Presenting Functors on Many-Sorted Varieties and Applications

Alexander Kurz, Daniela Petrişan

Department of Computer Science, University of Leicester, UK

Abstract

This paper studies several applications of the notion of a *presentation of a functor by operations and equations*. We show that the technically straightforward generalisation of this notion from the one-sorted to the many-sorted case has several interesting consequences. First, it can be applied to give equational logic for the binding algebras modelling abstract syntax. Second, it provides a categorical approach to algebraic semantics of first-order logic. Third, this notion links the uniform treatment of logics for coalgebras of an arbitrary type T with concrete syntax and proof systems. Analysing the many-sorted case is essential for achieving modular completeness proofs for coalgebraic logics.

Contents

1 Introduction **2**

2 Some preliminaries **3**

3 Presenting algebras and functors **5**

4 Equational logic for higher-order abstract syntax **10**
- 4.1 An equational presentation for $\mathcal{A} = \mathsf{Set}^{\mathbb{F}}$ 11
- 4.2 An equational presentation for binding algebras 18

5 First-Order Logic **20**
- 5.1 Algebraic Semantics of First-Order Logic 21
- 5.2 Coalgebraic Semantics of First-Order Logic 24

6 Modularity of Functorial Coalgebraic Logic **26**
- 6.1 Many-sorted composition of functors 27
- 6.2 Uniform completeness 29
- 6.3 Filtration and the finite distribution functor 32

6.3.1 A filtration method 33
6.3.2 The finite distribution functor 34
6.4 Modular Completeness 35

7 Conclusion **37**

1 Introduction

This paper describes several applications of finitary presentations of functors on many-sorted varieties. The notion of finitary presentations for functors on one-sorted varieties was introduced by [11] in the context of coalgebraic logic. It generalises the following situation from the category of sets: A finitary functor L on Set is a quotient of a polynomial functor. Indeed we have a quotient

$$\coprod_{n\in\mathbb{N}} L\underline{n} \times X^n \twoheadrightarrow LX \tag{1}$$

where $\underline{n}$ is the set $\{1,\ldots,n\}$, and a pair (σ, f) is mapped to $Lf(\sigma)$, (f can be thought of as a map from $\underline{n}$ to X). This is a quotient because L, as a filtered colimit preserving functor, is determined by its values on finite sets. The elements of $L\underline{n}$ can be regarded as the n-ary operations presenting L, satisfying the equations corresponding to the kernel of the above map (for a full account see Adámek and Trnková [7, III.4.9]). To summarise, (1) gives us a presentation $\langle\Sigma_L, E_L\rangle$ by operations Σ_L and equations E_L and, therefore, an *equational logic* for L-algebras: The category of L-algebras is isomorphic to the category of algebras for the signature Σ_L and equations E_L.

A functor on an arbitrary variety is said to have a finitary presentation by operations and equations, if it is isomorphic to a quotient of a polynomial functor (which captures the operations) by some rank 1 equations. To generalise (1), one can replace finite sets by finitely generated free algebras. But then, L should be determined by its values on finitely generated free algebras, that is, L should preserve sifted colimits [6]. As shown in [30] it is indeed the case that a functor L on a (finitary) variety $\mathcal{A}$ has a presentation by operations and equations in the sense of [11] if and only if L preserves sifted colimits. Although not as well known as filtered colimits, sifted colimits are the right concept when working with varieties (as opposed to locally finitely presentable categories): Each variety $\mathcal{A}$ is the free cocompletion by sifted colimits of the dual of the Lawvere theory of $\mathcal{A}$. The reason is that algebras for a Lawvere theory are set-valued product-preserving functors and sifted colimits are precisely those colimits that commute in Set with finite products.

This paper continues this line of research. We start by generalizing the results of [30] on functors on varieties from the one-sorted to the many sorted case (Section 3). This generalization in itself is not difficult, but it has interesting applications.

The first, maybe somewhat unexpected, is that it provides an equational logic for higher order abstract syntax. There are several mathematical models for abstract syntax with variable binding [15, 16, 22]. In particular, [15] showed that syntax can be specified by an algebraic signature even in the presence of binding constructors. However, this cannot be achieved in the usual way, via a Set functor. Instead, they have introduced binding signatures as functors on a presheaf category. In Section 4 we give an equational logic for the corresponding binding algebras.

Secondly, Halmos' polyadic algebras – introduced as an algebraic semantics for first-order logic – are characterized as algebras for a finitary presented functor on a category of Boolean algebra presheaves (Section 5). The second part of this section deals with coalgebraic semantics for first-order logic, making a transition towards the applications of this notion in coalgebraic logic.

Thirdly, we give a modular completeness theorem for functorial coalgebraic logic. Functorial coalgebraic logic started with [9, 25], which argued that logics for T-coalgebras (where T is an endofunctor on Set) are suitably described by endofunctors L on the category of Boolean algebras or some other suitable category of algebras. Syntactically, L specifies an extension of Boolean propositional logic by modal operators and axioms. [30] gave a uniform strong completeness result by generalizing the Jónsson-Tarski representation theorem for Boolean algebras with operators. As this required some assumptions on T, we show here that for completeness as opposed to strong completeness no condition on T is required. We then show how to extend completeness of basic logics in a modular way to large classes of inductively defined functors.

Acknowledgements We would like to thank Rick Thomas for pointing out a very useful reference [8].

2 Some preliminaries

For an endofunctor L on a category $\mathcal{A}$, we consider the category of *L-algebras*, denoted by $\mathsf{Alg}(L)$, whose objects are defined as pairs (A, α) such that $\alpha : LA \to A$ is a morphism in $\mathcal{A}$. A morphism of L-algebras $f : (A, \alpha) \to (A', \alpha')$ is a morphism $f : A \to A'$ of $\mathcal{A}$ such that $f \circ \alpha = \alpha' \circ Lf$. Dually, for an endofunctor $T : \mathcal{A} \to \mathcal{A}$ we consider the category of *T-coalgebras*, denoted by $\mathsf{Coalg}(T)$, whose objects are pairs (A, γ), such that $\gamma : A \to TA$. A morphism of T-coalgebras $f : (A, \gamma) \to (A', \gamma')$ is an arrow $f : A \to A'$ of $\mathcal{A}$ such that $Tf \circ \gamma = \gamma' \circ f$.

Let S be a set (of sorts). A *signature* Σ is a set of operation symbols together with an arity map $a : \Sigma \to S^* \times S$ which assigns to each element $\sigma \in \Sigma$ a pair $(s_1, \ldots, s_n; s)$ consisting of a finite word in the alphabet S indicating the sort of the arguments of σ and an element of S indicating the sort of the result of σ. To each signature we can associate an endofunctor on Set^S, which will be denoted for simplicity with the same symbol Σ:

$$(\Sigma X)_s = (\coprod_{k \in \omega_f^S} \Sigma_{k,s} \times X^k)_s$$

Here, by ω_f^S we denote the finite multisets on S. Note that a multiset $k \in \omega_f^S$ can be regarded as an S-sorted set. X^k denotes the set of presheaf morphisms $\mathsf{Set}^S(k, X)$. In detail, if $k \in \omega_f^S$ is the multiset $\{s_1, ..., s_n\}$ then $\Sigma_{k,s}$ is a set of operations of arity $(s_1...s_n; s)$ and X^k is isomorphic in Set with the finite product $X_{s_1} \times \cdots \times X_{s_n}$. Conversely, to each polynomial endofunctor on Set^S given as above corresponds a signature $\coprod\limits_{k \in \omega_f^S} \Sigma_{k,s}$. Throughout this paper we will make no notational difference between the signature and the corresponding functor, and it will be clear from the context when we refer to the set of operation symbols or to a Set^S endofunctor. The algebras for a signature Σ are precisely the algebras for the corresponding endofunctor, and form the category denoted by $\mathsf{Alg}(\Sigma)$. The terms over an S-sorted set of variables X are defined in the standard manner and form an S-sorted set denoted by $\mathrm{Term}_\Sigma(X)$, in fact this is the underlying set of the free Σ-algebra generated by X. An equation consists of a pair (τ_1, τ_2) of terms of the same sort, usually denoted $\tau_1 = \tau_2$. A Σ-algebra A satisfies this equation if and only if, for any interpretation of the variables of X, we obtain equality in A. A full subcategory $\mathcal{A}$ of $\mathsf{Alg}(\Sigma)$ is called a *variety* or an *equational class* if there exists a set of equations E such that an algebra lies in $\mathcal{A}$ if and only if it satisfies all the equations of E. In this case, the variety $\mathcal{A}$ will be denoted by $\mathsf{Alg}(\Sigma, E)$. The forgetful functor $U : \mathsf{Alg}(\Sigma, E) \to \mathsf{Set}^S$ preserves filtered colimits and has a left adjoint F. The variety $\mathsf{Alg}(\Sigma, E)$ is isomorphic to the Eilenberg-Moore category $(\mathsf{Set}^S)^T$ for the finitary monad $T = UF$ (see [4, Theorem 3.18]). In fact, the forgetful functor U preserves a wider class of colimits, namely *sifted colimits*, [3]. A *sifted category* $\mathcal{D}$ is a small category such that colimits over $\mathcal{D}$ commute in Set with finite products. A sifted colimit in a category $\mathcal{C}$ is a colimit over $\mathcal{D}$. The most important examples of sifted colimits are filtered colimits and reflexive coequalizers. An object in a category is called *strongly finitely presentable* if its hom-functor preserves sifted colimits. It is shown in [5] that any object in a variety is a sifted colimit of strongly finitely presentable algebras, which in a variety are the retracts of finitely generated free algebras. An important observation is that sifted colimit preserving functors on varieties are determined by their action on free algebras.

An important example of a (finitary) variety of algebras is the functor category $\mathsf{Set}^{\mathcal{C}}$ for any small category $\mathcal{C}$. The sorts are the objects of $\mathcal{C}$, the operations symbols are the morphisms of $\mathcal{C}$ (all of them with arity 1), and the equations are given by the commutative diagrams in $\mathcal{C}$.

Endofunctors may appear via composition of functors between different varieties. Therefore, it is useful to consider a slight generalization of the notion of signature. If S_1 and S_2 are sets of sorts we will consider operations with arguments of sorts in S_1 and returning a result of a sort in S_2, encompassed in the signature functor $\Sigma : \mathsf{Set}^{S_1} \to \mathsf{Set}^{S_2}$

$$\Sigma X = (\coprod_{k \in \omega_f^{S_1}} \Sigma_{k,s} \times X^k)_{s \in S_2} \tag{2}$$

3 Presenting algebras and functors

Before going into technicalities, we discuss a motivating example. It is relevant to Sections 5 and 6, but not to Section 4.

Motivation from Modal Logic The general idea is as follows. Just as coalgebras are given wrt a functor T on, say, Set, so are logics for coalgebras given by a functor L on, say, Boolean algebras. The following example shows how logics for coalgebras given in a more conventional style give rise to a functor on the category BA of Boolean algebras.

Example 3.1. *Let $T = \mathcal{P}$ be the covariant powerset functor. The modal logic $\mathbf{K}$ associated to $\mathcal{P}$-coalgebras (=Kripke frames) can be described by the functor L which maps a Boolean algebra A to the Boolean algebra LA freely generated by $\{\Box a \mid a \in A\}$ modulo the relations $\Box\top = \top$ and $\Box(a \wedge b) = \Box a \wedge \Box b$. We see that the modal operators appear as generators and the modal axioms as relations. Of course, from a logical point of view, we want the generators to be operations and the relations to be universally quantified equations. In other words, we need that the description of LA in terms of generators and relations is uniform in A. This is exactly captured by Definition 3.3 below.*

It is not difficult to see that the category $\mathsf{Alg}(L)$ of algebras for the functor L is isomorphic to the category of Boolean algebras with operators, which constitute the standard algebraic semantics of $\mathbf{K}$ in modal logic. In particular, the initial L-algebra is the Lindenbaum-Tarski algebra of the modal logic $\mathbf{K}$.

To simply replace a concrete modal logic by the corresponding functor is a powerful abstraction that makes a number of category theoretic methods available to modal logic. This section makes sure that the move from logics to functors is not an over-generalisation: Every suitable functor L will come from a modal logic in exactly the same way as in the example above. The reader who wants to know more about the relationship between T-coalgebras and L-algebras before reading this section might want to skip ahead to Section 6.2 or consult an introduction such as [27].

Presenting Algebras and Functors The notion of a finitary presentation by operations and equations for a functor was introduced in [11]. It generalises the notion of a presentation for an algebra, in the usual sense of universal algebra. An algebra A in a variety $\mathcal{A}$ is presented by a set of generators G and a set of equations E, if A is isomorphic to the free algebra on G, quotiented by the equations E. In a similar fashion, an endofunctor L on $\mathcal{A}$ is presented by operations Σ and equations E, if for each object A of $\mathcal{A}$, LA is isomorphic to the free algebra over ΣUA quotiented by the equations E. Below we extend this notion to the case of functors between possibly different many-sorted varieties.

A presentation for a (many-sorted) algebra in a variety $\mathcal{A}$ can be regarded as a coequalizer, as in the next definition. This category theoretical perspective will allow us to generalise this notion to functors.

Definition 3.2. *Let A be a many-sorted algebra in a variety $\mathcal{A}$. We say that (G, E) is a presentation for A if G is an S-sorted set of generators and $E = (E_s)_{s \in S}$, $E_s \subset (UFG)_s \times (UFG)_s$ is an S sorted set of equations such that q_A is the coequalizer of the following diagram:*

$$FE \underset{\pi_2^\sharp}{\overset{\pi_1^\sharp}{\rightrightarrows}} FG \xrightarrow{q_A} A \tag{3}$$

The maps $\pi_1^\sharp, \pi_2^\sharp$ are induced, via the adjunction, by the projections π_1, π_2 of E on UFG.

Next we want to define a presentation for a functor $L : \mathcal{A}_1 \to \mathcal{A}_2$ between many-sorted varieties. For $i \in \{1, 2\}$, denote by S_i the set of sorts for $\mathcal{A}_i$ respectively, by $U_i : \mathcal{A}_i \to \mathsf{Set}^{S^i}$ the corresponding forgetful functor, and by F_i its left adjoint. We will do this in the same fashion as in [30] and [11], keeping in mind that we need to extend (3) uniformly: this means that the generators and equations for each LA will depend functorially on A. Suppose A is a many-sorted algebra in $\mathcal{A}_1$. The generators $\Sigma U_1 A$ for the algebra LA will be given by a signature functor $\Sigma : \mathsf{Set}^{S_1} \to \mathsf{Set}^{S_2}$ as in (2). The equations that we will consider are of **rank 1**, meaning that in the terms involved every variable is under the scope of precisely one operation symbol in Σ, and are given by an S_2-sorted set E. In detail, for each sort $s \in S_2$ and each S_1-sorted set of variables V with the property that $\bigcup\limits_{t \in S_1} V_t$ is finite, we consider a set $E_{V,s}$ of equations over the set V, of terms of sort s, which is defined as a subset of $(U_2 F_2 \Sigma U_1 F_1 V)^2_s$. Now take $E_V = (E_{V,s})_{s \in S_2}$ and $E = \bigcup\limits_{V \in \omega_f^{S_1}} E_V$.

Definition 3.3. *Let S_1, S_2 be sets of sorts, $\mathcal{A}_1$ be an S_1-sorted variety and $\mathcal{A}_2$ be an S_2-sorted variety. A presentation for a functor $L : \mathcal{A}_1 \to \mathcal{A}_2$ is a pair $\langle \Sigma, E \rangle$ defined as above. A functor $L : \mathcal{A}_1 \to \mathcal{A}_2$ is presented by $\langle \Sigma, E \rangle$, if*

(i) for every algebra $A \in \mathcal{A}_1$, there exists a morphism $q_A : F_2 \Sigma U_1 A \to LA$ that is the joint coequalizer of the next diagram

$$F_2 E_V \underset{\pi_2^\sharp}{\overset{\pi_1^\sharp}{\rightrightarrows}} F_2 \Sigma U_1 F_1 V \xrightarrow{F_2 \Sigma U_1 v^\sharp} F_2 \Sigma U_1 A \xrightarrow{q_A} LA \tag{4}$$

taken after all finite sets of S_1-sorted variables V and all valuations $v : V \to U_1 A$. Here $v^\sharp$ denotes the adjoint transpose of a valuation v.

(ii) for all morphisms $f : A \to B$ the diagram commutes:

$$\begin{tikzcd} F_2 \Sigma U_1 A \arrow[r, "q_A"] \arrow[d, "F_2 \Sigma U_1 f"] & LA \arrow[d, "Lf"] \\ F_2 \Sigma U_1 B \arrow[r, "q_B"] & LB \end{tikzcd} \tag{5}$$

The Equational Logic Induced by a Presentation of L If $\mathcal{A} = \mathsf{Alg}(\Sigma_\mathcal{A}, E_\mathcal{A})$ is an S-sorted variety and the endofunctor $L : \mathcal{A} \to \mathcal{A}$ has a finitary presentation $\langle \Sigma_L, E_L \rangle$, we can obtain an equational calculus for $\mathsf{Alg}(L)$, regarding the equations $E_\mathcal{A}$ and E_L as equations containing terms in $\mathrm{Term}_{\Sigma_\mathcal{A}+\Sigma_L}$. First remark that formally, for an arbitrary set of variables V, $E_{L,V}$ is a subset of the S-sorted set $(UF\Sigma_L UFV)^2$. But for each set X, UFX is a quotient of $\mathrm{Term}_{\Sigma_\mathcal{A}}X$ modulo the equations. Thus, if we choose a representative for each equivalence class in $UF\Sigma_L UFV$, we can obtain a set of equations in $\mathrm{Term}_{\Sigma_\mathcal{A}}\Sigma_L\mathrm{Term}_{\Sigma_\mathcal{A}}V$. Using the natural map from $\mathrm{Term}_{\Sigma_\mathcal{A}}\Sigma_L\mathrm{Term}_{\Sigma_\mathcal{A}}V$ to $\mathrm{Term}_{\Sigma_\mathcal{A}+\Sigma_L}V$, we obtain a set of equations on terms $\mathrm{Term}_{\Sigma_\mathcal{A}+\Sigma_L}V$. By abuse of notation we will denote this set with E_L as well.

