Episode 8 - Monads and Algebras

Ralph Sarkis
November 30, 2020

Abstract

We introduce monads from three point of views: categorical, algebraic and computational.

1 POV: Category Theory

We will start from the concept of an adjunction which, as we hope was made clear in the previous lecture, is ubiquitous and powerful throughout mathematics. However, we will start with a great oversimplification, we will assume the categories concerned are posetal.

An adjunction between posets is more commonly called a Galois connection.

Definition 1 (Galois connection). Let $(P, \leq)$ and $(Q, \sqsubseteq)$ be posets. A **Galois connection** between them is a pair of order-preserving functions $L : P \to Q$ and $R : Q \to P$ satisfying for any $p \in P$ and $q \in Q$, $L(p) \sqsubseteq q \iff p \leq R(q)$.

We are now interested in the composition $R \circ L$. It is also a monotonic function but the results about adjoints we have seen yield a couple of interesting properties. First, the existence of the unit $\eta : \mathrm{id}_P \Rightarrow RL$ means that for any $p \in P$, there is $\eta_p : p \to RL(p)$, so RL is **extensive** (i.e.: $\forall p \in P, p \leq RL(p)$). Second, the existence of the counit $\varepsilon : RL \Rightarrow \mathrm{id}_P$ means that for any $p \in P$, there is $R(\varepsilon_{L(p)}) : RLRL(p) \to RL(p)$ and $RL(\eta_p) : RL(p) \to RLRL(p)$, so RL is idempotent (i.e.: $\forall p \in P, RL(p) = RLRL(p)$). We say that RL is a closure operator.

Definition 2 (Closure operator). Let $(P, \leq)$ be a poset, a **closure operator** on P is a monotone extensive and idempotent function $c : P \to P$.

Example 3. We can give a very simple example hinting at the origins of the terminology. Consider the real numbers $\mathbb{R}$ with the standard topology. We know that $(\mathcal{P}(\mathbb{R}), \subseteq)$ is a poset and we can define $c : \mathcal{P}(\mathbb{R}) \to \mathcal{P}(\mathbb{R})$ sending $U \subseteq \mathbb{R}$ to its closure $c(U)$ in the topological sense as in $c(U)$ is the set of limit points of U. Then, you probably have seen that for any $U, V \subseteq \mathbb{R}, U \leq V \implies c(U) \subseteq c(V), U \subseteq c(U)$ and $c(U) = c(c(U))$, thus operation of closure is a closure operator.

We will generalize this discussion to arbitrary categories now. Let $L : \mathbf{C} \rightleftarrows \mathbf{D} : R$ be an adjoint pair of functors, we have two natural transformations $\eta : \mathrm{id}_{\mathbf{C}} \Rightarrow RL$ and $R\varepsilon L : RLRL \Rightarrow RL$. Recall these may interact well together via the zig-zag identities that we reformulate in (1) and (2) and we add to those the commutativity diagram (3) in the definition of $\varepsilon \diamond \varepsilon$.

$$
\begin{array}{ccc}
L \xRightarrow{L\eta} LRL & RLR \xLeftarrow{\eta R} R & LRLR \xRightarrow{\varepsilon LR} LR \\
\end{array}
$$

$$
\begin{array}{lcr}
L \xRightarrow{L\eta} LRL \quad (1) & RLR \xLeftarrow{\eta R} R \quad (2) & LRLR \xRightarrow{\varepsilon LR} LR \quad (3) \\
\Vert_{\mathbb{1}_L} \;\; \Downarrow \varepsilon L & R\varepsilon \Downarrow \;\; \nearrow \mathbb{1}_L & LR\varepsilon \Downarrow \qquad \Downarrow \varepsilon \\
L & L & LR \xRightarrow{\varepsilon} \mathrm{id}_D
\end{array}
$$

With a bit of tinkering, we can make these diagrams all about the functor we are interested in, namely RL. Indeed, if we denote $M = RL$ and $\mu = R\varepsilon L$ before acting by R on the right of (1), L on the left of (2) and $R(\cdot)L$ on (3), we obtain the definition of a monad.

Definition 4 (Monad). A **monad** is a triple comprised of an endofunctor $M : \mathbf{C} \rightsquigarrow \mathbf{C}$ and two natural transformations $\eta : \mathrm{id}_{\mathbf{C}} \Rightarrow M$ and $\mu : M^2 \Rightarrow M$ called the **unit** and **multiplication** respectively that make (4) and (5) commute in $[\mathbf{C}, \mathbf{C}]$.

$$
\begin{array}{ccc}
M \xRightarrow{M\eta} M^2 \xLeftarrow{\eta M} M & \qquad & M^3 \xRightarrow{M\mu} M^2 \\
\mathbb{1}_M \searrow \;\; \Downarrow \mu \;\; \swarrow \mathbb{1}_M \qquad (4) & \qquad \mu M \Downarrow \qquad \Downarrow \mu \qquad (5) \\
M & \qquad & M^2 \xRightarrow{\mu} M
\end{array}
$$

Examples 5. Our discussion above tells us that any adjoint pair $L \dashv R$ corresponds to a monad $(RL, \eta, R\varepsilon L)$, so all the examples of adjunctions you have seen are suitable examples of monads. For instance, all closure operators are monads. Here, we describe three simple yet very useful examples and let you ponder on the adjunctions they might or might not originate from.

1. Suppose $\mathbf{C}$ has (binary) coproducts and a terminal object $\mathbf{1}$, then $(\cdot + \mathbf{1}) : \mathbf{C} \rightsquigarrow \mathbf{C}$ is a monad. We write inl^{X+Y} (resp. inr^{X+Y}) for the injection of X (resp. Y) into $X + Y$. First, note that for a morphism $f : X \to Y$,

$$
f + \mathbf{1} = [\mathsf{inl}^{Y+\mathbf{1}} \circ f, \mathsf{inr}^{Y+\mathbf{1}}] : X + \mathbf{1} \to Y + \mathbf{1}.
$$

The components of the unit are given by the coprojections, i.e.: $\eta_X = \mathsf{inl}^{X+\mathbf{1}} : X \to X + \mathbf{1}$, and the components of the multiplication are

$$
\mu_X = [\mathsf{inl}^{X+\mathbf{1}}, \mathsf{inr}^{X+\mathbf{1}}, \mathsf{inr}^{X+\mathbf{1}}] : X + \mathbf{1} + \mathbf{1} \to X + \mathbf{1}.
$$

Checking that (4) commutes, we have for any $X \in \mathbf{C}$:

$$
\begin{aligned}
\mu_X \circ (\eta_X + \mathbf{1}) &= [\mu_X \circ \mathsf{inl}^{(X+\mathbf{1})+\mathbf{1}} \circ \eta_X, \mu_X \circ \mathsf{inr}^{(X+\mathbf{1})+\mathbf{1}}] \\
&= [[\mathsf{inl}^{X+\mathbf{1}}, \mathsf{inr}^{X+\mathbf{1}}] \circ \mathsf{inl}^{X+\mathbf{1}}, \mathsf{inr}^{X+\mathbf{1}}] \\
&= [\mathsf{inl}^{X+\mathbf{1}}, \mathsf{inr}^{X+\mathbf{1}}] \\
&= \mathrm{id}_{X+\mathbf{1}} \\
&= [\mathsf{inl}^{X+\mathbf{1}}, \mathsf{inr}^{X+\mathbf{1}}] \\
&= \mu_X \circ \mathsf{inl}^{(X+\mathbf{1})+\mathbf{1}} \\
&= \mu_X \circ \eta_{X+\mathbf{1}}
\end{aligned}
$$

