimes X \, = \, \{(x, y) \in X \times X \mid x \geq y\}.$$

It carries the product dcpo structure, and forms a Plotkin algebra with:

$$(x, y) \amalg (x', y') \, = \, (x \vee x', y \wedge y') \qquad \text{and} \qquad \bowtie \, = \, (1, 0).$$

This operation $\amalg$ is continuous since X is a frame.

Explicitly, the projections form maps of Plotkin algebras in:

$$(X, 0, 1, \vee, 1) \xleftarrow{\pi_1} (X \ltimes X, (0, 0), (1, 1), \amalg, \bowtie) \xrightarrow{\pi_2} (X, 0, 1, \wedge, 0) \tag{4.1}$$

We shall also use functions $\mathrm{in}_1, \mathrm{in}_2 \colon X \to X \ltimes X$ defined by:

$$\mathrm{in}_1(x) \, = \, (x, 0) \qquad \text{and} \qquad \mathrm{in}_2(y) \, = \, (1, y).$$

These are *not* maps of Plotkin algebras, since $\mathrm{in}_1(1) = \bowtie \neq 1$ and $\mathrm{in}_2(0) = \bowtie \neq 0$. But we do have $\mathrm{in}_1(0) = 0$ and $\mathrm{in}_2(1) = 1$, and also the following structure is preserved.

$$(X, \vee, 1) \xrightarrow{\mathrm{in}_1} (X \ltimes X, \amalg, \bowtie) \xleftarrow{\mathrm{in}_2} (X, \wedge, 0) \tag{4.2}$$

The $\ltimes$ construction yields a three-element algebra $2 \ltimes 2$ that will be described more directly below, following [15]. We use it as dualising object.

Example 9. For the two-element frame $2 = \{0, 1\}$ the Plotkin algebra $2 \ltimes 2$ is a three-element set, which we can also describe as:

$$3 = \{0, \bowtie, 1\} \qquad \text{where} \qquad 0 \leq \bowtie \leq 1.$$

This order is obviously both complete and cocomplete. It is determined for $a, b \in 3$ by:

$$a \leq b \text{ iff } \text{ both } \begin{cases} a = 1 \Rightarrow b = 1 \\ b = 0 \Rightarrow a = 0. \end{cases} \tag{4.3}$$

The isomorphism $j = (j_1, j_2) \colon 3 \xrightarrow{\cong} 2 \ltimes 2$ is given by:

$$j(0) = (0, 0) \qquad j(\bowtie) = (1, 0) = \bowtie \qquad j(1) = (1, 1).$$

The two components $j_i = \pi_i \circ j \colon 3 \to 2$ are monotone, and satisfy $j_1 \geq j_2$.

This isomorphism $3 \cong 2 \ltimes 2$ makes 3 into a (directed complete) Plotkin algebra, via $\bowtie \in 3$ and $\amalg \colon 3 \times 3 \to 3$ determined by:

$$a \amalg b = \begin{cases} 0 & \text{if } a = b = 0 \\ 1 & \text{if } a = b = 1 \\ \bowtie & \text{otherwise.} \end{cases}$$

Finally we notice that the two maps $j_1, j_2 \colon 3 \to 2$ are maps of Plotkin algebras:

$$(2, 0, 1, \vee, 1) \xleftarrow{\;j_1\;} (3, 0, 1, \amalg, \bowtie) \xrightarrow{\;j_2\;} (2, 0, 1, \wedge, 0) \tag{4.4}$$

The following result is the analogue of Lemma 5, but with the dcpo 3 instead of 2. The correspondence is mentioned in [5], just before Lemma 4.11.

Lemma 10. For a dcpo X there is a bijective correspondence between:

$$\begin{array}{c} X \xrightarrow{\;f\;} 3 \text{ in } \mathbf{Dcpo} \\ \hline\hline X \overset{g_1}{\underset{g_2}{\rightrightarrows}} 2 \text{ in } \mathbf{Dcpo} \text{ with } g_1 \geq g_1 \\ \hline\hline (U_1, U_2) \in \mathcal{O}(X) \ltimes \mathcal{O}(X) \end{array}$$

As a result there is an isomorphism:

$$\mathbf{Dcpo}(X, 3) \cong \mathcal{O}(X) \ltimes \mathcal{O}(X) \qquad \text{given by} \qquad f \longmapsto (\, \{x \mid f(x) \neq 0\},\ \{x \mid f(x) = 1\}\,).$$

This is an isomorphism of Plotkin algebras, where the left hand side carries the pointwise Plotkin algebra structure inherited from 3.

The (equivalent) structures in this lemma form predicates on the dcpo X. The last description tells that such a predicate is a pair of opens $U_1, U_2 \in \mathcal{O}(X)$ with $U_1 \supseteq U_2$. This predicate is true if $U_1 = U_2 = X$ and false if $U_1 = U_2 = \emptyset$. In this 'logic', predicates come equipped with a binary operation $\amalg$; its logical interpretation is not immedidately clear.

Proof. The second, lower correspondence is given by Lemma 5, so we concentrate on the first one. It works as follows.

- Given $f \colon X \to 3$ in **Dcpo** we obtain continuous maps $\overline{f}_i = j_i \circ f \colon X \to 2$ by compositon, with $f_1 \geq f_2$, since $j_1 \geq j_2$, see Example 9.

- In the other direction, given $g = (g_1, g_2)$ we define $\overline{g}\colon X \to 3$ as:

$$\overline{g}(x) = \begin{cases} 0 & \text{if } g_1(x) = 0 \\ \bowtie & \text{if } g_1(x) = 1 \text{ and } g_2(x) = 0 \\ 1 & \text{if } g_2(x) = 1. \end{cases}$$

  We first show that $\overline{g}$ is monotone. So let $x \leq y$ in X. We use the characterisation (4.3).
  – Let $\overline{g}(x) = 1$, so that $g_2(x) = 1$. But then $g_2(y) \geq g_2(x) = 1$, so that $\overline{g}(y) = 1$.
  – If $\overline{g}(y) = 0$, then $g_1(x) \leq g_1(y) = 0$, so that $\overline{g}(x) = 0$.
  Next, let (x_i) be a directed collection in X. Since $\overline{g}$ is monotone we have $\bigvee_i \overline{g}(x_i) \leq \overline{g}(\bigvee_i x_i)$. For the reverse inequality we use (4.3) again.
  – Let $\overline{g}(\bigvee_i x_i) = 1$, so that $g_2(\bigvee_i x_i) = \bigvee_i g_2(x_i) = 1$. Then $g_2(x_i) = 1$ for some index i, for which then $\overline{g}(x_i) = 1$. Hence $\bigvee_i \overline{g}(x_i) = 1$.
  – Let $\bigvee_i \overline{g}(x_i) = 0$, so that $\overline{g}(x_i) = 0$ for all i, and thus $g_1(x_i) = 0$. But then $g_1(\bigvee_i x_i) = \bigvee_i g_1(x_i) = 0$. Hence $\overline{g}(\bigvee_i x_i) = 0$.

It is easy to see that $\overline{\overline{f}} = f$ and $\overline{\overline{g}} = g$. □

Here is another fundamental correspondence, see also [5, Obs. 4.10].

Lemma 11. For frames X, Y there is a bijective correspondence:

$$\begin{array}{c} X \ltimes X \xrightarrow{\;f\;} Y \ltimes Y \quad \text{in } \mathbf{DcPA} \\ \hline\hline X \xrightarrow[g_1]{} Y \text{ in } \mathbf{CL}_{\vee,1} \quad \text{and} \quad X \xrightarrow[g_2]{} Y \text{ in } \mathbf{PreFrm}_0 \quad \text{with } g_1 \geq g_2 \end{array}$$

Proof. The correspondence is given as follows.

