Presenting Distributive Laws

Marcello M. Bonsangue[1,3], Helle H. Hansen[2,3,⋆], Alexander Kurz[4], and Jurriaan Rot[1,3,⋆⋆]

[1] LIACS – Leiden University, Netherlands
[2] ICIS/IS – Radboud University Nijmegen, Netherlands
[3] Formal Methods – Centrum Wiskunde & Informatica, Amsterdam, Netherlands
[4] Dep. of Computer Science – University of Leicester, United Kingdom

Abstract. Distributive laws of a monad $\mathcal{T}$ over a functor F are categorical tools for specifying algebra-coalgebra interaction. They proved to be important for solving systems of corecursive equations, for the specification of well-behaved structural operational semantics and, more recently, also for enhancements of the bisimulation proof method. If $\mathcal{T}$ is a free monad, then such distributive laws correspond to simple natural transformations. However, when $\mathcal{T}$ is not free it can be rather difficult to prove the defining axioms of a distributive law. In this paper we describe how to obtain a distributive law for a monad with an equational presentation from a distributive law for the underlying free monad. We apply this result to show the equivalence between two different representations of context-free languages.

1 Introduction

The combination of algebraic structure and observable behaviour is fundamental in computer science. Examples include the operational models of structural operational semantics [1], denotational models of programming languages [21], finite stream circuits [12], linear and context-free systems of behavioural differential equations [16,22], and many types of automata such as nondeterministic and weighted automata [18].

In the categorical treatment of these examples, the algebraic structure is encoded by a monad $\mathcal{T} = \langle T, \eta, \mu \rangle$, and the system behaviour by coalgebras for a functor F. Often it is desirable that the algebraic and coalgebraic structure is compatible in some way. A general approach to specifying such algebra-coalgebra interaction is via a distributive law. There are several advantages of this structured approach. A distributive law λ of the monad $\mathcal{T}$ over F induces a $\mathcal{T}$-algebra on the final F-coalgebra of behaviours, yields solutions to corecursive equations $\phi\colon X \to FTX$ [2], and ensures that bisimulation is a congruence. Moreover

⋆ The research of this author has been funded by the Netherlands Organisation for Scientific Research (NWO), Veni project number 639.021.231.

⋆⋆ The research of this author has been funded by the Netherlands Organisation for Scientific Research (NWO), CoRE project, dossier number: 612.063.920.

it yields the soundness of techniques such as bisimulation-up-to-context [2] and extensions thereof [14,15].

Describing a distributive law explicitly and proving that it is one can, however, be rather complicated. Therefore, general methods for constructing distributive laws from simpler ingredients are very useful. An important example of this is given by abstract GSOS [19,2,10] where distributive laws of a free monad $\mathcal{T}$ over a (copointed) functor F are shown to correspond to plain natural transformations, called *abstract GSOS-rules* as they can be seen as specification formats. In [3] it was shown how an abstract GSOS-rule for a free monad $\mathcal{T}$ and functor F can be lifted to one for the functor $F(-)^A$ which describes F-systems with input in A. Another method which works for all monads $\mathcal{T}$, but only for certain polynomial behaviour functors F, produces a distributive law inducing a "pointwise lifting" of $\mathcal{T}$-algebra structure to F-behaviours, cf. [4,6,18].

But many examples do not fit into the abovementioned settings. An important motivating example for this paper is that of context-free grammars, where sequential composition is not a pointwise operation and whose formal semantics satisfies the axioms of idempotent semirings, i.e., the algebraic structure is not free. More generally, one may be interested in a monad arising from a free one by adding equations which one knows to hold in the final coalgebra, without having a particular concrete monad in mind.

The main contribution of this paper is to give a general approach for constructing a distributive law λ' for a monad $\mathcal{T}'$ with an equational presentation, from a distributive law λ for the underlying free monad $\mathcal{T}$. We have no constraints on the behaviour functor F. This λ' is obtained as a certain quotient of λ by the equations E of $\mathcal{T}'$, hence we say that λ' *is presented by a* λ *for the free monad and the equations* E. We show that such quotients exist precisely when the distributive law *preserves the equations* E, which roughly means that congruences generated by the equations are bisimulations. We also discuss how these quotients of distributive laws give rise to quotients of bialgebras, thereby giving a concrete operational interpretation, and a correspondence between solutions to corecursive equations with and without equations. As an illustration and application of our theory, we will show the existence of a distributive law of the monad for idempotent semirings over the deterministic automata functor. This result yields the equivalence betweeen the Greibach normal form representation of context-free languages and the coalgebraic representation via context-free expressions given in[22].

Outline. In Section 2 we recall the notions of monads and algebras, and give a concrete description of monad quotients. In Section 3 we recall distributive laws and their application to solving systems of equations. Then in Section 4 we prove our main results on quotients of distributive laws. In Section 5 we show that such quotients give rise to quotients of bialgebras. Finally in Section 6 we discuss related work, and provide some directions for future work.

Acknowledgements. We thank Neil Ghani, Bart Jacobs, Jan Rutten and Joost Winter for helpful discussions and suggestions.

2 Monads, Algebras and Equations

We start by recalling some basic definitions on monads, algebras, term equations and congruences. We will then proceed to give a concrete description of the quotient monad arising from a free monad and a set of equations. We consider only monads on Set.

A *monad* is a triple $\mathcal{T} = \langle T, \eta, \mu \rangle$ where T is a Set-endofunctor, and $\eta \colon \mathrm{Id} \Rightarrow T$ and $\mu \colon TT \Rightarrow T$ are natural transformations such that $\mu \circ T\eta = 1 = \mu \circ \eta_T$ and $\mu \circ \mu_T = \mu \circ T\mu$. A $\mathcal{T}$*-algebra* is a pair $\langle A, \alpha \rangle$ where A is a set and $\alpha \colon TA \to A$ is a function such that $\alpha \circ \eta_A = 1$ and $\alpha \circ \mu_A = \alpha \circ T\alpha$. A *(*$\mathcal{T}$*-algebra) homomorphism* from $\langle A, \alpha \rangle$ to $\langle B, \beta \rangle$ is a function $f \colon A \to B$ such that $f \circ \alpha = \beta \circ Tf$. The *free* $\mathcal{T}$*-algebra* over a set X is $\langle TX, \mu_X \rangle$. Given any $\mathcal{T}$-algebra $\langle A, \alpha \rangle$ and any function $f \colon X \to A$, there is a unique algebra homomorphism $f^\sharp \colon TX \to A$ such that $f^\sharp(x) = f(x)$ for all $x \in X$, given by $\alpha \circ Tf$.

Let $\langle T, \eta, \mu \rangle$ and $\langle K, \theta, \nu \rangle$ be monads. A *monad map* is a natural transformation $\sigma \colon T \Rightarrow K$ such that the following diagram commutes:

$$\begin{array}{ccccc} \mathrm{Id} & \xrightarrow{\eta} & T & \xleftarrow{\mu} & TT \\ & \searrow^{\theta} & \downarrow{\scriptstyle\sigma} & & \downarrow{\scriptstyle\sigma\sigma} \\ & & K & \xleftarrow{\nu} & KK \end{array} \tag{1}$$

where $\sigma\sigma = K\sigma \circ \sigma_T = \sigma_K \circ T\sigma$.

We fix a monad $\langle T, \eta, \mu \rangle$, a set of variables V, and a set of T*-equations (over* V*)* $E \subseteq TV \times TV$. Let $\mathbb{A} = \langle A, \alpha \rangle$ be a $\mathcal{T}$-algebra. We denote by $E_\mathbb{A}$ the relation on A induced by E when quantifying over all valuations $v \colon V \to A$:

$$E_\mathbb{A} = \bigcup_{v \colon V \to A} \{ \langle v^\sharp(s), v^\sharp(t) \rangle \mid \langle s, t \rangle \in E \} \tag{2}$$

By $q_\mathbb{A}$ we denote the coequalizer

$$TE_\mathbb{A} \underset{\pi_2^\sharp}{\overset{\pi_1^\sharp}{\rightrightarrows}} A \xrightarrow{q_\mathbb{A}} A/{\equiv_\mathbb{A}}$$

and we write $\equiv_\mathbb{A}$ for the kernel of $q_\mathbb{A}$, i.e.,

$$s \equiv_\mathbb{A} t \qquad \text{iff} \qquad q_\mathbb{A}(s) = q_\mathbb{A}(t)$$

which is the *least congruence* on $\mathbb{A}$ containing $E_\mathbb{A}$. Indeed $q_\mathbb{A}$ is the quotient map of $\equiv_\mathbb{A}$. For a free algebra $\mathbb{A} = \langle TX, \mu_X \rangle$ we write E_X, $\equiv_X$ and q_X for the corresponding equations, congruence and quotient map, respectively. Moreover we let $T'X = TX/{\equiv_X}$. Note that if T is finitary, the quotient is a $\mathcal{T}$-algebra and q is a $\mathcal{T}$-algebra homomorphism.