Theorem 3.4. *Let $\mathcal{A} = \mathsf{Alg}(\Sigma_\mathcal{A}, E_\mathcal{A})$ be an S-sorted variety and let $L : \mathcal{A} \to \mathcal{A}$ be a functor presented by operations Σ_L and equations E_L. Then $\mathsf{Alg}(L) \cong \mathsf{Alg}(\Sigma_\mathcal{A} + \Sigma_L, E_\mathcal{A} + E_L)$.*

Proof. We define a functor $H : \mathsf{Alg}(L) \to \mathsf{Alg}(\Sigma_\mathcal{A} + \Sigma_L, E_\mathcal{A} + E_L)$. Suppose $\alpha : LA \to A$ is an L-algebra. Then the underlying set of HA is defined to be UA. HA inherits the algebraic structure of A: the interpretation of the operation symbols of $\Sigma_\mathcal{A}$ is the same as in the algebra A and it satisfies the equations $E_\mathcal{A}$. As far as the operation symbols of Σ_L are concerned, their interpretation is given by the composition:

$$F\Sigma_L UA \xrightarrow{\quad q_A \quad} LA \xrightarrow{\quad \alpha \quad} A \tag{6}$$

Explicitly, the interpretation of an operation symbol σ of arity $(s_1 \dots s_n; s)$ is the morphism $\sigma^A : A_{s_1} \times \cdots \times A_{s_n} \to A_s$ defined by

$$\sigma^A(x_1, \dots, x_n) = \alpha(q_A((\sigma, x_1, \dots, x_n)))$$

Now it is clear that the equations E_L are satisfied in HA, because q_A is a coequalizer as in (4). If f is a morphism of L-algebras, we define $Hf = f$ and we only have to check that $f(\sigma(a_1, \dots, a_k)) = \sigma(f(a_1), \dots, f(a_k))$ for all $\sigma \in \Sigma_L$. But this follows from the fact the definition of the interpretation of the operations, the commutativity of diagram (5) and the fact that f is an L-algebra morphism.

Conversely, we define a functor $J : \mathsf{Alg}(\Sigma_\mathcal{A} + \Sigma_L, E_\mathcal{A} + E_L) \to \mathsf{Alg}(L)$. Suppose A is an algebra in $\mathsf{Alg}(\Sigma_\mathcal{A} + \Sigma_L, E_\mathcal{A} + E_L)$. The map $\rho_A : \Sigma_L UA \to UA$ defined by:

$$(\sigma_{(s_1 \dots s_n; s)}, x_{i_1}, \dots, x_{i_n}) \mapsto \sigma_{(s_1 \dots s_n; s)}(x_{i_1}, \dots, x_{i_n})$$

induces a map $\rho_A^\sharp : F\Sigma_L UA \to A$. The fact that equations E_L are satisfied implies that $\rho_A^\sharp \circ F\Sigma_L Uv^\sharp \circ \pi_1^\sharp = \rho_A^\sharp \circ F\Sigma_L Uv^\sharp \circ \pi_2^\sharp$ as depicted in (7). But q_A is a coequalizer in $\mathsf{Alg}(\Sigma_\mathcal{A}, E_\mathcal{A})$, therefore there exists a morphism $\alpha_A : LA \to A$ such that $\alpha_A \circ q_A = \rho_A^\sharp$. We define JA to be the L-algebra α_A. For any morphism $f : A \to B$ in $\mathsf{Alg}(\Sigma_\mathcal{A} + \Sigma_L, E_\mathcal{A} + E_L)$ we define $Jf = U_0 f$, where $U_0 : \mathsf{Alg}(\Sigma_\mathcal{A} + \Sigma_L, E_\mathcal{A} + E_L) \to \mathsf{Alg}(\Sigma_\mathcal{A}, E_\mathcal{A})$ is the forgetful functor. This is well defined and we can check this easily by proving that the rightmost square of diagram (7) is commutative:

$$
\begin{tikzcd}
 & & F\Sigma_L UA \arrow[r, "q_A"] \arrow[d, "F\Sigma Uf"] \arrow[rr, bend left, "\rho_A^\sharp"] & LA \arrow[r, dashed, "\alpha_A"] \arrow[d, "Lf"] & A \arrow[d, "f"] \\
FE_L \arrow[r, shift left, "\pi_1^\sharp"] \arrow[r, shift right, "\pi_2^\sharp"'] & F\Sigma_L UFV \arrow[ur, "F\Sigma Uv_1^\sharp"] \arrow[dr, "F\Sigma Uv_2^\sharp"] & F\Sigma_L UB \arrow[r, "q_B"] \arrow[rr, bend right, "\rho_B^\sharp"'] & LB \arrow[r, dashed, "\alpha_B"] & B
\end{tikzcd}
\tag{7}
$$

Now it is straightforward to check that $J \circ H$ and $H \circ J$ are the identities. $\square$

The Characterization Theorem The characterisation theorem of endofunctors having finitary presentation was given in [30] for monadic categories over Set and it can be easily extended if we replace Set with the presheaf category Set^S. The result holds even if we work with functors between different varieties.

Theorem 3.5. *Let S_1, S_2 be sets of sorts, $\mathcal{A}_1$ be an S_1-sorted variety and $\mathcal{A}_2$ be an S_2-sorted variety. For a functor $L : \mathcal{A}_1 \to \mathcal{A}_2$ the following conditions are equivalent:*
(i) L has a finitary presentation by operations and equations;
(ii) L preserves sifted colimits.

Proof. $(i) \Rightarrow (ii)$. Assume L has a finitary presentation $\langle \Sigma, E \rangle$. Let D be a sifted category and $a_i : A_i \to A$ be a sifted colimit in $\mathcal{A}_1$. Let $d_i : LA_i \to B$ be an arbitrary cocone. As we have seen in the preliminaries, the corresponding forgetful functors and their left adjoints U_1, U_2, F_1, F_2 preserve sifted colimits. Σ shares the same property because sifted colimits are computed point-wise and commute with finite products. Therefore we obtain that $F_2\Sigma U_1 a_i : F_2\Sigma U_1 A_i \to F_2 \Sigma U_1 A$ is a colimiting cocone in $\mathcal{A}_2$, hence there exists a map $d : F_2 \Sigma U_1 A \to B$ such that $d \circ F_2\Sigma U_1 a_i = d_i \circ q_{A_i}$ for all i in D.

Choose an arbitrary S_1-sorted set of variables $V = (V_s)_{s \in S_1}$ such that $\bigcup\limits_{s \in S_1} V_s$ is finite and a morphism $v : V \to U_1 A$. Since V is strongly finitely presented in the category Set^{S_1}, and U_1 preserves sifted colimits, we have that $\mathsf{Set}^{S_1}(V, U_1 A)$ is the sifted colimit of $\mathsf{Set}^{S_1}(V, U_1 A_i)$. In particular there exists an index i and a morphism $v_i : V \to U_1 A_i$ such that $v = U_1 a_i \circ v_i$. From the fact that q_{A_i} is a joint coequalizer, it follows that d makes the bottom line of diagram (8) commutative.

$$
\begin{tikzcd}
 & & F_2\Sigma U_1 A_i \arrow[d, "F_2\Sigma U_1 a_i"] \arrow[dr, "d_i \circ q_{A_i}"] & \\
F_2 E_V \arrow[r, shift left, "\pi_1^\sharp"] \arrow[r, shift right, "\pi_2^\sharp"'] & F_2\Sigma U_1 F_1 V \arrow[ur, "F_2\Sigma U_1 v_i^\sharp"] \arrow[r, "F_2\Sigma U_1 v^\sharp"] & F_2 \Sigma U_1 A \arrow[r, "d"] & B
\end{tikzcd}
\tag{8}
$$

Using that q_A is a joint coequalizer, we obtain $b : LA \to B$ such that $b \circ q_A = d$. Now it is immediate to check that diagram (9) is commutative, and this shows that the cocone $La_i : LA_i \to LA$ is universal.

$$\begin{tikzcd} F_2\Sigma U_1 A_i \arrow[r, "q_{A_i}"] \arrow[d, "F_2\Sigma U_1 a_i"] & LA_i \arrow[d, "La_i"] \arrow[ddr, "d_i"] & \\ F_2\Sigma U_1 A \arrow[r, "q_A"] \arrow[drr, "d"'] & LA \arrow[dr, "b"] & \\ & & B \end{tikzcd} \tag{9}$$

$(ii) \Rightarrow (i)$ Being a sifted colimit preserving functor, L is determined by its values on finitely generated free algebras. Given $k \in \omega_f^{S_1}$ with support $\{s_1, \ldots, s_n\}$ and given $s \in S_2$ we can view the elements of the set $(U_2LF_1k)_s$ as operations symbols which take $k(s_i)$ arguments of sort s_i for all $1 \leq i \leq n$ and return a result of sort s. More explicitly we can consider for all algebras A the map r_A given component-wise by:

$$\coprod_{k \in \omega_f^S} (U_2LF_1k)_s \times (U_1A)^k \xrightarrow{r_{A,s}} (U_2LA)_s \tag{10}$$

$$(\sigma, x) \mapsto (U_2L\epsilon_A \circ U_2LF_1x)_s(\sigma)$$

where $\epsilon_A : F_1U_1A \to A$ is the counit of the adjunction. In the definition of the map $r_{A,s}$ we have interpreted x as a morphism in $\mathsf{Set}^{S_1}(k, U_1A)$. Now the operations that we will consider are encompassed in the functor $\Sigma : \mathsf{Set}^{S_1} \to \mathsf{Set}^{S_2}$ defined by

$$\Sigma X = (\coprod_{k \in \omega_f^{S_1}} (U_2LF_1k)_s \times X^k)_{s \in S_2} \tag{11}$$

Note that r is a natural transformation from ΣU_1 to U_2L.

For an arbitrary S_1-sorted set of variables V, the equations are induced by the map $r_{F_1V} : \Sigma U_1F_1V \to U_2LF_1V$ as in (10), more precisely E_V is defined to be the kernel pair of the map $Ur^\sharp_{F_1V} : U_2F_2\Sigma U_1F_1V \to U_2LF_1V$. We will prove that L is presented by $\langle \Sigma, E \rangle$. For all $k \in \omega_f^{S_1}$ the following diagram is a split coequalizer because E_k is a kernel pair.

$$\begin{tikzcd} E_k \arrow[r, shift left, "\pi_1"] \arrow[r, shift right, "\pi_2"'] & U_2F_2\Sigma U_1F_1k \arrow[l, bend left, "t"] \arrow[r, "U_2r^\sharp_{F_1k}"] & U_2LF_1k \arrow[l, bend left, "s"] \end{tikzcd} \tag{12}$$

One can check that it follows that

$$\begin{tikzcd} U_2F_2E_k \arrow[r, shift left, "U_2\pi_1^\sharp"] \arrow[r, shift right, "U_2\pi_2^\sharp"'] & U_2F_2\Sigma U_1F_1k \arrow[l, bend left, "U_2F_2t \,\circ\, \eta_{U_2F_2\Sigma U_1F_1k}"] \arrow[r, "U_2r^\sharp_{F_1k}"] & U_2LF_1k \arrow[l, bend left, "s"] \end{tikzcd} \tag{13}$$

is again a split coequalizer. U_2 is a monadic functor, hence it creates split coequalizers, and we obtain that

$$F_2 E_k \underset{\pi_2^\sharp}{\overset{\pi_1^\sharp}{\rightrightarrows}} F_2 \Sigma U_1 F_1 k \xrightarrow{r^\sharp_{F_1 k}} L F_1 k \tag{14}$$

is a coequalizer. Now it is straightforward to show that

$$F_2 E_V \underset{\pi_2^\sharp}{\overset{\pi_1^\sharp}{\rightrightarrows}} F_2 \Sigma U_1 F_1 V \xrightarrow{F_2 \Sigma U_1 v^\sharp} F_2 \Sigma U_1 F_1 k \xrightarrow{r^\sharp_{F_1 k}} L F_1 k \tag{15}$$

is a joint coequalizer. This proves that L coincides on finitely generated algebras with the functor presented by the finitary presentation $\langle \Sigma, E \rangle$, and therefore it is presented by $\langle \Sigma, E \rangle$. $\square$

4 Equational logic for higher-order abstract syntax

Syntax with variable binders cannot be captured as an initial algebra in the usual way. But Fiore, Plotkin and Turi [15] (see also Hofmann [22] and Gabbay and Pitts [16]) showed that this is possible if one moves from algebras for a functor on Set to algebras for a functor on a suitable presheaf category. In particular, they showed that λ-terms up to α-equivalence form an initial algebra for a functor. These functors generalize the notion of a signature, but a notion of equational theory for these algebras is missing in [15] (but see the more recent work [14]).

This section starts from the observation that a category of presheaves is a many-sorted variety. From Theorem 3.5 we know that a large class of functors on presheaf categories have a presentation. To illustrate an application of Theorem 3.4 we give an equational presentation of the variety of 'λ-algebras' of [15]. Canonical representatives for λ-terms up to α-equivalence can be obtained in different ways, for example, using the method of De Bruijn levels or the method of De Bruijn indices. Using the method of De Bruijn levels, normal forms up to α-equivalence are obtained by specifying well-formedness rules for λ-terms within a context:

$$\frac{1 \le i \le n}{x_1, \dots, x_n \vdash x_i} \ , \ \frac{x_1, \dots, x_n, x_{n+1} \vdash t}{x_1, \dots, x_n \vdash \lambda x_{n+1}.t} \ , \ \frac{x_1, \dots, x_n \vdash t_1 \quad x_1, \dots, x_n \vdash t_2}{x_1, \dots, x_n \vdash t_1 t_2} \tag{16}$$

The appropriate notion to encompass contexts and the operations allowed on them is the full subcategory $\mathbb{F}$ of Set with objects $\underline{n} = \{1, \dots, n\}$ and $\underline{0} = \emptyset$. The equivalence classes of λ-terms over a countable set of variables $V = \{x_1, x_2, \dots\}$ form a presheaf in $\mathsf{Set}^{\mathbb{F}}$, which we will denote by ΛV_α. Explicitly $\Lambda V_\alpha(\underline{n})$ is defined as the set of equivalence classes of λ-terms with the free variables contained in the set $\{x_1, \dots, x_n\}$. For any morphism $\rho : \underline{n} \to \underline{m}$, $\Lambda V_\alpha(\rho)$ acts on an equivalence class of a term by substituting the free variables x_i with $x_{\rho(i)}$. More generally we can work with an arbitrary presheaf of variables V and again we can see that the λ-terms over V form a presheaf in $\mathsf{Set}^{\mathbb{F}}$. Contexts, which correspond to natural numbers, stratify λ-terms up to α-equivalence, and we can capture this by regarding them as the set of sorts. As we have seen in Section 2, $\mathcal{A} = \mathsf{Set}^{\mathbb{F}}$ is a many-sorted unary variety, the sorts being the set of objects of $\mathbb{F}$, which is isomorphic to the set of non-negative integers $\mathbb{N}$. For this many-sorted variety we denote by $U : \mathcal{A} \to \mathsf{Set}^{\mathbb{N}}$ the forgetful functor and by $F : \mathsf{Set}^{\mathbb{N}} \to \mathcal{A}$ its left adjoint.

We endow $\mathbb{F}$ with the coproduct structure:

$$\begin{tikzcd} & \underline{1} \arrow[d, "new"] \\ \underline{n} \arrow[r, "i"] & \underline{n+1} \end{tikzcd} \tag{17}$$

where i is the inclusion and $new(1) = n+1$. The type constructor for context extension can be defined as a functor $\delta : \mathcal{A} \to \mathcal{A}$ given by $\delta A(\underline{n}) = A(\underline{n+1})$ and

$$\delta(A)(f) = A(f + id_1) \tag{18}$$

for all $A \in \mathcal{A}$ and for all maps f in $\mathbb{F}$. Let $L : \mathcal{A} \to \mathcal{A}$ be the functor given by

$$LX = \delta X + X \times X \tag{19}$$

If V is a presheaf (of variables), then an immediate consequence of Theorem 2.1 of [15] states that ΛV_α is the free L-algebra over V. We obtain the algebraic structure of ΛV_α by giving an equational presentation for $\mathsf{Alg}(L)$, arising from a finitary presentation of the functor L and an equational presentation of the variety $\mathcal{A}$.

4.1 An equational presentation for $\mathcal{A} = \mathsf{Set}^{\mathbb{F}}$

An exhaustive presentation of $\mathcal{A}$ can be obtained if we take an operation symbol for each morphism in $\mathbb{F}$ and if we consider all the equations induced by the composition of morphisms. We can find a more elegant presentation of $\mathcal{A}$ with countably many operations and equations, if we can find a countable set of functions which generate all the functions of $\mathbb{F}$ and a countable set of equations, large enough to prove that any two representations of a function in terms of the generators are equivalent via these equations. Formally, the set of sorts will consist of the non-negative integers. We will consider a signature consisting only of unary operation symbols, whose arity is specified below:

$$\begin{array}{lll} \sigma_n^{(i)} : n \to n & 1 \leq i < n & n > 1 \\ w_n : n \to n+1 & & n \geq 0 \\ c_n : n+1 \to n & & n > 0 \\ \sigma_0 : 0 \to 0 & & \end{array} \tag{20}$$

The intended interpretation is the following: $\sigma_n^{(i)}$ corresponds to the transposition $\sigma_n^{(i)} = (i, i+1)$ of the set $\underline{n}$, c_n corresponds to the contraction $c_n : \underline{n+1} \to \underline{n}$ defined by $c_n(i) = i$

for $i \leq n$ and $c_n(n+1) = n$, and w_n to the inclusion w_n of $\underline{n}$ into $\underline{n+1}$. σ_0 corresponds to the empty map on $\emptyset$. In what follows, we will use the same notation for the operation symbols and for the corresponding morphisms in $\mathbb{F}$, and it should be clear from the context which one we refer to.