For (5), we have for any $X \in \mathbf{C}$:

$$
\begin{aligned}
\mu_X \circ (\mu_X + \mathbf{1}) &= [\mu_X \circ \mathsf{inl}^{(X+\mathbf{1})+\mathbf{1}} \circ \mu_X, \mu_X \circ \mathsf{inr}^{(X+\mathbf{1})+\mathbf{1}}] \\
&= [[\mathsf{inl}^{X+\mathbf{1}}, \mathsf{inr}^{X+\mathbf{1}}] \circ \mu_X, \mathsf{inr}^{X+\mathbf{1}}] \\
&= [[\mathsf{inl}^{X+\mathbf{1}}, \mathsf{inr}^{X+\mathbf{1}}, \mathsf{inr}^{X+\mathbf{1}}], \mathsf{inr}^{X+\mathbf{1}}] \\
&= [\mu_X, \mathsf{inr}^{X+\mathbf{1}}] \\
&= [[\mathsf{inl}^{X+\mathbf{1}}, \mathsf{inr}^{X+\mathbf{1}}], \mathsf{inr}^{X+\mathbf{1}}, \mathsf{inr}^{X+\mathbf{1}}] \\
&= [\mu_X \circ \mathsf{inl}^{(X+\mathbf{1})+\mathbf{1}}, \mu_X \circ \mathsf{inr}^{(X+\mathbf{1})+\mathbf{1}}, \mu_X \circ \mathsf{inr}^{(X+\mathbf{1})+\mathbf{1}}] \\
&= \mu_X \circ \mu_{X+\mathbf{1}}
\end{aligned}
$$

2. The covariant powerset functor $\mathcal{P} : \mathbf{Set} \rightsquigarrow \mathbf{Set}$ is a monad with the following unit and multiplication:

$$
\eta_X : X \to \mathcal{P}(X) = x \mapsto \{x\} \text{ and } \mu_X : \mathcal{P}(\mathcal{P}(X)) \to \mathcal{P}(X) = F \mapsto \bigcup_{s \in F} s.
$$

Checking that (4), we have for any $S \subseteq \mathcal{P}(X)$,

$$
\begin{aligned}
\mu_X(\mathcal{P}(\eta_X)(S)) &= \mu_X \left(\{\{x\} \mid x \in S\} \right) \\
&= \bigcup_{x \in S} \{x\} \\
&= S \\
&= \bigcup \{S\} \\
&= \mu_X(\{S\}) \\
&= \mu_X(\eta_{\mathcal{P}(X)}(S))
\end{aligned}
$$

Checking that (5) commutes, we have for any $\mathcal{F} \in \mathcal{P}(\mathcal{P}(\mathcal{P}(X)))$,

$$
\begin{aligned}
\mu_X(\mu_{\mathcal{P}(X)}(\mathcal{F})) &= \mu_X \left(\bigcup_{F \in \mathcal{F}} F \right) \\
&= \bigcup_{\substack{s \in \mathcal{P}(X) \\ \exists F \in \mathcal{F}, s \in F}} s \\
&= \{x \in X \mid \exists s \in \mathcal{P}(X), x \in s \text{ and } \exists F \in \mathcal{F}, s \in F\} \\
&= \bigcup_{F \in \mathcal{F}} \bigcup_{s \in F} s \\
&= \mu_X \left(\left\{ \bigcup_{s \in F} s \mid F \in \mathcal{F} \right\} \right) \\
&= \mu_X(\mathcal{P}(\mu_X)(\mathcal{F}))
\end{aligned}
$$

3. The functor $\mathcal{D} : \textbf{Set} \to \textbf{Set}$ sends a set X to the set of finitely supported distributions on X, i.e.:

$$\mathcal{D}(X) := \{\varphi \in [0,1]^X \mid \sum_{x \in X} \varphi(x) = 1 \text{ and } \varphi(x) \neq 0 \text{ for finitely many } x\text{'s}\}.$$

It sends a function $f : X \to Y$ to the function between distributions

$$\lambda \varphi^{\mathcal{D}(X)}.\lambda y^Y.\varphi(f^{-1}(y)).$$

More verbosely, the weight of $\mathcal{D}(f)(\varphi)$ at point y is equal to the total weight of φ on the preimage of y under f. It is a monad with unit $\eta_X = x \mapsto \delta_x$, where δ_x is the Dirac distribution at x (all the weight is at x), and multiplication

$$\mu_X = \Phi \mapsto \lambda x^X. \sum_{\phi \in \mathrm{supp}(\Phi)} \Phi(\phi) \cdot \phi(x),$$

where $\mathrm{supp}(\Phi)$ is the support of Φ, i.e.: $\mathrm{supp}(\Phi) := \{\varphi \mid \Phi(\varphi) \neq 0\}$.

After looking long enough for adjunctions giving rise to the monads in Example 5 (even if you found one), a natural question can be asked. Does every monad arise from an adjunction? If yes, is that adjunction unique?

The second question might not be as natural to novices in category theory but it is almost as important as the first one. Indeed, uniqueness is a very strong property and if every monad had a unique corresponding adjunction, one might expect it to be fairly easy to find. This is part of the beauty of category theory. We are working with very little data M, η and μ so if it completely determined an adjunction L, R and $\mathrm{Hom}(L(-), -) \cong \mathrm{Hom}(-, R(-))$, it could not do so in a very convoluted way merely because there is not that many ways to manipulated the original data.

In any case, we will respectively give a positive and negative answer to these questions. Fortunately, while we might not benefit from the power of uniqueness, there are two special adjunctions arising from a monad whose descriptions are fairly straightforward. In the order we present them, the first is due to Kleisli and the second to Eilenberg and Moore. In the rest of this section, (M, η, μ) will be a monad on a category $\textbf{C}$.

1.1   Kleisli Category $\textbf{C}_M$

An intuitive way to think about monads is through *generalized elements*. Given an object $A \in \textbf{C}_0$, we can view MA as extending A with more general or structured elements built from A.

In this picture, the morphisms $\eta_A : A \to MA$ give a way to understand anything inside A trivially as a general element of A. The morphisms $\mu_A : M^2 A \to MA$ imply that higher order structures can collapsed so that generalized elements over generalized elements of A are generalized elements of A. The functoriality of M implies that the new structures in MA are somewhat independent of A. Indeed, for every morphisms $f : A \to B$, there is a morphism $Mf : MA \to MB$ which, by naturality of η ($Mf(\eta_A(-)) = \eta_B(f(-))$), acts just like f on the trivial generalization of elements in A. Commutativity of (4) says that the trivial generalization (there are two ways to do it) of a generalized element is indeed trivial as after collapsing via μ, we end up with what we started with.

Finally, the associativity of μ (i.e.: commutativity of (5)) corresponds to the fact that in higher order of generalizations, one can collapse the structure at every level in any order and end up with the same thing.

Now, we can also consider *generalized morphisms*. Let us say we were given an ill-defined morphism $f : A \to B$ that sends some of the stuff in A outside of B. One way to fix this might be to consider general elements of B and see f as a morphism $A \to MB$. We will call such morphisms **Kleisli morphisms** and write $f : A \to MB$ or $f : A \nrightarrow B$.

With an arbitrary functor F, you might have a hard time to come up with a way to compose them Kleisli morphisms $A \to FB$ and $B \to FC$ or define the identity Kleisli morphism $A \to FA$, but the data of a monad lets you do just that. We end up with the category $\mathbf{C}_M$.

Definition 6 ($\mathbf{C}_M$). Let $\mathbf{C}$ be a category and (M, η, μ) a monad on $\mathbf{C}$. The **Kleisli category** of M, denoted $\mathbf{C}_M$ has objects $\mathbf{C}_0$ and morphisms $f : A \to MB \in \mathbf{C}_1$. The identity for A is $\eta_A : A \to MA$ and composition is given by

$$g \circ_M f = \mu_C \circ Mg \circ f : A \nrightarrow C,$$

where $f : A \nrightarrow B$ and $g : B \nrightarrow C$.

Checking that this is indeed a category is left as an exercise and you should remark tha it requires using all the properties M, η and μ satisfy.