We will use that all epis in Set are split, so every quotient map q_X has a section $r_X \colon T'X \rightarrowtail TX$ (which picks representatives) such that $q_X \circ r_X = 1_{TX/\equiv}$. The following basic result will be useful.

Lemma 1. *Let $f\colon A \to B$ be an algebra homomorphism from $\mathbb{A} = \langle A, \alpha\rangle$ to $\mathbb{B} = \langle B, \beta\rangle$. Then for all $x, y \in A$: $x \equiv_{\mathbb{A}} y$ implies $f(x) \equiv_{\mathbb{B}} f(y)$.*

Proof. Let f, $\mathbb{A}$ and $\mathbb{B}$ be as above. We proceed as follows: first, we prove that

$$(f \times f)(E_{\mathbb{A}}) \subseteq E_{\mathbb{B}} \tag{3}$$

and subsequently we show that this extends to the respective congruences, i.e., that

$$(f \times f)(\equiv_{\mathbb{A}}) \subseteq \equiv_{\mathbb{B}} \tag{4}$$

which is what we need to prove.

For (3), suppose $\langle x, y\rangle \in E_{\mathbb{A}}$. By definition of $E_{\mathbb{A}}$ there exists a map $v\colon V \to A$ and $\langle s, t\rangle \in E$ such that $v^\sharp(s) = x$ and $v^\sharp(t) = y$. Define $w\colon V \to B$ as $w = f \circ v$. By definition of $E_{\mathbb{B}}$ then $\langle w^\sharp(s), w^\sharp(t)\rangle \in E_{\mathbb{B}}$. But

$$\begin{aligned} \langle w^\sharp(s), w^\sharp(t)\rangle &= \langle (f \circ v)^\sharp(s), (f \circ v)^\sharp(t)\rangle \\ &= \langle f \circ v^\sharp(s), f \circ v^\sharp(t)\rangle \\ &= \langle f(x), f(y)\rangle \end{aligned}$$

so $\langle f(x), f(y)\rangle \in E_{\mathbb{B}}$ as desired.

Now consider the following diagram:

$$\begin{array}{ccccc} E_{\mathbb{A}} & \overset{\pi_1}{\underset{\pi_2}{\rightrightarrows}} & A & & \\ {\scriptstyle f\times f}\downarrow & & \downarrow{\scriptstyle f} & & \\ E_{\mathbb{B}} & \overset{\pi_1'}{\underset{\pi_2'}{\rightrightarrows}} & B & \xrightarrow{q_{\mathbb{B}}} & B/{\equiv_{\mathbb{B}}} \end{array}$$

where π_1' and π_2' are the projections of $E_{\mathbb{B}}$. By (3) $f \times f$ is well-defined; moreover it holds that $f \circ \pi_i = \pi_i' \circ f \times f$ for $i \in \{1, 2\}$. Recall from Section 2 that $\equiv_{\mathbb{B}}$ is the kernel of $q_{\mathbb{B}}$, so since $E_{\mathbb{B}} \subseteq \equiv_{\mathbb{B}}$ we have in particular $q_{\mathbb{B}} \circ \pi_1' = q_{\mathbb{B}} \circ \pi_2'$, and it follows that the following diagram commutes:

$$E_{\mathbb{A}} \overset{\pi_1}{\underset{\pi_2}{\rightrightarrows}} A \xrightarrow{f} B \xrightarrow{q_{\mathbb{B}}} B/{\equiv_{\mathbb{B}}}$$

Now since $q_{\mathbb{B}}$ and f are algebra homomorphisms, $q_{\mathbb{B}} \circ f$ is as well, so

$$TE_{\mathbb{A}} \overset{\pi_1^\sharp}{\underset{\pi_2^\sharp}{\rightrightarrows}} A \xrightarrow{f} B \xrightarrow{q_{\mathbb{B}}} B/{\equiv_{\mathbb{B}}}$$

commutes, by the fact that there is a unique homomorphism extending $q_{\mathbb{B}} \circ f \circ \pi_i$. Thus, by the universal property of the coequalizer $q_{\mathbb{A}}$ there is a unique $f'\colon A/{\equiv_{\mathbb{A}}} \to B/{\equiv_{\mathbb{B}}}$ such that $f' \circ q_{\mathbb{A}} = q_{\mathbb{B}} \circ f$, and consequently $q_{\mathbb{A}}(s) = q_{\mathbb{A}}(t)$ implies $q_{\mathbb{B}} \circ f(s) = q_{\mathbb{B}} \circ f(t)$, which proves (4). $\square$

Let $f\colon X \to Y$ be any map. Then Tf is an algebra homomorphism from $\langle TX, \mu_X\rangle$ to $\langle TY, \mu_Y\rangle$, so Lemma 1 implies that for any x, y such that $x \equiv_X y$ we have $q_Y(Tf(x)) = q_Y(Tf(y))$. Consequently by the universal property of the coequalizer q_X there is a unique map $T'X \to T'Y$ in the following square:

$$\begin{array}{ccc} TX & \xrightarrow{q_X} & T'X \\ {\scriptstyle Tf}\downarrow & & \downarrow{\scriptstyle \exists!} \\ TY & \xrightarrow{q_Y} & T'Y \end{array} \tag{5}$$

We define $T'f$ to be this uniquely induced map. Since $q_X \circ r_X = 1_{T'X}$, it follows that we may describe $T'f$ concretely as $T'f = q_Y \circ T(f) \circ r_X$. The uniqueness of $T'f$ means that the definition of $T'f$ does not depend on (the particular choice of representatives made by) r_X.

Definition 1 (Quotient monad). *Given a monad $\langle T, \eta, \mu\rangle$ and equations $E \subseteq TV \times TV$, we define the "quotient monad" $\langle T', \eta', \mu'\rangle$ as follows.*

$$\begin{aligned} T'X &= TX/{\equiv_X}, \\ T'(f\colon X \to Y) &= q_Y \circ T(f) \circ r_X, \\ \eta'_X &= q_X \circ \eta_X, \\ \mu'_X &= q_X \circ \mu_X \circ r_{TX} \circ T'(r_X) \end{aligned}$$

◁

In order to show that $\mathcal{T}'$ is, in fact, a monad (Proposition 1), the following result [11, Lemma 2.3.2] will be useful.

Lemma 2 ([11]). *Let T, K be endofunctors, and $\sigma\colon T \Rightarrow K$ a natural transformation with epic components. Let $\eta\colon 1 \to T$, $\mu\colon TT \to T$, $\theta\colon 1 \to K$ and $\nu\colon KK \to K$ be maps (not assumed to be natural transformations) satisfying the diagram* (1)*. If $\langle T, \eta, \mu\rangle$ is a monad then $\langle K, \theta, \nu\rangle$ is a monad as well.*

Proposition 1. *Given any monad $\langle T, \eta, \mu\rangle$, the quotient monad $\mathcal{T}' = \langle T', \eta', \mu'\rangle$ is indeed a monad, and $q\colon T \Rightarrow T'$ is a monad map.*

Proof. We first check that T' is a functor.

$$\begin{aligned} T'(1_X) &= q_X \circ T(1_X) \circ r_X \\ &= q_X \circ 1_{TX} \circ r_X \\ &= q_X \circ r_X \\ &= 1_{T'X} \end{aligned}$$

Let $f\colon X \to Y$ and $g\colon Y \to Z$.

$$\begin{aligned} T'g \circ T'f &= T'g \circ q_Y \circ Tf \circ r_X && (5)_f \\ &= q_Z \circ Tg \circ Tf \circ r_X && (5)_g \\ &= q_Z \circ T(g \circ f) \circ r_X \\ &= T'(g \circ f) \end{aligned}$$

Having verified that T' is a functor, the square (5) says that $q\colon T \Rightarrow T'$ is a natural transformation. Clearly q has epic components. We proceed to show that q, T and T' satisfy diagram (1). For the unit law, we have $\eta' = q \circ \eta$ by definition. In order to proceed with the associativity, we note that since μ_X is a $\mathcal{T}$-algebra homomorphism for any X, from Lemma 1 we obtain that for any $s, t \in TTX$:

$$s \equiv_{TX} t \qquad \text{implies} \qquad \mu_X(s) \equiv_X \mu_X(t) \tag{6}$$

In particular, since $s \equiv_X (r_{TX} \circ q_{TX})(s)$ this means that

$$\mu_X(s) \equiv_X (\mu_X \circ r_{TX} \circ q_{TX})(s) \tag{7}$$

Now

$$\begin{aligned}
q \circ \mu &= q \circ \mu \circ r_T \circ q_T && \text{by (7)}\\
&= q \circ r \circ q \circ \mu \circ r_T \circ q_T && \text{since } q \circ r = 1\\
&= q \circ \mu \circ T(r \circ q) \circ r_T \circ q_T && \text{naturality of } \mu\\
&= q \circ \mu \circ r_T \circ q_T \circ T(r \circ q) \circ r_T \circ q_T && \text{by (7)}\\
&= q \circ \mu \circ r_T \circ T'r \circ T'q \circ q_T && \text{def. of } T'(r \circ q) \text{ and functoriality}\\
&= \mu' \circ T'q \circ q_T && \text{def. of } \mu'
\end{aligned}$$

Now, all of the conditions of Lemma 2 are satisfied in order to conclude that $\langle T', \eta', \mu' \rangle$ is a monad and, of course, q is a monad map. □

The above construction yields a concrete monad $\mathcal{T}'$ given a set of operations and equations. Intuitively, any monad which is isomorphic to $\mathcal{T}'$ is presented by these same operations and equations; this is captured by the following definition.