Firstly, we consider the equations coming from the presentation of the symmetric group, see for example [34]:

$$\begin{array}{lll} (\sigma_n^{(i)})^2 = id_n & 1 \leq i < n & \\ \sigma_n^{(i)}\sigma_n^{(j)} = \sigma_n^{(j)}\sigma_n^{(i)} & j \neq i \pm 1; 1 \leq i, j < n & (\mathrm{E}_1) \\ (\sigma_n^{(i)}\sigma_n^{(i+1)})^3 = id_n & 1 \leq i < n-1 & \end{array}$$

Each permutation of the set $\underline{n}$ can be written as a composition of transpositions $\sigma_n^{(i)}$ and we choose for each permutation such a representation. The permutations that will appear in equation (E_9) below, should be regarded as abbreviations of their representation in terms of the corresponding $\sigma_n^{(i)}$.

Secondly, we use the next set of equations:

$$\begin{array}{lll} c_n\sigma_{n+1}^{(n)} = c_n & & (\mathrm{E}_2) \\ c_n w_n = id_n & & (\mathrm{E}_3) \\ \sigma_{n+1}^{(i)} w_n = w_n \sigma_n^{(i)} & 1 \leq i < n & (\mathrm{E}_4) \\ \sigma_{n+2}^{(n+1)} w_{n+1} w_n = w_{n+1} w_n & & (\mathrm{E}_5) \\ \sigma_n^{(i)} c_n = c_n \sigma_{n+1}^{(i)} & i < n-1 & (\mathrm{E}_6) \\ c_n \sigma_{n+1}^{(n-1)} \sigma_{n+1}^{(n)} w_n = \sigma_n^{(n-1)} w_{n-1} c_{n-1} & n \geq 2 & (\mathrm{E}_7) \\ c_n c_{n+1} \sigma_{n+2}^{(n)} = c_n c_{n+1} & & (\mathrm{E}_8) \\ ((2, n-1)(1, n) w_{n-1} c_{n-1})^2 = (w_{n-1} c_{n-1} (2, n-1)(1, n))^2 & n \geq 4 & (\mathrm{E}_9) \end{array}$$

We define $E_{\mathcal{A}}$ to be the set of all the equations of the form (E_1)-(E_9).

In order to prove that we have indeed a presentation of $\mathcal{A}$ we will use a known presentation of the monoid of functions from $\underline{n}$ to $\underline{n}$, for $n \geq 4$, given by Aizenštat [8].

Proposition 4.1. *$\mathcal{A}$ is isomorphic to* $\mathsf{Alg}(\Sigma_{\mathcal{A}}, E_{\mathcal{A}})$.

Proof. Given a functor $G : \mathbb{F} \to \mathsf{Set}$ we will construct a $(\Sigma_{\mathcal{A}}, E_{\mathcal{A}})$-algebra. For each $n \in \mathbb{N}$ consider $G(\underline{n})$ as the set of elements of sort n. The operations corresponding to $\sigma_n^{(i)}, c_n$ and w_n are given by $G(\sigma_n^{(i)}), G(c_n)$ and $G(w_n)$ respectively. It is not difficult to check that $(G(\underline{n}))_{n\in\mathbb{N}}$ is indeed a $(\Sigma_{\mathcal{A}}, E_{\mathcal{A}})$-algebra, as all the equations (E_1)-(E_9) are satisfied by the corresponding functions in $\mathbb{F}$.

Conversely, starting with a $(\Sigma_{\mathcal{A}}, E_{\mathcal{A}})$-algebra $(A_n)_{n\in\mathbb{N}}$ we will construct a functor $G : \mathbb{F} \to \mathsf{Set}$. On objects we define $G(\underline{n})$ to be A_n. We define $G(\sigma_n^{(i)}), G(c_n)$ and $G(w_n)$ to be the interpretation of $\sigma_n^{(i)}, c_n$ and w_n respectively. We can define $G(f)$ for any map f of $\mathbb{F}$ if we prove that any morphism in $\mathbb{F}$ can be written as a composition of functions of the form $\sigma_n^{(i)}$,

c_n and w_n. We can conclude that G is a well-defined functor once we show that any two such representations of a function f as composition of the generators $\sigma_n^{(i)}$, c_n and w_n, are equivalent using the equations (E$_1$)-(E$_9$).

Let us prove that any function in $\mathbb{F}$ can be generated using only functions of the form $\sigma_n^{(i)}, c_n, w_n$. We will use several lemmas, which will be proved at the end of the section.

Lemma 4.2. *Let n be a natural number such that $n \geq 4$. For each function $f : \underline{n} \to \underline{n}$ we can choose a canonical representation in terms of $\sigma_n^{(i)}$ and $a_n = w_{n-1}c_{n-1}$, and any other representation in terms of $\sigma_n^{(i)}$ and a_n can be reduced to this canonical one using $E_{\mathcal{A}}$.*

For all positive integers n, k such that $n \geq k$, we call a k-partition of n a k-tuple $p = (i_1, \ldots, i_k)$ such that $i_1 + \cdots + i_k = n$ and $1 \leq i_1 \leq \cdots \leq i_k$. For any k-partition p of n we denote by $N^p_{n,k} : \underline{n} \to \underline{n}$ the function which maps the first i_1 elements of $\underline{n}$ to 1, the next i_2 elements to 2 and so on, the last i_k elements to k. By Lemma 4.2, if $n \geq 4$ then $N^p_{n,k}$ has a canonical representation in terms of $\sigma_n^{(i)}$ and $w_{n-1}c_{n-1}$.

Each function $f : \underline{n} \to \underline{m}$ determines a k-partition of n denoted by p_f, where k is the cardinal of the image of f. There are exactly k nonempty sets among $f^{-1}(1), \ldots, f^{-1}(m)$, and the sum of their cardinalities is n. It is easy to see that there exist permutations π_n of the set $\underline{n}$ and π_m of the set $\underline{m}$, such that:

$$f = \pi_m w_{m-1} \ldots w_k c_k \ldots c_{n-1} N^{p_f}_{n,k} \pi_n \qquad \text{if } n \geq 4$$

or

$$f = \pi_m w_{m-1} \ldots w_k c_k \ldots c_{n-1} \pi_n \qquad \text{if } 1 \leq n < 4$$

We use here the convention that if $k = m$ then no inclusions will appear in the equation above, and similarly if $k = n$ we don't have any contractions. We have shown that any map f has a representation using the generators.

Let us consider another representation of f as composition of generators. We have a corresponding sequence of operation symbols, let us denote it by R.

Lemma 4.3. *R can be reduced to a sequence of the form $\pi'_m w_{m-1} \ldots w_k g$ where g is a sequence of transpositions and contractions and π'_m is a sequence of transpositions of sort m.*

Let us first consider the case $n \geq 4$. There exists a permutation $\tau : \underline{k} \to \underline{k}$ such that we have the following equalities in $\mathbb{F}$:

$$\pi'_m w_{m-1} \ldots w_k \tau = \pi_m w_{m-1} \ldots w_k \tag{21}$$

$$\tau^{-1} g = c_k \ldots c_{n-1} N^{p_f}_{n,k} \pi_n \tag{22}$$

This holds because both $\pi'_m w_{m-1} \ldots w_k$ and $\pi_m w_{m-1} \ldots w_k$ are injective maps from $\underline{k}$ to $\underline{m}$, and their image is the image of f. If we restrict the co-domain of these maps to the image of

f, we obtain two bijective maps, and τ is obtained by composing the latter with the inverse of the former. Let us notice that (22) implies that the following equality is also true in $\mathbb{F}$. Here we use the fact that the image of $N_{n,k}^{p_f}$ is exactly $\underline{k}$.

$$w_{n-1}\dots w_k\tau^{-1}g\pi_n^{-1} = N_{n,k}^{p_f} \tag{23}$$

Notice that if (23) is derivable from the equations, then so is (22); we have to apply (E_3) for $n-k$ times. Therefore it is enough to show that we can derive (21) and (23) from the equations $E_{\mathcal{A}}$. The first follows from the next lemma. We only have to apply (E_4) and multiply with $(\pi_m)^{-1}$.

Lemma 4.4. *If the equality $\tau_m w_{m-1}\dots w_k = w_{m-1}\dots w_k$ holds in $\mathbb{F}$, where τ_m is a permutation on $\underline{m}$, then the equality can be deduced from $E_{\mathcal{A}}$.*

In order to prove (23), recall that g is a sequence of transpositions and contractions, and since it has arity $n\to k$, it must contain precisely $n-k$ contractions. From the next lemma it follows that $w_{n-1}\dots w_k\tau^{-1}g\pi_n^{-1}$ can be reduced to an expression in terms of $\sigma_n^{(i)}$ and $w_{n-1}c_{n-1}$.

Lemma 4.5. *If h is a sequence which contains only transpositions and exactly $n-k$ contractions, then $w_{n-1}\dots w_k h$ can be reduced to an expression written only in terms of transpositions of the form $\sigma_n^{(i)}$ and $a_n = w_{n-1}c_{n-1}$.*

Now we can apply Lemma 4.2 and we obtain that $w_{n-1}\dots w_k\tau^{-1}g\pi_n^{-1}$ can be reduced to $N_{n,k}^{p_f}$ via the equations $E_{\mathcal{A}}$. The proof is complete for $n\geq 4$.

If $n=1$ then the fact that any two representations of f are equivalent via $E_{\mathcal{A}}$ derives from Lemma 4.4.

Assume $n=2$. Using Lemma 4.3, $c_1 = c_1\sigma_2^{(1)}$ and (E_4) we obtain that any representation of f can be reduced to the form $f=\pi_m w_{m-1}\dots w_1 c_1$ where π_m is a sequence of transpositions of the form $\sigma_m^{(i)}$. If $\tau_m w_{m-1}\dots w_1 c_1$ is a different representation of f, then the equality $\pi_m w_{m-1}\dots w_1 = \tau_m w_{m-1}\dots w_1$ holds in $\mathbb{F}$, because c_1 is surjective. The conclusion of the proposition follows easily from Lemma 4.4.

If $n=3$ then we have three possible cases:

1. If the image of f has three elements then any representation of f can be reduced to the form $\pi_m w_{m-1}\dots w_3$, and we can apply Lemma 4.4.

2. The image of f has two elements. Then any representation of f can be reduced to one of the form $\pi_m w_{m-1}\dots w_3 c_2\sigma_3^{(1)}$ if $f(1)=f(3)$ or $\pi_m w_{m-1}\dots w_3 c_2\sigma_3^{(1)}\sigma_3^{(2)}$ if $f(1)=f(2)$ or $\pi_m w_{m-1}\dots w_3 c_2$ if $f(2)=f(3)$. In either case we can deduce that the representations are equivalent by a similar reasoning as above, using Lemma 4.4.

3. Finally if the image of f has one element, then any representation of f can be reduced to the form $\pi_m w_{m-1}\dots w_1 c_1 c_2$ using (E_8) and (E_2). We conclude again using Lemma 4.4.

□

Proof of Lemma 4.2 Let n be a positive integer such that $n \geq 4$. Aizenštat [8], gives a presentation of the monoid of functions from $\underline{n}$ to $\underline{n}$ using the generators of the symmetric group and an additional generator:

$$A = \begin{pmatrix} 1 & 2 & 3 & \dots & n \\ 1 & 1 & 3 & \dots & n \end{pmatrix}$$

Apart from the relations used in the presentation of the symmetric group, Aizenštat proves that the following seven relations are enough to present the monoid:

$$A\sigma_n^{(1)} = \sigma_n^{(3)} A \sigma_n^{(3)} = (3, 4, \dots, n)A(3, 4, \dots, n) = [(1, n)A]^2 = A$$

$$[\sigma_n^{(2)} A]^2 = A\sigma_n^{(2)} A = [A\sigma_n^{(2)}]^2 \tag{24}$$

$$[\sigma_n^{(2)}(1, n)A]^2 = [A\sigma_n^{(2)}(1, n)]^2$$

We will use the fact that $A = (1, n-1)(2, n)w_{n-1}c_{n-1}(1, n-1)(2, n)$. Using Aizenštat's result we can find a representation of f in terms of $\sigma_n^{(i)}$ and $a_n = w_{n-1}c_{n-1}$. It only remains to check that the relations (24) can be deduced from $E_{\mathcal{A}}$. Indeed, $A\sigma_n^{(1)} = A$ can be deduced from (E_2). From (E_4) and (E_6) one can deduce that:

$$\sigma_n^{(i)} w_{n-1}c_{n-1} = w_{n-1}c_{n-1}\sigma_n^{(i)} \qquad \text{for} \;\; 1 \leq i < n-2 \tag{25}$$

But $(1, n-1)(2, n)\sigma_n^{(3)}(1, n-1)(2, n)$ and $(1, n-1)(2, n)(3, 4, \dots n)(1, n-1)(2, n)$ are permutations which leave invariant $n-1$ and n, so they can be represented in terms of $\sigma_n^{(i)}$ for $1 \leq i < n-2$. Therefore we obtain:

$$(1, n-1)(2, n)\sigma_n^{(3)}(1, n-1), (2, n)w_{n-1}c_{n-1} = w_{n-1}c_{n-1}(1, n-1)(2, n)\sigma_n^{(3)}(1, n-1), (2, n)$$

or equivalently $\sigma_n^{(3)} A \sigma_n^{(3)} = A$. Similarly $(3, 4, \dots, n)A(3, 4, \dots, n) = A$ can be derived from $E_{\mathcal{A}}$.

It is not difficult to prove that $((1, n)A)^2 = A$ can be obtained from (25) and $(\sigma_n^{(n-2)} w_{n-1}c_{n-1})^2 = w_{n-1}c_{n-1}$. The latter can be deduced as follows:

$$\begin{aligned} &\sigma_n^{(n-2)} w_{n-1}c_{n-1}\sigma_n^{(n-2)} w_{n-1}c_{n-1} = \\ &\overset{(\mathrm{E}_4)}{=} w_{n-1}\sigma_{n-1}^{(n-2)} c_{n-1}w_{n-1}\sigma_{n-1}^{(n-2)} c_{n-1} = \\ &\overset{(\mathrm{E}_3)}{=} w_{n-1}\sigma_{n-1}^{(n-2)}\sigma_{n-1}^{(n-2)} c_{n-1} = \\ &\overset{(\mathrm{E}_1)}{=} w_{n-1}c_{n-1} \end{aligned} \tag{26}$$

Similarly the relations $[\sigma_n^{(2)} A]^2 = A\sigma_n^{(2)} A = [A\sigma_n^{(2)}]^2$ follow from (25) and:

$$[(n-2, n)w_{n-1}c_{n-1}]^2 = [w_{n-1}c_{n-1}(n-2, n)]^2 = w_{n-1}c_{n-1}(n-2, n)w_{n-1}c_{n-1}$$

which are proved below. Firstly, we have:

$$\begin{aligned} & w_{n-1}c_{n-1}\sigma_n^{(n-2)}\sigma_n^{(n-1)}\sigma_n^{(n-2)}w_{n-1}c_{n-1} = \\ & \stackrel{(\mathrm{E}_4)}{=} w_{n-1}(c_{n-1}\sigma_n^{(n-2)}\sigma_n^{(n-1)}w_{n-1})\sigma_{n-1}^{(n-2)}c_{n-1} = \\ & \stackrel{(\mathrm{E}_7)}{=} w_{n-1}\sigma_{n-1}^{(n-2)}w_{n-2}c_{n-2}\sigma_{n-1}^{(n-2)}c_{n-1} = \\ & \stackrel{(\mathrm{E}_4),(\mathrm{E}_2)}{=} \sigma_n^{(n-2)}w_{n-1}w_{n-2}c_{n-2}c_{n-1} \end{aligned} \tag{27}$$

Now, using (27) we get:

$$\begin{aligned} & (\sigma_n^{(n-2)}\sigma_n^{(n-1)}\sigma_n^{(n-2)}w_{n-1}c_{n-1})^2 = \\ & \stackrel{(27)}{=} \sigma_n^{(n-2)}\sigma_n^{(n-1)}\sigma_n^{(n-2)}\sigma_n^{(n-2)}w_{n-1}w_{n-2}c_{n-2}c_{n-1} = \\ & \stackrel{(\mathrm{E}_1)}{=} \sigma_n^{(n-2)}\sigma_n^{(n-1)}w_{n-1}w_{n-2}c_{n-2}c_{n-1} = \\ & \stackrel{(\mathrm{E}_5)}{=} \sigma_n^{(n-2)}w_{n-1}w_{n-2}c_{n-2}c_{n-1} = \\ & \stackrel{(27)}{=} w_{n-1}c_{n-1}\sigma_n^{(n-2)}\sigma_n^{(n-1)}\sigma_n^{(n-2)}w_{n-1}c_{n-1} \end{aligned} \tag{28}$$

Similarly:

$$\begin{aligned} & (w_{n-1}c_{n-1}\sigma_n^{(n-2)}\sigma_n^{(n-1)}\sigma_n^{(n-2)})^2 = \\ & \stackrel{(27)}{=} \sigma_n^{(n-2)}w_{n-1}w_{n-2}c_{n-2}c_{n-1}\sigma_n^{(n-2)}\sigma_n^{(n-1)}\sigma_n^{(n-2)} = \\ & \stackrel{(\mathrm{E}_8)}{=} \sigma_n^{(n-2)}w_{n-1}w_{n-2}c_{n-2}c_{n-1}\sigma_n^{(n-1)}\sigma_n^{(n-2)} = \\ & \stackrel{(\mathrm{E}_2)}{=} \sigma_n^{(n-2)}w_{n-1}w_{n-2}c_{n-2}c_{n-1}\sigma_n^{(n-2)} = \\ & \stackrel{(\mathrm{E}_8)}{=} \sigma_n^{(n-2)}w_{n-1}w_{n-2}c_{n-2}c_{n-1} = \\ & \stackrel{(27)}{=} w_{n-1}c_{n-1}\sigma_n^{(n-2)}\sigma_n^{(n-1)}\sigma_n^{(n-2)}w_{n-1}c_{n-1} \end{aligned} \tag{29}$$

The last of Aizenštat's equations follows from (E_9) and (25). □

Proof of Lemma 4.3 Let us consider the first w_l which appears from left to right in R. We will prove that we can reduce R such that either w_l disappears or in front of it there are only transpositions. Assume that there are some contractions in front of w_l. If R has the form $\ldots c_l w_l \ldots$ then it can be reduced by (E_3) and w_l disappears. Otherwise, R has the form $\ldots c_l \sigma_{l+1}^{(i_1)} \ldots \sigma_{l+1}^{(i_j)} w_l \ldots$. We will prove that this can be reduced to $\ldots c_l w_l \ldots$ or to $c_l \sigma_{l+1}^{(l-1)} \sigma_{l+1}^{(l)} w_l \ldots$. We are in at least one of the following six possible cases:

1. $i_j < l$ or $i_1 < l-1$. Then R can be further transformed using either (E_4) or (E_6) to an expression such that the number of transpositions between c_l and w_l is reduced by one.