As we have constructed $\mathbf{C}_M$ with objects and morphisms in $\mathbf{C}$, we would like to describe a forgetful functor $U_M : \mathbf{C}_M \rightsquigarrow \mathbf{C}$. A first guess to send objects A to themselves will fail when it is time to defining the action of U_M on morphisms for $f : A \nrightarrow B$ can send some stuff into generalized elements of B, and we have no way to get this back into B. Instead, we need to generalize everything before going into $\mathbf{C}$, namely, $U_M(A) = MA$ and $U_M f = \mu_B \circ Mf : MA \to MB$.

We can check that for any A, $U_M(\eta_A) = \mu_A \circ M(\eta_A) \overset{(4)}{=} \mathrm{id}_A$ and for any for any $f : A \nrightarrow B$ and $g : B \nrightarrow C$,

$$
\begin{aligned}
U_M(g \circ_M f) &= U_M(\mu_C \circ Mg \circ f) \\
&= \mu_C \circ M(\mu_C \circ Mg \circ f) \\
&= \mu_C \circ M(\mu_C) \circ MMg \circ Mf \\
&= \mu_C \circ \mu_{MC} \circ MMg \circ Mf && \text{by (5)} \\
&= \mu_C \circ Mg \circ \mu_B \circ Mf && \text{by naturality of } \mu \\
&= U_M(g) \circ U_M(f).
\end{aligned}
$$

We conclude that U_M is a functor.

In the opposite direction, we want a functor $F_M : \mathbf{C} \rightsquigarrow \mathbf{C}_M$. This time, we send A to itself because we can make any morphism $f : A \to B$ generalized by post-composing with η_B, namely $F_M(A) = A$ and $F_M(f) = \eta_B \circ f$ for any $f : A \to B$. For functoriality,

$F_M(\mathrm{id}_A)$ is trivial as η_A is the identity on A in $\mathbf{C}_M$ and

$$
\begin{aligned}
F_M(g \circ f) &= \eta_C \circ g \circ f \\
&= Mg \circ \eta_B \circ f & \text{by naturality of } \eta \\
&= Mg \circ \mu_B \circ M(\eta_B) \circ \eta_B \circ f & \text{by (4)} \\
&= \mu_C \circ MMg \circ M(\eta_B) \circ \eta_B \circ f & \text{by naturality of } \mu \\
&= \mu_C \circ M(\eta_C) \circ Mg \circ \eta_B \circ f & \text{by naturality of } \eta \\
&= F_M(g) \circ_M F_M(f)
\end{aligned}
$$

Before checking that this is indeed an adjunction, we need to make sure it composes to M, that is $U_M F_M = M$. On objects, it is clear. On a morphism $f : A \to B$, we have

$$
U_M(F_M(f)) = U_M(\eta_B \circ f) = \mu_B \circ M(\eta_B) \circ Mf \overset{(4)}{=} Mf.
$$

Let us now verify that $F_M \dashv U_M$. Let $A, B \in \mathbf{C}_0$ (we view B as an object of $\mathbf{C}_M$), we need to exhibit a natural isomorphism $\mathrm{Hom}_{\mathbf{C}_M}(F_M A, B) \cong \mathrm{Hom}_{\mathbf{C}}(A, U_M B)$. The isomorphism is clear as $U_M B = MB$, $F_M A = A$ and a morphism $A \to MB$ is precisely a Kleisli morphism $A \nrightarrow B$, we will denote it id. For the naturality, we need to show the following square commutes for any $f : A' \to A$ and $g : B \nrightarrow B'$.

$$
\begin{array}{ccc}
\mathrm{Hom}_{\mathbf{C}_M}(A, B) & \xrightarrow{\ \mathrm{id}\ } & \mathrm{Hom}_{\mathbf{C}}(A, MB) \\
{\scriptstyle g \circ_M (-) \circ_M F_M f} \downarrow & & \downarrow {\scriptstyle U_M g \circ (-) \circ f} \\
\mathrm{Hom}_{\mathbf{C}_M}(A', B') & \xrightarrow[\ \mathrm{id}\ ]{} & \mathrm{Hom}_{\mathbf{C}}(A', MB')
\end{array}
\tag{6}
$$

It follows from the following derivation starting with a morphism $k : A \to MB$.

$$
\begin{aligned}
g \circ_M k \circ_M F_M f &= \mu_{B'} \circ M(g) \circ \mu_B \circ M(k) \circ \eta_A \circ f \\
&= \mu_{B'} \circ M(g) \circ \mu_B \circ \eta_{MB} \circ k \circ f & \text{by naturality of } \eta \\
&= \mu_{B'} \circ M(g) \circ \mathrm{id}_{MB} \circ k \circ f & \text{by (4)} \\
&= \mu_{B'} \circ M(g) \circ k \circ f \\
&= U_M g \circ k \circ f
\end{aligned}
$$

Recall that we claimed (F_M, U_M) was special in some way and that this was the (informal) reason why it was relatively easy to find, the next proposition will make this precise.

Definition 7 (Adj$_M$). Let $\mathbf{C}$ be a category and (M, η, μ) a monad on $\mathbf{C}$. The category of adjunctions inducing M is denoted Adj$_M$. Its objects are adjoint pairs (L, R) with $R \circ L = M$ whose unit is η and whose counit ε satisfies $R \varepsilon L = \mu$. Its morphisms $(L, R) \to (L', R')$ are functors K satisfying $K \circ L = L'$ and $R' \circ K = R$ as in (7).

$$
\begin{array}{ccc}
\mathbf{D} & \xrightarrow{\ \ K\ \ } & \mathbf{D}' \\
& & \\
\end{array}
\tag{7}
$$

Let us first check that (F_M, U_M) induces (M, η, μ), we already know that $U_M F_M = M$, but it remains to find the unit and counit of this adjunction. We will need to use the correspondence between the definitions of adjoint pairs seen in the last lecture. Recall that the natural isomorphism $\mathrm{Hom}_{\mathbf{C}_M}(F_M-, -) \cong \mathrm{Hom}_{\mathbf{C}}(-, U_M-)$ is simply the identity function between these two sets. Hence, the unit of the adjunction is $\mathbf{C}_0 \ni A \mapsto \mathrm{id}(u_{\mathbf{C}_M}(A)) = \eta_A$ and the counit is $(\mathbf{C}_M)_0 \ni A \mapsto \mathrm{id}(u_{\mathbf{C}}(U_M A)) =: \varepsilon_A$. Note that ε_A has type $MA \twoheadrightarrow B$. We check that $U_M(\varepsilon_{F_M A}) = \mu_A$:

$$U_M(\varepsilon_{F_M A}) = \mu_A \circ \varepsilon_{F_M A} = \mu_A \circ u_{\mathbf{C}}(MA) = \mu_A.$$

Proposition 8. *The adjunction (F_M, U_M) is initial in* Adj_M.

Proof. Let $\mathbf{C} : L \dashv R : \mathbf{D} \in \mathrm{Adj}_M$ with unit η and counit ε, we claim there is a unique functor $K : \mathbf{C}_M \rightsquigarrow \mathbf{D}$ satisfying $K \circ F_M = L$ and $R \circ K = U_M$ as in (8).

$$
\begin{array}{c}
\mathbf{C}_M \xrightarrow{\quad K \quad} \mathbf{D} \\
\text{(diagram with } U_M, L, F_M, R, \mathbf{C})
\end{array}
\tag{8}
$$

On objects, K is determined by $KA = KF_M A = LA$. To a morphism $f : A \twoheadrightarrow B$, we need to assign a morphism in $Kf \in \mathrm{Hom}_{\mathbf{D}}(LA, LB)$. Denote $\Phi_{A,B} : \mathrm{Hom}_{\mathbf{C}_M}(F_M A, B) \cong \mathrm{Hom}_{\mathbf{C}}(A, U_M B)$ and $\Psi_{A,B} : \mathrm{Hom}_{\mathbf{D}}(LA, B) \cong \mathrm{Hom}_{\mathbf{C}}(A, RB)$ be given by the adjunctions $F_M \dashv U_M$ and $L \dashv R$ respectively. Using the fact that $RK = U_M$, we obtain the following commutative diagram.

