Mathematical Structures in Computer Science (2021), 1–18
doi:10.1017/S0960129521000268

CAMBRIDGE
UNIVERSITY PRESS

PAPER

On generalized algebraic theories and categories with families

Marc Bezem[1], Thierry Coquand[2], Peter Dybjer[3]* and Martín Escardó[4]

[1] Department of Informatics, University of Bergen, Norway, [2] Department of Computer Science and Engineering, University of Göteborg, Sweden, [3] Department of Computer Science and Engineering, Chalmers University of Technology, Sweden and [4] School of Computer Science, University of Birmingham, UK

*Corresponding author. Email: peterd@chalmers.se

(Received 16 August 2020; revised 17 July 2021; accepted 19 August 2021)

Abstract

We give a syntax independent formulation of finitely presented generalized algebraic theories as initial objects in categories of categories with families (cwfs) with extra structure. To this end, we simultaneously define the notion of a *presentation* Σ of a generalized algebraic theory and the associated category $\mathbf{CwF}_\Sigma$ of small cwfs with a Σ-structure and cwf-morphisms that preserve Σ-structure on the nose. Our definition refers to the purely semantic notion of *uniform family* of contexts, types, and terms in $\mathbf{CwF}_\Sigma$. Furthermore, we show how to syntactically construct an initial cwf with a Σ-structure. This result can be viewed as a generalization of Birkhoff's completeness theorem for equational logic. It is obtained by extending Castellan, Clairambault, and Dybjer's construction of an initial cwf. We provide examples of generalized algebraic theories for monoids, categories, categories with families, and categories with families with extra structure for some type formers of Martin-Löf type theory. The models of these are internal monoids, internal categories, and internal categories with families (with extra structure) in a small category with families. Finally, we show how to extend our definition to some generalized algebraic theories that are not finitely presented, such as the theory of contextual cwfs.

Keywords: Dependent type theory; generalized algebraic theory; category with families; initial model; internal category; Martin-Löf type theory

1. Introduction

Martin-Löf type theory can be characterized in a syntax independent way as the initial category with families (cwf) with extra structure for the type formers (Castellan et al., 2015, 2017). The main contribution of this paper is a similar syntax independent characterization of finitely presented generalized algebraic theories as initial cwfs with extra structure.

Generalized algebraic theories were introduced by Cartmell in his PhD thesis (Cartmell, 1978) as a dependently typed generalization of many sorted algebraic theories. Each generalized algebraic theory is presented by (possibly infinite) sets of sort symbols, operator symbols, and equations. Cartmell's definition of generalized algebraic theories (Cartmell, 1978, 1986) is based on a notion of *derived rule* expressed in terms of a traditional syntactic system for dependent type theory. He also defines a notion of model whereby sort symbols are interpreted as families of sets.

Categories with families (cwfs) (Dybjer, 1996) were introduced as a new notion of model of dependent type theory. Cwfs arise by reformulating the notion of category with attributes in

Martin Hofmann's sense (Hofmann, 1994). The key point is that cwfs arise as models of a certain generalized algebraic theory closely related to Martin-Löf's substitution calculus (Martin-Löf, 1992). As such the notion of cwf becomes a useful intermediary between traditional syntactic systems for dependent type theory and a variety of categorical notions of model.

The generalized algebraic theory of cwfs can be seen as a kind of idealized formal system for dependent type theory. In contrast to Martin-Löf's substitution calculus and other syntactic systems for dependent type theory, it is *not* formulated in terms of grammars and inference rules for the forms of judgment of type theory. Instead it is formulated in terms of the sort symbols (corresponding to the judgment forms), operator symbols (corresponding to the formation, introduction, and elimination rules), and equations (corresponding to the equality rules for the type formers) of the generalized algebraic theory. Some of the general reasoning (about equality, substitution, and assumptions) is taken care of by the underlying infrastructure of dependent types. This makes it possible to abstract away from details in the formulation of grammars and inference rules.

In this paper, we explore the interdependence between generalized algebraic theories and cwfs. We already explained that cwfs can be defined as models of a generalized algebraic theory. In the other direction, the notion of generalized algebraic theory relies on the notion of cwf, in the sense that the latter models the underlying infrastructure of dependent types.