- For $f\colon X \ltimes X \to Y \ltimes Y$ in **DcPA** we take the following continuous functions.

$$\overline{f}_1 = \Big(X \xrightarrow{\text{in}_1} X \ltimes X \xrightarrow{f} Y \ltimes Y \xrightarrow{\pi_1} Y\Big) \quad \text{and} \quad \overline{f}_2 = \Big(X \xrightarrow{\text{in}_2} X \ltimes X \xrightarrow{f} Y \ltimes Y \xrightarrow{\pi_2} Y\Big)$$

  They preserve $0, 1, \bowtie, \amalg$ by (4.1) and (4.2). For instance,

$$\overline{f}_1(1) = \pi_1\big(f(\text{in}_1(1))\big) = \pi_1\big(f(\bowtie)\big) = \pi_1(\bowtie) = 1.$$

  And:

$$\begin{aligned} \overline{f}_1(x \vee y) = \pi_1\big(f(\text{in}_1(x \vee y))\big) &= \pi_1\big(f(\text{in}_1(x) \amalg \text{in}_1(y))\big) \\ &= \pi_1\big(f(\text{in}_1(x)) \amalg f(\text{in}_1(y))\big) \\ &= \pi_1\big(f(\text{in}_1(x))\big) \vee \pi_1\big(f(\text{in}_1(y))\big) \\ &= \overline{f}_1(x) \vee \overline{f}_1(y). \end{aligned}$$

  We claim that for $(x, x') \in X \ltimes X$ the following two equations hold.

$$\overline{f}_1(x) = \pi_1(f(x, x')) \qquad \text{and} \qquad \overline{f}_2(x') = \pi_2(f(x, x')). \tag{$*$}$$

  We only prove the first one, since the second one works analogously. We have to prove $\overline{f}_1(x) = \pi_1(f(x, 0)) = \pi_1(f(x, x'))$. The inequality $\leq$ holds by monotonicity, so it suffices to prove $\geq$. In $Y \ltimes Y$ we have:

$$f(x, 0) \amalg f(x, x') = f\big((x, 0) \amalg (x, x')\big) = f(x \vee x, 0 \wedge x') = f(x, 0)$$

  By applying the first projection we obtain:

$$\pi_1(f(x, 0)) \vee \pi_1(f(x, x')) = \pi_1\big(f(x, 0) \amalg f(x, x')\big) = \pi_1(f(x, 0)).$$

Hence $\pi_1(f(x,x')) \leq \pi_1(f(x,0))$.

We use these equations $(*)$ to prove $\overline{f}_1 \geq \overline{f}_2$. For an arbitrary $x \in X$ we have $(x,x) \in X \ltimes X$, and so:

$$\overline{f}_1(x) \stackrel{(*)}{=} \pi_1(f(x,x)) \geq \pi_2(f(x,x)) \stackrel{(*)}{=} \overline{f}_2(x).$$

- In the other direction, given $g_1 \colon X \to Y$ in $\mathbf{CL}_{\vee,1}$ and $g_2 \colon X \to Y$ in $\mathbf{PreFrm}_0$ we define $\overline{g} \colon X \ltimes X \to Y \ltimes Y$ by:
$$\overline{g}(x,x') = (\, g_1(x),\ g_2(x')\,).$$
This is well-defined: we have $x \geq x'$, so $g_1(x) \geq g_1(x') \geq g_2(x')$. It is easy to see that $\overline{g}$ is a continuous map of Plotkin algebras.

We prove that these operations yield a bijective correspondence. First,

$$\overline{\overline{g}}_1(x) = \pi_1\big(\overline{g}(\mathrm{in}_1(x))\big) = \pi_1\big(\overline{g}(x,0)\big) = \pi_1(g_1(x), g_2(0)) = g_1(x).$$

Similarly we get $\overline{\overline{g}}_2(x) = g_2(x)$. Next, in the other direction,

$$\overline{\overline{f}}(x,x') = (\,\overline{f}_1(x),\ \overline{f}_2(x')\,) \stackrel{(*)}{=} (\,\pi_1\big(f(x,x')\big),\ \pi_2\big(f(x,x')\big)\,) = f(x,x'). \qquad \square$$

As announced, we will use the dcpo 3 as dualising object, in:

$$\begin{tikzcd} \mathbf{DcPA}^{\mathrm{op}} \arrow[d, bend left, "\mathrm{Hom}(-,3)"] \arrow[d, phantom, "\dashv"] \\ \mathbf{Dcpo} \arrow[u, bend left, "\mathrm{Hom}(-,3)"] \arrow[loop below, "\wp"] \end{tikzcd} \qquad \frac{Y \xrightarrow{\mathbf{DcPA}} \mathrm{Hom}(X,3)}{X \xrightarrow[\mathbf{Dcpo}]{} \mathrm{Hom}(Y,3)} \qquad \begin{tikzcd} \mathbf{DcPA}^{\mathrm{op}} \arrow[rr, bend left, "\top"] & & \mathcal{EM}(\wp) \arrow[ll, bend left] \\ & \mathcal{K}\ell(\wp) \arrow[ul, "\mathrm{Pred}"] \arrow[ur, "\mathrm{Stat}"'] & \end{tikzcd}$$

For directed complete Plotkin algebra $Y \in \mathbf{DcPA}$ the homset $\mathrm{Hom}(Y,3)$ of maps in $\mathbf{DcPA}$ is a dcpo, via the pointwise ordering. The above adjunction is then obtained via the usual swapping of arguments.

We call the induced monad the Plotkin power domain on $\mathbf{Dcpo}$. It can be described as:

$$\begin{aligned} \wp(X) &= \mathbf{DcPA}\big(\mathbf{Dcpo}(X,3),3\big) \\ &\cong \mathbf{DcPA}\big(\mathcal{O}(X) \ltimes \mathcal{O}(X), 2 \ltimes 2\big) \\ &\cong \{(f_1,f_2) \mid f_1 \in \mathbf{CL}_{\vee,1}(\mathcal{O}(X),2), f_2 \in \mathbf{PreFrm}_0(\mathcal{O}(X),2),\ \text{with } f_1 \geq f_2\}. \end{aligned}$$

The first isomorphism is based on Lemma 10 and Example 9. The second one comes from Lemma 11.

The map $f_1 \colon \mathcal{O}(X) \to 2$ in $\mathbf{CL}_{\vee,1}$ corresponds to a non-empty closed subset of X, see Lemma 6. The function $f_2 \colon \mathcal{O}(X) \to 2$ in $\mathbf{PreFrm}_0$ correponds to a proper Scott open filter, and in the sober case, to a non-empty compact saturated subset, as discussed already in Subsection 3.9.

In [15] 'valuations' of the form $\mathcal{O}(X) \to 2 \ltimes 2$, for a topological space X, form the elements of a monad. In contrast, here we arrive at maps of the form $\mathcal{O}(X) \ltimes \mathcal{O}(X) \to 2 \ltimes 2$.

5. Dualising with $[0,1]$

The next series of examples starts from adjunctions that are obtained by homming into the unit interval $[0,1]$. The quantitative logic that belongs to these examples is given in terms of effect modules. These can be seen as "probabilistic vector spaces", involving scalar multiplication with scalars from the unit interval $[0,1]$, instead of from $\mathbb{R}$ or $\mathbb{C}$. We provide a crash course for these structures, and refer to [27, 21, 7] or [10] for more information. A systematic description of the 'probability' monads below can be found in [25].