$$
\begin{array}{ccccccc}
\mathrm{Hom}_{\mathbf{C}_M}(F_M A, A) & \xrightarrow{\Phi_{A,A}} & \mathrm{Hom}_{\mathbf{C}}(A, U_M A) & == & \mathrm{Hom}_{\mathbf{C}}(A, RLA) & \xleftarrow{\Psi_{A,LA}} & \mathrm{Hom}_{\mathbf{D}}(LA, LA) \\
\Big\downarrow{\scriptstyle f \circ_M (-)} & & \Big\downarrow{\scriptstyle U_M f \circ (-)} & & \Big\downarrow{\scriptstyle RKf \circ (-)} & & \Big\downarrow{\scriptstyle Kf \circ (-)} \\
\mathrm{Hom}_{\mathbf{C}_M}(F_M A, B) & \xrightarrow[\Phi_{A,B}]{} & \mathrm{Hom}_{\mathbf{C}}(A, U_M B) & == & \mathrm{Hom}_{\mathbf{C}}(A, RLB) & \xleftarrow[\Psi_{A,LB}]{} & \mathrm{Hom}_{\mathbf{D}}(LA, LB)
\end{array}
$$

Because these adjunctions both have the same unit, we have $\Phi_{A,A}(u_{\mathbf{C}_M}(F_M A)) = \eta_A = \Psi_{A,A}(u_{\mathbf{D}}(LA))$. Therefore, if we start with $\eta_A \in \mathrm{Hom}_{\mathbf{C}_M}(F_M A, A)$ and follow the diagram, we infer that $Kf = \Psi_{A,LB}^{-1}(\Phi_{A,B}(f))$. Using the fact that $\Phi_{A,B}(f) = f$ and $\Psi_{A,LB}^{-1}(f) = \varepsilon_{LB} \circ Lf$ (follow the proof of the Yoneda lemma), we end up with $Kf = \varepsilon_{LB} \circ Lf$ and we can verify it is indeed functorial.

$$K(u_{\mathbf{C}_M}(A)) = K(\eta_A) = \varepsilon_{LB} \circ L(\eta_A) \overset{\text{zig-zag}}{=} \mathrm{id}_A$$

$$
\begin{aligned}
K(g \circ_M f) &= K(\mu_{\mathbf{C}} \circ RLg \circ f) \\
&= \varepsilon_{LC} \circ L(\mu_{\mathbf{C}}) \circ LRLg \circ Lf \\
&= \varepsilon_{LC} \circ LR\varepsilon_{LC} \circ LRLg \circ Lf & \text{by hypothesis on } \varepsilon \\
&= \varepsilon_{LC} \circ \varepsilon_{LRLC} \circ LRLg \circ Lf & \text{by naturality of } \varepsilon \\
&= \varepsilon_{LC} \circ Lg \circ \varepsilon_{LB} \circ Lf & \text{by naturality of } \varepsilon \\
&= Kg \circ Kf
\end{aligned}
$$

$\square$

1.2 Eilenberg-Moore Category $\mathbf{C}^M$

For the second solution to the problem of finding an adjunction inducing a given monad, we look at the more structural side of monads.

Definition 9 (*M*–algebra)**.** Let (M, η, μ) be a monad, an **Eilenberg-Moore algebra** for *M* or simply *M*–**algebra** is a pair (A, α) consisting of an object $A \in \mathbf{C}_0$ and morphism $\alpha : MA \to A$ such that (9) and (10) commute.

$$
\begin{array}{ccc}
A \xrightarrow{\eta_A} MA \\
\;\;\searrow_{\mathrm{id}_A} \;\downarrow{\alpha} \\
A
\end{array}
\qquad (9)
\qquad
\begin{array}{ccc}
M^2A \xrightarrow{\mu_A} MA \\
M(\alpha)\downarrow \qquad \downarrow{\alpha} \\
MA \xrightarrow{\;\;\alpha\;\;} A
\end{array}
\qquad (10)
$$

We will often denote an *M*–algebra using only its underlying set or morphism.

Definition 10 (*M*–algebra homomorphism)**.** Given two *M*–algebras (A, α) and (B, β), an *M*–algebra homomorphism $f : (A, \alpha) \to (B, \beta)$ is a morphism $h : A \to B$ making (11) commute.

$$
\begin{array}{ccc}
MA \xrightarrow{Mh} MB \\
\alpha\downarrow \qquad \downarrow{\beta} \\
A \xrightarrow{\;\;h\;\;} B
\end{array}
\qquad (11)
$$

For a monad *M*, the category of *M*–algebras is called the **Eilenberg-Moore** category of *M* and denoted $\mathbf{C}^M$, composition and identities are induced by the composition and identities in $\mathbf{C}$. Again, since $\mathbf{C}^M$ was built from objects and morphisms in $\mathbf{C}$, there is an obvious candidate for a forgetful functor $U^M : \mathbf{C}^M \rightsquigarrow \mathbf{C}$ sending an *M*–algebra (A, α) to its underlying object *A* and a homomorphism to its underlying morphism. Next, we need a left adjoint to U^M, $F^M : \mathbf{C} \rightsquigarrow \mathbf{C}^M$. Since we want $U^M F^M = M$ to hold, F^M must send $A \in \mathbf{C}_0$ to an *M*–algebra over MA and $h \in \mathbf{C}_1$ to Mh. There is only one choice given to us by the data of *M*, that is, $F^M A = (MA, \mu_A))$ and it turns out naturality of μ_A yields

$$
\begin{array}{ccc}
M^2A \xrightarrow{M^2h} M^2B \\
\mu_A\downarrow \qquad \downarrow{\mu_B} \\
MA \xrightarrow{\;\;Mh\;\;} MB
\end{array}
\qquad ,
\qquad (12)
$$

which implies *Mh* is indeed an *M*–algbera homomorphism. Let us now show that $F^M \dashv U^M$ with unit η and counit ε satisfying $U^M \varepsilon F^M = \mu$.

We want to exhibit an isomorphism $\mathrm{Hom}_{\mathbf{C}^M}(\mu_A, \beta) \cong \mathrm{Hom}_{\mathbf{C}}(A, B)$ natural in *A* and $\beta : MB \to B$. In the forward direction, we send $h : MA \to B$ to $h \circ \eta_A : A \to B$ which ensure the unit of this adjunction is η. In the backwards direction, we send $h : A \to B$ to $\beta \circ Mh$ which is a homomorphism by the following diagram where (a) commutes by

naturality of μ and (b) by β being an M–algebra.

$$
\begin{array}{ccccc}
M^2A & \xrightarrow{M^2h} & M^2B & \xrightarrow{M\beta} & MB \\
\mu_A \downarrow & \text{(a)} & \downarrow \mu_B & \text{(b)} & \downarrow \beta \\
MA & \xrightarrow[Mh]{} & MB & \xrightarrow{\;\;\beta\;\;} & B
\end{array}
\tag{13}
$$

One can easily check that these operations are inverses of each other. Next, we turn to naturality. Let $f : A' \to A \in \mathbf{C}_1$ and $g : (B,\beta) \to (B',\beta') \in \mathbf{C}_1^M$, we claim that (14) commutes.

$$
\begin{array}{ccc}
\mathrm{Hom}_{\mathbf{C}^M}(\mu_A,\beta) & \xrightarrow{(-)\circ\eta_A} & \mathrm{Hom}_{\mathbf{C}}(A,B) \\
{\scriptstyle g\circ(-)\circ Mf}\downarrow & & \downarrow{\scriptstyle g\circ(-)\circ f} \\
\mathrm{Hom}_{\mathbf{C}^M}(\mu_{A'},\beta') & \xrightarrow[(-)\circ\eta_{A'}]{} & \mathrm{Hom}_{\mathbf{C}}(A',B')
\end{array}
\tag{14}
$$

Starting with a homomorphism $h : (MA,\mu_A) \to (B,\beta)$ in the top-left object, we need to show $g \circ h \circ Mf \circ \eta_{A'} = g \circ h \circ \eta_A \circ f$ which holds by naturality of η.