2. $i_j = l$ and $j = 1$. Then w_l disappears as $c_l \sigma_{l+1}^{(l)} w_l$ can be reduced using (E_2) to $c_l w_l$ and further to id_l, using (E_3).

3. $i_j = l$ and $j = 2$ and R has the form $\ldots c_l \sigma_{l+1}^{(l-1)} \sigma_{l+1}^{(l)} w_l \ldots$.

4. $j > 1$ and there exists h such that $1 \leq h < j-1$, $i_{h+1} = i_h - 1$ and $i_{r+1} = i_r + 1$ for all $r > h$. In this case using (E_1) we have that

$$\begin{aligned}
&\sigma_{l+1}^{(i_h)}\sigma_{l+1}^{(i_{h+1})}\sigma_{l+1}^{(i_{h+2})}\sigma_{l+1}^{(i_{h+3})}\dots\sigma_{l+1}^{(i_j)}w_l = \\
&= \sigma_{l+1}^{(i_h)}\sigma_{l+1}^{(i_h-1)}\sigma_{l+1}^{(i_h)}\sigma_{l+1}^{(i_h+1)}\dots\sigma_{l+1}^{(i_j)}w_l = \\
&= \sigma_{l+1}^{(i_h-1)}\sigma_{l+1}^{(i_h)}\sigma_{l+1}^{(i_h-1)}\sigma_{l+1}^{(i_h+1)}\dots\sigma_{l+1}^{(i_j)}w_l = \\
&= \sigma_{l+1}^{(i_h-1)}\sigma_{l+1}^{(i_h)}\sigma_{l+1}^{(i_h+1)}\dots\sigma_{l+1}^{(i_j)}\sigma_{l+1}^{(i_h-1)}w_l = \\
&= \sigma_{l+1}^{(i_h-1)}\sigma_{l+1}^{(i_h)}\sigma_{l+1}^{(i_h+1)}\dots\sigma_{l+1}^{(i_j)}w_l\sigma_{l+1}^{(i_h-1)}
\end{aligned} \tag{30}$$

so the number of transpositions between c_l and w_l is reduced by one.

5. $j > 1$ and there exists h such that $1 \leq h < j$ and $i_{h+1} > i_h + 1$ and $i_{r+1} = i_r + 1$ for all $h < r \leq j-1$. In this case we have that $i_h < l$ so we can apply (E_1) and (E_4) to get:

$$\begin{aligned}
\sigma_{l+1}^{(i_h)}\sigma_{l+1}^{(i_{h+1})}\dots\sigma_{l+1}^{(i_j)}w_l &= \sigma_{l+1}^{(i_{h+1})}\sigma_{l+1}^{(i_h)}\dots\sigma_{l+1}^{(i_j)}w_l \\
&= \sigma_{l+1}^{(i_{h+1})}\sigma_{l+1}^{(i_{h+2})}\dots\sigma_{l+1}^{(i_j)}\sigma_{l+1}^{(i_h)}w_l \\
&= \sigma_{l+1}^{(i_{h+1})}\sigma_{l+1}^{(i_{h+2})}\dots\sigma_{l+1}^{(i_j)}w_l\sigma_{l+1}^{(i_h)}
\end{aligned} \tag{31}$$

6. $j > 1$ and there exists h such that $1 \leq h < j-1$ and $i_{h+1} < i_h - 1$ and $i_{r+1} = i_r + 1$ for all $h < r \leq j-1$. In this case we have that $\sigma_{l+1}^{(i_h)}$ commutes with $\sigma_{l+1}^{(i_{h+s})}$ for s such that $i_{h+s} < i_h - 1$. This case can be reduced to the case (4) above and so, also in this case the number of transpositions between c_l and w_l decreases.

We can conclude that repeating the reductions as above we can either make w_l disappear or reduce R to the form $\dots c_l\sigma_{l+1}^{(l-1)}\sigma_{l+1}^{(l)}w_l\dots$. In the latter case we can apply (E_7) and R can be transformed to $\dots\sigma_l^{(l-1)}w_{l-1}c_{l-1}\dots$. So either w_l disappears, or the first w to appear in the new sequence is w_{l-1} and it has a smaller number of contractions in front of it, compared to w_l. Continuing this procedure we can reduce R to a sequence in which the first w which appears (if there is any) has no contractions in front of it. Now we can apply the same algorithm to reduce the remaining part of R. So we will get an expression of the form $\tau_{m'}w_{m'-1}\dots\tau_{k'}w_{k'}g$, where g has only transpositions and contractions and $\tau_{m'},\dots,\tau_{k'}$ are sequences of transpositions of the form $\sigma_{m'}^{(i)},\dots,\sigma_{k'}^{(i)}$ respectively. But whenever we have w followed by a transposition, we can apply (E_4) to move the transposition in front of w. So in the end R can be reduced to an expression of the form $\theta_{m'}w_{m'-1}\dots w_{k'}g$. It only remains to notice that we must have $m = m'$ and $k = k'$. The first equality holds because of the arity of R. Moreover we must have that $f = \theta_{m'}w_{m'-1}\dots w_{k'}g$. Since g is a surjective map and $\theta_{m'}w_{m'-1}\dots w_{k'}$ is injective, we must have that the cardinality of the image of g is the same as that of the image of f, namely k. We conclude the $w_{k'}$ must have arity $k \to k+1$, thus $k = k'$. □

Proof of Lemma 4.4 Since the equality $\tau_m w_{m-1}\dots w_k = w_{m-1}\dots w_k$ holds in $\mathbb{F}$, we deduce that τ_m leaves invariant the elements $1,\dots,k$, therefore, using (E_1) we can express τ_m as a composition of j transposition of the form $\sigma_m^{(i)}$ where $i \geq k+1$. The proof is by induction on j. It is obvious for $j = 0$. Assume the statement of the lemma holds for j, and let us prove that it also holds for $j+1$. Assume that $\sigma_m^{(i_1)} \dots \sigma_m^{(i_{j+1})} w_{m-1} \dots w_k = w_{m-1} \dots w_k$ holds in $\mathbb{F}$, where $i_1, \dots, i_{j+1} > k$. Applying repeatedly (E$_4$), we can deduce that $\sigma_m^{(i_1)} \dots \sigma_m^{(i_{j+1})} w_{m-1} \dots w_k = \sigma_m^{(i_1)} \dots \sigma_m^{(i_j)} w_m \dots \sigma_{i_{j+1}+1}^{(i_{j+1})} w_{i_{j+1}} w_{i_{j+1}-1} \dots w_k$. Since $i_{j+1} > k$ and $\sigma_{i_{j+1}+1}^{(i_{j+1})} w_{i_{j+1}} w_{i_{j+1}-1} = w_{i_{j+1}} w_{i_{j+1}-1}$ by (E$_5$), we obtain that $\sigma_m^{(i_1)} \dots \sigma_m^{(i_{j+1})} w_{m-1} \dots w_k = \sigma_m^{(i_1)} \dots \sigma_m^{(i_j)} w_{m-1} \dots w_k$. The conclusion follows from the induction hypothesis. □

Proof of Lemma 4.5 We use induction on $n-k$. If $n-k = 0$ we have nothing to prove. Now assume the statement of the lemma is valid for $n-k = l$ and let us prove it for $n-k = l+1$. We have that $w_{n-1} \dots w_k h = w_{n-1} \dots w_k \sigma_k^{(i_1)} \dots \sigma_k^{(i_j)} c_k h'$ where h' is a sequence containing transpositions and l contractions. We have:

$$\begin{aligned}
& w_{n-1} \dots w_k \sigma_k^{(i_1)} \dots \sigma_k^{(i_j)} c_k h' = \\
& \overset{(\mathrm{E}_4)}{=} \sigma_n^{(i_1)} \dots \sigma_n^{(i_j)} w_{n-1} \dots w_k c_k h' = \\
& \overset{(\mathrm{E}_7)}{=} \sigma_n^{(i_1)} \dots \sigma_n^{(i_j)} w_{n-1} \dots w_{k+1} \sigma_{k+1}^{(k)} c_{k+1} \sigma_{k+2}^{(k)} \sigma_{k+2}^{(k+1)} w_{k+1} h' = \\
& \overset{(\mathrm{E}_4),(\mathrm{E}_7)}{=} \dots = \\
& \overset{(\mathrm{E}_4),(\mathrm{E}_7)}{=} \sigma_n^{(i_1)} \dots \sigma_n^{(i_j)} \sigma_n^{(k)} \dots \sigma_n^{(n-1)} w_{n-1} c_{n-1} \tau_n w_{n-1} \dots w_{k+1} h'
\end{aligned}$$

where $\tau_n : n \to n$ is a sequence of transpositions of the form $\sigma_n^{(i)}$. To finalize the proof we just have to apply the induction hypothesis for $w_{n-1} \dots w_{k+1} h'$. □

4.2 An equational presentation for binding algebras

An equational presentation for the binding algebras $\mathsf{Alg}(L)$ can be obtained from the presentation of $\mathcal{A}$ and a presentation for L. We can easily check that the functor L described at the beginning of this section preserves sifted colimits. Therefore, by Theorem 3.5, L has a finitary presentation. However the presentation obtained in the proof of this theorem is an exhaustive one, the set of operation symbols for each sort is infinite. In this section we give a more efficient presentation for L, considering for each $n \in \mathbb{N}$ the operation symbols $\mathsf{lam}_n, \mathsf{app}_n$ which semantically correspond to λ-abstraction and application. The respective signature functor $\Sigma_L : \mathsf{Set}^{\mathbb{N}} \to \mathsf{Set}^{\mathbb{N}}$ is given by

$$(\Sigma_L X)_n = \{\mathsf{lam}_{n+1}\} \times X_{n+1} + \{\mathsf{app}_n\} \times X_n \times X_n \tag{32}$$

For simplicity we will denote $(\mathsf{lam}_{n+1}, t) \in (\Sigma_L X)(n)$ by $\mathsf{lam}_{n+1} t$ and an element $(\mathsf{app}_n, t_1, t_2) \in (\Sigma_L X)(n)$ by $\mathsf{app}_n(t_1, t_2)$. For any presheaf $A \in \mathcal{A}$ let $\rho_A : \Sigma U A \to U L A$ be the map defined by

$$\begin{array}{ll}
\mathsf{lam}_{n+1} t \mapsto t & \forall t \in A(n+1) = (\delta A)(n) \\
\mathsf{app}_n(t_1, t_2) \mapsto (t_1, t_2) & \forall t_1, t_2 \in A(n)
\end{array}$$

The equations E_L should correspond to the kernel pair of the adjoint transpose $\rho_A^\sharp : F\Sigma U A \to LA$. For any $(X_n)_n \in \mathsf{Set}^{\mathbb{N}}$ we have that $F(X_n)_n = \coprod_{n \in \mathbb{N}} X_n \cdot \hom(\underline{n}, -)$ where $\cdot$ denotes the copower. For example $X_n \cdot \hom(\underline{n}, \underline{m})$ consists of $|X_n|$ copies of $\hom(\underline{n}, \underline{m})$. In the remaining of this section, for $x \in X_n$ and $f : \underline{n} \to \underline{m}$, we will denote by fx the element of $X_n \cdot \hom(\underline{n}, \underline{m})$ which is the copy of f corresponding to x. Now we can give the map $\rho^\sharp_A$ explicitly:

$$\begin{aligned} &f(\mathsf{lam}_{n+1}\alpha) \mapsto (\delta A)(f)(\alpha) && \forall\ \alpha \in A(\underline{n+1}) \\ &f\mathsf{app}_n(\alpha_1, \alpha_2) \mapsto (A(f)(\alpha_1), A(f)(\alpha_2)) && \forall\ \alpha_1, \alpha_2 \in A(\underline{n}) \end{aligned} \tag{33}$$

We will consider E_L to be the set of equations of the following form

$$\begin{aligned} &\sigma^{(i)}_n \mathsf{lam}_{n+1} t = \mathsf{lam}_{n+1} \sigma^{(i)}_{n+1} t \\ &w_n \mathsf{lam}_{n+1} t = \mathsf{lam}_{n+2} \sigma^{(n+1)}_{n+2} w_{n+1} t \\ &c_n \mathsf{lam}_{n+2} t' = \mathsf{lam}_{n+1} \sigma^{(n)}_{n+1} c_{n+1} \sigma^{(n)}_{n+2} \sigma^{(n+1)}_{n+2} t' \\ &\sigma^{(i)}_n \mathsf{app}_n(t_1, t_2) = \mathsf{app}_n(\sigma^{(i)}_n t_1, \sigma^{(i)}_n t_2) \\ &w_n \mathsf{app}_n(t_1, t_2) = \mathsf{app}_{n+1}(w_n t_1, w_n t_2) \\ &c_n \mathsf{app}_{n+1}(t'_1, t'_2) = \mathsf{app}_n(c_n t'_1, c_n t'_2) \end{aligned} \tag{34}$$

where t is a variable of sort $n+1$, t' is a variable of sort $n+2$, t_1, t_2 are variables of sort n, t'_1,t'_2 are variables of sort $n+1$ and n is an arbitrary positive integer. The equations for lam are obtained from (17) and (18).

Proposition 4.6. *L is presented by $\langle \Sigma_L, E_L \rangle$.*

Proof. Suppose $A \in \mathcal{A}$. We have to check that LA is a coequalizer as in diagram (4). First let us check that the equations E_L are satisfied. Let t be a variable of sort $n+1$ and let $v : V \to UA$ be a valuation such that $v(t) = \alpha \in (UA)(n+1)$. Observe that $\rho^\sharp_A(F\Sigma_L U v^\sharp(f(\mathsf{lam}_{n+1}t))) = \rho^\sharp_A(f(\mathsf{lam}_{n+1}\alpha)) = \delta(A)(f)(\alpha) = A(f + id_1)(\alpha)$. On the other hand, $\rho^\sharp_A(F\Sigma_L U v^\sharp(\mathsf{lam}_{n+1}(f + id_1)t)) = \rho^\sharp_A(\mathsf{lam}_{n+1} A(f + id_1)(\alpha)) = \delta(A)(id_n)(A(f + id_1)(\alpha)) = A(f+id_1)(\alpha)$. Therefore, the first three of the equations (34) are satisfied because we have:

$$\sigma^{(i)}_{n+1} = \sigma^{(i)}_n + id_1$$

$$\sigma^{(n+1)}_{n+2} w_{n+1} = w_n + id_1$$

$$\sigma^{(n)}_{n+1} c_{n+1} \sigma^{(n)}_{n+2} \sigma^{(n+1)}_{n+2} = c_n + id_1$$

As for the latter three it suffice to notice that

$$\rho^\sharp_A(F\Sigma_L U v^\sharp(f\mathsf{app}_n(t_1, t_2))) = (A(f)(v(t_1)), A(f)(v(t_2)))$$

$$\rho^\sharp_A(F\Sigma_L U v^\sharp(\mathsf{app}_n(ft_1, ft_2))) = (A(f)(v(t_1)), A(f)(v(t_2)))$$

Conversely, suppose that $(f(\mathsf{lam}_{n+1}\alpha), g(\mathsf{lam}_{m+1}\beta))$ is in the kernel pair of $\rho^\sharp_A$, for some $f : \underline{n} \to \underline{k}$, $g : \underline{m} \to \underline{k'}$, $\alpha \in A(\underline{n+1})$ and $\beta \in A(\underline{m+1})$. This means that

$$A(f + id_1)(\alpha) = A(g + id_1)(\beta) \tag{35}$$

so in particular we deduce that $k = k'$. We have to show that $f(\mathsf{lam}_{n+1}\alpha)$ and $g(\mathsf{lam}_{m+1}\beta)$ can be proved equal in $F\Sigma_L UA$ using the equations (34). Let t be a variable of sort $n+1$.

Then the pair $(f(\mathsf{lam}_{n+1}t), \mathsf{lam}_{k+1}(f + id_1)t)$ belongs to the congruence relation generated by the equations E_L in $F\Sigma_L UFV$, and so does $(g(\mathsf{lam}_{m+1}s), \mathsf{lam}_{k+1}(g + id_1)s)$ for a variable s of sort $m+1$. In order to prove this, we can write f and g as compositions of $\sigma_n^{(i)}$,c_n and w_n and use the following observations:

1. For any functions h, h' which can be composed in $\mathbb{F}$, we have that $h \circ h' + id_1 = (h + id_1) \circ (h' + id_1)$.