Our development can be formulated in a constructive set theory, as described for instance by Aczel (1978, 1998), although the set theory we use for formulating the notion of cwf with a Σ-structure is probably much weaker. In the general theory, we need to distinguish between small and large sets, and hence we assume that our set theory comes with a Grothendieck universe V of *small sets*. Furthermore, in order to give examples of small cwfs with internal monoids, categories and cwfs, we shall assume two more Grothendieck universes V' and V'', where $\mathrm{V}'' \in \mathrm{V}' \in \mathrm{V}$.

Plan of the paper

In Section 2, we recall the definition of the category **CwF** of small cwfs and morphisms preserving cwf-structure on the nose. Section 3 contains our main definition of a syntax independent notion of presentation Σ of a generalized algebraic theory and the category $\mathbf{CwF}_\Sigma$ of small cwfs with a Σ-structure. In Section 4, we construct an initial object $\mathcal{T}_\Sigma$ in $\mathbf{CwF}_\Sigma$. In Section 5 we show several examples of generalized algebraic theories: for monoids, categories, cwfs, and cwfs with extra structure for Π-types, a type of natural numbers, and a universe. We point out that small cwfs with extra structure for generalized algebraic theories of monoids, categories, and cwfs have an internal monoid, internal category, and internal cwf, respectively. We also sketch how to extend our approach to some countably presented generalized algebraic theories and show the example of contextual cwfs, a variant of Cartmell's contextual categories (Cartmell, 1978, 1986). Finally, in Section 6, we discuss related work with connections to Voevodsky's initiality conjecture (Voevodsky, 2017) and to Altenkirch and Kaposi's quotient inductive-inductive types (Altenkirch and Kaposi, 2016).

Remarks on terminology and notation

Like Cartmell, we have chosen to use the term *sort symbol* from many-sorted universal algebra. However, in our semantic notion of presentation sort symbols are interpreted by *type families* in a cwf. A cwf consists of a base category where the objects are (semantic) contexts and the morphisms are (semantic) substitutions. Moreover, we have a family-valued presheaf mapping contexts to families of (semantic) terms indexed by (semantic) types. Thus, the reader should be aware of the mismatch between the word *sort* from universal algebra and the word *type* in the cwf semantics.

Another possible source of confusion is that cwfs appear on two different levels. In Section 2, we recall the definition of cwf in set-theoretic metalanguage, where we use Ty to denote the family of types indexed by contexts and Tm to denote the family of terms indexed by contexts and types. This notion of cwf is then used to define the semantic notions of presentation and category of models of a generalized algebraic theory. Then in Section 5.3, we define the generalized algebraic theory of cwfs. This generalized algebraic theory has sort symbols ty for *internal types* and tm for *internal terms* using lower case to highlight the difference from Ty and Tm in the model cwf.

Furthermore, we often use the same notation both on the semantic and the syntactic level. For example, in Section 3, where we are syntax independent, we use the same letter S as in Section 4, where we syntactically construct the initial model.

As emphasized by Voevodsky (2017), we study structures invariant under *isomorphisms* and not under *equivalences*, and it is actually misleading to call them "category" (and this is why Voevodsky used the term "C-system" for what Cartmell called "contextual category"). As he also noticed, this important distinction between categories and notions invariant under isomorphisms becomes precise in the setting of univalent foundations where not all collections of objects are constructed from sets.

To the memory of Martin Hofmann

We have written this paper to honour the memory of Martin Hofmann. The topic is categorical models of dependent type theory, an area that Martin made seminal contributions to. In particular, he did much to clarify the relationship between intensional and extensional type theory. His thesis was the first investigation of the setoid model (Hofmann, 1997). His and Streicher's groupoid model (Hofmann and Streicher, 1994) refutes uniqueness of identity proofs and identity reflection, the two rules that separate extensional from intensional type theory. The groupoid model also validates the principle of universe extensionality, a special case of Voevodsky's univalence axiom. As a consequence, this work is a forerunner to Voevodsky's univalent foundations.