We find the counit of this adjunction is $\varepsilon_\alpha = \alpha : (MA,\mu_A) \to (A,\alpha)$ which is a homomorphism because (10) commutes and we verify

$$
U^M(\varepsilon_{F^MA}) = U^M(\varepsilon_{\mu_A}) = U^M(\mu_A) = \mu_A.
$$

Dually to Proposition 8, we show that this adjunction is special in a precise way.

Proposition 11. *The adjunction* (F^M,U^M) *is terminal in* Adj_M.

Proof. Let $\mathbf{C} : L \dashv R : \mathbf{D} \in \mathrm{Adj}_M$ with unit η and counit ε, we claim there is a unique functor $K : \mathbf{D} \rightsquigarrow \mathbf{C}^M$ satisfying $K \circ L = F^M$ and $U^M \circ K = R$ as in (15).

$$
\begin{array}{ccc}
\mathbf{D} & \xrightarrow{\quad K \quad} & \mathbf{C}^M \\
 & R \quad F^M & \\
 L & & U^M \\
 & \mathbf{C} &
\end{array}
\tag{15}
$$

As before, we can determine K by the equation $U^MK = R$ which means it sends $A \in \mathbf{D}_0$ to an algebra on RA and $f : A \to B \in \mathbf{D}_1$ to an algebra homomorphism $Rf : KA \to KB$. The only missing piece of this puzzle is the algebra structure on KA. The only clue we have is the fact that Rf is a homomorphism so is $KA = \alpha$ and $KB = \beta$, then (16) commutes.

$$
\begin{array}{ccc}
MRA & \xrightarrow{MRf} & MRB \\
\alpha \downarrow & & \downarrow \beta \\
RA & \xrightarrow[Rf]{} & RB
\end{array}
\tag{16}
$$

Replacing M with RL, we recognize this as the naturality diagram for ε acted on by $R(\cdot)$. Hence, we find $KA = (RA, R\varepsilon_A)$ is a suitable candidate after showing it is indeed an

M–algebra. For the unit diagram, we have $R\varepsilon_A \circ \eta_A = \mathrm{id}_A$ by the zig-zag identities. For the multiplication diagram, we have

$$R\varepsilon_A \circ \mu_A = R\varepsilon_A \circ R\varepsilon_{LA} = R(\varepsilon_A \circ \varepsilon_{LA}) = R(\varepsilon_A \circ LR(\varepsilon_A)) = R\varepsilon_A \circ MR\varepsilon_A.$$

As K acts like R on morphisms, it is obviously functorial. We leave the proof that K is unique as an exercise as the argument should be somewhat similar to what we have done in the dual proposition. $\qquad\square$

In summary, you can keep in mind the following diagram.

$$(17)$$

One last thing we want to mention is the embedding of the Kleisli category inside the Eilenberg-Moore category. After a bit more work, we can infer from the discussion above that the unique morphism of adjunctions $K : \mathbf{C}_M \rightsquigarrow \mathbf{C}^M$ is a fully faithful functor sending an object A in $\mathbf{C}_M$ to the algebra (MA, μ_A), called the **free algebra** on A, and a Kleisli morphism $f : A \nrightarrow B$ to the homomorphism $\mu_B \circ Mf : (A, \mu_A) \to (B, \mu_B)$.

2 POV: Universal Algebra

In this section, we will highlight the link between algebraic structures as you have encountered them in other classes with the Eilenberg-Moore algebras discussed above. We will only work over the category **Set**. We start by developing an example.

Example 12 ($\mathcal{P}_{\mathrm{ne}}$). Consider the non-empty finite powerset functor $\mathcal{P}_{\mathrm{ne}}$ sending X to $\{S \in \mathcal{P}(X) \mid S \text{ is finite and non-empty}\}$. The same unit and multiplication defined for $\mathcal{P}$ make $\mathcal{P}_{\mathrm{ne}}$ into a monad. A $\mathcal{P}_{\mathrm{ne}}$–algebra is a function $\alpha : \mathcal{P}_{\mathrm{ne}}(A) \to A$ satisfying the equations $\alpha\{a\} = a$ and $\alpha(\mathcal{P}_{\mathrm{ne}}(\alpha)(S)) = \alpha(\bigcup S)$. From this, we can extract a binary operation $\oplus_\alpha : A \times A \to A$ by defining $x \oplus_\alpha y = \alpha\{x, y\}$. This operation is clearly commutative and idempotent, i.e.: $x \oplus_\alpha y = y \oplus_\alpha y$ and $x \oplus_\alpha x = x$, but it is also associative by the following derivation.

$$
\begin{aligned}
(x \oplus_\alpha y) \oplus_\alpha z &= \alpha\{x, y\} \oplus_\alpha z \\
&= \alpha\{\alpha\{x, y\}, z\} \\
&= \alpha\{\alpha\{x, y\}, \alpha\{z\}\} \\
&= \alpha\{\mathcal{P}_{\mathrm{ne}}\alpha\{\{x, y\}, \{z\}\}\} \\
&= \alpha\{\mu_A\{\{x, y\}, \{z\}\}\} \\
&= \alpha\{x, y, z\}.
\end{aligned}
$$

Since a $\mathcal{P}_{\mathrm{ne}}$–algebra homomorphism $h : (A, \alpha) \to (B, \beta)$ commutes with α and β it also commutes with $\oplus_\alpha$ and $\oplus_\beta$.

Conversely, if $(A, \oplus)$ is an idempotent, associative and commutative binary operation on A, we can define $\alpha_\oplus$ on non-empty finite sets of A by iterating $\oplus$. Namely,

$$\alpha_\oplus\{x\} = x \oplus x \quad \text{and} \quad \alpha_\oplus\{x_1, \ldots, x_n\} = x_1 \oplus x_2 \oplus \cdots \oplus x_n.$$

It is well-defined by associativity and commutativity and we can check that it is the inverse of the operation described in the previous paragraph. That is to say, we can check that $\alpha_{\oplus_\alpha} = \alpha$ and $\oplus_{\alpha_\oplus} = \oplus$. For the former, it is clear for singleton sets and for any $n > 1$, we have the following derivation.

$$
\begin{aligned}
\alpha_{\oplus_\alpha}\{x_1, \ldots, x_n\} &= x_1 \oplus_\alpha \cdots \oplus_\alpha x_n \\
&= \alpha\{x_1, x_2 \oplus_\alpha \cdots \oplus_\alpha x_n\} \\
&= \vdots \\
&= \alpha\{x_1, \alpha\{x_2, \alpha\{\cdots, \alpha\{x_n\}\}\}\} \\
\text{using } \alpha \circ \mathcal{P}_{\mathrm{ne}}(\alpha) = \alpha \circ \mu_A &= \alpha\{x_1, x_2, \alpha\{\cdots, \alpha\{x_n\}\}\} \\
&= \vdots \\
&= \alpha\{x_1, \ldots, x_n\}
\end{aligned}
$$

For the latter, we have

$$x \oplus_{\alpha_\oplus} y = \alpha_\oplus\{x, y\} = x \oplus y.$$

A set equipped with an idempotent, commutative and associative binary operation is called a **(join)-semilattice** and we have shown above that $\mathcal{P}_{\mathrm{ne}}$–algebras are in correspondence with semilattices. Through the introduction of basic notions in universal algebra, we will explain how this correspondence is functorial and generalize the core idea behind it.

Definition 13 (Algebraic theory). An **algebraic signature** is a set Σ of operation symbols along with arities in $\mathbb{N}$, we denote $f : n \in \Sigma$ for an n–ary operation f in Σ. Given a set X, one constructs the set of Σ–**terms** with variables in X, denoted $T_\Sigma(X)$ by iterating operations symbols:

$$\forall x \in X, x \in T_\Sigma(X)$$
$$\forall t_1, \ldots, t_n \in T_\Sigma(X), f : n \in \Sigma, f(t_1, \ldots, t_n) \in T_\Sigma(X).$$

An **equation** (sometimes **axiom**) E over Σ is a pair of Σ–terms over a set of dummy variables. We will call the tuple (Σ, E) an **algebraic theory**.