2. For any functions h, h', which can be composed in $\mathbb{F}$, we have that $h \circ h'(\mathsf{lam}_{n+1}t)$ and $h(\mathsf{lam}_{k+1}(h' + id_1)t))$ are congruent via E_L.

Let $v : V \to UA$ such that $v(t) = \alpha$ and $v(s) = \beta$. We can see that $F\Sigma_L Uv^\sharp(\mathsf{lam}_{k+1}(f + id_1)t) = \mathsf{lam}_{k+1}A(f + id_1)(\alpha)$ and analogously $F\Sigma_L Uv^\sharp(\mathsf{lam}_{k+1}(g + id_1)s) = \mathsf{lam}_{k+1}A(g + id_1)(\beta)$. Using (35) we can conclude that $F\Sigma_L Uv^\sharp(f(\mathsf{lam}_{n+1}t)) = f(\mathsf{lam}_{n+1}\alpha)$ and $F\Sigma_L Uv^\sharp(g(\mathsf{lam}_{m+1}s)) = g(\mathsf{lam}_{m+1}\beta)$ can be proved equal via the equations E_L. A similar, but easier argument works for the elements in the kernel pair of $\rho_A^\sharp$ of the form $(f\mathsf{app}_n(\alpha_1, \alpha_2), g\mathsf{app}_m(\beta_1, \beta_2))$. $\square$

Remark 4.7. *The presentation of L depends on the operations $\Sigma_{\mathcal{A}}$ used to describe $\mathcal{A} = \mathsf{Set}^{\mathbb{F}}$ but is independent of the equations $E_{\mathcal{A}}$.*

Representing different implementations of λ-terms If V is the presheaf defined by $V(\rho) = \rho$ for all morphisms ρ in $\mathbb{F}$, the free L-algebra over V gives an implementation of λ-terms by the De Bruijn levels method. In [15] it is suggested that different implementations of λ-terms can be obtained by equipping $\mathbb{F}$ with different coproduct structures. But this implies working with a different functor than L. Instead, we can use another approach, namely to consider the free L-algebra over different presheaves of variables. For example, if W is the presheaf of variables defined explicitly by

$$W(\underline{n}) = \underline{n} \quad W(c_n)(1) = 1 \quad W(c_n)(i) = i - 1; i > 1$$
$$W(w_n)(i) = i + 1$$
$$W(\sigma_n^{(i)}) = \sigma_n^{(n-i)}$$

we obtain the presheaf ΛW_α of λ-terms implemented by the De Bruijn indices method.

5 First-Order Logic

The second application (a treatment of first-order logic) builds on the first (presenting functors on presheaf-categories) and leads us to the third (coalgebraic logic). The basic observation, presumably going back to Lawvere, is that presheaves taking values in the category BA of Boolean algebras

$$A : \mathbb{F}_+ \to \mathsf{BA}$$

where the weakenings w_n have left-adjoints $\exists_n$

$$\exists_n a \leq b \;\Leftrightarrow\; a \leq w_n b \tag{36}$$

are (algebraic) models of first-order logic (we write $w_n b$ for $A(w_n)(b)$).

In this section we will: (1) show how to obtain algebraic models of first-order logic as algebras for a functor Q on $\mathsf{BA}^{\mathbb{F}_+}$ satisfying the additional equations (36); (2) show that these Q-algebras are equivalent to the polyadic algebras of Halmos [19]; (3) dualise these Q-algebras to obtain a coalgebraic semantics of first-order logic.

Note that $\mathsf{BA}^{\mathbb{F}_+}$ is a many-sorted subvariety of $\mathsf{Set}^{\mathbb{F}_+}$. A presentation for $\mathsf{Set}^{\mathbb{F}_+}$ can be obtained from the presentation of $\mathsf{Set}^{\mathbb{F}}$ just by dropping the operation symbols, σ_0 and w_0 and the equations involving them. $\mathsf{BA}^{\mathbb{F}_+}$ is the subvariety of $\mathsf{Set}^{\mathbb{F}_+}$, obtained by adding for each sort the Boolean connectives $\vee_n$ and $\neg_n$, satisfying the usual axioms for Boolean algebras, plus $w_n \vee_n = \vee_{n+1} w_n$, $c_n \vee_{n+1} = \vee_n c_n$ and $\sigma_n^{(i)}(x \vee_n y) = \sigma_n^{(i)} x \vee_n \sigma_n^{(i)} y$, and the analogous equations for $\neg_n$.

5.1 Algebraic Semantics of First-Order Logic

We are looking for algebras $\mathsf{Q}A \to A$ where the structure at sort $\underline{n}$, $(\mathsf{Q}A)(\underline{n}) \to A(\underline{n})$ interprets the quantifier $\exists_n$ binding the new name in $\underline{n+1}$. Thus, the quantifier corresponds to a map $A(\underline{n+1}) \to A(\underline{n})$ and as it is the case for an existential quantifier, this map preserves joins but not meets. Since arrows in BA are Boolean homomorphism, we account for this by letting $(\mathsf{Q}A)(\underline{n})$ be the free BA over the finite-join-semilattice $A(\underline{n+1})$, or, explicitly

Definition 5.1. *Define $\mathsf{Q} : \mathsf{BA}^{\mathbb{F}_+} \to \mathsf{BA}^{\mathbb{F}_+}$ as the functor mapping $A \in \mathsf{BA}^{\mathbb{F}_+}$ to the presheaf*

- *generated, at sort $\underline{n}$, by $\exists_n a$, $a \in A(\underline{n+1})$*
- *modulo equations specifying that $\exists_n$ preserves finite joins, explicitly $\exists_n(0) = 0$ and $\exists_n(a \vee b) = \exists_n a \vee \exists_n b$.*

Remark 5.2. *Boolean algebra homomorphisms $\mathsf{Q}A(\underline{n}) \to A(\underline{n})$ are in bijective correspondence with finite-join preserving maps $A(\underline{n+1}) \to A(\underline{n})$.*

Furthermore, using the (co)unit of the adjunction, the two implications (36) are easily transformed into to equations (recall $a \leq b \;\Leftrightarrow\; a = a \wedge b$), leading to

Definition 5.3. *The category of FOL-algebras is the category of those Q-algebras satisfying the additional equations $\phi \leq w_n \exists_n \phi$ and $\exists_n w_n \psi \leq \psi$, where ϕ is a variable of sort $\underline{n+1}$ and ψ is a variable of sort $\underline{n}$.*

Algebraic semantics of first-order logic was first studied by Tarski [21] and Halmos [19]. A polyadic algebra on a set of variables V is a Boolean algebra with some additional structure that captures quantifiers and an action of the set of transformations of V, subject to several axioms. If A be a Boolean algebra, a map $\exists : A \to A$ is called a **quantifier** if

$\exists 0 = 0$
$\exists p \geq p$ for all $p \in A$
$\exists p \vee \exists q = \exists(p \vee q)$ for all $p, q \in A$
$\exists\exists p = \exists p$ for all $p \in A$
$\exists\neg\exists p = \neg\exists p$ for all $p \in A$

Definition 5.4. *A polyadic algebra $\mathbb{A}$ over a set of variables V is a tuple $(A, V, \mathcal{S}, \exists)$ such that A is a Boolean algebra, $\mathcal{S} : V^V \to \mathsf{End}A$ and $\exists$ is a map from $\mathcal{P}V$ to the set of quantifiers on A, such that*

(P1) $\exists(\emptyset)$ *is the identity map on A.*
(P2) $\exists(J \cup K) = \exists(J)\exists(K)$ *for all $J, K \subseteq V$*
(P3) $\mathcal{S}$ *maps the identity on V to the identity on A.*
(P4) $\mathcal{S}(\sigma\tau) = \mathcal{S}(\sigma)\mathcal{S}(\tau)$ *for all $\sigma, \tau \in V^V$*
(P5) $\mathcal{S}(\sigma)\exists(J) = \mathcal{S}(\tau)\exists(J)$, *if σ and τ coincide on $V \setminus J$.*
(P6) $\exists(J)\mathcal{S}(\tau) = \mathcal{S}(\tau)\exists(\tau^{-1}J)$ *for all transformations τ which are injective when restricted to $\tau^{-1}J$.*

Definition 5.5. *A polyadic algebra $\mathbb{A} = (A, V, \mathcal{S}, \exists)$ is called locally finite if for each $P \in A$ there exists a finite set $W \subseteq V$ such that $\exists(J)P = P$ for all $J \subset V$ such that J and W are disjoint.*

Ouellet [32] reformulated Halmos's polyadic algebras using Boolean-valued presheaves. He characterized the locally finite polyadic algebras on a set of variables V as Boolean algebra objects in the category of *locally finite V-actions* that admit suprema indexed by V. Note that any locally finite polyadic algebra is equipped with a V-action given by $(P3)$ and $(P4)$, which is locally finite because of $(P5)$. Locally finite V-actions also appear (under the name of nominal substitutions) in Staton [37] in his study of the open bisimulation of π-calculus.

Ouellet uses the equivalence [31] between the category of locally finite V-actions and $\mathsf{Set}^{\mathbb{F}_+}$, given by two functors $\sharp$ and $\flat$. Based on this, we show how this equivalence restricts to an equivalence between FOL-algebras and locally finite polyadic algebras.

Theorem 5.6. *The category of FOL-algebras is equivalent to the category of locally finite polyadic algebras.*

Proof. First we construct a functor from FOL-algebras to locally finite polyadic algebras. Let $\alpha : \mathsf{Q}A \to A$ be a FOL-algebra. Let us fix an infinite set of variables V.

We consider the Boolean algebra $A^\flat = \mathsf{Lan}_i A(V)$, where $\mathsf{Lan}_i A$ is the left Kan extension of A along the inclusion $i : \mathbb{F}_+ \to \mathsf{Set}$. Notice that $A^\flat$ is computed as a colimit in the comma category (i, V), more explicitly, it is isomorphic to $\varinjlim_{f:\underline{n}\to V} A(\underline{n})$. If P is an element of $A(\underline{n})$, we will denote by $[P(v_1, \ldots, v_n)]$ the equivalence class of the P which corresponds to the function $f : \underline{n} \to V$ that maps $i \in \underline{n}$ to v_i.

For any transformation $\xi : V \to V$ we define $\mathcal{S}(\xi)$ to be the Boolean algebra morphism $\mathsf{Lan}_i(A)(\xi)$. Notice that each element of $A^\flat$ is finitely supported and has a minimal support denoted by $\mathsf{supp}(x)$. Moreover if $x \in A^\flat$ has the support $\{v_1, \ldots, v_n\}$ for some $n \geq 1$, then there exists $P \in A(\underline{n})$ such that $x = [P(v_1, \ldots, v_n)]$. If x has empty support, then for any tuple of variables $(v_1, \ldots, v_n)$ there exists $P \in A(\underline{n})$ such that $x = [P(v_1, \ldots, v_n)]$.

Next, for each subset $W \subseteq V$ we define an existential quantifier $\exists W$. First we do this for singleton sets. Assume $v \in V$, and $x \in A^\flat$ is such that $\mathsf{supp}(x) \setminus \{v\} = \{v_1, \ldots, v_n\}$. There exists a unique $P \in A(\underline{n+1})$ such that $x = [P(v_1, \ldots, v_n, v)]$. We define $\exists v(x) = [(\exists_n P)(v_1, \ldots, v_n)]$. Note that $\exists_n P$ is just an abbreviation for $\alpha_n(\exists_n P)$.

Remark 5.7. *Note that* $\mathsf{supp}(\exists v(x)) = \mathsf{supp}(x) \setminus \{v\}$.

We need to show that $\exists v$ is indeed an existential quantifier on $A^\flat$.

1. $\exists v 0 = 0$ follows from $\exists_n 0_{n+1} = 0_n$.

2. Let us prove that $\exists v(x) \geq x$. With the notations above we have that

$$\begin{aligned} \exists v(x) &= [(\exists_n P)(v_1, \ldots, v_n)] \\ &= [(w_n \exists_n P)(v_1, \ldots, v_n, v)] \\ &\geq [P(v_1, \ldots, v_n, v)] \\ &= x \end{aligned} \tag{37}$$

3. The fact that $\exists v(x \vee y) = \exists v(x) \vee \exists v(y)$ follows from the corresponding equation for $\exists_n$.

4. Let us prove that $\exists v(\exists v(x)) = \exists v(x)$. Using Remark 5.7 it is enough to show that $\exists v(x) = x$ for all x whose support does not contain v. Indeed, if x is such that $v \notin \mathsf{supp}(x) = \{v_1, \ldots, v_n\}$, then $x = [P(v_1, \ldots, v_n)]$ for some $P \in A(\underline{n})$. Then $\exists v(x) = [(\exists_n w_n P)(v_1, \ldots, v_n)] \leq [P(v_1, \ldots, v_n)] = x$. On the other hand we know that $\exists v(x) \geq x$.

5. In order to prove that $\exists v(\neg \exists v(x)) = \neg \exists v(x)$ we use the same argument as above, plus the observation that $\mathsf{supp}(\neg x) = \mathsf{supp}(x)$ for all $x \in A^\flat$.

Lemma 5.8. *For $u, v \in V$ and $x \in A^\flat$ we have $\exists v(\exists u(x)) = \exists u(\exists v(x))$*

Proof. There exists $P \in A(\underline{n+2})$ such that $x = [P(v_1, \ldots, v_n, u, v)$, where $\{v_1, \ldots, v_n\} = \mathsf{supp}(x) \setminus \{u, v\}$. It remains to show that

$$\exists_n \exists_{n+1} \sigma_{n+2}^{(n+1)}(P) = \exists_n \exists_{n+1}(P) \tag{38}$$

From the equations it follows that

$$\begin{aligned} & w_{n+1} w_n \exists_n \exists_{n+1}(P) \geq P \\ \Leftrightarrow \quad & \sigma_{n+2}^{(n+1)} w_{n+1} w_n \exists_n \exists_{n+1}(P) \geq \sigma_{n+2}^{(n+1)} P \\ \Leftrightarrow \quad & w_{n+1} w_n \exists_n \exists_{n+1}(P) \geq \sigma_{n+2}^{(n+1)} P \\ \Leftrightarrow \quad & \exists_n \exists_{n+1}(P) \geq \exists_n \exists_{n+1} \sigma_{n+2}^{(n+1)}(P) \end{aligned} \tag{39}$$

Applying the last inequality for $\sigma_{n+2}^{(n+1)}(P)$ instead of P, we get that $\exists_n \exists_{n+1}(P) \leq \exists_n \exists_{n+1} \sigma_{n+2}^{(n+1)}(P)$, so in fact we have equality. $\square$

Now we can define the existential quantifier $\exists W$ for an arbitrary subset $W \subseteq V$. If $x \in A^\flat$ is such that $\mathsf{supp}(x) \cap W = \{v_1, \ldots, v_n\}$, then we define $\exists W(x) = \exists v_1 \ldots \exists v_n(x)$. The above lemma implies that $\exists W$ is well defined.

We have to show that these existential quantifiers satisfy the equations defining a polyadic algebra. It is straightforward to check (P1)-(P5), so we will only give the proof for (P6). Assume $W \subseteq V$ and $\xi \in V^V$ is injective when restricted to $\xi^{-1}(W)$. We need to show that $\exists W \circ \xi = \xi \circ \exists \xi^{-1}(W)$. This is immediate using the observation that $\mathsf{supp}(\xi(x)) \subseteq \xi(\mathsf{supp}(x))$.

Conversely, let $(\mathbb{A}, V, \mathcal{S}, \exists)$ be a locally finite polyadic algebra and let us construct a FOL-algebra $\mathbb{A}^\sharp$. The map $\mathcal{S} : V^V \to \mathsf{End}(\mathbb{A})$ determines a V-action structure on $\mathbb{A}$ such that each element is finitely supported. For each $n > 0$ define $\mathbb{A}^\sharp(n)$ to be the set of V-action morphisms from V^n to $\mathbb{A}$, where V^n is endowed with the component-wise evaluation action. If $f : \underline{n} \to \underline{m}$ is a morphism in $\mathbb{F}$, and $P : V^n \to \mathbb{A}$ is an element of $\mathbb{A}^\sharp(\underline{n})$ then

$$\mathbb{A}^\sharp(f)(P)(v_1, \ldots, v_m) = P(v_{f(1)}, \ldots, v_{f(n)})$$

We have to construct an algebra $\alpha : \mathsf{Q}\mathbb{A}^\sharp \to \mathbb{A}^\sharp$. This will be determined by the maps $\exists_n : \mathbb{A}^\sharp(\underline{n+1}) \to \mathbb{A}^\sharp(\underline{n})$ defined as follows: For $P \in \mathbb{A}^\sharp(\underline{n+1})$ define $(\exists_n P)(v_1, \ldots, v_n) = \exists v(P(v_1, \ldots, v_n, v))$ for some v distinct from all the v_i-s. From (P6) it follows that this is well-defined. It is trivial to check that $\exists_n$ preserves joins.

We can check that for all $P \in \mathbb{A}^\sharp(\underline{n})$ we have that $\exists_n w_n P = P$. Indeed $\exists_n w_n P(v_1, \ldots, v_n) = (\exists v)((w_n P)(v_1, \ldots, v_n, v))$ for some v different than $v_1, \ldots, v_n$. Therefore $\exists_n w_n P(v_1, \ldots, v_n) = (\exists v)(P(v_1, \ldots, v_n)) = P(v_1, \ldots, v_n)$. The last equality holds because $\mathsf{supp}(P(v_1, \ldots, v_n)) \subseteq \{v_1, \ldots, v_n\}$ does not contain v.

For $P \in \mathbb{A}^\sharp(\underline{n+1})$ we have that $(w_n \exists_n P)(v_1, \ldots, v_n, v_{n+1}) = (\exists_n P)(v_1, \ldots, v_n) = (\exists v)(P(v_1, \ldots, v_n, v) \geq P(v_1, \ldots, v_n, v_{n+1})$ for some $v \notin \{v_1, \ldots, v_n\}$. The last inequality follows from (P5) and the fact that $(\exists v)(P(v_1, \ldots, v_n, v)) \geq P(v_1, \ldots, v_n, v)$.