Definition 2. *Let Σ be an endofunctor, $\mathcal{T}$ the corresponding free monad, $E \subseteq TV \times TV$ a set of equations and $\mathcal{T}'$ the quotient monad. We say that a monad $\mathcal{K} = \langle K, \theta, \nu \rangle$ is* presented by Σ and E *if there is a monad map $i\colon T' \Rightarrow K$ which is a natural isomorphism.* ◁

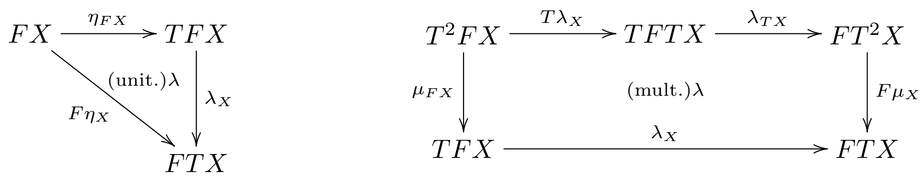
Example 1. The *idempotent semiring monad* is defined by the functor mapping a set X to the set $\mathcal{P}_\omega(X^*)$ of finite languages over X and, for morphisms $f\colon X \to Y$ in Set we define $\mathcal{P}_\omega(f^*)(L) = \bigcup\{f(x_1)\cdots f(x_n) \mid x_1 \cdots x_n \in L\}$. Further, $\eta_X\colon X \to \mathcal{P}_\omega(X^*)$ is given by $\eta_X(x) = \{x\}$ and $\mu_X\colon \mathcal{P}_\omega(\mathcal{P}_\omega(X^*)^*) \to \mathcal{P}_\omega(X^*)$ by $\mu_X(\mathcal{L}) = \bigcup_{L_1 \cdots L_n \in \mathcal{L}}\{w_1 \cdots w_n \mid w_i \in L_i\}$. It is presented by two constants $\mathtt{0}$ and $\mathtt{1}$, two binary operations $+$ and $\cdot$, and the idempotent semiring axioms. The witnessing isomorphism can easily be given based on the observation that every semiring term is equivalent with respect to the idempotent semiring equations to a sum of products of variables. ◁

3 Distributive Laws and Bialgebras

We briefly recall the basic definitions of distributive laws and bialgebras; for a more thorough introduction we refer to [8,2,19].

3.1 Basic Definitions

Let $\mathcal{T} = \langle T, \eta, \mu \rangle$ be a finitary Set-monad, and F a Set-functor. A *distributive law* λ of the monad $\mathcal{T}$ over the functor F is a natural transformation $\lambda\colon TF \Rightarrow FT$ which is compatible with the monad structure, meaning that $\lambda \circ \eta_F = F\eta$ and $\lambda \circ \mu_F = F\mu \circ \lambda_T \circ T\lambda$, i.e., for all X the following diagrams commute:



We recall that every distributive law $\lambda\colon TF \Rightarrow FT$ corresponds to a *lifting* F_λ of F to the category of $\mathcal{T}$-algebras (see, e.g., [7,8]), defined as

$$F_\lambda\langle A, \alpha\rangle = \langle FA, F\alpha \circ \lambda_A\rangle \qquad\qquad F_\lambda(f) = Ff \tag{8}$$

Note that the compatibility of λ_X with μ_X means precisely that λ_X is a $\mathcal{T}$-algebra homomorphism from $\langle TFX, \mu_{FX}\rangle$ to $F_\lambda\langle TX, \mu_X\rangle$.

An F-*coalgebra* is a pair $\langle X, c\rangle$ where X is a set and $c\colon X \to FX$ is a map. An F-*coalgebra morphism* from $\langle X, c\rangle$ to $\langle Y, d\rangle$ is a map $f\colon X \to Y$ such that $d \circ f = Tf \circ c$. A λ-*bialgebra* is a triple $\langle X, \alpha, \beta\rangle$ where $\alpha\colon TX \to X$ is a $\mathcal{T}$-algebra and $\beta\colon X \to FX$ is an F-coalgebra such that $\beta \circ \alpha = F\alpha \circ \lambda_X \circ T\beta$. A *morphism of* λ-*bialgebras* from $\langle X_1, \alpha_1, \beta_1\rangle$ to $\langle X_2, \alpha_2, \beta_2\rangle$ is a function $f\colon X_1 \to X_2$ which is both a $\mathcal{T}$-algebra morphism and an F-coalgebra morphism.

The following results are well known (e.g., [2,8]). If $\langle Z, \zeta\rangle$ is a final F-coalgebra, then a distributive law $\lambda\colon TF \Rightarrow FT$ yields a final λ-bialgebra $\langle Z, \alpha, \zeta\rangle$ where $\alpha\colon TZ \to Z$ is defined by coinduction from the F-coalgebra $\langle TZ, \lambda_Z \circ T\zeta\rangle$.

We will need the notion of distributive laws of monads over *copointed* functors. A copointed functor is a pair $\langle F, \epsilon\rangle$ where F is an endofunctor and $\epsilon\colon F \Rightarrow \mathrm{Id}$ a natural transformation. A distributive law of $\mathcal{T}$ over $\langle F, \epsilon\rangle$ is a distributive law of $\mathcal{T}$ over F additionally satisfying $\epsilon_T \circ \lambda = T\epsilon$. For any Set-functor F, the *cofree copointed functor generated by* F is the pair $\langle \mathrm{Id} \times F, \pi_1\colon \mathrm{Id} \times F \to \mathrm{Id}\rangle$ where π_1 is the natural left-projection.

When $\mathcal{T}$ is the free monad generated by a signature functor Σ, then distributive laws involving $\mathcal{T}$ can be reduced to "plain" natural transformations using recursion, namely, there is a 1-1 correspondence between distributive laws $\lambda\colon TF \Rightarrow FT$ of $\mathcal{T}$ over F and natural transformations $\rho\colon \Sigma F \Rightarrow FT$ (cf. [2]). Such a ρ corresponds to a specification format of operational rules, and is sometimes referred to as a *simple SOS rule*. Similarly, for cofree copointed functors, if $\mathcal{T}$ is freely generated by Σ, then there is a 1-1 correspondence between distributive laws $\lambda\colon T(\mathrm{Id} \times F) \Rightarrow (\mathrm{Id} \times F)T$ of $\mathcal{T}$ over $\langle \mathrm{Id} \times F, \pi_1\rangle$ and natural transformations $\rho\colon \Sigma(\mathrm{Id} \times F) \Rightarrow FT$ (cf. [10,4]). Such a natural transformation ρ is also referred to as an *abstract GSOS-rule* since it generalises the GSOS-format for labelled transition systems where $F = \mathcal{P}_\omega(-)^A$, cf. [2,19].

3.2 Solutions to Corecursive Equations

An important application of distributive laws is in solving *corecursive equations* which are maps of the type $\phi\colon X \to FTX$ where F is a functor and T is (the functor component of) a monad. These include many interesting and useful structures such as linear and context-free systems of behavioural differential equations [16,22], as well as linear, nondeterministic and weighted automata cf. [4,18]. These are all instances of $\mathcal{T}$*-automata* [4] which have the type $X \to B \times (TX)^A$ where A is a set and B carries a $\mathcal{T}$-algebra $\beta\colon TB \to B$, i.e., in particular, $F = B \times (-)^A$ whose final coalgebra carrier is B^{A^*}.

In the presence of a distributive law $\lambda\colon TF \Rightarrow FT$ one obtains a λ*-coinduction principle* [2] which provides unique solutions in the final λ-bialgebra $\langle Z, \alpha, \zeta\rangle$ to corecursive equations of the form $\phi\colon X \to FTX$. Ordinary coinduction is the special case where $\mathcal{T}$ is the identity monad. Formally, a solution to $\phi\colon X \to FTX$ in a λ-bialgebra $\langle A, \alpha, \beta\rangle$ is a map $f\colon X \to A$ such that

$$\begin{array}{ccccc} X & & \xrightarrow{\quad f \quad} & & A \\ {\scriptstyle\phi}\downarrow & & & & \downarrow{\scriptstyle\beta} \\ FTX & \xrightarrow{FTf} & FTA & \xrightarrow{F\alpha} & FA \end{array} \tag{9}$$

commutes. More precisely, λ-coinduction is coinduction in the category of λ-bialgebras, and we have the following fact.