Other notable contributions to dependent type theory include the interpretation of extensional type theory in locally cartesian closed categories (Curien et al., 2014; Hofmann, 1994), the use of a presheaf model to prove that the Logical Framework version of Martin-Löf type theory is a conservative extension of the original version (Hofmann, 1996), and a method for eliminating extensional identity types (Hofmann, 1995). Martin also wrote a widely read introduction to the syntax and semantics of dependent types (Hofmann, 1996).

Martin was an extremely gifted and generous person, many researchers have benefited from his collaboration. He is truly missed.

2. Categories with Families

Definition 1. ***Fam*** *is a category whose objects are set-indexed families of sets* $(U_x)_{x\in X}$*. A morphism of* ***Fam*** *with source* $(U_x)_{x\in X}$ *and target* $(V_y)_{y\in Y}$ *consists of a re-indexing function* $f : X \to Y$ *together with a family* $(g_x)_{x\in X}$ *of functions* $g_x : U_x \to V_{f(x)}$.

The next step is to define the category **CwF**. We split this definition in two: first the objects, which are called *categories with families*, in Definition 2, and then the morphisms in Definition 3. Since **CwF** has been developed as a categorical framework for the semantics of type theory, much of the terminology (contexts, substitutions, types, terms) refers to the syntax of type theory, suggesting the intended interpretation of this syntax in the so-called **CwF**-semantics.

The main novelty of this paper is to use **CwF** as a framework for defining a notion of finitely presented generalized algebraic theory. Contexts, substitutions, types, and terms also make sense in relation to generalized algebraic theories.

Definition 2. *A category with families (cwf) consists of the following data:*

- *A category* $\mathcal{C}$*;*
- *A* ***Fam****-valued presheaf on* $\mathcal{C}$*, that is, a functor* $T : \mathcal{C}^{op} \to \textbf{\textit{Fam}}$*;*
- *A terminal object* $1 \in \mathcal{C}$*, and unique maps* $\langle\rangle_\Gamma \in \mathcal{C}(\Gamma, 1)$ *for all objects* Γ *of* $\mathcal{C}$*;*
- *Operations* $\,.\,,\ \langle \_,\_ \rangle,\ \mathrm{p}$ *and* q *explained in the following paragraphs. These four operations and their associated equations are referred to as* context comprehension.

We let $\Gamma, \Delta, \ldots$ *range over objects of* $\mathcal{C}$*, and refer to them as* contexts. *We let* $\delta, \gamma, \ldots$ *range over morphisms, and refer to them as* substitutions. *We refer to* 1 *as the* empty *context; the terminal maps* $\langle\rangle_\Gamma$ *represent the* empty *substitutions.*

If $T(\Gamma) = (U_x)_{x \in X}$*, we write* $\mathrm{Ty}(\Gamma)$ *for the set* X*. We call the elements of* $\mathrm{Ty}(\Gamma)$ types in context Γ*, and let* A, B, C *range over such types. Furthermore, for* $A \in \mathrm{Ty}(\Gamma)$*, we write* $\mathrm{Tm}(\Gamma, A)$ *for the set* U_A *and call the elements of* $\mathrm{Tm}(\Gamma, A)$ terms of type A in context Γ.

For $\gamma : \Delta \to \Gamma$*, the functorial action of* T *yields a morphism*

$$T(\gamma) \in \textbf{\textit{Fam}}\left((\mathrm{Tm}(\Gamma, A))_{A \in \mathrm{Ty}(\Gamma)}, (\mathrm{Tm}(\Delta, B))_{B \in \mathrm{Ty}(\Delta)}\right)$$

consisting of a reindexing function $\_[\gamma] : \mathrm{Ty}(\Gamma) \to \mathrm{Ty}(\Delta)$ *referred to as* substitution in types, *and for each* $A \in \mathrm{Ty}(\Gamma)$ *a function* $\_[\gamma] : \mathrm{Tm}(\Gamma, A) \to \mathrm{Tm}(\Delta, A[\gamma])$*, referred to as* substitution in terms.