A partial commutative monoid (PCM) consists of a set M with a partial binary operation $\ovee$ and a zero element $0 \in M$. The operation $\ovee$ is commutative and associative, in an appropriate partial sense. One writes $x \perp y$ if $x \ovee y$ is defined.

An *effect algebra* is a PCM with an orthosupplement $(-)^\perp$, so that $x \ovee x^\perp = 1$, where $1 = 0^\perp$, and $x \perp 1$ implies $x = 0$. An effect algebra is automatically a poset, via the definition $x \leq y$ iff $x \ovee z = y$ for some z. The main example is the unit interval $[0,1]$, with $x \perp y$ iff $x + y \leq 1$, and in that case $x \ovee y = x + y$; the orthosupplement is $x^\perp = 1 - x$. A map of effect algebras $f\colon E \to D$ is a function that preserves 1 and $\ovee$, if defined. We write $\mathbf{EA}$ for the resulting category. Each Boolean algebra is an effect algebra, with $x \perp y$ iff $x \wedge y = 0$, and in that case $x \ovee y = x \vee y$. This yields a functor $\mathbf{BA} \to \mathbf{EA}$, which is full and faithful.

An *effect module* is an effect algebra E with an action $[0,1] \times E \to E$ that preserves $\ovee, 0$ in each argument separately. A map of effect modules f is a map of effect algebras that preserves scalar multiplication: $f(r \cdot x) = r \cdot f(x)$. We thus get a subcategory $\mathbf{EMod} \hookrightarrow \mathbf{EA}$. For each set X, the set $[0,1]^X$ of fuzzy predicates on X is an effect module, with $p \perp q$ iff $p(x)+q(x) \leq 1$ for all $x \in X$, and in that case $(p \ovee q)(x) = p(x) + q(x)$. Orthosupplement is given by $p^\perp(x) = 1 - p(x)$ and scalar multiplication by $r \cdot p \in [0,1]^X$, for $r \in [0,1]$ and $p \in [0,1]^X$, by $(r \cdot p)(x) = r \cdot p(x)$. This assignment $X \mapsto [0,1]^X$ yields a functor $\mathbf{Sets} \to \mathbf{EMod}^{\mathrm{op}}$ that will be used below. Important examples of effect modules arise in quantum logic. For instance, for each Hilbert space $\mathscr{H}$, the set $\mathcal{E}f(\mathscr{H}) = \{A\colon \mathscr{H} \to \mathscr{H} \mid 0 \leq A \leq \mathrm{id}\}$ of effects is an effect module. More generally, for a (unital) C^*-algebra A, the set of effects $[0,1]_A = \{a \in A \mid 0 \leq a \leq 1\}$ is an effect module. In [11] it is shown that taking effects yields a full and faithful functor:

$$\mathbf{Cstar}_{\mathrm{PU}} \xrightarrow{[0,1]_{(-)}} \mathbf{EMod} \tag{5.1}$$

Here we write $\mathbf{Cstar}_{\mathrm{PU}}$ for the category of C^*-algebras with positive unital maps.

An *MV-algebra* [8] can be understood as a 'commutative' effect algebra. It is an effect algebra with a join $\vee$, and thus also a meet $\wedge$, via De Morgan, in which the equation $(x \vee y)^\perp \ovee x = y^\perp \ovee (x \wedge y)$ holds. There is a subcategory $\mathbf{MVA} \hookrightarrow \mathbf{EA}$ with maps additionally preserving joins $\vee$ (and hence also $\wedge$). Within an MV-algebra one can define (total) addition and subtraction operations as $x + y = x \ovee (x^\perp \wedge y)$ and $x - y = (x^\perp + y)^\perp$. The unit interval $[0,1]$ is an MV-algebra, in which $+$ and $-$ are truncated (to 1 or 0), if needed.

There is a category $\mathbf{MVMod}$ of *MV-modules*, which are MV-algebras with $[0,1]$-scalar multiplication. Thus $\mathbf{MVMod}$ is twice a subcategory in: $\mathbf{MVA} \hookleftarrow \mathbf{MVMod} \hookrightarrow \mathbf{EMod}$. The effect module $[0,1]^X$ of fuzzy predicates is an MV-module. For a commutative C^*-algebra A the set of effects $[0,1]_A$ is an MV-module. In fact there is a full and faithful functor:

$$\mathbf{CCstar}_{\mathrm{MIU}} \xrightarrow{[0,1]_{(-)}} \mathbf{MVMod} \tag{5.2}$$

where $\mathbf{CCstar}_{\mathrm{MIU}}$ is the category of commutative C^*-algebras, with MIU-maps, preserving multiplication, involution and unit (aka. $*$-homomorphisms).

Having seen this background information we continue our series of examples.

5.1. **Sets and effect modules.** As noted above, fuzzy predicates yield a functor $\mathbf{Sets} \to \mathbf{EMod}^{\mathrm{op}}$. This functor involves homming into $[0,1]$, and has an adjoint that is used as starting point for several variations.

$$\begin{tikzcd}
\mathbf{EMod}^{\mathrm{op}} \arrow[d, bend right, "\mathrm{Hom}(-,[0,1])"'] \arrow[d, phantom, "\dashv"] \\
\mathbf{Sets} \arrow[u, bend right, "\mathrm{Hom}(-,[0,1])"'] \arrow[loop below, "\mathcal{E}=\mathbf{EMod}([0,1]^{(-)},[0,1])"]
\end{tikzcd}
\qquad
\dfrac{Y \xrightarrow{\mathbf{EMod}} [0,1]^X}{X \xrightarrow[\mathbf{Sets}]{} \mathbf{EMod}(Y,[0,1])}
\qquad
\begin{tikzcd}
\mathbf{EMod}^{\mathrm{op}} \arrow[rr, bend left, ""] \arrow[rr, phantom, "\top"] & & \mathcal{EM}(\mathcal{E}) = \mathbf{CCH}_{\mathrm{sep}} \arrow[ll, bend left] \\
& \mathcal{K}\ell(\mathcal{E}) \arrow[ul, "\mathrm{Pred}"] \arrow[ur, "\mathrm{Stat}"'] &
\end{tikzcd}$$

The induced monad $\mathcal{E}$ is the *expectation* monad introduced in [26]. It can be understood as an extension of the (finite probability) distribution monad $\mathcal{D}$, since $\mathcal{E}(X) \cong \mathcal{D}(X)$ if X is a finite set. The triangle corollary on the right says in particular that Kleisli maps $X \to \mathcal{E}(Y)$ are in bijective correspondence with effect module maps $[0,1]^Y \to [0,1]^X$ acting as predicate transformers, on fuzzy predicates.

The category of algebras $\mathcal{EM}(\mathcal{E})$ of the expectation monad is the category $\mathbf{CCH}_{\mathrm{sep}}$ of convex compact Hausdorff spaces, with a separation condition (see [26, 28] for details). State spaces in quantum computing are typically such convex compact Hausdorff spaces.