One can check that the functors $\flat$ and $\sharp$ give an equivalence of categories.

$\square$

5.2 Coalgebraic Semantics of First-Order Logic

Based on the duality (dual adjunction) between Boolean algebras and Sets, we will exhibit the coalgebraic dual of FOL-algebras and relate them to (standard) models of first-order logic. We start by dualising the algebraic side described above. The technical justification of the next two definitions will be the duality theorem 5.13 below.

Definition 5.9. *Define* $\mathsf{P} : \mathsf{Set}^{\mathbb{F}_+^{\mathrm{op}}} \to \mathsf{Set}^{\mathbb{F}_+^{\mathrm{op}}}$ *via* $(\mathsf{P}X)(\underline{n}) = \mathcal{P}(X(\underline{n+1}))$.

We consider coalgebras $X \to \mathsf{P}X$ as models of first-order logic. $X(\underline{1})$ is the carrier of the model and formulas in one free variable will be interpreted as subsets of $X(\underline{1})$. Similarly, formulas in two free variables will be interpreted as subsets of $X(\underline{2})$. The weakening $w_1 : \underline{1} \to \underline{2}$ gives rise to a 'projection' $X(\underline{2}) \to X(\underline{1})$. This terminology derives from the case where $X(\underline{2}) = X(\underline{1}) \times X(\underline{1})$. This observation leads to the following

Example 5.10. *A **standard model** is given by*

- *a binary-product preserving functor $X : \mathbb{F}_+^{\mathrm{op}} \to \mathsf{Set}$. Writing X_1 for $X(\underline{1})$, this means that we can identify $X(\underline{n})$ with X_1^n and that $X(f : \underline{n} \to \underline{m})$ is given by $X_1^m \cong X_1^{\underline{m}} \xrightarrow{X_1^f} X_1^{\underline{n}} \cong X_1^n$. For example,*
  - *$X(w_1)$ is the projection $X_1^2 \to X_1$ mapping (x_1, x_2) to x_1,*
  - *$X(c_1)$ is the diagonal $X_1 \to X_1^2$ mapping x to (x, x),*
  - *$X(\sigma_2^{(1)})$ is the swap $X_1^2 \to X_1^2$ mapping (x_1, x_2) to (x_2, x_1).*

  The fact that X is completely determined by $X(\underline{1})$ can be expressed more abstractly: $\mathbb{F}_+^{\mathrm{op}}$ is the free category with binary products on one generator.

- *a structure map $\xi : X \to \mathsf{P}X$ ('cylindrification') such that $\xi(\underline{n})(x_1, \ldots x_n) = \{(x_1, \ldots x_n, x) \mid x \in X_1\}$. Note how $\xi(\underline{n})$ is determined by the projection $X(w_n)$.*

The dual of (36) is formulated in

Definition 5.11. *The category of FOL-coalgebras is the category of those P-coalgebras $\xi : X \to \mathsf{P}X$ satisfying $y \in \xi(x) \;\Leftrightarrow\; X(w_n)(y) = x$ for all $x \in X(\underline{n}), y \in X(\underline{n+1})$.*

The algebraic and coalgebraic semantics are related by the duality (dual adjunction) between BA and Set (see [27] for an introduction on this point of view). In more detail, we have

$$\begin{tikzcd} \mathsf{BA} \arrow[r, bend right, "\mathrm{Uf}"'] & \mathsf{Set} \arrow[l, bend right, "P"'] \end{tikzcd} \tag{40}$$

mapping a Boolean algebra A to the set $\mathrm{Uf}(A) = \mathsf{BA}(A, \mathbb{2})$ and mapping a set X to its powerset $PX = \mathsf{Set}(X, 2)$. This adjunction becomes a dual equivalence if restricted to finite BAs and finite sets. The adjunction lifts pointwise to presheaves

$$\begin{tikzcd} \mathsf{BA}^{\mathbb{F}_+} \arrow[r, bend right, "\mathrm{Uf}"'] & \mathsf{Set}^{\mathbb{F}_+^{\mathrm{op}}} \arrow[l, bend right, "P"'] \end{tikzcd} \tag{41}$$

where we continue to write P, Uf for the lifted functors.

Example 5.12. *1. Consider a standard model $\xi : X \to \mathsf{P}X$. The dual algebra $A = P(X, \xi)$ has sorts $A(\underline{n}) = 2^{X_1^n}$. Denoting by $\phi(x_0, \ldots, x_{n-1}, x_n)$ an element in $A(\underline{n})$, we can write $w_n\phi(x_0, \ldots x_{n-1}, x_n) = \phi(x_0, \ldots x_{n-1}, x_n, x_{n+1})$ (adding a dummy variable) and $c_{n-1}\phi(x_0, \ldots x_n) = \phi(x_0, \ldots, x_{n-1}, x_{n-1})$ (fusing two variables).*

2. *Algebras arising from duals of standard models are two-valued functional algebras in the sense of Halmos [19, p.102].*

We call a presheaf $A : \mathsf{BA}^{\mathbb{F}_+} \to \mathsf{Set}$, or $X : \mathsf{Set}^{\mathbb{F}_+^{\mathrm{op}}}$, **sort-finite** if $A(\underline{n})$, or $X(\underline{n})$, is finite for all $n \in \mathbb{N}$.

Theorem 5.13. *For sort-finite $X \in \mathsf{Set}^{\mathbb{F}_+^{\mathrm{op}}}$, we have $\mathsf{Q}PX \cong P\mathsf{P}X$. This induces a dual equivalence between sort-finite Q-algebras and sort-finite P-coalgebras. Moreover, this dual equivalence restricts to a dual equivalence between the sort-finite FOL-algebras and the sort-finite FOL-coalgebras.*

Proof. First, the isomorphism $\delta : \mathsf{Q}PX \to P\mathsf{P}X$ is given, at sort $\underline{n}$, by $\delta_n(\exists_n a) = \{b \subseteq X(\underline{n+1}) \mid b \cap a \neq \emptyset\}$. That this is an isomorphism for finite $X(\underline{n+1})$ is well-known in modal logic, see eg Halmos [18, Theorem 8] (and recalling Remark 5.2). The second sentence's claim then follows from the dual equivalence of finite Boolean algebras and finite sets. The final claim follows from the observation that, whenever there is an adjunction between posets (wrt inclusion)

$$\begin{tikzcd} 2^Y \arrow[r, bend right, "\Diamond"'] & 2^Z \arrow[l, bend right, "f"'] \end{tikzcd}$$

where Y, Z are finite, f preserves Boolean operations and $\Diamond$ is left-adjoint to f, then $f = w^{-1}$ for some $w : Y \to Z$ and $z \in \Diamond b \iff \exists y. w(y) = z \wedge y \in b$. $\square$

6 Modularity of Functorial Coalgebraic Logic

The functorial approach to coalgebraic logic considers logics as functors: Whereas coalgebras are given wrt a functor T on a category of 'state spaces' (the category Set of sets in this section), logics for T-coalgebras are given by functors L on a category of algebras representing a propositional logic (the category BA of Boolean algebras in this section). Syntactically, L specifies an extension of Boolean propositional logic by modal operators and axioms. Semantically, L gives a logical description of the 'transition type' T of the coalgebras.

Two slogans of Coalgebraic Logic are: Coalgebraic Logic is uniform and Coalgebraic Logic is modular. This section discusses these notions in connection with completeness. The key to a modular composition of logics is the notion of many-sorted (or symmetric) composition of functors (Section 6.1). We then give a uniform proof of completeness for arbitrary $T : \mathsf{Set} \to \mathsf{Set}$ (Section 6.2). For the case where T does not restrict to finite sets, we devise a method of filtration and illustrate it with the finite distribution functor (Section 6.3). Finally (Section 6.4), we prove completeness of the modular logics from Section 6.1 by using the uniform completeness of Section 6.2.

The material has been organised in such a way that Section 6.1 on many-sorted composition is purely logical/algebraic and does not refer to coalgebras. Sections 6.2 - 6.4 concern the application to coalgebras.

6.1 Many-sorted composition of functors

In this section we are interested in composing logics. In particular, given logics L_1, L_2, we want to form the logics

$$+(L_1, L_2) \qquad \times (L_1, L_2) \qquad L_2 \circ L_1. \tag{42}$$

Above, talking about logics, we conflate logics, functors and presentations. This is often convenient, but let us recall that, more precisely, any sifted-colimits preserving functor on a variety has a presentation and hence gives rise to what we call a logic of rank 1, that is an equational logic where the axioms are of the special format of Definition 3.3; conversely, every logic of rank 1 gives rise to a sifted-colimits preserving functor (see Theorem 3.5). The constructions (42) are easily described for functors, but need to be extended to presentations. We first show how to obtain a presentation for the functor $L = L_2 \circ L_1$ from presentations $\langle \Sigma_2, E_2 \rangle$ and $\langle \Sigma_1, E_1 \rangle$.

We know that such a presentation exists: Given $\langle \Sigma_2, E_2 \rangle$ and $\langle \Sigma_1, E_1 \rangle$, take the canonical presentation of $L_2 \circ L_1$, which exists due to Theorem 3.5. But this in itself does not give us a recipe to compute a good presentation $\langle \Sigma, E \rangle$ from the presentations $\langle \Sigma_i, E_i \rangle$ *in a simple modular way*:

Remark 6.1. *For example, in the case that $L_i : \mathsf{BA} \to \mathsf{BA}$, even if the Σ_i contain only one unary operation symbol $\Box_i$, one may need an infinite set of operation symbols of arbitrary (finite) arities to present $L = L_2 \circ L_1 : \mathsf{BA} \to \mathsf{BA}$. The reason is that operation symbols for L are of the form $\Box_2 \phi$ where ϕ can be* any Boolean combination *of terms of the kind $\Box_1 \psi$, or, more formally, in the notation of Section 3, operation symbols for L are terms in G_2UFG_1UFV.*

The solution is to replace L by a two-sorted functor $\bar{L} : \mathsf{BA}^S \to \mathsf{BA}^S$ where we write $S = \{\mathsf{s}, \mathsf{i}\}$, the idea being that L_1-formulas are now of sort[1] (or type) i and L_2-formulas of sort s. In fact, we consider the more general case $L_1 : \mathcal{A}_s \to \mathcal{A}_i$ and $L_2 : \mathcal{A}_i \to \mathcal{A}_s$, which allows us to obtain $+(L_1, L_2)$ and $\times(L_1, L_2)$ as particular examples.

Definition 6.2 (two-sorted, or symmetric, composition of functors)**.** *Given two functors $L_1 : \mathcal{A}_s \to \mathcal{A}_i$ and $L_2 : \mathcal{A}_i \to \mathcal{A}_s$ between any two categories, the two-sorted composition of L_1 with L_2 is the functor $\bar{L} : \mathcal{A}_i \times \mathcal{A}_s \to \mathcal{A}_i \times \mathcal{A}_s$ mapping $A = (A_\mathsf{i}, A_\mathsf{s})$ to $(\bar{L}A)_\mathsf{s} = L_2(A_\mathsf{i})$ and $(\bar{L}A)_\mathsf{i} = L_1 A_\mathsf{s}$.*

Example 6.3. *If $L_1 : \mathsf{BA}^{S_2} \to \mathsf{BA}^{S_1}$ and $L_2 : \mathsf{BA}^{S_1} \to \mathsf{BA}^{S_2}$, then the two-sorted composition is a functor $\mathsf{BA}^{S_1+S_2} \to \mathsf{BA}^{S_1+S_2}$.*

This composition is symmetric: Swapping L_1 and L_2 just means that the indices i and s change role. It is therefore tempting to suppress the distinction between 1 and i and between 2 and s in our notation. We do not do this because we want to use the notation $(-)_1$ to refer to the functor L_1 and the notation $(-)_\mathsf{i}$ to refer to a projection onto sort i.

[1] i for intermediate; also $\mathsf{i}, 1$ and $\mathsf{s}, 2$ go together.

The next proposition ensures that we can extract the initial $L_2 \circ L_1$-algebra from the initial algebra of the symmetric composition. (We continue to write $L_2 L_1$ instead of $L_2 \circ L_1$).

Proposition 6.4. *Consider categories $\mathcal{A}_{\mathsf{i}}, \mathcal{A}_{\mathsf{s}}$ which are lfp and two finitary functors $L_1 : \mathcal{A}_s \to \mathcal{A}_i$ and $L_2 : \mathcal{A}_i \to \mathcal{A}_s$. Let $\bar{L}$ be the symmetric composition of L_1 with L_2. Then the* s*-component of the initial $\bar{L}$-algebra is the initial L_2L_1-algebra.*

Proof. $\mathcal{A}_i \times \mathcal{A}_{\mathsf{s}}$ is lfp and $\bar{L}$ is finitary. Therefore, the initial $\bar{L}$-algebra is the colimit of the initial algebra chain $\bar{L}^n 0$ where 0 denotes the initial object and n runs through finite ordinals. As colimits are calculated sort-wise, it is enough to show that the projected sequence $(\bar{L}^n 0)_{\mathsf{s}}$ has the same colimit as the initial sequence of L_2L_1, which is easy to see as the latter sequence is a subsequence of the former. ◻

The proposition tells us that we can present $\bar{L}$ instead of L_2L_1. It is obvious how to do this:

Theorem 6.5. *Consider (many-sorted) varieties $\mathcal{A}_{\mathsf{i}} \times \mathcal{A}_{\mathsf{s}}$ and two functors $L_1 : \mathcal{A}_s \to \mathcal{A}_i$ and $L_2 : \mathcal{A}_i \to \mathcal{A}_s$ with presentations $\langle \Sigma_1, E_1 \rangle$ and $\langle \Sigma_2, E_2 \rangle$, respectively. Consider $\langle \bar{\Sigma}, \bar{E} \rangle$ given as follows.*

$$(\bar{\Sigma} X)_{\mathsf{s}} = \Sigma_2 X_{\mathsf{i}}$$
$$(\bar{\Sigma} X)_{\mathsf{i}} = \Sigma_1 X_{\mathsf{s}}$$

where we use that the signatures Σ_1, Σ_2 are given by functors $\mathsf{Set}^{S_2} \to \mathsf{Set}^{S_1}, \mathsf{Set}^{S_1} \to \mathsf{Set}^{S_2}$ and $X = (X_{\mathsf{i}}, X_{\mathsf{s}})$ denotes and element of $\mathsf{Set}^{S_1} \times \mathsf{Set}^{S_2}$. Equations are given by $\bar{E}_{\mathsf{s}} = E_2$, $\bar{E}_{\mathsf{i}} = E_1$. Then $\langle \bar{\Sigma}, \bar{E} \rangle$ is a presentation of the symmetric composition $\bar{L}$ of L_1 with L_2.

Let us illustrate this theorem using more familiar notation.

Example 6.6. *Assume that $\mathcal{A}_i$ and $\mathcal{A}_s$ are both* BA. *We write $\vdash_{\mathsf{i}} \psi$ and $\vdash_{\mathsf{s}} \phi$ to assert that ψ, ϕ are formulas of sort* i, s, *respectively. The theorem then states that formulas of both sorts are closed under Boolean operations and, for all n-ary operation symbols σ_i in Σ_i, formulas are closed under*

$$\frac{\vdash_{\mathsf{i}} \psi_1, \ldots, \vdash_{\mathsf{i}} \psi_n}{\vdash_{\mathsf{s}} \sigma_2(\psi_1, \ldots, \psi_n)} \qquad \frac{\vdash_{\mathsf{s}} \phi_1, \ldots, \vdash_{\mathsf{s}} \phi_n}{\vdash_{\mathsf{i}} \sigma_1(\phi_1, \ldots, \phi_n)}$$

The axioms are given by equations E_1 and E_2 and the laws of Boolean algebra. The rules of the calculus are those of equational logic. The only rules that make the two sorts interact are the congruence rules:

$$\frac{\vdash_{\mathsf{i}} \psi_1 = \psi_1', \ldots, \vdash_{\mathsf{i}} \psi_n = \psi_2'}{\vdash_{\mathsf{s}} \sigma_2(\psi_1, \ldots, \psi_n) = \sigma_2(\psi_1', \ldots, \psi_n')} \qquad \frac{\vdash_{\mathsf{s}} \phi_1 = \phi_1', \ldots, \vdash_{\mathsf{s}} \phi_n = \phi_n'}{\vdash_{\mathsf{i}} \sigma_1(\phi_1, \ldots, \phi_n) = \sigma_1(\phi_1', \ldots, \phi_n')}$$

Here, we use $\vdash_{\mathsf{i}} \psi = \psi'$ and $\vdash_{\mathsf{s}} \phi = \phi'$ to denote derivability of equations of the respective sorts.

The above approach of sorting a logic was found useful eg in the work [10] on the π-calculus which has process-formulas and capability-formulas. The example above shows how it arises in a systematic way from a symmetric composition of functors. The next two examples give the constructions of $+(L_1, L_2)$ and $\times(L_1, L_2)$.