Proposition 2 (Lemmas 4.3.3 and 4.3.4 of [2]). *Let $\phi\colon X \to FTX$ be a corecursive equation. Taking $\phi^\lambda = F\mu_X \circ \lambda_{TX} \circ T\phi$ then $\langle TX, \mu_X, \phi^\lambda\rangle$ is a λ-bialgebra, and $\eta_X\colon X \to TX$ is a solution of ϕ. Moreover, for any λ-bialgebra $\langle A, \alpha, \beta\rangle$, there is a 1-1 correspondence between solutions of ϕ in $\langle A, \alpha, \beta\rangle$ and λ-bialgebra morphisms from $\langle TX, \mu_X, \phi^\lambda\rangle$ to $\langle A, \alpha, \beta\rangle$.*

A "pointwise distributive law" λ for $\mathcal{T}$-automata can be obtained (cf. [4,6]) by taking $\lambda_X = (\beta \times \mathsf{st}) \circ \langle T\pi_1, T\pi_2\rangle$ where $\mathsf{st}\colon T \circ (-)^A \Rightarrow (-)^A \circ T$ is the strength natural transformation. This λ is named so, since it induces the pointwise extension of $\beta\colon TB \to B$ on B^{A^*}. In the context-free and streams examples below, however, the desired algebraic structure on B^{A^*} uses the convolution product which is not the pointwise extension of the semiring product of B. So for these examples a different λ must be given.

4 Quotients of Distributive Laws

In Section 2 we saw how term equations give rise to quotients of algebras, and we gave an explicit construction of the resulting quotient monad. In this section, we investigate conditions under which distributive laws and equations give rise to quotients of distributive laws.

As before, let Σ be a finitary signature functor generating the free monad $\mathcal{T} = \langle T, \eta, \mu\rangle$, and let E be a set of T-equations with associated T-congruence $\equiv_X$ on terms, and quotient monad $\mathcal{T}' = \langle T', \eta', \mu'\rangle$.

4.1 Distributive Laws over Plain Behaviour Functors

In this subsection, we assume that $\lambda\colon TF \Rightarrow FT$ is a distributive law of $\mathcal{T}$ over a plain behaviour functor F. We will provide a condition on λ and the equations E that ensures that we get a distributive law $\lambda'\colon T'F \Rightarrow FT'$ for the quotient monad. To this end, it is convenient to use the notion of a morphism of distributive laws from [20].

Definition 3. *Let $\langle T, \eta, \mu\rangle$ and $\langle K, \theta, \nu\rangle$ be monads, and let $\lambda\colon TF \Rightarrow FT$ and $\kappa\colon KF \Rightarrow FK$ be distributive laws. A natural transformation $\tau\colon T \Rightarrow K$ is a morphism from λ to κ (notation $\tau\colon \lambda \Rightarrow \kappa$) if τ is a monad morphism and the following square commutes:*

$$\begin{array}{ccc} TF & \xrightarrow{\tau F} & KF \\ {\scriptstyle\lambda}\downarrow & & \downarrow{\scriptstyle\kappa} \\ FT & \xrightarrow{F\tau} & FK \end{array} \tag{10}$$

◁

We note that there are generalisations of the above definition that allow natural transformations between behaviour functors, cf. [20]. For our purposes, we do not need to change the behaviour type.

Definition 4. *We say that $\lambda\colon TF \Rightarrow FT$* preserves (equations in) E *if for all $g\colon V \to FX$, and for all $s, t \in TV$:*

$$s\ E\ t \quad \Rightarrow \quad Fq_X(\lambda_X(Tg(s))) \ = \ Fq_X(\lambda_X(Tg(t))). \tag{11}$$

◁

Equation (11) can be conveniently formulated in terms of relation lifting as

$$s\ E\ t \quad \Rightarrow \quad \lambda_X(Tg(s))\ \overline{F}(\equiv_X)\ \lambda_X(Tg(t)). \tag{12}$$

where the F-lifting of a relation $R \subseteq Y \times Y$ is defined as

$$\overline{F}(R) = \{\langle F\pi_1(u), F\pi_2(u)\rangle \in FY \times FY \mid u \in F(R)\}$$

and noticing that $u\ \overline{F}(\equiv_X)\ v$ iff $Fq_X(u) = Fq_X(v)$.

We can now state our main result.

Theorem 1. *The following are equivalent.*

1. *$\lambda\colon TF \Rightarrow FT$ preserves equations E.*
2. *there is a (unique) distributive law $\lambda'\colon T'F \Rightarrow FT'$ such that the quotient map $q\colon T \Rightarrow T'$ is a morphism of distributive laws from λ to λ'.*

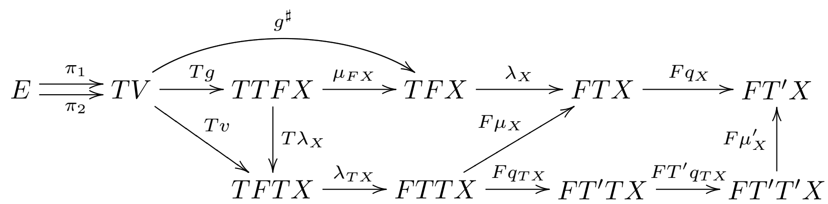
Remark 1. Using that distributive laws correspond to functor liftings on $\mathcal{T}$-algebras (cf. (8)), the distributive law λ' in Theorem 1 exists if and only if the functor F_λ restricts to $\mathcal{T}'$-algebras. A similar statement for the case when F is a monad is made in [11, Corollary 3.4.2].

In order to prove Theorem 1, we first prove a lemma.

Lemma 3. *If $\lambda\colon TF \Rightarrow FT$ preserves equations in E then for all valuations $g\colon V \to TFX$, and for all $s,t \in TV$:*

$$s\;E\;t \quad\Rightarrow\quad \lambda_X(g^\sharp(s))\;\overline{F}(\equiv_X)\;\lambda_X(g^\sharp(t)). \tag{13}$$

Proof. Let $g\colon V \to TFX$ be any valuation. Define $v\colon V \to FTX$ as $v = \lambda_X \circ g$.



The right half of the rectangle commutes by the fact that q is a monad map and functoriality, the left half commutes by compatibility of λ with μ. The lower two paths from E to $FT'TX$ (and thus on to $FT'X$) commute by assumption that E preserves equations. The triangle commutes by definition of v and functoriality, and the upper bubble commutes by definition of $g^\sharp$. Thus we obtain that $Fq_X \circ \lambda_X \circ g^\sharp \circ \pi_1 = Fq_X \circ \lambda_X \circ g^\sharp \circ \pi_2$, which proves (13). □

Proof of Theorem 1. For the implication from 1 to 2, assume that λ preserves E. As a first step, we show that for any X, the following diagram commutes:

$$E_{FX} \underset{\pi_2}{\overset{\pi_1}{\rightrightarrows}} TFX \xrightarrow{\lambda_X} FTX \xrightarrow{Fq_X} FT'X \tag{14}$$

To see this, let X be a set and $\langle u, v\rangle \in E_{FX}$. By definition of E_{FX} there are $s,t \in TV$ with $\langle s,t\rangle \in E$ and a $g\colon V \to TFX$ such that $u = g^\sharp(s)$ and $v = g^\sharp(t)$. As λ preserves E, it follows by Lemma 3 that $Fq_X(\lambda_X(g^\sharp(s))) = Fq_X(\lambda_X(g^\sharp(t)))$, and hence (14) commutes.

Next, we show that (14) extends to the congruence $\equiv_{FX}$, i.e., that

$$TE_{FX} \underset{\pi_2^\sharp}{\overset{\pi_1^\sharp}{\rightrightarrows}} TFX \xrightarrow{\lambda_X} FTX \xrightarrow{Fq_X} FX/{\equiv} \tag{15}$$

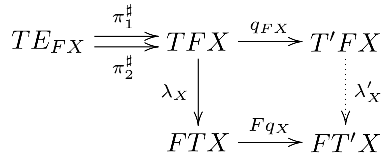
To obtain (15) it suffices to show that $Fq_X \circ \lambda_X$ is a $\mathcal{T}$-algebra homomorphism, since then $\pi_i^\sharp \circ Fq_X \circ \lambda_X$ is a $\mathcal{T}$-algebra homorphism extending $\pi_i \circ Fq_X \circ \lambda_X$, for $i = 1, 2$. Since $\pi_1 \circ Fq_X \circ \lambda_X = \pi_2 \circ Fq_X \circ \lambda_X$ due to (14), and homomorphic extensions are uniqe, we get (15).