Using the full and faithfulness of the functor $[0,1]_{(-)} \colon \mathbf{Cstar}_{\mathrm{PU}} \to \mathbf{EMod}$ from (5.1), the expectation monad can alternatively be described in terms of the states of the commutative C^*-algebra $\ell^\infty(X)$ of bounded functions $X \to \mathbb{C}$, via:

$$\begin{array}{rcl}
\mathrm{Stat}(\ell^\infty(X)) \stackrel{\mathrm{def}}{=} \mathbf{Cstar}_{\mathrm{PU}}\big(\ell^\infty(X), \mathbb{C}\big) & \stackrel{(5.1)}{\cong} & \mathbf{EMod}\big([0,1]_{\ell^\infty(X)}, [0,1]_{\mathbb{C}}\big) \\
& = & \mathbf{EMod}\big([0,1]^X, [0,1]\big) \;=\; \mathcal{E}(X).
\end{array} \tag{5.3}$$

In this way one obtains the result from [11] that there is a full & faithful functor:

$$\mathcal{K}\ell(\mathcal{E}) \xrightarrow{\qquad\qquad} \big(\mathbf{CCstar}_{\mathrm{PU}}\big)^{\mathrm{op}} \tag{5.4}$$

embedding the Kleisli category $\mathcal{K}\ell(\mathcal{E})$ of the expectation monad into commutative C^*-algebras with positive unital maps. On objects this functor (5.4) is given by $X \mapsto \ell^\infty(X)$.

5.2. **Compact Hausdorff spaces and effect modules.** In the previous example we have used the *set* $\mathbf{EMod}(E,[0,1])$ of effect module maps $E \to [0,1]$, for an effect module E. It turns out that this homset has much more structure: it is a compact Hausdorff space. The reason is that the unit interval $[0,1]$ is compact Hausdorff, and so the function space $[0,1]^E$ too, by Tychonoff. The homset $\mathbf{EMod}(E,[0,1]) \hookrightarrow [0,1]^E$ can be described via a closed subset of maps satisfying the effect module map requirements. Hence $\mathbf{EMod}(E,[0,1])$ is compact Hausdorff itself. We thus obtain the following situation.

$$\begin{tikzcd}
\mathbf{EMod}^{\mathrm{op}} \arrow[d, bend right, "\mathrm{Hom}(-,[0,1])"'] \arrow[d, phantom, "\dashv"] \\
\mathbf{CH} \arrow[u, bend right, "\mathrm{Hom}(-,[0,1])"'] \arrow[loop below, "\mathcal{R}=\mathbf{EMod}(\mathrm{C}(-,[0,1]),[0,1])"]
\end{tikzcd}
\qquad
\dfrac{Y \xrightarrow{\mathbf{EMod}} \mathrm{C}(X,[0,1])}{X \xrightarrow[\mathbf{CH}]{} \mathbf{EMod}(Y,[0,1])}
\qquad
\begin{tikzcd}
\mathbf{EMod}^{\mathrm{op}} \arrow[rr, bend left, ""] \arrow[rr, phantom, "\top"] & & \mathcal{EM}(\mathcal{R}) = \mathbf{CCH}_{\mathrm{sep}} \arrow[ll, bend left] \\
& \mathcal{K}\ell(\mathcal{R}) \arrow[ul, "\mathrm{Pred}"] \arrow[ur, "\mathrm{Stat}"'] &
\end{tikzcd}$$

For a compact Hausdorff space X, the subset $\mathrm{C}(X,[0,1]) \hookrightarrow [0,1]^X$ of continuous maps $X \to [0,1]$ is a (sub) effect module. The induced monad $\mathcal{R}(X) = \mathbf{EMod}\big(\mathrm{C}(X,[0,1]),[0,1]\big)$ is the *Radon* monad. Using the full & faithful functor (5.1) the monad can equivalently be described as $X \mapsto \mathrm{Stat}(\mathrm{C}(X))$, where $\mathrm{C}(X)$ is the commutative C^*-algebra of functions $X \to \mathbb{C}$. The monad occurs in [40] as part of a topological and domain-theoretic approach to information theory. The main result of [11] is the equivalence of categories

$$\mathcal{K}\ell(\mathcal{R}) \simeq \big(\mathbf{CCstar}_{\mathrm{PU}}\big)^{\mathrm{op}}$$

between the Kleisli category of this Radon monad $\mathcal{R}$ and the category of commutative C^*-algebras and positive unital maps. This shows how (commutative) C^*-algebras appear in state-and-effect triangles (see also [21, 7]).

The algebras of the Radon monad are convex compact Hausdorff spaces (with separation), like for the expectation monad $\mathcal{E}$, see [26] for details.

5.3. **Compact Hausdorff spaces and MV-modules.** The adjunction $\mathbf{EMod}^{\mathrm{op}} \rightleftarrows \mathbf{CH}$ can be restricted to an adjunction $\mathbf{MVMod}^{\mathrm{op}} \rightleftarrows \mathbf{CH}$, involving MV-modules instead of effect modules. This can be done since continuous functions $X \to [0,1]$ are appropriately closed under joins $\vee$, and thus form an MV-module. Additionally, for an MV-module E, the MV-module maps $E \to [0,1]$ form a compact Hausdorff space (using the same argument as in the previous subsection).

Via this restriction to an adjunction $\mathbf{MVMod}^{\mathrm{op}} \rightleftarrows \mathbf{CH}$ we hit a wall again.

Lemma 12. For a compact Hausdorff space X, the unit $\eta\colon X \to \mathbf{MVMod}\big(\mathrm{C}(X,[0,1]),[0,1]\big)$, given by $\eta(x)(p) = p(x)$, is an isomorphism in $\mathbf{CH}$.

This result can be understood as part of the Yosida duality for Riesz spaces. It is well-known in the MV-algebra community, but possibly not precisely in this form. For convenience, we include a proof.

Proof. We only show that the unit η is an isomorphism, not that it is also a homeomorphism. Injectivity is immediate by Urysohn. For surjectivity, we first establish the following two auxiliary results.

(1) For each $p \in \mathrm{C}(X,[0,1])$ and $\omega \in \mathbf{MVMod}\big(\mathrm{C}(X,[0,1]),[0,1]\big)$, if $\omega(p) = 0$, then there is an $x \in X$ with $p(x) = 0$.

If not, then $p(x) > 0$ for all $x \in X$. Hence there is an inclusion $X \subseteq \bigcup_{r>0} p^{-1}\big((r,1]\big)$. By compactness there are finitely many r_i with $X \subseteq \bigcup_i p^{-1}\big((r_i,1]\big)$. Thus for $r = \bigwedge_i r_i > 0$ we have $p(x) > r$ for all $x \in X$. Find an $n \in \mathbb{N}$ with $n \cdot r \geq 1$. The n-fold sum $n \cdot p$ in the MV-module $\mathrm{C}(X,[0,1])$ then satisfies $p(x) = 1$ for all x, so that $n \cdot p = 1$ in $\mathrm{C}(X,[0,1])$. But now we get a contradiction: $1 = \omega(1) = \omega(n \cdot p) = n \cdot \omega(p) = 0$.

(2) For each finite collection of maps $p_1, \ldots, p_n \in \mathrm{C}(X,[0,1])$ and for each function $\omega \in \mathbf{MVMod}\big(\mathrm{C}(X,[0,1]),[0,1]\big)$ there is an $x \in X$ with $\omega(p_i) = p_i(x)$ for all $1 \leq i \leq n$.

For the proof, define $p \in \mathrm{C}(X,[0,1])$ using the MV-structure of $\mathrm{C}(X,[0,1])$ as:

$$p = \bigvee\nolimits_i \big(p_i - \omega(p_i) \cdot 1\big) \vee \big(\omega(p_i) \cdot 1 - p_i\big).$$

Since the state $\omega\colon \mathrm{C}(X,[0,1]) \to [0,1]$ preserves the MV-structure we get in $[0,1]$:

$$\omega(p) = \bigvee\nolimits_i \big(\omega(p_i) - \omega(p_i) \cdot 1\big) \vee \big(\omega(p_i) \cdot 1 - \omega(p_i)\big) = 0.$$

Hence by the previous point there is an $x \in X$ with $p(x) = 0$. But then $p_i(x) = \omega(p_i)$, as required.