Example 6.7. *We define $+(L_1, L_2)$ as the logic given by the composition* $\mathsf{BA} \xrightarrow{\langle L_1,L_2\rangle} \mathsf{BA} \times \mathsf{BA} \xrightarrow{L_+} \mathsf{BA}$*, where $L_+(A_1, A_2)$ is presented by unary operation symbols $\langle 1\rangle$ and $\langle 2\rangle$ (where $\langle i\rangle$ takes arguments from A_i). Equations specify that the $\langle i\rangle$ preserve finite joins and binary meets and that (i) $\langle 1\rangle a_1 \wedge \langle 2\rangle a_2 = \bot$, (ii) $\langle 1\rangle\top \vee \langle 2\rangle\top = \top$, (iii) $\neg\langle 1\rangle a_1 = \langle 2\rangle\top \vee \langle 1\rangle\neg a_1$, $\neg\langle 2\rangle a_2 = \langle 1\rangle\top \vee \langle 2\rangle\neg a_2$.*

The modal operators $\langle 1\rangle$, $\langle 2\rangle$ describe a situation of choice between two alternatives called 1 and 2. For example, $\langle 1\rangle a$ can be read as 'alternative 1 is chosen and then a holds'. The axioms express that (i) the alternatives exclude each other, (ii) one of the alternatives has to be chosen, (iii) $\neg\langle 1\rangle a_1$ means that either alternative 2 is chosen or 1 is chosen but then not a_1.

A good way of thinking about $+(L_1, L_2)$ is as a logic for systems that have to output exactly one of 1 or 2 and then continue. Dually, one can think of $\times(L_1, L_2)$ as a logic for systems that read input from a two-element set $\{1, 2\}$:

Example 6.8. *We define $\times(L_1, L_2)$ as the logic given by the composition* $\mathsf{BA} \xrightarrow{\langle L_1,L_2\rangle} \mathsf{BA} \times \mathsf{BA} \xrightarrow{L_\times} \mathsf{BA}$*, where $L_\times(A_1, A_2)$ is presented by unary operation symbols $[1]$ and $[2]$ (where $[i]$ takes arguments from A_i). Equations specify that the $[i]$ preserve all Boolean operations.*

For example, if both L_i are the modal logic $\mathbf{K}$ from Example 3.1 with modal operators $\Box_i$ and $\Diamond_i$, respectively, then, because of the equations, every formula can be written in a way such that the only modalities are $[1]\Box_1, [1]\Diamond_1$ and $[2]\Box_2, [2]\Diamond_2$. This means that we can elide $[1], [2]$ from formulas, and we obtain what is known in modal logic as the fusion of L_1 and L_2, see eg Kurucz [26]. The reason for our notation $\times(L_1, L_2)$ will be apparent from Definition 6.18.

6.2 Uniform completeness

In this section we show how to associate to an arbitrary set-functor T a functor L on BA and a semantics $\delta : LP \to PT$ so that the logic given by any presentation of L is complete. Construction and proof are syntax free because, by Theorem 3.5 we can conflate the distinction between functors and logics of rank 1. On the other hand, Theorem 3.5 does not tell us how to find good presentations of functors. How to build them in a modular way will be discussed in Section 6.4.

The definition of L from T is the same as in [30, 29], but as we do not insist on strong completeness[2] here, *we don't need to put any assumptions on T*. Instead we use an induction along the final sequence as first done in Pattinson [33] and adapted to the setting of functorial logics over BA in [25].

[2]A logic is strongly complete if, whenever ϕ holds in all models satisfying a possibly infinite set of formulas Γ, then one can also derive ϕ from Γ. Strong completeness is closely related to compactness. So, for example, the procedure below will not give rise to strongly complete logics if T is the probability distribution functor or if $TX = A \times X$ for an infinite set A.

Definition of L. First, let us recall from [30, 29] the definition of L from T (see also Klin [24]). The essential ingredients are as follows. Two contravariant functors P and S that are adjoint on the right

$$
\begin{tikzcd}
\mathcal{A} \arrow[loop left, "L"] \arrow[r, bend right, "S"'] & \mathcal{X} \arrow[l, bend right, "P"'] \arrow[loop right, "T"] \\
\mathcal{A}_0 \arrow[u, "I"] &
\end{tikzcd}
\tag{43}
$$

where $\mathcal{A}$ is lfp with a small subcategory $\mathcal{A}_0$ of finitely presentable objects. We then define L as

$$
LIA = PTSIA \tag{44}
$$

and extend L continuously from $\mathcal{A}_0$ to $\mathcal{A}$. Note that L thus defined preserves filtered colimits, whereas PTS need not to do so.

Example 6.9. *Take $\mathcal{A} = \mathsf{BA}$ and $\mathcal{X} = \mathsf{Set}$. P is contravariant powerset and S takes ultrafilters. On arrows, P and S map a function to its inverse image. The adjunction restricts to a dual equivalence between finite Boolean algebras and finite sets. The ultrafilters of a finite Boolean algebra A are the atoms of A, that is, those elements $a \in A$ such that there are no elements strictly between $\perp$ and a. Thus, on finite Boolean algebras, the duality reduces to the well known fact that every finite Boolean algebra is isomorphic to the powerset of its atoms. We will also make use of the fact that the finitely presentable Boolean algebras coincide with the finite ones.*

Definition of $\delta : LP \to PT$. The idea that the semantics of a logic for coalgebras should be described by a natural transformation $LP \to PT$ goes back to [25, 9]. The following definition is again from [30].

$$
\begin{tikzcd}
PX & LPX \arrow[r, "\delta_X"] & PTX \\
A_i \arrow[u, "c_i"] & LA_i \arrow[u, "Lc_i"] \arrow[r, "\cong"] & PTSA_i \arrow[u, "PTc_i^\sharp"']
\end{tikzcd}
\tag{45}
$$

PX is a filtered colimit $c_i : A_i \to PX$. Under the adjunction, this cocone corresponds to a cone $c_i^\sharp : X \to SA_i$ which is turned into a cocone under PT (recall that P is contravariant). Now δ_X exists uniquely, since L preserves filtered colimits .

Example 6.10. *In the situation of Example 6.9, consider an arbitrary functor $T : \mathsf{Set} \to \mathsf{Set}$ and let L be the functor defined above. Let $\langle \Sigma, E \rangle$ be the presentation of L given in the proof of Theorem 3.5. According to (11), the set of operations of Σ of arity k is $ULFk \cong UPTSFk \cong UPTUPk = 2^{T(2^k)} \cong \mathrm{Nat}((2^k)^X, 2^{TX})$, the latter denoting the set of natural transformations $(2^k)^X \to 2^{TX})$. But the natural transformations $(2^k)^X \to 2^{TX}$ are precisely the (k-ary) predicate liftings of Pattinson [33]. It follows that the logic given by Σ is the logic of all* *(finitary) predicate liftings investigated by Schröder [36]. In addition L also incorporates a complete axiomatization of the logic of all predicate liftings. Conversely, any logic for T-coalgebras given by predicate liftings and axioms of rank 1 defines a functor $L : \mathsf{BA} \to \mathsf{BA}$ and a natural transformation $\delta : LP \to PT$.*

Intuitively, the logic L is complete if any two formulas identified in the semantics, are already identified in the syntax, or, more technically, if δ is injective. We turn this into

Definition 6.11 (one-step completeness [25])**.** *(L, δ) is one-step complete if δ is injective.*

Assumption: From now on we take $\mathcal{A} = \mathsf{BA}$ and $\mathcal{X} = \mathsf{Set}$ with the functors P and S sort-wise as in Example 6.9.

Lemma 6.12. *δ_X as defined in (45) is injective.*

Proof. Consider two distinct $\phi_1, \phi_2 \in LPX$. By filteredness, we find some A_i and $\phi'_j \in A_i$ such that $c_i(\phi'_j) = \phi_j$. Moreover, since in BA the finitely presentable objects are closed under quotients, we can assume c_i to be injective. The following fact is easily proved.

Claim 6.13. *Let A be finite. $c : A \to PX$ is injective iff the adjoint transpose $c^\sharp : X \to SA$ is surjective.*

Indeed, by the laws of adjunction and A being finite, we have that c is $A \cong PSA \xrightarrow{Pc^\sharp} PX$; now $Pc^\sharp = (c^\sharp)^{-1}$ is injective iff $c^\sharp$ is surjective, which proves the claim. Using that T, as any functor on Set, preserves surjective maps and that P maps surjective maps to injective BA-homomorphisms, we conclude that $PTc_i^\sharp$ is injective, hence $\delta_X(\phi_1) \neq \delta_X(\phi_2)$. □

Lemma 6.14. *Finitary functors $L : \mathsf{BA} \to \mathsf{BA}$ preserve injective maps.*

Proof. Consider an injective BA-morphism $f : A \to B$. A is a filtered colimit $c_i : A_i \to A$ where A_i are finite and c_i are injective. Let $a_1, a_2 \in LA$ and $Lf(a_1) = Lf(a_2)$. Since L preserves filtered colimits, we find i and $a'_1, a'_2 \in LA_i$ with $Lc_i(a'_j) = a_j$. Hence $L(f \circ c_i)(a'_1) = L(f \circ c_i)(a'_1)$. Since $f \circ c_i$ is injective and A_i is finite, the claim now follows from the fact that for any BA-morphism $g : C \to D$ with C finite and g injective there is $h : D \to C$ with $h \circ g = \mathrm{id}_G$ (and hence Lg is injective). □

Theorem 6.15. *Let L be a finitary functor and $\delta : LP \to PT$ be injective. Then the logic given by any presentation of L is complete for T-coalgebras. In particular, the logic given by (44), (45) is complete for T-coalgebras.*

Proof. (The proof is essentially the one from [25], where the reader can find the missing technical details.) L preserves filtered colimits and therefore, using a special property of BA and following [30, Proposition 3.4], L preserves sifted colimits. It follows that L has a presentation, which induces an equational logic, which in turn can be written in the usual modal-logic style, using the correspondences between equations $\phi = \psi$ and formulas $\phi \leftrightarrow \psi$ and between formulas ϕ and equations $\phi = \top$.

The semantics of an L-formula wrt a coalgebra $\xi : X \to TX$ is determined by the arrow $[\![-]\!]_{(X,\xi)}$ from the initial L-algebra to the algebra $LPX \to PTX \to PX$. Because of the naturality of δ, the semantics wrt to all coalgebras is determined by the semantics wrt to the final coalgebra. Since we don't assume that the final coalgebra exists, we replace it by the corresponding final sequence $T^n\mathbb{1}$ which is defined as follows. We denote by $\mathbb{1} = T^0\mathbb{1}$ the final object in Set. $p_0 : T\mathbb{1} \to \mathbb{1}$ is given by finality and $p_{n+1} : T(T^n\mathbb{1}) \to T^n\mathbb{1}$ is defined to be Tp_n. We think of the $T^n\mathbb{1}$ as approximating the final coalgebra.[3] In the same way as any coalgebra $\xi : X \to TX$ has a unique arrow into the final coalgebra, there are canonical arrows $\xi_n : X \to T^n\mathbb{1}$ to the approximants of the final coalgebra, defined inductively by $\xi_{n+1} = T(\xi_n) \circ \xi$. The idea now is to interpret a formula ϕ 'of depth n' as a subset $[\![\phi]\!]_n$ of $T^n\mathbb{1}$. The semantics of ϕ in X is then $\xi_n^{-1}([\![\phi]\!]_n)$.[4] To say what it means for a formula to be of depth n we need the initial sequence of L, which we define next.

Since L is finitary the initial algebra is the colimit of the sequence $L^n\mathbb{2}$ defined as follows. We denote by $\mathbb{2} = L^0\mathbb{2}$ the initial object in BA. $e_0 : \mathbb{2} \to L\mathbb{2}$ is given by initiality and $e_{n+1} : L^n\mathbb{2} \to L(L^n\mathbb{2})$ is defined to be Le_n. Since L preserves sifted colimits and hence injective maps [30, Corollary 4.10], all maps in the sequence are injective. This means that we can consider the initial L-algebra as a union of its approximants $L^n\mathbb{2}$. We call the elements of $L^n\mathbb{2}$ formulas of depth n. The semantics of a formula of depth n is given by a BA-morphism $[\![-]\!]_n : L^n\mathbb{2} \to PT^n\mathbb{1}$ as follows.

$$\begin{tikzcd}
P\mathbb{1} \arrow[r, "Pp_0"] & \cdots & PT^n\mathbb{1} \arrow[r, "Pp_n"] & PT^{n+1}\mathbb{1} & \cdots \\
\mathbb{2} \arrow[u, "{[\![-]\!]_0}"] \arrow[r, "e_0"'] & \cdots & L^n\mathbb{2} \arrow[u, "{[\![-]\!]_n}"] \arrow[r, "e_n"'] & L^{n+1}\mathbb{2} \arrow[u, "{[\![-]\!]_{n+1}}"] & \cdots
\end{tikzcd} \tag{46}$$

$[\![-]\!]_0$ is given by initiality (and is actually the identity). $[\![-]\!]_{n+1}$ is defined to be $\delta_{T^n\mathbb{1}} \circ L([\![-]\!]_n)$. Observe that the semantics of a formula is independent of the particular approximant we choose (all squares in the diagram commute). Moreover, given a coalgebra $\xi : X \to TX$ and a formula of depth n, the semantics via the initial L-algebra and the semantics via the final sequence coincide: $[\![\phi]\!]_{(X,\xi)} = \xi_n^{-1}([\![\phi]\!]_n)$. Since δ is injective and L preserves injective maps, all $[\![-]\!]_n$, $n \in \mathbb{N}$, are injective.

To show completeness, suppose $\phi_1 \neq \phi_2$ in the initial L-algebra. We find an approximant $L^n\mathbb{2}$, in which ϕ_1 and ϕ_2 are different. Any one-sided inverse i of p_0 gives rise to a T-coalgebra $\xi = T^n(i)$ with carrier $T^n\mathbb{1}$. We have $[\![\phi]\!]_{(T^n\mathbb{1},\xi)} = [\![\phi]\!]_n$. Now injectivity of $[\![-]\!]_n$ shows that $(T^n\mathbb{1}, \xi)$ provides a counter-example for $\phi_1 = \phi_2$. □

6.3 Filtration and the finite distribution functor

If T preserves finite sets the logic L discussed in the previous section has good claims of being *the* finitary logic for T. Otherwise—following Example 6.10—the language of all predicate

[3]Indeed, if we let run the final sequence through all ordinals, we obtain the final coalgebra as a limit if it exists, see Adámek and Koubek [2].

[4]This point of view has been elaborated in [28].

liftings has $2^{T(2^k)}$-many modal operators of arity k, which is uncountable if $T(2^k)$ is infinite.

In this section we first describe a method to find a functor L so that the corresponding logic has only countably many modal operators of arity k. Second, we illustrate this construction with the important example of the finite distribution functor (for which the logic of all predicate liftings is strictly more expressive than the logic discussed here).

6.3.1 A filtration method

As in the previous section, given T, we want to find L and a (componentwise) injective δ as in

$$\begin{tikzcd} LPX \arrow[r, "\delta_X"] & PTX \\ L_kPX \arrow[u] \arrow[ur, "\delta_{k,X}"'] & \end{tikzcd} \tag{47}$$

To this end we propose to find L_k such that

- L is a filtered colimit of the L_k
- the L_k preserve finite sets (so that the initial L_k-algebras are countable)
- the δ_k are injective

Lemma 6.16. *In the situation of Diagram (47) assume that L is a filtered colimit of the L_k. Then δ is injective if the δ_k are injective.*

Proof. Suppose $\delta_X(x) = \delta_X(y)$. Then there is k such that $\delta_{k,X}(x) = \delta_{k,X}(y)$, hence $x = y$. □

Using the above lemma to prove completeness of L resembles the filtration method in modal logic: instead of using the whole language L, one restricts to a sublanguage L_k. This aspect is emphasized by the following lemma:

Lemma 6.17. *In the situation of Diagram (47) assume that the L_k preserve finite sets. Then δ_k is injective iff, for all finite X, the transpose $\delta_k^\sharp : T \to SL_kP$ is surjective .*

Proof. As in Lemma 6.12, it is enough to show that $\delta_{k,X} : L_kPX \to PTX$ is injective for finite X. But L_kPX is finite for finite X (since L_k preserves finite sets) and we can apply Claim 6.13. □

The above lemmas could be called the *one-step filtration* method: For a finite X, SL_kPX is the finite set of maximal consistent theories over the formulas in L_kPX. To show that δ_k is surjective means to find a model in TX for each maximal consistent theory in SL_kPX.

In the usual filtration method one would not work with one-step theories but instead consider maximal consistent theories over all formulas of depth bounded by some $n < \omega$. These theories are elements in SL^n2. The task then is to find a model in T^n1 for each theory. But this is exactly how the proof of the completeness theorem 6.15 goes.

6.3.2 The finite distribution functor

We will now apply the method of this section to the finite distribution functor $\mathcal{D} : \mathsf{Set} \to \mathsf{Set}$ defined by

$$\mathcal{D}(X) = \{\mu : X \to [0,1] \mid \mu(x) \neq 0 \text{ for finitely many } x \in X \text{ and } \sum_{x \in X} \mu(x) = 1\}$$

Notation. From now we will follow standard practice and denote by L_q modal operators of the probability logic. Where L before referred to a functor we will write $\mathcal{L}$ now.

The syntax. We consider the endofunctor $\mathcal{L} : \mathsf{BA} \to \mathsf{BA}$ given by the following finitary presentation. The signature will consist of unary operations L_q for all rational numbers $q \in [0,1]$. The intended meaning of $L_q\phi$ is that formula ϕ has probability at least q. The following abbreviations will be used: $M_q = L_{1-q}\neg$ and $E_q = L_q \wedge M_q$ with thew intended meaning 'probability at most q' and 'probability exactly q', respectively.