We now show $Fq_X \circ \lambda_X$ is a $\mathcal{T}$-algebra homomorphism. Let F_λ be the lifting of F to the category of $\mathcal{T}$-algebras, and recall that λ_X is a $\mathcal{T}$-algebra homomorphism from $\langle TFX, \mu_{FX}\rangle$ to $F_\lambda\langle TX, \mu_X\rangle$ (cf. Section 3.1). Notice also that since q is a monad map, $\langle TX, \mu_X\rangle \xrightarrow{q_X} \langle T'X, \mu'_X \circ q_{T'X}\rangle$ is a $\mathcal{T}$-algebra homomorphism, and thus by applying the lifting F_λ we obtain a $\mathcal{T}$-algebra homomorphism

$$F_\lambda\langle TX, \mu_X\rangle \xrightarrow{Fq_X} F_\lambda\langle T'X, \mu'_X \circ q_{T'X}\rangle$$

Thus $Fq_X \circ \lambda_X$ is a $\mathcal{T}$-algebra homomorphism from the free $\mathcal{T}$-algebra $\langle TFX, \mu_{FX} \rangle$.

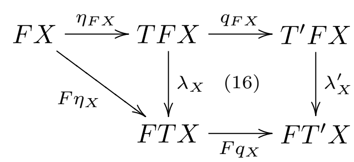
Finally by the universal property of the coequalizer q_{FX} there is a (unique) function $\lambda' \colon T'FX \to FT'X$ such that $\lambda' \circ q_{FX} = Fq_X \circ \lambda_X$:



(16)

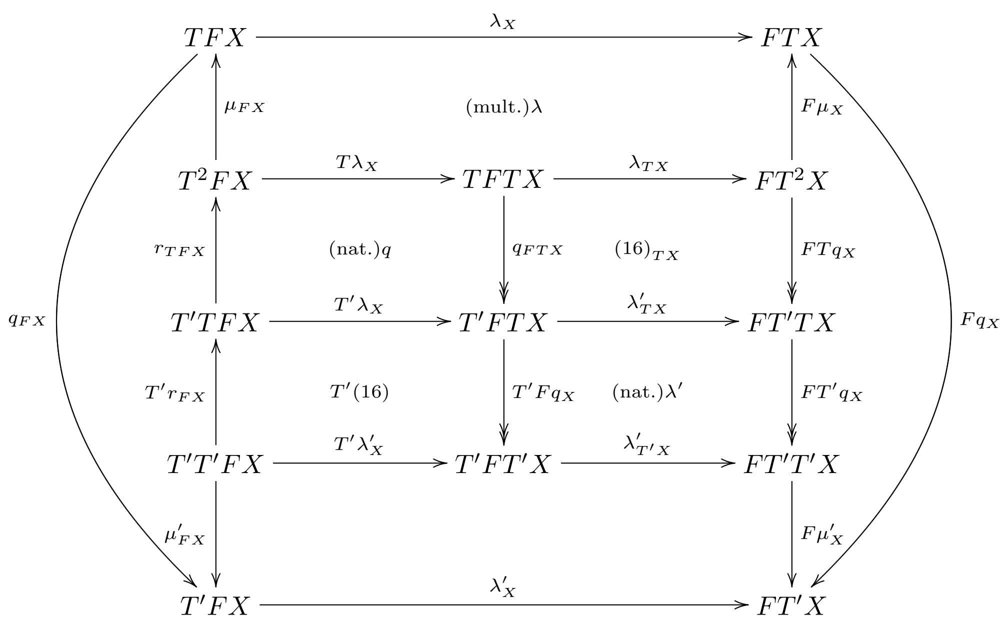
The naturality of λ' follows from the definition of $\mathcal{T}'$ and the naturality of λ and q. Due to the commutativity of the square in (16), q is a morphism of distributive laws from λ to λ' once we show that λ' is, in fact, a distributive law.

The unit law for λ' holds due to the unit law for λ and (16):



(17)

Multiplication law for λ':



(18)

The small upper-left square commutes by naturality of q and precomposing with r_{TFX}. Similarly, the small lower-left square commutes by applying T' to (16) and precomposing with $T'r_{FX}$. The outer crescents commute since q is a monad morphism, and the outermost part due to (16).

The implication from 2 to 1 follows from the fact that (16) implies (14). $\square$

As a corollary we obtain the analogue of Theorem 1 for monads presented by operations and equations.

Corollary 1. *Suppose $\mathcal{K} = \langle K, \theta, \nu \rangle$ is presented by operations Σ and equations E with natural isomorphism $i\colon T' \Rightarrow K$, and suppose we have a distributive law $\lambda\colon TF \Rightarrow FT$ of $\mathcal{T}$ over F. Then there exists a unique distributive law $\kappa\colon KF \to FK$ of $\mathcal{K}$ over F such that $i \circ q\colon \lambda \Rightarrow \kappa$ is a morphism of distributive laws.*

Proof. The distributive law $\kappa\colon KF \to KF$ is defined as $\kappa = Fi \circ \lambda \circ i^{-1}$. The proof proceeds by checking that κ indeed satisfies the defining axioms of a distributive law, which is an easy but tedious exercise. □

Theorem 1 says that if λ preserves the equations in E, then we can *present λ' as "λ modulo equations"*. We illustrate this with an example.

Example 2 (Stream calculus). Behavioural differential equations are used extensively in [16,17] to define streams and stream operations. Here, the behaviour functor is $F(X) = \mathbb{R} \times X$ whose final coalgebra $\langle \mathbb{R}^\omega, \zeta \rangle$ consists of streams over the real numbers together with the map $\zeta(\sigma) = \langle \sigma(0), \sigma' \rangle$ which maps a stream σ to its initial value $\sigma(0)$ and derivative σ'.

The following behavioural specification defines the constant streams $[a] = (a, 0, 0, \ldots)$ for all $a \in \mathbb{R}$, $\mathtt{X} = (0, 1, 0, 0, \ldots)$, pointwise addition and convolution product of streams. We point out that the convolution product is defined here by a simple stream SOS-rule rather than a stream GSOS-rule, which is used in [16,17]. We explain this choice at the end of the example.

$$\begin{array}{rll} [a](0) = a, & [a]' = [0], & \forall a \in \mathbb{R} \\ \mathtt{X}(0) = 0, & \mathtt{X}' = [1], & \\ (\sigma + \tau)(0) = \sigma(0) + \tau(0), & (\sigma + \tau)' = \sigma' + \tau', & \\ (\sigma \times \tau)(0) = \sigma(0) \cdot \tau(0), & (\sigma \times \tau)' = (\sigma' \times [\tau(0)]) + (\sigma' \times \mathtt{X} \times \tau') + ([\sigma(0)] \times \tau') & \end{array}$$

The signature functor is thus $\Sigma(X) = \mathbb{R} + 1 + (X \times X) + (X \times X)$, and the above specification induces a distributive law $\lambda\colon TF \Rightarrow FT$. We would like to apply Theorem 1 to obtain a distributive law λ' for the quotient monad $\mathcal{T}'$ arising from $\mathcal{T}$ and E. Let E consist of the following axioms where $V = \{v, u, w\}$ and $a, b \in \mathbb{R}$:

$$\begin{array}{lll} (v + u) + w = v + (u + w) & [0] + v = v & v + u = u + v \\ (v \times u) \times w = v \times (u \times w) & [1] \times v = v & v \times u = u \times v \\ v \times (u + w) = (v \times u) + (v \times w) & [0] \times v = [0] & \\ [a + b] = [a] + [b] & [a \cdot b] = [a] \times [b] & \end{array} \tag{19}$$

E consists of the *commutative* semiring axioms together with axioms stating the inclusion of the underlying semiring of the reals. We show that λ preserves E. Let $g\colon V \to FX$ be arbitrary and suppose $g(v) = \langle a, x \rangle, g(u) = \langle b, y \rangle, g(z) = \langle c, z \rangle$. First note that for $F = \mathbb{R} \times \mathrm{Id}$, $\langle r_1, t_1 \rangle \; \overline{F}(\equiv_X) \; \langle r_2, t_2 \rangle$ iff $r_1 = r_2$ and $t_1 \equiv_X t_2$. It is straightforward to check preservation of the axioms that only concern addition, as well as of $[1]\times v = v$, $[0]\times v = [0]$ and $v\times u = u\times v$. We show that $[a\cdot b] = [a]\times[b]$ is preserved:

$$\lambda_X([a] \times [b]) \;=\; \langle a \cdot b, [0] \times [b] + [0] \times \mathtt{X} \times [0] + [a] \times [0]\rangle \;\overline{F}(\equiv_X)\; \langle a \cdot b, [0]\rangle = \lambda_X([a \cdot b])$$

We check that λ preserves the distribution axiom:

$$\begin{aligned}
&\lambda_X(\langle a, x\rangle \times (\langle b, y\rangle + \langle c, z\rangle)) \\
&\quad = \langle a \cdot (b + c), (x \times [b + c]) + (x \times X \times (y + z)) + [a] \times (y + z)\rangle \\
&\quad \overline{F}(\equiv_X)\ \langle a \cdot (b + c), (x \times [b + c]) + (x \times X \times y) + (x \times X \times z) + \\
&\qquad\qquad ([a] \times y) + ([a] \times z)\rangle \\
&\quad \overline{F}(\equiv_X)\ \langle (a \cdot c) + (b \cdot c), (x \times [b]) + (x \times X \times y) + ([a] \times y) + \\
&\qquad\qquad (x \times [c]) + (x \times X \times z) + ([a] \times z)\rangle \\
&\quad = \\
&\lambda_X((\langle a, x\rangle \times \langle b, y\rangle) + (\langle a, x\rangle \times \langle c, z\rangle))
\end{aligned}$$

Note that we used $[a + b] = [a] + [b]$. Similarly, preservation of $\times$-associativity can be verified, and it uses the axiom $[a\cdot b] = [a]\times[b]$. We have thus shown that λ preserves E, and it follows, in particular, that $\langle \mathbb{R}^\omega, +, \times, [0], [1]\rangle$ is a commutative semiring. This was shown directly in [17], but the proof uses bisimulation-up-to as well as the fundamental theorem of stream calculus, which cannot be added as an equation. In our approach we construct a distributive law, and obtain not only this result but also the soundness of the bisimulation-up-to technique [15], and the existence of unique solutions to corecursive equations $\phi\colon X \to FT'X$ (see Section 3.2).