Now we can prove surjectivity of the unit map $\eta\colon X \to \mathbf{MVMod}\big(\mathrm{C}(X,[0,1]),[0,1]\big)$. Let $\omega\colon \mathrm{C}(X,[0,1]) \to [0,1]$ be an MV-module map. Define for each $p \in \mathrm{C}(X,[0,1])$ the subset $U_p = \{x \in X \mid \omega(p) \neq p(x)\}$. This subset $U_p \subseteq X$ is open since it can be written as $f^{-1}(\mathbb{R} - \{0\})$, for the continuous function $f(x) = p(x) - \omega(p)$.

Suppose towards a contradiction that $\omega \neq \eta(x)$ for all $x \in X$. Thus, for each $x \in X$ there is a $p \in \mathrm{C}(X,[0,1])$ with $\omega(p) \neq \eta(x)(p) = p(x)$. This means $X \subseteq \bigcup_p U_p$. By compactness of X there are finitely many $p_i \in \mathrm{C}(X,[0,1])$ with $X \subseteq \bigcup_i U_{p_i}$. The above second point however gives an $x \in X$ with $\omega(p_i) = p_i(x)$ for all i. But then $x \notin \bigcup_i U_{p_i}$. $\square$

5.4. **Sets and directed complete effect modules.** In the remainder of this paper we shall consider effect modules with additional completeness properties (w.r.t. its standard order), as in [29]. Specifically, we consider ω-complete, and directed-complete effect modules. In the first case each ascending ω-chain $x_0 \leq x_1 \leq \cdots$ has a least upperbound $\bigvee_n x_n$; and in the second case each directed subset D has a join $\bigvee D$. We write the resulting subcategories as:

$$\mathbf{DcEMod} \hookrightarrow \omega\text{-}\mathbf{EMod} \hookrightarrow \mathbf{EMod}$$

where maps are required to preserve the relevant joins $\bigvee$.

We start with the directed-complete case. The adjunction $\mathbf{EMod}^{\mathrm{op}} \rightleftarrows \mathbf{Sets}$ from Subsection 5.1 can be restricted to an adjunction as on the left below.

$$\begin{tikzcd}
\mathbf{DcEMod}^{\mathrm{op}} \arrow[d, bend right, "\mathrm{Hom}(-,[0,1])"'] \arrow[d, phantom, "\dashv"] \\
\mathbf{Sets} \arrow[u, bend right, "\mathrm{Hom}(-,[0,1])"'] \arrow[loop below, "\mathcal{E}_\infty = \mathbf{DcEMod}([0,1]^{(-)},[0,1])"]
\end{tikzcd}
\qquad
\frac{Y \xrightarrow{\mathbf{DcEMod}} [0,1]^X}{X \xrightarrow[\mathbf{Sets}]{} \mathbf{DcEMod}(Y,[0,1])}
\qquad
\begin{tikzcd}
\mathbf{DcEMod}^{\mathrm{op}} \arrow[rr, bend left] \arrow[rr, phantom, "\top"] & & \mathcal{EM}(\mathcal{E}_\infty) = \mathbf{Conv}_\infty \arrow[ll, bend left] \\
& \mathcal{K}\ell(\mathcal{E}_\infty) \arrow[ul, "\mathrm{Pred}"] \arrow[ur, "\mathrm{Stat}"'] &
\end{tikzcd}$$

The resulting monad $\mathcal{E}_\infty = \mathbf{DcEMod}\big([0,1]^{(-)},[0,1]\big)$ on $\mathbf{Sets}$ is in fact isomorphic[1] to the infinite (discrete probability) distribution monad $\mathcal{D}_\infty$, see [24]. We recall, for a set X,

$$\mathcal{D}_\infty(X) = \{\omega\colon X \to [0,1] \mid \mathrm{supp}(\omega) \text{ is countable, and } \textstyle\sum_x \omega(x) = 1\}.$$

The subset $\mathrm{supp}(\omega) \subseteq X$ contains the elements $x \in X$ with $\omega(x) \neq 0$. The requirement in the definition of $\mathcal{D}_\infty(X)$ that $\mathrm{supp}(\omega)$ be countable is superfluous, since it follows from the requirement $\sum_x \omega(x) = 1$. Briefly, $\mathrm{supp}(\omega) \subseteq \bigcup_{n>0} X_n$, where $X_n = \{x \in X \mid \omega(x) > \frac{1}{n}\}$ contains at most $n-1$ elements (see *e.g.* [44, Prop. 2.1.2]).

Proposition 13. There is an isomorphism of monads $\mathcal{D}_\infty \cong \mathcal{E}_\infty$, where $\mathcal{E}_\infty$ is the monad induced by the above adjunction $\mathbf{DcEMod}^{\mathrm{op}} \rightleftarrows \mathbf{Sets}$.

Proof. For a subset $U \subseteq X$ we write $\mathbf{1}_U\colon X \to [0,1]$ for the 'indicator' function, defined by $\mathbf{1}_U(x) = 1$ if $x \in U$ and $\mathbf{1}_U(x) = 0$ if $x \notin U$. We write $\mathbf{1}_x$ for $\mathbf{1}_{\{x\}}$. This function $\mathbf{1}_{(-)}\colon \mathcal{P}(X) \to [0,1]^X$ is a map of effect algebras that preserves all joins.

---

[1] This isomorphism $\mathcal{E}_\infty \cong \mathcal{D}_\infty$ in Proposition 13 is inspired by work of Robert Furber (PhD Thesis, forthcoming): he noticed the isomorphism $\mathrm{NStat}(\ell^\infty(X)) \cong \mathcal{D}_\infty(X)$ in (5.7), which is obtained here as a corollary to Proposition 13.

Let $h \in \mathcal{E}_\infty(X)$, so h is a Scott continuous map of effect modules $h\colon [0,1]^X \to [0,1]$. Define $\overline{h}\colon X \to [0,1]$ as $\overline{h}(x) = h(\mathbf{1}_x)$. Notice that if $U \subseteq X$ is a finite subset, then:

$$1 = h(1) = h(\mathbf{1}_X) \geq h(\mathbf{1}_U) = h(\bigovee_{x\in U} \mathbf{1}_x) = \bigovee_{x\in U} h(\mathbf{1}_x) = \bigovee_{x\in U} \overline{h}(x).$$

We can write X as directed union of its finite subsets, and thus also $\mathbf{1}_X = \bigvee\{\mathbf{1}_U \mid U \subseteq X \text{ finite}\}$. But then $\overline{h} \in \mathcal{D}_\infty(X)$, because h preserves directed joins:

$$1 = h(\mathbf{1}_X) = \bigvee\{h(\mathbf{1}_U) \mid U \subseteq X \text{ finite}\} = \bigvee\{\textstyle\sum_{x\in U} \overline{h}(x) \mid U \subseteq X \text{ finite}\} = \textstyle\sum_{x\in X} \overline{h}(x).$$

Conversely, given $\omega \in \mathcal{D}_\infty(X)$ we define $\overline{\omega}\colon [0,1]^X \to [0,1]$ as $\overline{\omega}(p) = \sum_{x\in X} p(x)\cdot\omega(x)$. It is easy to see that $\overline{\omega}$ is a map of effect modules. It is a bit more challenging to see that it preserves directed joins $\bigvee_i p_i$, for $p_i \in [0,1]^X$.