We consider the following set of equations:

$$\begin{array}{lll}
L_0 x = \top & & (E1) \\
L_\alpha \top = \top & & (E2) \\
L_\alpha(x \wedge y) \wedge L_\beta(x \wedge \neg y) \to L_{\alpha+\beta} x = \top & \alpha + \beta \leq 1 & (E3) \\
\neg L_\alpha(x \wedge y) \wedge \neg L_\beta(x \wedge \neg y) \to \neg L_{\alpha+\beta} x = \top & \alpha + \beta \leq 1 & (E4) \\
L_\alpha x \to \neg L_\beta \neg x = \top & \alpha + \beta > 1 & (E5) \\
L_\alpha x \leftrightarrow L_\alpha y = \top & x \leftrightarrow y & (E6)
\end{array}$$

So far, this is an adaptation of a sound axiomatization system for probability logic for type spaces considered by Aumann. This system was completed by Heifetz and Mongin in [20] considering another axiom, essentially expressed by the following equation:

$$(\bigwedge_{i=1}^{m} L_{\alpha_i} x_i) \wedge (\bigwedge_{j=2}^{n} M_{\beta_j} y_j) \to L_{(\alpha_1 + \cdots + \alpha_m) - (\beta_2 + \cdots + \beta_n)} y_1 = \top \qquad (E7)$$

whenever $m, n \geq 1$, $(\alpha_1 + \cdots + \alpha_m) - (\beta_2 + \cdots + \beta_n) \in [0,1]$ and $\bigwedge_{k=1}^{\max m,n} x^{(k)} \leftrightarrow y^{(k)}$. Here by $x^{(k)}$ denotes $\bigvee_{1 \leq l_1 < \ldots l_k \leq m} (x_{l_1} \wedge \cdots \wedge x_{l_k})$.

The main idea in the proof of completeness is to use the method of filtration. Completeness is proved for each formula, restricting the language to a finite one. Suppose X is a Boolean algebra and ϕ is an element of $\mathcal{L}X$. In the formula ϕ appear only a finite number of operators L_q. Let k be the least common multiple of the denominators of these rational numbers q. Then ϕ can be regarded as an element of $\mathcal{L}_k X$ where $\mathcal{L}_k : \mathsf{BA} \to \mathsf{BA}$ is the endofunctor defined by the following finitary presentation. As generators we consider only the unary operations L_r such that the rational number r has k as its denominator. The equations will be the same as for $\mathcal{L}$. The advantage of considering the functor $\mathcal{L}_k$ consists in the fact that it sends finite sets to finite sets. We can easily see that $\mathcal{L}$ is a filtered colimit of the functors $\mathcal{L}_k$ taken after all positive integers k.

The semantics. As in the previous section, the semantics of this coalgebraic logic will be described by a natural transformation $\delta : \mathcal{L}P \to P\mathcal{D}$. For each positive integer k consider the map $\delta_{k,X} : \mathcal{L}_k PX \to P\mathcal{D}X$ given by:

$$L_r Y \mapsto \{\mu \in \mathcal{D}X \mid \sum_{y \in Y} \mu(y) \geq r\}$$

for all $Y \subseteq X$. It is not difficult to check that equations $(E1)$-$(E7)$ are satisfied and that $\delta_{k,X}$ is a well-defined Boolean algebra morphism.

Completeness. We have to show the surjectivity of the maps $\delta^\sharp_{k,X} : \mathcal{D}X \to S\mathcal{L}_k PX$ defined by

$$\delta^\sharp_{k,X}(\mu) = \theta_\mu$$

where $\theta_\mu : \mathcal{L}_k PX \to 2$ is given by $\theta_\mu(L_r Y) = \top$ iff $\sum_{y\in Y} \mu(y) \geq r$. Suppose $\theta \in S\mathcal{L}_k PX$. We need to find a probability μ such that $\theta = \theta_\mu$. Let us consider $r_x = \max\{r \mid \theta(L_r\{x\}) = \top\}$ and $l_x = \min\{r \mid \theta(M_r\{x\}) = \top\}$. From the axioms one can derive that $r_x \leq l_x$. We should find a probability μ such that $\mu(x) = r_x$ if $r_x = l_x$ or such that $\mu(x) \in (r_x, l_x)$ if $r_x < l_x$. But this is exactly the content of Lemma A.5 of [20].

6.4 Modular Completeness

We study the modular construction of logics for T-coalgebras, where T is constructed from a number of 'basic' functors $B : \mathsf{Set} \to \mathsf{Set}$ together with binary constructors $+, \times$ and composition $\circ$. Thus we consider **functor expressions**

$$\begin{aligned} B &::= Id \mid K_C \mid \mathcal{P} \mid \mathcal{D} \mid \ldots \\ H &::= B \mid H + H \mid H \times H \mid H \circ H \end{aligned} \tag{48}$$

Functor expressions correspond to Abramsky's notion of meta-language [1]. They can be interpreted on the semantic side over Set and on the logic side over BA:

Definition 6.18 (T_H, L_H, δ_H)**.** *Each functor expression H gives rise to functors $T_H : \mathsf{Set} \to \mathsf{Set}$ and $L_H : \mathsf{BA} \to \mathsf{BA}$. To obtain T_H, one interprets $\mathcal{P}$ as powerset, $\mathcal{D}$ as the finite distribution functor, $+$ as disjoint union, $\times$ as cartesian product. To obtain L_H one interprets $\mathcal{P}$ as in Example 3.1, $\mathcal{D}$ as the $\mathcal{L}$ of Section 6.3.2, $+$ as cartesian product, $\times$ as coproduct. $\circ$ is composition in both interpretations. Assuming a semantics $\delta_B : L_B P \to PT_B$ for the basic functors is given, one obtains inductively $\delta_H : L_H P \to PT_H$.*

It is convenient to continue to write $\mathcal{P}$ instead of $T_\mathcal{P}$, etc, or to write T_2, T_1 instead of T_{H_2}, T_{H_1} if the precise nature of the H_i does not matter for the issue under discussion.

We can think of a coalgebra $X \to T_2 \circ T_1(X)$ as first making a step to an intermediate state in $T_1(X)$ and then to a (proper) state in $T_2(T_1(X))$. This point of view introduces a

$\{\mathsf{s},\mathsf{i}\}$-sorted semantics: (proper) states, of sort s, and intermediate states, of sort i. We could make this explicit using a two sorted functor $\bar{T} : \mathsf{Set}^{\{\mathsf{s},\mathsf{i}\}} \to \mathsf{Set}^{\{\mathsf{s},\mathsf{i}\}}$, but we do not need to do this here. On the other hand, on the dual side, to construct the logics, introducing new sorts for intermediate states will allow us to compose presentations in a modular way.

Definition 6.19 ($\bar{L}_H$)**.** *Each functor expression H gives rise to a functor $\bar{L}_H : \mathsf{BA}^S \to \mathsf{BA}^S$ for some finite set S as follows. The interpretation of basic expressions B is L_B as in Definition 6.18. The symmetric composition of Definition 6.2 is used to interpret $\circ$, but also $+$ and $\times$ as follows. $H = H' + H''$ is interpreted as*

$$L_+^{H',H''} \circ \langle \bar{L}_{H'}, \bar{L}_{H''} \rangle \tag{49}$$

where

$$\begin{aligned}
\bar{L}_{H'} &: \mathsf{BA}^{\{\mathsf{s}',\mathsf{i}'_1,\ldots\mathsf{i}'_{n'}\}} \to \mathsf{BA}^{\{\mathsf{s}',\mathsf{i}'_1,\ldots\mathsf{i}'_{n'}\}} \\
\bar{L}_{H''} &: \mathsf{BA}^{\{\mathsf{s}'',\mathsf{i}''_1,\ldots\mathsf{i}''_{n''}\}} \to \mathsf{BA}^{\{\mathsf{s}'',\mathsf{i}''_1,\ldots\mathsf{i}''_{n''}\}} \\
L_+^{H',H''} &: \mathsf{BA}^{\{\mathsf{s}',\mathsf{i}'_1,\ldots\mathsf{i}'_{n'},\mathsf{s}'',\mathsf{i}''_1,\ldots\mathsf{i}''_{n''}\}} \to \mathsf{BA}^{\{\mathsf{s},\mathsf{i}'_1,\ldots\mathsf{i}'_{n'},\mathsf{i}''_1,\ldots\mathsf{i}''_{n''}\}}
\end{aligned}$$

and $L_+^{H',H''}$ is defined as

$$\begin{aligned}
(L_+^{H',H''}(A))_{\mathsf{i}} &= A_{\mathsf{i}} & \mathsf{i} \in \{\mathsf{i}'_1, \ldots \mathsf{i}'_{n'}, \mathsf{i}''_1, \ldots \mathsf{i}''_{n''}\} \\
(L_+^{H',H''}(A))_{\mathsf{s}} &= A_{\mathsf{s}'} \times A_{\mathsf{s}''} \cong L_+(A_{\mathsf{s}'}, A_{\mathsf{s}''})
\end{aligned}$$

where L_+ is as in Example 6.7. The interpretation of $\times$ is obtained by permuting $+$ with $\times$.

Remark 6.20. *Applying Definition 6.2 to (49) yields a functor*

$$\begin{aligned}
\bar{L}_H : \mathsf{BA}^{\{\mathsf{s},\mathsf{s}',\mathsf{i}'_1,\ldots\mathsf{i}'_{n'},\mathsf{s}'',\mathsf{i}''_1,\ldots\mathsf{i}''_{n''}\}} &\to \mathsf{BA}^{\{\mathsf{s},\mathsf{s}',\mathsf{i}'_1,\ldots\mathsf{i}'_{n'},\mathsf{s}'',\mathsf{i}''_1,\ldots\mathsf{i}''_{n''}\}} \\
(\bar{L}_H(A))_{\mathsf{s}} &= (\bar{L}_H(A))_{\mathsf{s}'} \times (\bar{L}_H(A))_{\mathsf{s}''} \\
(\bar{L}_H(A))_{\mathsf{i}} &= (\bar{L}_{H'}(A'))_{\mathsf{i}} \qquad \mathsf{i} \in \{\mathsf{s}', \mathsf{i}'_1, \ldots, \mathsf{i}'_{n'}\} \\
(\bar{L}_H(A))_{\mathsf{i}} &= (\bar{L}_{H''}(A''))_{\mathsf{i}} \qquad \mathsf{i} \in \{\mathsf{s}'', \mathsf{i}''_1, \ldots \mathsf{i}''_{n''}\}
\end{aligned}$$

where A', A'' are the restrictions of A to $\{\mathsf{s}', \mathsf{i}'_1, \ldots, \mathsf{i}'_{n'}\}$ and $\{\mathsf{s}'', \mathsf{i}''_1, \ldots \mathsf{i}''_{n''}\}$, respectively.

Definition 6.21 ($\mathcal{L}og_H$)**.** *Assume presentations $\langle \Sigma_B, E_B \rangle$ for basic functors are given and recall the presentations of $L_+, L_\times$ given in Examples 6.7, 6.8. Then $\mathcal{L}og_H$ is the many-sorted equational logic obtained by representing $\bar{L}_H$ as in Theorem 6.5.*

Remark 6.22. *The many-sorted approach to coalgebraic logic goes back to Rößiger [35] and was further developed, in technically different styles, by Jacobs [23] and Cîrstea and Pattinson [12]. The above definitions can be seen as reformulating their approaches by conflating logics (of rank 1) with (sifted-colimits preserving) functors on* BA*. This functorial formulation has the advantage of clearly separating the abstract treatment (in terms of category theoretic properties of L and δ) from concrete syntactic considerations. For example, to use* *the theorem below, all the work will go into verifying one-step completeness of the basic logics given by presentations $\langle \Sigma_B, E_B \rangle$, the rest then coming from the abstract machinery. Other examples of the usefulness of the abstract category theoretic approach to coalgebraic logic have been given in [30, 29, 10].*

Theorem 6.23 (modular completeness)**.** *If the logics $\langle \Sigma_B, E_B \rangle$ are one-step complete, then $\mathcal{L}og_H$ is complete.*

Proof. Consider two $\mathcal{L}og_H$-formulas ϕ, ψ of sort s and suppose the equation $\phi = \psi$ is not derivable. By Theorem 3.4, ϕ and ψ are in different equivalence classes of the initial $\bar{L}_H$-algebra. By Proposition 6.4, ϕ and ψ are also different in the initial L_H-algebra. Now the claim follows from uniform completeness, Theorem 6.15, once we know that δ_H is injective. But this is a consequence of Lemma 6.14. $\square$

7 Conclusion

In this paper we have seen that the notion of a finitary presentation of a functor on a many-sorted variety by operations and equations allows a systematic treatment to syntax involving binding constructors, algebraic and coalgebraic semantics of first-order logic and modular completeness in coalgebraic logic. Our work can be extended in several directions.

Regarding Section 4, it will be interesting to move from $\mathsf{Set}^{\mathbb{F}}$ to $\mathsf{Set}^{\mathbb{I}}$(where $\mathbb{I}$ is the category of finite ordinals and injective maps), making available standard (many-sorted) universal algebra to study certain nominal logics [13, 17]. Applying this to Section 5 should correspond to replacing polyadic algebras by cylindric algebras. Polyadic and cylindric algebras would then be related by the observation that presheaves on $\mathbb{F}^{op}$ are monadic over presheaves on $\mathbb{I}^{op}$.

It will also be of interest to further explore the duality between the algebraic and coalgebraic models of first-order logic, for example: To use bisimulations to prove the equivalence of points (or tuples) in (standard) first-order models; to use the Jónsson-Tarski Theorem for presenting modal algebras to prove completeness of first-order logic. It is straight-forward to extend the models of Section 5 by adding equality, relation symbols, and function symbols; the latter will replace $\mathbb{F}$ by a suitable Lawvere-theory, a move not available in traditional polyadic algebra.

In Section 6.2, the construction of the logic L and its semantics is purely category theoretic, but the completeness theorems use special properties of the category of Boolean algebras. The generalisation to, at least, (presheaves over) distributive lattices is an important task.

References

[1] S. Abramsky. Domain theory in logical form. *Ann. Pure Appl. Logic*, 51, 1991.

[2] J. Adámek and V. Koubek. On the greatest fixed point of a set functor. *Theoret. Comput. Sci.*, 150, 1995.

[3] J. Adámek, F. Lawvere, and J. Rosický. On the duality between varieties and algebraic theories. *Algebra universalis*, 49, 2003.

[4] J. Adámek and J. Rosický. *Locally Presentable and Accessible Categories*. CUP, 1994.

[5] J. Adámek and J. Rosický. On sifted colimits and generalized varieties. *Th. Appl. Categ.*, 8, 2001.

[6] J. Adámek, J. Rosický, and E. M. Vitale. *Algebraic Theories: a Categorical Introduction to General Algebra*. Draft available at `http://www.iti.cs.tu-bs.de/~adamek/algebraic.theories.pdf`.

[7] J. Adámek and V. Trnková. *Automata and Algebras in Categories*. Kluwer, 1990.

[8] A. Y. Aizenštat. Defining relations of finite symmetric semigroups. *Math. Sb. N. S.*, 45, 1958.

[9] M. Bonsangue and A. Kurz. Duality for logics of transition systems. In *FoSSaCS'05*.

[10] M. Bonsangue and A. Kurz. Pi-calculus in logical form. In *LICS'07*.

[11] M. Bonsangue and A. Kurz. Presenting functors by operations and equations. In *FoSSaCS'06*.

[12] C. Cîrstea and D. Pattinson. Modular construction of modal logics. In *CONCUR'04*.

[13] R. Clouston and A. Pitts. Nominal equational logic. In *Computation, Meaning and Logic, Articles dedicated to Gordon Plotkin*, volume 172 of *Electronic Notes in Theoretical Computer Science*. 2007.

[14] M. Fiore and C.-K. Hur. Equational systems and free constructions. In *ICALP'07*.

[15] M. Fiore, G. Plotkin, and D. Turi. Abstract syntax and variable binding. In *LICS'99*.

[16] M. Gabbay and A. Pitts. A new approach to abstract syntax involving binders. In *LICS'99*.

[17] M. J. Gabbay and A. Mathijssen. One-and-a-halfth-order logic. *Journal of Logic and Computation*, 18(4):521–562, 2008.

[18] P. Halmos. Algebraic Logic I. Monadic Boolean algebras. *Compositio Mathematica*, 12:217–249, 1955.

[19] P. Halmos. *Algebraic Logic*. Chelsea Publishing, 1962.

[20] A. Heifetz and P. Mongin. Probabilistic logic for type spaces. *Games and Economic Behavior*, 35, 2001.

[21] L. Henkin, J. D. Monk, and A. Tarski. *Cylindric Algebra.* North Holland Publishing Company., 1971.

[22] M. Hofmann. Semantical analysis of higher-order abstract syntax. In *LICS'99*.

[23] B. Jacobs. Many-sorted coalgebraic modal logic: a model-theoretic study. *Theor. Inform. Appl.*, 35, 2001.

[24] B. Klin. Coalgebraic modal logic beyond sets. In *MFPS'07*, 2007.

[25] C. Kupke, A. Kurz, and D. Pattinson. Algebraic semantics for coalgebraic logics. In *CMCS'04*.

[26] A. Kurucz. *Handbook of Modal Logic*, chapter Combining modal logics. Elsevier, 2007.

[27] A. Kurz. Coalgebras and their logics. *SIGACT News*, 37, 2006.

[28] A. Kurz and D. Pattinson. Coalgebraic modal logic of finite rank. *Math. Structures Comput. Sci.*, 15, 2005.

[29] A. Kurz and J. Rosický. The Goldblatt-Thomason-theorem for coalgebras. In *CALCO'07*.

[30] A. Kurz and J. Rosický. Strongly complete logics for coalgebras. July 2006. Submitted.

[31] R. Ouellet. Inclusive first-order logic. *Studia Logica*, 40, 1981.

[32] R. Ouellet. A categorical approach to polyadic algebras. *Studia Logica*, 41, 1982.

[33] D. Pattinson. Coalgebraic modal logic: Soundness, completeness and decidability of local consequence. *Theoret. Comput. Sci.*, 309, 2003.

[34] D. Robinson. *A Course in the Theory of Groups*. Springer, 1996.

[35] M. Rößiger. Coalgebras and modal logic. In *CMCS'00*.

[36] L. Schröder. Expressivity of Coalgebraic Modal Logic: The Limits and Beyond. In *FoSSaCS'05*.

[37] S. Staton. Relating coalgebraic notions of bisimulation with applications to name-passing process calculi. In *CALCO'09*.