Definition 14 $((\Sigma, E)$–algebras). Given a signature Σ and a set of equations E over this signature, a (Σ, E)–**algebra** is a set A along with operations $f^A : A^n \to A$ for all $f : n \in \Sigma$ such that the pair of terms in E are always equal when the dummy variables are quantified over all of A. We usually denote Σ^A for the set operations f^A.

Examples 15. As is suggested by the terminology, the common algebraic structures can be define with simple algebraic theories.

1. We can define a monoid as an algebra for the signature $\{\cdot : 2, 1 : 0\}$ and the equations $x \cdot (y \cdot z) = (x \cdot y) \cdot z, 1 \cdot x = x, x \cdot 1 = x$.

2. Adding the unary operation $(-)^{-1}$ and the equations $x \cdot x^{-1} = 1$ and $x^{-1} \cdot x = 1$, we obtain the theory of groups.

3. Adding the equation $x \cdot y = y \cdot x$ yields the theory of abelian groups.

4. With the signature $\{+ : 2, \cdot : 2, 1 : 0, 0 : 0\}$, we can add the abelian group equations for the operation $+$ (identity is 0), the monoid equations for $\cdot$ (identity is 1) and the distributivity equation $x \cdot (y + z) = (x \cdot y) + (x \cdot z)$ and thus obtain the theory of rings.

We also have homomorphisms.

Definition 16 ((Σ, E)–algebra homomorphisms)**.** Given two (Σ, E)–algebras A and B, a **homomorphism** between them is a map $h : A \to B$ commuting with all operations in Σ, that is $\forall f : n \in \Sigma, h \circ f^A = f^B \circ h^n$.

The category of (Σ, E)–algebras and their homomorphisms (with the obvious composition and identities) is denoted $\mathrm{Alg}(\Sigma, E)$.

Example 17 (Σ_S, E_S)**.** Recall in Example 12 that $\mathcal{P}_{ne}$–algebras correspond with semilattices. Formally, the theory of semilattices has a single binary operation $\Sigma_S = \{\oplus : 2\}$ satisfying the following equations in E_S:

$$x \oplus x = x \qquad\qquad\qquad I$$
$$x \oplus y = y \oplus x \qquad\qquad\qquad C$$
$$(x \oplus y) \oplus z = x \oplus (y \oplus z). \qquad\qquad\qquad A$$

Up to a couple of missing functoriality argument, we have shown that the categories $\mathbf{Set}^{\mathcal{P}_{ne}}$ and $\mathrm{Alg}(\Sigma_S, E_S)$ are isomorphic. We say that (Σ_S, E_S) is an **algebraic presentation** of the monad $\mathcal{P}_{ne}$.

It turns out all algebraic theories present at least one monad.

Definition 18 (Term monad)**.** Let (Σ, E) be an algebraic theory, one can assign to any set X, the set $T_{\Sigma,E}(X)$ of terms in $T_\Sigma(X)$ modulo the equations in E. This can be extended to functions $f : X \to Y$, by variable substitution, i.e.: $T_\Sigma(f)$ acts on a term t by replacing all occurrences of $x \in X$ by $f(x) \in Y$ and $T_{\Sigma,E}(f)$ acts on equivalence classes by $[t] \mapsto [T_\Sigma(f)(t)]$. We obtain a functor $T_{\Sigma,E}$ on which we can put a monad structure.

The unit map is obvious because any element of X is a Σ–term, thus $\eta_X : X \to T_{\Sigma,E}(X)$ maps x to the equivalence class containing the term x. The multiplication map is derived from the fact that applying operations in Σ to Σ–terms yields Σ–terms. More explicitly, μ_X is a flattening operation defined recursively by

$$\forall t \in T_\Sigma(X), \mu_X([[t]]) = [t]$$
$$\forall f : n \in \Sigma, t_1, \ldots, t_n \in T_\Sigma T_{\Sigma,E}(X), \mu_X([f(t_1, \ldots, t_n)]) = [f(\mu_X([t_1]), \ldots, \mu_X([t_n]))]$$

One can show that $\mathbf{Set}^{T_{\Sigma,E}}$ is the category of (Σ, E) algebras.

Unfortunately, the terms monads are not very simple to work with and it is often desirable to find other simpler monads which are presented by the same theory or conversely to find an algebraic presentation for a given monad.

Examples 19.   1. The algebraic theory for presenting $\mathcal{D}$ is called the theory of **convex algebras** and is denoted $(\Sigma_{\mathsf{CA}}, E_{\mathsf{CA}})$, it consists of a binary operation $+_p : 2$ for any $p \in (0,1)$ which is meant to represent a choice between the two terms in the operation, the left one being chosen with probability p and the second one with probability $1 - p$. There are three equations in the theory that morally ensure that terms representing the same choice are equal.

$$
\begin{aligned}
x +_p x &= x & I_p&: \text{idempotence} \\
x +_p y &= y +_{\overline{p}} x & C_P&: \text{skew-commutativity} \\
(x +_q y) +_p z &= x +_{pq} (y +_{\frac{p\overline{q}}{\overline{pq}}} z) & A_p&: \text{skew-associativity}
\end{aligned}
$$

These equations are necessary for every distribution in $\mathcal{D}X$ to correspond uniquely to an equivalence class in $T_{\Sigma_{\mathsf{CA}}, E_{\mathsf{CA}}}(X)$.

2. The monad $(\cdot + \mathbf{1})$ is particular because it is really simple and combines very well with other monads.

   Proposition 20. *For any monad M, there is a monad structure on the composition $M(\cdot + \mathbf{1})$. Moreover, if M is presented by (Σ, E) the monad $M(\cdot + \mathbf{1})$ is presented by $(\Sigma \cup \{* : 0\}, E)$, that is, the new theory only has an additional constant which is neutral with respect to the equations.*

   We often qualify theories with an added constant as **pointed**. For instance, the theories presented by $\mathcal{P}_{\mathrm{ne}}(\cdot + \mathbf{1})$ and $\mathcal{D}(\cdot + \mathbf{1})$ are those of **pointed semilattices** (PS) and **pointed convex algebras** (PCA) respectively.

Remark 21 (Lawvere's way)*.* There is another way to do universal algebra *more categorically* still very much linked to monads: Lawvere theories. Algebras over a Lawvere theory are defined more abstractly using the categorical language and on this account, they enjoy straightforward generalization through enrichment or lifting to higher order categories.

3   POV: Computer Programs

In this section, we will develop on an original idea by Eugenio Moggi that monads are suitable models for a general notion of *computation*. In the sequel, we will use the terms *type* and *set* interchangeably.

Moggi gave a justification for using monads in computer science (particularly in programming semantics) via the informal intuition of *computational types*. For a type A, the computational type of A should contain all computations which return a value of type A. It is intended for the interpretation of *computation* to be made explicit by an instance of a monad. In most cases, it can be thought of as a piece of code which returns some value, but for now, we start by building the intuition in an abstract sense.

Let MA denote the computational type of A and MMA the computational type of MA, that is computations returning values which are themselves computations of type A. The following items should coincide with our intuition of computation.

1. For any $x \in A$, there is a trivial computation return $x \in MA$.

2. For any $C \in MMA$, we can reduce C to flatten$(C) \in MA$ which executes C and the computation returned by C to obtain a final return value of type A.

3. If $C \in MA$, then flatten$($return $C) = C$.

4. If $C \in MA$ and $C' \in MMA$ does the same computation as C but instead of returning a value x, it returns the computation return x, then flatten$(C') = C$.

5. If $MMMA$ is the computational type of MMA and $C \in MMMA$, then there are two ways to flatten C. First, there is the computation C_1 which executes C and executes the returned computation (of type MMA) to obtain a final value of type MA, hence $C_1 \in MMA$ and flatten$(C_1) \in MA$. Second, C_2 executes C and flattens the returned computation to obtain a final value of type MA, C_2 is also of type MMA and flatten$(C_2) \in MA$. These two operations should yield the same result.