First we write the countable support of ω as $\mathrm{supp}(\omega) = \{x_0, x_1, x_2, \ldots\} \subseteq X$ in such a way that $\omega(x_0) \geq \omega(x_1) \geq \omega(x_2) \geq \cdots$. We have $1 = \sum_{x\in X}\omega(x) = \sum_{n\in\mathbb{N}}\omega(x_n)$. Hence, for each $N \in \mathbb{N}$ we get:

$$\textstyle\sum_{n>N}\omega(x_n) = 1 - \sum_{n\leq N}\omega(x_n).$$

By taking the limit $N \to \infty$ on both sides we get:

$$\lim_{N\to\infty} \textstyle\sum_{n>N}\omega(x_n) = 1 - \displaystyle\lim_{N\to\infty} \textstyle\sum_{n\leq N}\omega(x_n) = 1 - \sum_{n\in\mathbb{N}}\omega(x_n) = 1 - 1 = 0.$$

We have to prove $\overline{\omega}(\bigvee_i p_i) = \bigvee_i \overline{\omega}(p_i)$. The non-trivial part is $(\leq)$. For each $N \in \mathbb{N}$ we have:

$$\begin{aligned}
\overline{\omega}(\textstyle\bigvee_i p_i) &= \textstyle\sum_{n\in\mathbb{N}}(\bigvee_i p_i)(x_n)\cdot\omega(x_n) \\
&= \textstyle\sum_{n\in\mathbb{N}}(\bigvee_i p_i(x_n))\cdot\omega(x_n) \\
&= \textstyle\sum_{n\in\mathbb{N}}\bigvee_i p_i(x_n)\cdot\omega(x_n) \\
&= \Big(\textstyle\sum_{n\leq N}\bigvee_i p_i(x_n)\cdot\omega(x_n)\Big) + \Big(\textstyle\sum_{n>N}\bigvee_i p_i(x_n)\cdot\omega(x_n)\Big) \\
&= \Big(\textstyle\bigvee_i \sum_{n\leq N} p_i(x_n)\cdot\omega(x_n)\Big) + \Big(\textstyle\sum_{n>N}\bigvee_i p_i(x_n)\cdot\omega(x_n)\Big) \\
&\leq \Big(\textstyle\bigvee_i \sum_{n\leq N} p_i(x_n)\cdot\omega(x_n)\Big) + \Big(\textstyle\sum_{n>N}\omega(x_n)\Big) \qquad \text{since } p_i(x) \in [0,1].
\end{aligned}$$

Hence we are done by taking the limit $N \to \infty$. Notice that we use that the join $\bigvee$ can be moved outside a finite sum. This works precisely because the join is taken over a directed set.

What remains is to show that these mappings $h \mapsto \overline{h}$ and $\omega \mapsto \overline{\omega}$ yield an isomorphism $\mathcal{D}_\infty(X) \cong \mathcal{E}_\infty(X)$, which is natural in X, and forms an isomorphism of monads. This is left to the interested reader. □

As a result, the Eilenberg-Moore category $\mathcal{EM}(\mathcal{E}_\infty)$ is isomorphic to $\mathcal{EM}(\mathcal{D}_\infty) = \mathbf{Conv}_\infty$, where $\mathbf{Conv}_\infty$ is the category of countably-convex sets X, in which convex sums $\sum_{n\in\mathbb{N}} r_n x_n$ exist, where $x_n \in X$ and $r_n \in [0,1]$ with $\sum_n r_n = 1$.

We briefly look at the relation with C^*-algebras (actually W^*-algebras), like in Subsection 5.1. We write $\mathbf{Wstar}_{\mathrm{NPU}}$ for the category of W^*-algebras with normal positive unital maps. The term 'normal' is used in the operator algebra community for what is called 'Scott continuity' (preservation of directed joins) in the domain theory community. This means that taking effects yields a full and faithful functor:

$$\mathbf{Wstar}_{\mathrm{NPU}} \xrightarrow{[0,1]_{(-)}} \mathbf{DcEMod} \tag{5.5}$$

This is similar to the situation in (5.1) and (5.2). One could also use AW^*-algebras here. Next, there is now a full and faithful functor to the category of commutative W^*-algebras:

$$\mathcal{K}\ell(\mathcal{D}_\infty) \cong \mathcal{K}\ell(\mathcal{E}_\infty) \longrightarrow \mathbf{CWstar}_{\mathrm{NPU}} \tag{5.6}$$

On objects it is given by $X \mapsto \ell^\infty(X)$. This functor is full and faithful since there is a bijective correspondence:

$$\begin{array}{ll} \ell^\infty(X) \longrightarrow \ell^\infty(Y) & \text{in } \mathbf{CWstar}_{\mathrm{NPU}} \\ \hline\hline Y \longrightarrow \mathrm{NStat}(\ell^\infty(X)) \cong \mathcal{E}_\infty(X) \cong \mathcal{D}_\infty(X) & \text{in } \mathbf{Sets} \end{array}$$

where the isomorphism $\cong$ describing normal states is given, like in (5.3), by:

$$\begin{aligned} \mathrm{NStat}(\ell^\infty(X)) \stackrel{\text{def}}{=} \mathbf{Wstar}_{\mathrm{NPU}}\big(\ell^\infty(X), \mathbb{C}\big) &\stackrel{(5.5)}{\cong} \mathbf{DcEMod}\big([0,1]_{\ell^\infty(X)}, [0,1]_{\mathbb{C}}\big) \\ &= \mathbf{DcEMod}\big([0,1]^X, [0,1]\big) \\ &= \mathcal{E}_\infty(X) \\ &\cong \mathcal{D}_\infty(X). \end{aligned} \tag{5.7}$$

5.5. **Measurable spaces and ω-complete effect modules.** In our final example we use an adjunction between effect modules and measurable spaces (instead of sets or compact Hausdorff spaces). We write **Meas** for the category of measurable spaces (X, Σ_X), where $\Sigma_X \subseteq \mathcal{P}(X)$ is the σ-algebra of measurable subsets, with measurable functions between them (whose inverse image maps measurable subsets to measurable subsets). We use the unit interval $[0,1]$ with its standard Borel σ-algebra (the least one that contains all the usual opens). A basic fact in this situation is that for a measurable space X, the set $\mathbf{Meas}(X, [0,1])$ of measurable functions $X \to [0,1]$ is an ω-effect module. The effect module structure is inherited via the inclusion $\mathbf{Meas}(X,[0,1]) \hookrightarrow [0,1]^X$. Joins of ascending ω-chains $p_0 \leq p_1 \leq \cdots$ exists, because the (pointwise) join $\bigvee_n p_n$ is a measurable function again. In this way we obtain a functor $\mathbf{Meas}(-,[0,1])\colon \mathbf{Meas} \to \omega\text{-}\mathbf{EMod}^{\mathrm{op}}$.