The derivative of the convolution product is usually (cf. [16,17]) specified as:

$$(\sigma \times \tau)' = (\sigma' \times \tau) + ([\sigma(0)] \times \tau') \tag{20}$$

which corresponds to a stream GSOS-rule $\Sigma(\mathrm{Id} \times \mathbb{R} \times \mathrm{Id}) \Rightarrow \mathbb{R} \times T(-)$, and thus to a distributive law over the cofree copointed functor. However, with this definition, we could not show that the commutativity of $\times$ is preserved although all other axioms remain preserved. Hence a given λ does not necessarily satisfy all equations that are valid on the final F-coalgebra. ◁

In the above example, we did not have a concrete monad in mind; we simply considered a free monad and a set of equations. In Example 4 below we give an example for the concrete idempotent semirings monad.

Remark 2. The concrete proof method for preservation of equations bears a close resemblance to *bisimulation up to congruence* [15], in that one must show that for every pair in E_{FX} its derivatives are related by the least congruence $\equiv_X$ instead of E_X.

Example 3. We have seen in the discussion of (20) that equations that hold in the final coalgebra are not necessarily preserved by λ. Now we give another concrete example of this fact. This example again concerns stream systems, i.e., coalgebras for the functor $FX = \mathbb{R} \times X$. We define the constant stream of zeros by three different constants n_1, n_2 and n_3 by the following behavioural differential equations:

$$n_1(0) = 0,\, n_1' = n_1 \qquad n_2(0) = 0,\, n_2' = n_3 \qquad n_3(0) = 0,\, n_3' = n_3$$

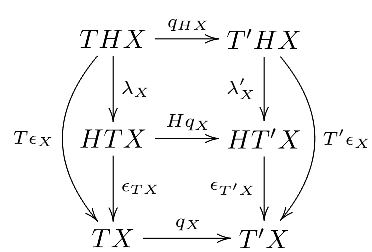
The corresponding signature functor is thus $\Sigma X = 1+1+1$, and the above specification gives rise to a distributive law $\lambda\colon TF \Rightarrow FT$ where T is (the functorial component of) the free monad over Σ. Now consider the equation $n_1 = n_2$; this clearly holds when interpreted in the final coalgebra. However, this equation is not preserved by λ. To see this, notice that $\lambda(n_1) = \langle 0, n_1\rangle$ and $\lambda(n_2) = \langle 0, n_3\rangle$, but $n_1 \not\equiv_X n_3$, so $\lambda(n_1)$ and $\lambda(n_2)$ are not related by $\overline{F}(\equiv_X)$. ◁

4.2 Distributive Laws over Copointed Functors

We now show that our main results hold as well for distributive laws of monads over *copointed* functors. This extends our method to deal with operations specified in abstract GSOS format, such as language concatenation.

Proposition 3. *Theorem 1 and Corollary 1 hold as well for any distributive law of a monad over a copointed functor.*

Proof. Let $\langle H, \epsilon\rangle$ be a copointed functor and $\lambda\colon TH \Rightarrow HT$ a distributive law of $\mathcal{T}$ over $\langle H, \epsilon\rangle$. Suppose λ preserves equations E. By Theorem 1 then there is a distributive law λ' of $\mathcal{T}'$ over H such that $q\colon T \to T'$ is a morphism of distributive laws. In order to show that λ' is a distributive law of $\mathcal{T}'$ over $\langle H, \epsilon\rangle$ we only need to prove that λ' satisfies the additional axiom, i.e., that the right crescent in the following diagram commutes:



The outermost part commutes by naturality of q, the two squares commute by naturality of λ and ϵ, and the left crescent commutes by the fact that λ is a distributive law of $\mathcal{T}$ over $\langle H, \epsilon\rangle$. Consequently we have $\epsilon_{T'X} \circ \lambda'_X \circ q_{HX} = T'\epsilon_X \circ q_{HX}$, and since q_{HX} is epi we obtain $\epsilon_{T'X} \circ \lambda'_X = T'\epsilon_X$ as desired.

For Corollary 1 one needs to add to its proof a check that the distributive law satisfies the additional axiom as well, which is again rather easy to do. □

Example 4 (Context-free languages). A context free grammar (in Greibach normal form) consists of a finite set A of terminal symbols, a (finite) set X of non-terminal symbols, and a map $\langle o, t\rangle \colon X \to 2 \times \mathcal{P}_\omega(X^*)^A$, i.e., it is a coalgebra for the behavior functor $F = 2 \times \mathrm{Id}^A$ composed with the idempotent semiring monad $\mathcal{P}_\omega(\mathrm{Id}^*)$ from Example 1. Intuitively, $o(x) = 1$ means that the variable x can generate the empty word, whereas $w \in t(x)(a)$ if and only if x can generate aw, cf. [22].

It is a rather difficult task to describe concretely a distributive law of $\mathcal{P}_\omega(\mathrm{Id}^*)$ over F (or $\mathrm{Id} \times F$) defining the sum $+$ and sequential composition $\cdot$ of context-free grammars. More conveniently, since we have seen in Example 1 that the monad $\mathcal{P}_\omega(\mathrm{Id}^*)$ can be presented by the operations and axioms of idempotent semirings, we proceed by defining a distributive law λ of the free monad $\mathcal{T}$ generated by the semiring signature functor $\Sigma(X) = 1 + 1 + (X \times X) + (X \times X)$ over the cofree copointed functor $\langle \mathrm{Id} \times F, \pi_1\rangle$, and show that λ preserves the semiring axioms. We define λ as the distributive law that corresponds to the natural transformation $\rho \colon \Sigma(\mathrm{Id} \times F) \Rightarrow FT$ whose components are given by:

$$\begin{aligned}
\rho^0_X &= \langle 0, a \mapsto \emptyset\rangle \\
\rho^1_X &= \langle 1, a \mapsto \emptyset\rangle \\
\rho^+_X(\langle x, o, f\rangle, \langle y, p, g\rangle) &= \langle \max\{o, p\}, a \mapsto f(a) + g(a)\rangle \\
\rho^{\cdot}_X(\langle x, o, f\rangle, \langle y, p, g\rangle) &= \left\langle \min\{o, p\}, a \mapsto \begin{cases} f(a) \cdot y & \text{if } p = 0 \\ f(a) \cdot y + g(a) & \text{if } p = 1 \end{cases} \right\rangle
\end{aligned} \tag{21}$$

We proceed to show that λ preserves the defining equations of idempotent semirings. We treat here only the case of distributivity, i.e., $u \cdot (v + w) = u \cdot v + u \cdot w$. To this end let $g \colon V \to X \times FX$ be arbitrary and suppose $g(v) = \langle x, o, d\rangle, g(u) = \langle y, p, e\rangle$ and $g(z) = \langle z, q, f\rangle$. Notice that either $o = 0$ or $o = 1$; we treat both cases separately:

$$\begin{aligned}
&\lambda(\langle x, 0, d\rangle \cdot (\langle y, p, e\rangle + \langle z, q, f\rangle)) \\
&\quad = \quad (x \cdot (y + z), 0, a \mapsto d(a) \cdot (y + z)) \\
&\quad \overline{F}(\equiv_X)\ (x \cdot y + x \cdot z, 0, a \mapsto d(a) \cdot y + d(a) \cdot z) \\
&\quad = \quad \lambda(\langle x, 0, d\rangle \cdot \langle y, p, e\rangle + \langle x, 0, d\rangle \cdot \langle z, q, f\rangle)
\end{aligned}$$

$$\begin{aligned}
&\lambda(\langle x, 1, d\rangle \cdot (\langle y, p, e\rangle + \langle z, q, f\rangle)) \\
&\quad = \quad (x \cdot (y + z), p + q, a \mapsto d(a) \cdot (y + z) + (e(a) + f(a))) \\
&\quad \overline{F}(\equiv_X)\ (x \cdot y + x \cdot z, p + q, a \mapsto (d(a) \cdot y + d(a) \cdot z) + (e(a) + f(a))) \\
&\quad \overline{F}(\equiv_X)\ (x \cdot y + x \cdot z, p + q, a \mapsto (d(a) \cdot y + e(a)) + (d(a) \cdot z + f(a))) \\
&\quad = \quad \lambda(\langle x, 1, d\rangle \cdot \langle y, p, e\rangle + \langle x, 1, d\rangle \cdot \langle z, q, f\rangle)\,.
\end{aligned}$$

In a similar way one can show that λ preserves the other idempotent semiring equations. Thus, from Proposition 3 and Corollary 1 we obtain a distributive law κ of $\mathcal{P}_\omega(\mathrm{Id}^*)$ over $2 \times \mathrm{Id}^A$ such that $i \circ q \colon \lambda \Rightarrow \kappa$ is a morphism of distributive laws, i.e., κ is presented by λ and the equations of idempotent semirings. $\triangleleft$

5 Morphisms and Solutions

In this section, we show that morphisms of distributive laws commute with solving corecursive equations. In the case of monads with equations, this means that first solving equations ϕ with respect to $\mathcal{T}$ and then quotienting the solution bialgebra is the same as first quotienting $\mathcal{T}$ and solving with respect to the quotient monad $\mathcal{T}'$.