Now, a monad M is a description of computational types that is general, namely, for any type A, the monad M gives a type MA behaving as expected. You can check that $x \mapsto$ return x is the unit of this monad and flatten is the multiplication.

Examples 22. Here, we list more examples commonly used in computer science.

List monad: For any set X, let $L(X)$ denote the set of all finite lists whose elements are chosen in X. This is a functor that sends a function $f : X \to Y$ to its extension on lists $L(f) : L(X) \to L(Y)$ which applies f to all elements on the list (in lots of programming languages, one writes $L(f) := \mathtt{map}(f, -)$).Then, we can put a monad structure on L. The unit maps send an element $x \in X$ to the list containing only that element: $\eta_X = x \mapsto [x]$. The multiplication maps concatenate all the lists in a lists of lists: $\mu_X = [\ell_1, \ldots, \ell_n] \mapsto \ell_1 \ell_2 \cdots \ell_n$. It is easy to check diagrams (4) to (5) commute.

Termination: In order to model computations that might terminate with no output, the monad $(\cdot + \mathbf{1})$ is often used. For any type X, the type $X + \mathbf{1}$ has all the values of type X and an additional termination value denoted $*$. The behavior of the unit and multiplication of the monad can be interpreted as the fact that the stage of the computation that leads to a termination is irrelevant. This monad is also known as the Maybe monad.

Non-deterministic choice: The model for nondeterministic choice is given by the monad $\mathcal{P}_{\mathrm{ne}}$. The elements of $S \in \mathcal{P}_{\mathrm{ne}}(X)$ are seen as the possible outcomes of a nondeterministic choice. The unit is basically viewing a deterministic choice as a nondeterministic choice. The multiplication reduces the number of choices without changing the behavior. For instance, consider a process that nondeterministically chooses between two boxes containing two coins each and then chooses a coin in the box. By simply observing the final choice, we would not be able to distinguish it from a process that nondeterministically chooses between the four coins from the start.

Probabilistic choice: In the same vein, probabilistic choice can be interpreted with the monad $\mathcal{D}$ of finitely supported distributions.

Exceptions: As a generalization of termination, we can put a monad structure on the functor $(\cdot + E)$ where E is a set of exceptions that the computation can raise.

This view sheds light on one important features of monads we have not yet explored. If M and $\widehat{M}$ are monads describing computational effects, it is natural to ask for a way

to combine them. Indeed, it does not seem too ambitious to have a model for programs which, for instance, make nondeterministic choices and also might terminate with no output. It turns out there is a very useful tool to deal with this at the level of monads.

Definition 23 (Monad distributive law). Let (M, η, μ) and $(\widehat{M}, \widehat{\eta}, \widehat{\mu})$ be two monads on **C**, a natural transformation $\lambda : M\widehat{M} \Rightarrow \widehat{M}M$ is called a **monad distributive law of M over $\widehat{M}$** if it makes (18), (19) commute.

$$
\begin{array}{ccc}
M & \xrightarrow{M\widehat{\eta}} & M\widehat{M} \xleftarrow{\eta\widehat{M}} \widehat{M} \\
& \searrow{\widehat{\eta}M} & \Big\Vert\lambda \quad \swarrow{\widehat{M}\eta} \\
& & \widehat{M}M
\end{array}
\tag{18}
$$

$$
\begin{array}{ccccc}
MM\widehat{M} & \xrightarrow{\mu\widehat{M}} & M\widehat{M} & \xleftarrow{M\widehat{\mu}} & M\widehat{M}\widehat{M} \\
M\lambda\Big\Vert & & \Big\Vert\lambda & & \Big\Vert\lambda\widehat{M} \\
M\widehat{M}M \xrightarrow{\lambda M} \widehat{M}MM & \xrightarrow{\widehat{M}\mu} & \widehat{M}M & \xleftarrow{\widehat{\mu}M} \widehat{M}\widehat{M}M & \xleftarrow{\widehat{M}\lambda} \widehat{M}M\widehat{M}
\end{array}
\tag{19}
$$

Proposition 24. *If $\lambda : M\widehat{M} \Rightarrow \widehat{M}M$ is a monad distributive law, then the composite $\overline{M} = \widehat{M}M$ is a monad with unit $\overline{\eta} = \widehat{\eta} \diamond \eta$ and multiplication $\overline{\mu} = (\widehat{\mu} \diamond \mu) \cdot \widehat{M}\lambda M$.*

Proof. We have to show that the following instances of (4) and (5) commute.

$$
\begin{array}{ccc}
\overline{M} & \xrightarrow{\overline{M}(\widehat{\eta}\diamond\eta)} & \overline{M}^2 \xleftarrow{(\widehat{\eta}\diamond\eta)\overline{M}} \overline{M} \\
& \searrow & \Big\Vert\widehat{M}\lambda M \\
\mathbb{1}_{\overline{M}} & & \widehat{M}^2M^2 \quad \mathbb{1}_{\overline{M}} \\
& \searrow & \Big\Vert\widehat{\mu}\diamond\mu \quad \swarrow \\
& & \overline{M}
\end{array}
\tag{20}
$$

$$
\begin{array}{ccc}
\overline{M}^3 & \xrightarrow{\overline{M}\widehat{M}\lambda M} \overline{M}\widehat{M}^2M^2 \xrightarrow{\overline{M}(\widehat{\mu}\diamond\mu)} & \overline{M}^2 \\
\widehat{M}\lambda M\overline{M}\Big\Vert & & \Big\Vert\widehat{M}\lambda M \\
\widehat{M}^2M^2\overline{M} & & \widehat{M}^2M^2 \\
(\widehat{\mu}\diamond\mu)\overline{M}\Big\Vert & & \Big\Vert\widehat{\mu}\diamond\mu \\
\overline{M}^2 & \xrightarrow{\widehat{M}\lambda M} \widehat{M}^2M^2 \xrightarrow{\widehat{\mu}\diamond\mu} & \overline{M}
\end{array}
\tag{21}
$$

For the left part of (20), we have the following diagram, the justifications of each part is given below with what diagram has to be considered and what functors should be applied to it (recall that acting on the diagrams does not affect commutativity). The no-

tation ([4]).L (resp. .R) means only the left (resp. right) part of the diagram is considered.


$$(22)$$
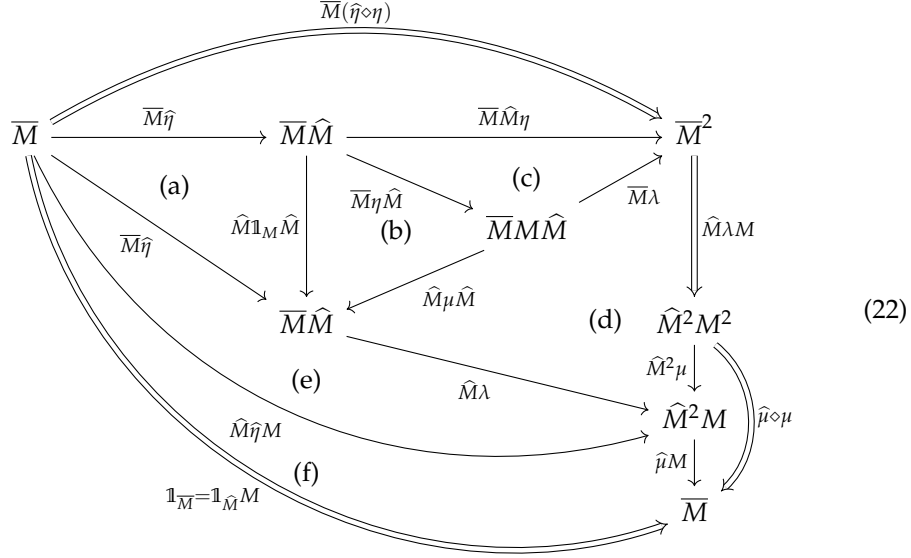
(a) $\widehat{M}\mathbb{1}_M\widehat{M}$ is the identity transformation.