In the other direction there is also a hom-functor $\omega\text{-}\mathbf{EMod}(-,[0,1])\colon \omega\text{-}\mathbf{EMod}^{\mathrm{op}} \to \mathbf{Meas}$. For an ω-effect module E we can provide the set of maps $\omega\text{-}\mathbf{EMod}(E,[0,1])$ with a σ-algebra, namely the least one that makes all the evaluation maps $\mathrm{ev}_x\colon \omega\text{-}\mathbf{EMod}(E,[0,1]) \to [0,1]$ measurable, for $x \in E$. This function ev_x is given by $\mathrm{ev}_x(p) = p(x)$. This gives the following situation.

$$\begin{tikzcd} \omega\text{-}\mathbf{EMod}^{\mathrm{op}} \arrow[d, bend right, "\mathrm{Hom}(-,[0,1])"'] \arrow[d, phantom, "\dashv"] \\ \mathbf{Meas} \arrow[u, bend right, "\mathrm{Hom}(-,[0,1])"'] \arrow[loop below, "\mathcal{G}=\omega\text{-}\mathbf{EMod}(\mathbf{Meas}(-,[0,1]),[0,1])"] \end{tikzcd} \qquad \begin{array}{c} Y \xrightarrow{\omega\text{-}\mathbf{EMod}} \mathbf{Meas}(X,[0,1]) \\ \hline\hline X \xrightarrow[\mathbf{Meas}]{} \omega\text{-}\mathbf{EMod}(Y,[0,1]) \end{array} \qquad \begin{tikzcd} \omega\text{-}\mathbf{EMod}^{\mathrm{op}} \arrow[rr, bend left, shift left] \arrow[rr, phantom, "\top"] & & \mathcal{EM}(\mathcal{G}) \arrow[ll, bend left, shift left] \\ & \mathcal{K}\ell(\mathcal{G}) \arrow[ul, "\mathrm{Pred}"] \arrow[ur, "\mathrm{Stat}"'] & \end{tikzcd}$$

We use the symbol $\mathcal{G}$ for the induced monad because of the following result.

Proposition 14. The monad $\mathcal{G} = \omega\text{-}\mathbf{EMod}\big(\mathbf{Meas}(-,[0,1]),[0,1]\big)$ on **Meas** in the above situation is (isomorphic to) the Giry monad [12], given by probability measures:

$$\mathrm{Giry}(X) \stackrel{\text{def}}{=} \{\phi\colon \Sigma_X \to [0,1] \mid \phi \text{ is a probability measure}\} = \omega\text{-}\mathbf{EA}(\Sigma_X,[0,1]).$$

Proof. The isomorphism involves Lebesgue integration:

$$\begin{tikzcd}
\mathcal{G}(X) = \omega\text{-}\mathbf{EMod}\big(\mathbf{Meas}(X,[0,1]),[0,1]\big) \arrow[r, bend left, "I\mapsto(M\mapsto I(\mathbf{1}_M))"] \arrow[r, phantom, "\cong"] & \omega\text{-}\mathbf{EA}(\Sigma_X,[0,1]) = \mathrm{Giry}(X) \arrow[l, bend left, "\phi\mapsto(p\mapsto\int p\,\mathrm{d}\phi)"]
\end{tikzcd}$$

See [20] or [29] for more details. □

The above triangle is further investigated in [20]. It resembles the situation described in [6] for Markov kernels (the ordinary, not the abstract, ones).

5.6. **Dcpo's and directed complete effect modules.** In our final example we briefly consider another variation of the adjunction $\mathbf{DcEMod}^{\mathrm{op}} \leftrightarrows \mathbf{Sets}$ in Subsection 5.4, now with an adjunction $\mathbf{DcEMod}^{\mathrm{op}} \leftrightarrows \mathbf{Dcpo}$ between the categories of directed complete effect modules and partial orders. This brings us into the realm of probabilistic power domains, which has its own thread of research, see *e.g.* [14, 31, 32, 33, 36, 42, 46]. Our only aim at this stage is to show how the current approach connects to that line of work. The most significant difference is that we use the unit interval $[0,1]$, whereas it is custom for probabilistic power domains to use the extended non-negative real numbers $\{r \in \mathbb{R} \mid r \geq 0\} \cup \{\infty\}$. Consequently, we use effect modules instead of cones.

Using that the unit interval $[0,1]$, with its usual order, is a dcpo, and that its multiplication, and also its partial addition, is Scott continuous in each variable, we obtain:

$$\begin{tikzcd}
\mathbf{DcEMod}^{\mathrm{op}} \arrow[d, bend left, "\mathrm{Hom}(-,[0,1])"] \arrow[d, phantom, "\dashv"] \\
\mathbf{Dcpo} \arrow[u, bend left, "\mathrm{Hom}(-,[0,1])"] \arrow[loop below, "\mathcal{V}=\mathbf{DcEMod}([0,1]^{(-)},[0,1])"]
\end{tikzcd}
\qquad
\begin{array}{c}
Y \xrightarrow{\mathbf{DcEMod}} \mathbf{Dcpo}(X,[0,1]) \\
\hline\hline
X \xrightarrow[\mathbf{Dcpo}]{} \mathbf{DcEMod}(Y,[0,1])
\end{array}
\qquad
\begin{tikzcd}
\mathbf{DcEMod}^{\mathrm{op}} \arrow[rr, bend left] \arrow[rr, phantom, "\top"] & & \mathcal{EM}(\mathcal{V}) \arrow[ll, bend left] \\
& \mathcal{K}\ell(\mathcal{V}) \arrow[ul, "\mathrm{Pred}"] \arrow[ur, "\mathrm{Stat}"'] &
\end{tikzcd}$$

The induced monad $\mathcal{V}$ is a restricted version of the monad of valuations, that uses the extended real numbers, as mentioned above. It is unclear what its category of Eilenberg-Moore algebras is.

Acknowledgements. Several people have contributed to the ideas and examples presented here, including, in alphabetical order: Kenta Cho, Robert Furber, Helle Hansen, Klaus Keimel, Bas and Bram Westerbaan. Thanks to all of them!

References

[1] S. Abramsky. Domain theory in logical form. *Ann. Pure & Appl. Logic*, 51(1/2):1–77, 1991.
[2] S. Abramsky and A. Jung. Domain theory. In S. Abramsky, Dov M. Gabbai, and T. Maibaum, editors, *Handbook of Logic in Computer Science*, volume 3, pages 1–168. Oxford Univ. Press, 1994.
[3] J. Adámek, H. Herrlich, and G.E. Stecker. *Abstract and Concrete Categories. The Joy of Cats.* John Wiley and Sons, New York, 1990. Republished in: *Reprints in Theory and Appl. of Categories* 17, see `www.tac.mta.ca/tac/reprints/articles/17/tr17.pdf`.
[4] M. Barr and Ch. Wells. *Toposes, Triples and Theories.* Springer, Berlin, 1985. Revised and corrected version available from URL: `www.cwru.edu/artsci/math/wells/pub/ttt.html`.
[5] I. Battenfeld, K. Keimel, and T. Streicher. Observationally-induced algebras in domain theory. *Logical Methods in Comp. Sci.*, 10(3):1–26, 2014.
[6] P. Chaput, V. Danos, P. Panangaden, and G. Plotkin. Approximating Markov processes by averaging. *Journ. ACM*, 61(1), 2014.