We first describe some functors that link the relevant categories of bialgebras and corecursive equations. Throughout this Section, we let $\mathcal{T} = \langle T, \eta, \mu \rangle$ and $\mathcal{K} = \langle K, \theta, \nu \rangle$ be monads; and $\lambda\colon TF \Rightarrow FT$ and $\kappa\colon KF \Rightarrow FK$ be distributive laws of $\mathcal{T}$ and $\mathcal{K}$ over F, respectively.

If $\tau\colon \lambda \Rightarrow \kappa$ is a morphism of distributive laws, then precomposing with τ yields a functor:

$$\begin{array}{llcl} I\colon & \mathsf{Bialg}(\kappa) & \to & \mathsf{Bialg}(\lambda) \\ & KX \xrightarrow{\alpha} X \xrightarrow{\beta} FX & \mapsto & TX \xrightarrow{\alpha \circ \tau_X} X \xrightarrow{\beta} FX \end{array} \tag{22}$$

I takes a κ-bialgebra to a λ-bialgebra follows from the naturality of τ and $F\tau \circ \lambda = \kappa \circ \tau F$. Similarly, postcomposing with $F\tau$ yields a functor between corecursive equations:

$$\begin{array}{rrcl} Q\colon & \mathsf{Coalg}(FT) & \to & \mathsf{Coalg}(FK) \\ & \phi\colon X \to FTX & \mapsto & F\tau_X \circ \phi\colon X \to FKX \end{array} \tag{23}$$

Recall from Section 3.2, that given a distributive law $\lambda\colon TF \Rightarrow FT$, the solutions of a corecursive equation $\phi\colon X \to FTX$ are characterised by morphisms from the λ-bialgebra $\langle TX, \mu_X, \phi^\lambda \rangle$ whose F-coalgebra structure given by

$$\phi^\lambda = F\mu_X \circ \lambda_{TX} \circ T\phi \tag{24}$$

This yields a functor (see e.g. [5, Lem. 5.4.11]):

$$\begin{array}{rrcl} G_\lambda\colon & \mathsf{Coalg}(FT) & \to & \mathsf{Bialg}(\lambda) \\ & \langle X, \phi \rangle & \mapsto & \langle TX, \mu_X, \phi^\lambda \rangle \end{array} \tag{25}$$

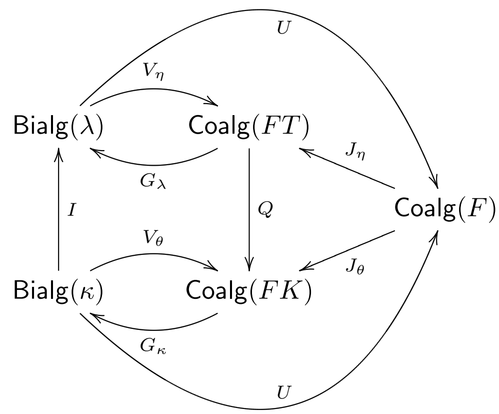
We can go in the opposite direction by using the monad unit,

$$\begin{array}{rrcl} V_\eta\colon & \mathsf{Bialg}(\lambda) & \to & \mathsf{Coalg}(FT) \\ & \langle X, \alpha, \beta \rangle & \mapsto & \langle X, F\eta_X \circ \beta \rangle \end{array} \tag{26}$$

which decomposes into the functor $U\colon \mathsf{Bialg}(\lambda) \to \mathsf{Coalg}(F)$ that forgets algebra structure, and

$$\begin{array}{rrcl} J_\eta\colon & \mathsf{Coalg}(F) & \to & \mathsf{Coalg}(FT) \\ & \langle X, \beta \rangle & \mapsto & \langle X, F\eta_X \circ \beta \rangle \end{array} \tag{27}$$

The following diagram summarises the situation:



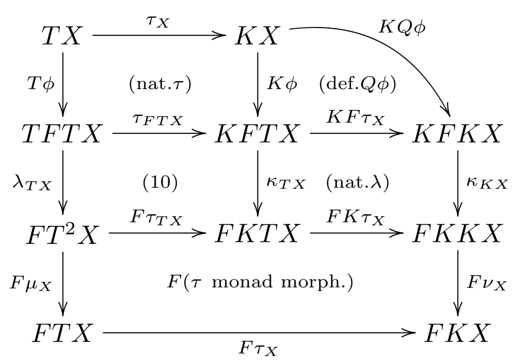
(28)

We mention that $QV_\eta I = V_\theta$ since τ is compatible with the units of $\mathcal{T}$ and $\mathcal{K}$.

Morphisms of distributive laws are defined to be monad maps, and hence respect the algebraic structure. The next proposition shows that, as one might expect, they also respect the coalgebraic structure, and hence morphisms of distributive laws induce morphisms between bialgebras.

Proposition 4. *If $\tau\colon \lambda \Rightarrow \kappa$ is a morphism of distributive laws, then for all $\phi : X \to FTX$ we have that τ_X is a λ-bialgebra morphism $\tau_X\colon G_\lambda(\phi) \to IG_\kappa Q(\phi)$ or, equivalently, an F-coalgebra morphism $\tau_X\colon \langle TX, \phi^\lambda\rangle \to \langle KX, (Q\phi)^\kappa\rangle$.*

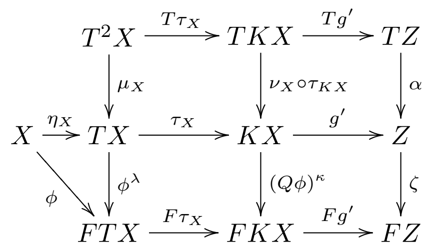
Proof. We show that $\tau_X\colon \langle TX, \phi^\lambda\rangle \to \langle KX, (Q\phi)^\kappa\rangle$ is an F-coalgebra morphism:



□

It follows that the unique λ-bialgebra morphism $g\colon \langle TX, \mu_X, \phi^\lambda\rangle \to \langle Z, \alpha, \zeta\rangle$ into the final λ-bialgebra $\langle Z, \alpha, \zeta\rangle$ factors as $g = g' \circ \tau_X$, where g' is the final

λ-bialgebra morphism from $IG_\kappa Q(\phi)$, as shown here:

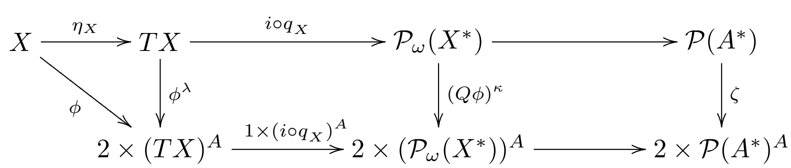


(29)

Hence by Proposition 2, every solution of ϕ in the final λ-bialgebra yields a solution of $Q\phi$, and vice versa.

When $\tau\colon \lambda \Rightarrow \kappa$ arises from a set of preserved equations E as in Section 4 (with $\kappa = \lambda'$), then Proposition 4 says that $IG_\kappa Q(\phi)$ is a quotient of the "free" λ-bialgebra $\langle TX, \mu_X, \phi^\lambda \rangle$, and in particular, the congruence $\equiv_X$ is an F-behavioural equivalence. In this case, $Q\phi$ is the corecursive equation obtained by reading the right-hand side of ϕ modulo equations in E. In other words, quotienting the solution of the equation ϕ is the same as solving the quotiented equation $Q\phi$.

Example 5. Recall from Example 4 that $i \circ q\colon T \Rightarrow \mathcal{P}_\omega(X^*)$ is a morphism of distributive laws. By Proposition 4 we have the following commuting diagram for any corecursive equation $\phi\colon X \to 2 \times (TX)^A$:



(30)

Notice that a context-free grammar $\langle o, t\rangle\colon X \to 2 \times \mathcal{P}_\omega(X^*)^A$ can be represented by a $\phi\colon X \to 2 \times (TX)^A$ such that $Q\phi = \langle o, t\rangle$, since $i \circ q$ is surjective. This gives the expected correspondence between two of the three different coalgebraic approaches to context-free languages introduced in [22] (the third approach is about fixed-point expressions and as such is outside the scope of this paper). $\triangleleft$

Similarly, the algebraic structure induced by λ on the final F-coalgebra factors uniquely through the algebraic structure induced by κ.