(b) Act on ([4]).L with $\widehat{M}$ on the left and right.

(c) Act on ([18]).R with $\overline{M}$ on the left.

(d) Act on ([19]).L with $\widehat{M}$ on the left.

(e) Act on ([18]).L with $\widehat{M}$ on the left.

(f) Act on ([4]) with M on the right.

Without a diagram, the derivation is this (we use ; to denote the opposite of $\circ$, i.e.: composition in the order read on the diagram):

$$
\begin{aligned}
\overline{M}(\widehat{\eta}\diamond\eta);\widehat{M}\lambda M;\widehat{\mu}\diamond\mu &= \overline{M}\widehat{\eta};\overline{M}\widehat{M}\eta;\widehat{M}\lambda M;\widehat{M}^2\mu;\widehat{\mu}M && \text{def of } \diamond \\
&= \overline{M}\widehat{\eta};\overline{M}\eta\widehat{M};\overline{M}\lambda;\widehat{M}\lambda M;\widehat{M}\widehat{M}\mu;\widehat{\mu}M && \overline{M}(18).\text{R} \\
&= \overline{M}\widehat{\eta};\overline{M}\eta\widehat{M};\widehat{M}\mu\widehat{M};\widehat{M}\lambda;\widehat{\mu}M && \widehat{M}(19).\text{L} \\
&= \overline{M}\widehat{\eta};\widehat{M}\mathbb{1}_M\widehat{M};\widehat{M}\lambda;\widehat{\mu}M && \widehat{M}(4).\text{L}\widehat{M} \\
&= \overline{M}\widehat{\eta};\widehat{M}\lambda;\widehat{\mu}M && \\
&= \widehat{M}\widehat{\eta}M;\widehat{\mu}M && \widehat{M}(18).\text{R} \\
&= \mathbb{1}_{\widehat{M}}M = \mathbb{1}_{\overline{M}} && (4).\text{L}M
\end{aligned}
$$

For the right part of ([20]), the derivation is very similar.

For ([21]), we do the same thing.
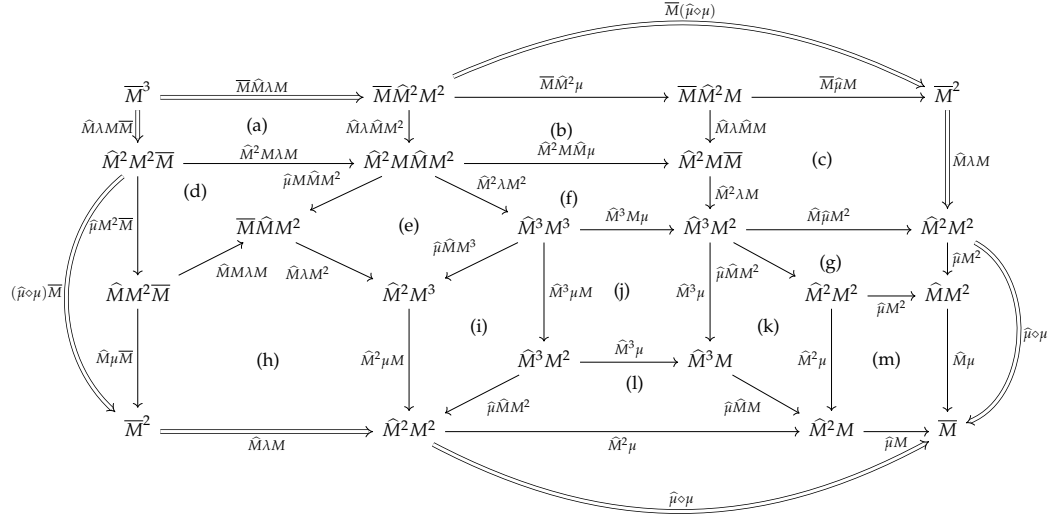
(a) Def of $\widehat{M}\lambda \diamond \lambda M$.

(b) Def of $\widehat{M}\lambda\widehat{M} \diamond \mu$.

(c) Apply $\widehat{M}(\cdot)M$ to (19).R.

(d) Def of $\widehat{\mu} \diamond M\lambda M$.

(e) Def of $\widehat{\mu} \diamond \lambda M^2$.

(f) Def of $\widehat{M}^2\lambda \diamond \mu$.

(g) Apply $(\cdot)M^2$ to associativity of $\widehat{\mu}$ (5).

(h) Apply $\widehat{M}(\cdot)M$ to (19).L.

(i) Def of $\widehat{\mu}\widehat{M} \diamond \mu M$.

(j) Apply $\widehat{M}^3$ to associativity of μ (5).

(k) Def of $\widehat{\mu}\widehat{M} \diamond \mu$.

(l) Same as (k): Def of $\widehat{\mu}\widehat{M} \diamond \mu$.

(m) Def of $\widehat{\mu} \diamond \mu$.

$\square$

Corollary 25. *If* **C** *has (binary) coproducts and a terminal object* **1** *and* M *is a monad, then* $M(\cdot + \mathbf{1})$ *is also monad.*

Proof. We will exhibit a monad distributive law of M over $(\cdot + \mathbf{1})$. We claim

$$\iota_X : MX + \mathbf{1} \to M(X+1) = [M(\mathsf{inl}^{X+\mathbf{1}}), \eta_{X+\mathbf{1}} \circ \mathsf{inr}^{X+\mathbf{1}}]$$

is a monad distributive law $\iota : (\cdot + \mathbf{1})M \Rightarrow M(\cdot + \mathbf{1})$. Then, it follows by Proposition 24. $\square$

Example 26 (Rings)**.** Consider the term monads for the theory of monoids and abelian groups $T_{\mathbf{Mon}}$ and $T_{\mathbf{Ab}}$. You can check that they are the monads induced by the free-forgetful adjunctions between **Mon** and **Set** and **Ab** and **Set**. Also, $T_{\mathbf{Mon}}$ is the same thing as the list monad. Call the binary operation of $T_{\mathbf{Mon}}$ and $T_{\mathbf{Ab}}$ the product and sum respectively.

Then, by identifying products of sums (elements of $T_{\mathbf{Mon}}T_{\mathbf{Ab}}X$) with sums of products (elements of $T_{\mathbf{Ab}}T_{\mathbf{Mon}}X$) by *distributing* the product over the sum as we are used to do with, say, real numbers, we obtain a monad distributive law of $T_{\mathbf{Mon}}$ over $T_{\mathbf{Ab}}$. The resulting composite monad $T_{\mathbf{Ab}}T_{\mathbf{Mon}}$ is the term monad for the theory of rings. The term distributive law comes from this example.

Remark 27. It is not always possible to combine monads in such a natural way. For instance, it was shown that no distributive law exist between $\mathcal{P}_{\mathrm{ne}}$ and $\mathcal{D}$ and even that no monad structure can exist on $\mathcal{P}_{\mathrm{ne}}\mathcal{D}$ or $\mathcal{D}\mathcal{P}_{\mathrm{ne}}$. Thus, modelling combined probabilistic and nondeterministic effects has been quite a hard endeavor and is still an active area of research I discovered in an internship with Matteo Mio and Valeria Vignudelli at ENS de Lyon last summer.

If you are looking for more applications of this perspective on monads and especially if you enjoyed the assignment on Brzozowski's algorithm, I suggest you look into the paper *Generalizing Determinization From Automata to Coalgebras* available at `https://arxiv.org/abs/1302.1046`.

4 Exercises

1. Show that the triple $(\mathcal{D}, \eta, \mu)$ described in Example 5.3 is a monad.

2. Show that the Kleisli category of the powerset monad is the category **Rel** of relations.

3. Show that ι defined in the proof of Corollary 25 is a monad distributive law.

4. Show Proposition 20 with the monad structure on $M(\cdot + \mathbf{1})$ given in Corollary 25.