[7] K. Cho, B. Jacobs, A. Westerbaan, and B. Westerbaan. An introduction to effectus theory. see `arxiv.org/abs/1512.05813`, 2015.
[8] R. Cignoli, I D'Ottaviano, and D. Mundici. *Algebraic Foundations of Many-Valued Reasoning*, volume 7 of *Trends in Logic*. Springer, 2000.
[9] E. Dijkstra and C. Scholten. *Predicate Calculus and Program Semantics*. Springer, Berlin, 1990.
[10] A. Dvurečenskij and S. Pulmannová. *New Trends in Quantum Structures*. Kluwer Acad. Publ., Dordrecht, 2000.
[11] R. Furber and B. Jacobs. From Kleisli categories to commutative C^*-algebras: Probabilistic Gelfand duality. In R. Heckel and S. Milius, editors, *Conference on Algebra and Coalgebra in Computer Science (CALCO 2013)*, number 8089 in Lect. Notes Comp. Sci., pages 141–157. Springer, Berlin, 2013.
[12] M. Giry. A categorical approach to probability theory. In B. Banaschewski, editor, *Categorical Aspects of Topology and Analysis*, number 915 in Lect. Notes Math., pages 68–85. Springer, Berlin, 1982.
[13] H. H. Hansen, C. Kupke, and R. Leal. Strong completeness for iteration-free coalgebraic dynamic logics. In J. Diaz, I. Lanese, and D. Sangiorgi, editors, *Theoretical Computer Science*, number 8705 in Lect. Notes Comp. Sci., pages 281–295. Springer, Berlin, 2014.
[14] R. Heckmann. Probabilistic power domains, information systems, and locales. In S. Brookes, M. Main, A. Melton, M. Mislove, and D. Schmidt, editors, *Math. Found. of Programming Semantics*, number 802 in Lect. Notes Comp. Sci., pages 410–437. Springer, Berlin, 1994.
[15] R. Heckmann. Abstract valuations: A novel representation of Plotkin power domain and Vietoris hyperspace. In *Math. Found. of Programming Semantics*, number 6 in Elect. Notes in Theor. Comp. Sci., pages 160–173. Elsevier, Amsterdam, 1997.
[16] T. Heinosaari and M. Ziman. *The Mathematical Language of Quantum Theory. From Uncertainty to Entanglement*. Cambridge Univ. Press, 2012.
[17] W. Hino, H. Kobayashi I. Hasuo, and B. Jacobs. Healthiness from duality. Logic in Computer Science, 2016.
[18] K. Hofmann and M. Mislove. Local compactness and continuous lattices. In B. Banaschewski and R.-E. Hoffman, editors, *Continuous Lattices*, number 871 in Lect. Notes Math., pages 209–248. Springer, Berlin, 1981.
[19] B. Jacobs. Semantics of weakening and contraction. *Ann. Pure & Appl. Logic*, 69(1):73–106, 1994.
[20] B. Jacobs. Measurable spaces and their effect logic. In *Logic in Computer Science*. IEEE, Computer Science Press, 2013.
[21] B. Jacobs. Dijkstra and Hoare monads in monadic computation. *Theor. Comp. Sci.*, 604:30–45, 2015.
[22] B. Jacobs. New directions in categorical logic, for classical, probabilistic and quantum logic. *Logical Methods in Comp. Sci.*, 11(3):1–76, 2015.
[23] B. Jacobs. A recipe for state and effect triangles. In L. Moss and P. Sobocinski, editors, *Conference on Algebra and Coalgebra in Computer Science (CALCO 2015)*, volume 35 of *LIPIcs*, pages 116–129. Schloss Dagstuhl, 2015.
[24] B. Jacobs. *Introduction to Coalgebra. Towards Mathematics of States and Observations*. Number 59 in Tracts in Theor. Comp. Sci. Cambridge Univ. Press, 2016.
[25] B. Jacobs. From probability monads to commutative effectuses. *Journ. of Logical and Algebraic Methods in Programming*, 156, 2017, to appear.
[26] B. Jacobs and J. Mandemaker. The expectation monad in quantum foundations. In B. Jacobs, P. Selinger, and B. Spitters, editors, *Quantum Physics and Logic (QPL) 2011*, number 95 in Elect. Proc. in Theor. Comp. Sci., pages 143–182, 2012.
[27] B. Jacobs and J. Mandemaker. Relating operator spaces via adjunctions. In J. Chubb, A. Eskandarian, and V. Harizanov, editors, *Logic and Algebraic Structures in Quantum Computing*, volume 45 of *Lect. Notes in Logic*, pages 123–150. Cambridge Univ. Press, 2016.
[28] B. Jacobs, J. Mandemaker, and R. Furber. The expectation monad in quantum foundations. Inf. & Comp., 2016.
[29] B. Jacobs and A. Westerbaan. An effect-theoretic account of Lebesgue integration. In D. Ghica, editor, *Math. Found. of Programming Semantics*, number 319 in Elect. Notes in Theor. Comp. Sci., pages 239–253. Elsevier, Amsterdam, 2015.
[30] P. Johnstone. *Stone Spaces*. Number 3 in Cambridge Studies in Advanced Mathematics. Cambridge Univ. Press, 1982.

[31] C. Jones and G. Plotkin. A probabilistic powerdomain of evaluations. In *Logic in Computer Science*, pages 186–195. IEEE, Computer Science Press, 1989.

[32] K. Keimel. The monad of probability measures over compact ordered spaces and its Eilenberg-Moore algebras. *Topology and its Applications*, 156:227–239, 2008.

[33] K. Keimel. Abstract ordered compact convex sets and algebras of the (sub)probabilistic power domain monad over ordered compact spaces. *Algebra an Logic*, 48(5):330–343, 2009.

[34] K. Keimel. On the equivalence of state transformer semantics and predicate transformer semantics. In *Proc. Informatics and Information Technologies in Education: Theory, Practice, Didactics*, pages 78–104, Novosibirsk, 2012. See `arxiv.org/abs/1410.7930`.

[35] K. Keimel and J. Paseka. A direct proof of the Hofmann-Mislove theorem. *Proc. Amer. Math. Soc.*, 120(1):301–303, 1994.

[36] K. Keimel and G. Plotkin. Predicate transformers for extended probability and non-determinism. *Math. Struct. in Comp. Sci.*, 19(3):501–539, 2009.

[37] J. Lambek and P. Scott. *Introduction to higher order Categorical Logic*. Number 7 in Cambridge Studies in Advanced Mathematics. Cambridge Univ. Press, 1986.

[38] E. Manes. A triple-theoretic construction of compact algebras. In B. Eckman, editor, *Seminar on Triples and Categorical Homology Theory*, number 80 in Lect. Notes Math., pages 91–118. Springer, Berlin, 1969.

[39] G. Markowsky. Free completely distributive complete lattices. *Proc. Amer. Math. Soc.*, 74(2):227–228, 1979.

[40] M. Mislove. Probabilistic monads, domains and classical information. In E. Kashefi, J. Krivine, and F. van Raamsdonk, editors, *Developments of Computational Methods (DCM 2011)*, number 88 in Elect. Proc. in Theor. Comp. Sci., pages 87–100, 2012.

[41] E. Moggi. Notions of computation and monads. *Inf. & Comp.*, 93(1):55–92, 1991.

[42] N. Saheb-Djahromi. CPO's of measures for nondeterminism. *Theor. Comp. Sci.*, 12:19–37, 1980.

[43] M. Smyth. Power domains and predicate transformers: a topological view. In J. Díaz, editor, *Int. Coll. on Automata, Languages and Programming*, number 154 in Lect. Notes Comp. Sci., pages 662–675. Springer, Berlin, 1983.

[44] A. Sokolova. *Coalgebraic Analysis of Probabilistic Systems*. PhD thesis, Techn. Univ. Eindhoven, 2005.

[45] P. Taylor. Subspaces in abstract Stone duality. *Theory and Appl. of Categories*, 10(13):301–368, 2002.

[46] R. Tix, K. Keimel, and G. Plotkin. *Semantic Domains for Combining Probability and Non-Determinism*. Number 129 in Elect. Notes in Theor. Comp. Sci. Elsevier, Amsterdam, 2005.

[47] O. Wyler. Algebraic theories of continuous lattices. In B. Banaschewski and R.-E. Hoffman, editors, *Continuous Lattices*, number 871 in Lect. Notes Math., pages 390–413. Springer, Berlin, 1981.