Now we turn to the explanation of the operations $\,.\,,\ \langle \_,\_ \rangle,\ \mathrm{p},\ \mathrm{q}$*. Given* $\Gamma \in \mathcal{C}$*,* $A \in \mathrm{Ty}(\Gamma)$*,* $\gamma : \Delta \to \Gamma$*, and* $a \in \mathrm{Tm}(\Delta, A[\gamma])$*, we have*

$$\Gamma.A \in \mathcal{C} \qquad \mathrm{p}_{\Gamma,A} : \Gamma.A \to \Gamma \qquad \mathrm{q}_{\Gamma,A} \in \mathrm{Tm}(\Gamma.A, A[\mathrm{p}_{\Gamma,A}]) \qquad \langle \gamma, a \rangle_A : \Delta \to \Gamma.A.$$

We call $\Gamma.A$ *the* extended *context and* $\langle \gamma, a \rangle_A$ *the* extended *substitution.*

The operations $\,.\,,\ \langle \_,\_ \rangle,\ \mathrm{p},\ \mathrm{q}$ *satisfy the following universal property:* $\langle \gamma, a \rangle_A$ *is the unique substitution satisfying*

$$\mathrm{p}_{\Gamma,A} \circ \langle \gamma, a \rangle_A = \gamma \qquad \textit{and} \qquad \mathrm{q}_{\Gamma,A}[\langle \gamma, a \rangle_A] = a\,.$$

We refer (colloquially) to p *as the* first projection, *and to* q *as the* second projection. *Note that the first equation implies that* $\mathrm{Tm}(\Delta, A[\mathrm{p}_{\Gamma,A}][\langle \gamma, a \rangle]) = \mathrm{Tm}(\Delta, A[\gamma])$ *so that* $\mathrm{q}_{\Gamma,A}[\langle \gamma, a \rangle]$ *and* a *are elements of the same set. Here and below, subscripts are omitted from* $\,.\,,\ \langle \_,\_ \rangle,\ \mathrm{p},\ \mathrm{q}$ *when they can be reconstructed from the context (no pun intended). (End Definition 2.)*

A cwf is thus a structure $(\mathcal{C}, 1, \langle\rangle, T, .\,, \langle \_,\_ \rangle, \mathrm{p}, \mathrm{q})$, subject to equations, for the category and the presheaf, and universal properties, formulated purely equationally, for the terminal object and for context comprehension. The morphisms to be defined next preserve this structure, even in a strict way, "on the nose". We often shorten the notation of a cwf to $(\mathcal{C}, T)$, or even just $\mathcal{C}$, leaving the remaining structure implicit.

If V is a Grothendieck universe, then $\mathbf{Set}_\mathrm{V}$ is the cwf where $\mathcal{C}$ is the category of V-small sets and functions, $\mathrm{Ty}(\Gamma) = \mathrm{V}$ and $\mathrm{Tm}(\Gamma, A) = \Gamma \to A$.

Definition 3. *A* (strict) cwf-morphism F between cwfs $(\mathcal{C}, T_\mathcal{C})$ and $(\mathcal{D}, T_\mathcal{D})$ *consists of*

- *A functor* $F_{\mathrm{fun}} : \mathcal{C} \to \mathcal{D}$*;*
- *A natural transformation* $F_{\mathrm{nat}} : T_\mathcal{C} \Rightarrow (T_\mathcal{D} \circ F_{\mathrm{fun}}^{op})$*;*
- *The terminal object is preserved on the nose:* $F_{\mathrm{fun}}(1_\mathcal{C}) = 1_\mathcal{D}$*;*
- *Context comprehension is preserved on the nose, see below.*