Proposition 5. *Let $\tau\colon \lambda \Rightarrow \kappa$ be a morphism of distributive laws, and let $\alpha\colon TZ \to Z$ and $\alpha'\colon KZ \to Z$ be the algebras induced by λ and κ respectively on the final coalgebra $\langle Z, \zeta\rangle$. Then $\alpha = \alpha' \circ \tau_Z$.*

Proof. Consider the following diagram:

$$
\begin{array}{ccccccc}
TZ & \xrightarrow{\tau_Z} & KZ & \xrightarrow{\alpha'} & Z & \xleftarrow{\alpha} & TZ \\
\downarrow{\scriptstyle T\zeta} & & \downarrow{\scriptstyle K\zeta} & & & & \downarrow{\scriptstyle T\zeta} \\
TFZ & \xrightarrow{\tau_{FZ}} & KFZ & & \downarrow{\scriptstyle \zeta} & & TFZ \\
\downarrow{\scriptstyle \lambda_Z} & & \downarrow{\scriptstyle \kappa_Z} & & & & \downarrow{\scriptstyle \lambda_Z} \\
FTZ & \xrightarrow{F\tau_Z} & FKZ & \xrightarrow{F\kappa} & FZ & \xleftarrow{F\alpha} & FTZ
\end{array}
$$

The upper left square commutes by naturality of τ, whereas the lower left square commutes since τ is a morphism of distributive laws. The two rectangles commute by definition of α and α' (see Section 3). Thus $\alpha' \circ \tau_Z$ and α are both coalgebra homomorphisms from $\langle TZ, \lambda_Z \circ T\zeta \rangle$ to $\langle Z, \zeta \rangle$ and consequently $\alpha' \circ \tau_Z = \alpha$ by finality. □

Example 6. Continuing Example 5, it follows from Proposition 5 that the algebra $\alpha \colon T\mathcal{P}(A^*) \to \mathcal{P}(A^*)$ induced by the distributive law for $\mathcal{T}$ can be decomposed as $i \circ q \circ \alpha'$, where α' is the algebra on $\mathcal{P}(A^*)$ induced by the distributive law for $\mathcal{P}_\omega(\mathrm{Id}^*)$. It can be shown by induction that α is the algebra on languages given by union and concatenation product. Now $\alpha' \colon \mathcal{P}_\omega(\mathcal{P}(A^*)^*) \to \mathcal{P}(A^*)$ can be given by selecting a representative term and applying α, and it follows that $\alpha'(\mathcal{L}) = \bigcup_{L_1 \cdots L_n \in \mathcal{L}} \{w_1 \cdots w_n \mid w_i \in L_i\}$. ◁

6 Discussion and Conclusion

We have presented a preservation condition that is necessary and sufficient for the existence of a distributive law λ' for a monad with equations given a distributive law λ for the underlying free monad. This condition consists of checking that the base equations are preserved by λ. For concrete monads, checking preservation is often much easier than describing and verifying the distributive law requirements directly. We demonstrated our method by applying it to obtain distributive laws for stream calculus over commutative semirings, and for context-free grammars which use the monad of idempotent semirings.

In [20] the notion of morphisms of distributive laws is studied as a general approach to translations between operational semantics. In this paper we investigate in detail the case of quotients of distributive laws. Distributive laws for monad quotients and equations are also studied in [10,11]. The setting and motivation of [11] is different as they study distributive laws of one monad over another with the aim to compose these monads. We study distributive laws of a monad over a plain or copointed functor. The approach in [10] is more general as they consider monads on arbitrary categories, but it also differs from ours in that the desired distributive law is contingent on two given distributive laws and the existence of the coequaliser (in the category of monads) which encodes equations. We have given a more direct analysis for monads in Set and a practical proof principle, which covers many known examples. We leave as future work to find out precisely how their Theorem 31 relates to our Theorem 1.

While in this work we have focused on adding equations which already hold in the final bialgebra, it is often useful to use equations to *induce* behaviour, next to a behavioural specification in terms of a distributive law. In process theory this idea is captured by the notion of structural congruences [13]; at the more general level of distributive laws there is work on adding recursive equations [9]. We leave a general study of structural congruences for distributive laws, and the precise relation to the present results, as future work.

More technically, it remains an open problem whether a converse of Proposition 4 holds. Finally, as mentioned by one of the referees, it would be desirable to be able construct the quotient monad without the assumption of right-inverses by using a more categorical treatment of congruences as spans. Another abstract formulation of our results in terms of right Kan extensions has also been suggested. We intend to investigate these matters in future work.

References

1. Aceto, L., Fokkink, W., Verhoef, C.: Structural operational semantics. In: Bergstra, J., Ponse, A., Smolka, S. (eds.) Handbook of Process Algebra, pp. 197–292. Elsevier (2001)
2. Bartels, F.: On Generalised Coinduction and Probabilistic Specification Formats. Ph.D. thesis, Vrije Universiteit Amsterdam (2004)
3. Hansen, H., Klin, B.: Pointwise extensions of GSOS-defined operations. Math. Struct. in Comp. Sci. 21, 321–361 (2011)
4. Jacobs, B.: A bialgebraic review of deterministic automata, regular expressions and languages. In: et al., K.F. (ed.) Algebra, Meaning and Computation, LNCS, vol. 4060, pp. 375–404. Springer (2006)
5. Jacobs, B.: Introduction to coalgebra. towards mathematics of states and observations. version 2.0. (2012), unpublished book draft
6. Jacobs, B.: Distributive laws for the coinductive solution of recursive equations. Inf. Comput. 204(4), 561–587 (2006)
7. Johnstone, P.: Adjoint lifting theorems for categories of algebras. Bull. London Math. Society 7, 294–297 (1975)
8. Klin, B.: Bialgebras for structural operational semantics: An introduction. Theor. Comp. Sci. 412, 5043–5069 (2011)
9. Klin, B.: Adding recursive constructs to bialgebraic semantics. J. Logic and Algebraic Programming 60-61, 259–286 (2004)
10. Lenisa, M., Power, J., Watanabe, H.: Category theory for operational semantics. Theor. Comp. Sci. 327(1-2), 135–154 (2004)
11. Manes, E., Mulry, P.: Monad compositions I: General constructions and recursive distributive laws. Theory and Applications of Categories 18(7), 172–208 (2007)
12. Milius, S.: A sound and complete calculus for finite stream circuits. In: Proceedings of LICS 2012. pp. 421–430. IEEE Computer Society (2010)
13. Mousavi, M.R., Reniers, M.A.: Congruence for structural congruences. In: Sassone, V. (ed.) FoSSaCS. LNCS, vol. 3441, pp. 47–62. Springer (2005)
14. Rot, J., Bonchi, F., Bonsangue, M., Pous, D., Rutten, J., Silva, A.: Enhanced coalgebraic bisimulation, `http://www.liacs.nl/~jrot/papers/up-to.pdf`

15. Rot, J., Bonsangue, M., Rutten, J.: Coalgebraic bisimulation-up-to. In: van Emde Boas, P., Groen, F., Italiano, G., Nawrocki, J., Sack, H. (eds.) SOFSEM. LNCS, vol. 7741, pp. 369–381. Springer (2013)
16. Rutten, J.: Behavioural differential equations: a coinductive calculus of streams, automata and power series. Theor. Comp. Sci. 308(1), 1–53 (2003)
17. Rutten, J.: A coinductive calculus of streams. Math. Struct. in Comp. Sci. 15, 93–147 (2005)
18. Silva, A., Bonchi, F., Bonsangue, M., Rutten, J.: Generalizing the powerset construction, coalgebraically. In: Lodaya, K., Mahajan, M. (eds.) FSTTCS. LIPIcs, vol. 8, pp. 272–283. Schloss Dagstuhl - Leibniz-Zentrum für Informatik (2010)
19. Turi, D., Plotkin, G.: Towards a mathemathical operational semantics. In: Proceedings of LICS'97. pp. 280–291. IEEE Computer Society (1997)
20. Watanabe, H.: Well-behaved translations between structural operational semantics. In: Moss, L. (ed.) Proceedings of CMCS 2002. ENTCS, vol. 65, pp. 337–357. Elsevier (2002)
21. Winskel, G.: The formal semantics of programming languages - an introduction. Foundation of computing series, MIT Press (1993)
22. Winter, J., Bonsangue, M., Rutten, J.: Context-free languages, coalgebraically. In: Corradini, A., Klin, B., Cirstea, C. (eds.) Proceedings of CALCO 2011. LNCS, vol. 6859, pp. 359–376. Springer (2011)