Since F_{nat} *is a natural transformation between* ***Fam****-valued presheaves,* F_{nat} *has a component for any object* Γ *of* $\mathcal{C}$*, and these components are morphisms in* $\textbf{\textit{Fam}}(T_\mathcal{C}(\Gamma), T_\mathcal{D}(F_{\mathrm{fun}}(\Gamma)))$*. Recall that* *morphisms in **Fam** consist of a reindexing function and a family of functions. It is convenient to denote F_{fun}, all reindexing functions, as well as all members of the families of functions, simply by F. Thus, we have $F(A) \in \mathrm{Ty}_{\mathcal{D}}(F(\Gamma))$ and $F(a) \in \mathrm{Tm}_{\mathcal{D}}(F(\Gamma), F(A))$, for all Γ and $A \in \mathrm{Ty}_{\mathcal{C}}(\Gamma)$ and $a \in \mathrm{Tm}_{\mathcal{C}}(\Gamma, A)$.*

Naturality of F_{nat} amounts to preservation of substitution, i.e., for all $\gamma : \Delta \to \Gamma$ in $\mathcal{C}$, we have

$$F(A[\gamma]) = F(A)[F(\gamma)] \qquad\qquad F(a[\gamma]) = F(a)[F(\gamma)]\,.$$

Last but not least, we turn to the preservation of context comprehension on the nose, and require

$$F(\Gamma.A) = F(\Gamma).F(A) \qquad F(\mathrm{p}_{\Gamma,A}) = \mathrm{p}_{F(\Gamma),F(A)} \qquad F(\mathrm{q}_{\Gamma,A}) = \mathrm{q}_{F(\Gamma),F(A)}\,.$$

Note that the universal property implies that $F(\langle \gamma, a\rangle) = \langle F(\gamma), F(a)\rangle$. The same is true for the terminal maps: $F(\langle\rangle_\Gamma) = \langle\rangle_{F(\Gamma)}$. (End Definition 3.)

V-small cwfs with strict cwfs-morphisms form a category, written **CwF**. If $\mathrm{V}' \in \mathrm{V}$ is another Grothendieck universe, then $\mathbf{Set}_{\mathrm{V}'}$ is a V-small cwf.

3. Presentations and Models of Generalized Algebraic Theories

In universal algebra one has the notion of a *signature*, which consists of a set of sort symbols and a set of typed operator symbols. Using the vocabulary of the signature one then specifies the set of equational axioms.

For generalized algebraic theories, the situation is more complicated. First, sorts may depend on other sorts and even on operators, so that sorts and operators cannot be presented as two separated sets. Even more so, sorts and operators may depend on equations to be well-typed, so that separation of these three syntactic categories is not possible. We give an example of the interdependency in the next paragraph, more examples can be found in Section 5.

For readability, we give this example in the language of type theory. Consider a sort X with two operators $x_0 : X$ and $x_1 : X$ and an axiom $x_0 = x_1$, all in the empty context. Consider now a sort $Y(x)$ in the context $x : X$, and a sort $Z(y, y')$ in the context $y : Y(x_0), y' : Y(x_1)$. Finally, consider a sort $W(y, z)$ in the context $y : Y(x_0), z : Z(y, y)$. The sort $W(y, z)$ is only well-typed since $x_0 = x_1$. The same is true for an operator $w(y, z) : W(y, z)$ in the same context as $W(y, z)$. The equation $x_0 = x_1$ could come after the sorts $Y(x)$ and $Z(y, y')$, but has to come before $W(y, z)$.

For reasons mentioned above, we need a more general notion than that of signature in universal algebra, encompassing not only the types of the sort and operator symbols but also the equations. We call this notion the *presentation* of a generalized algebraic theory, or presentation for short. It is nontrivial to define what a presentation is and how it presents a generalized algebraic theory. This requires several steps.

We first define the notion of a presentation Σ and the associated category $\mathbf{CwF}_\Sigma$ of V-small cwfs with a Σ-structure. Each object of $\mathbf{CwF}_\Sigma$ is a V-small cwf with extra structure and each morphism is a cwf-morphism preserving Σ-structure. For this definition, we will need the following auxiliary notions.

A *uniform family of contexts* is a family $\Gamma = (\Gamma_{\mathcal{C}})$ with $\Gamma_{\mathcal{C}}$ a context in $\mathcal{C}$ for each $\mathcal{C} \in \mathbf{CwF}_\Sigma$, such that $F(\Gamma_{\mathcal{C}}) = \Gamma_{\mathcal{D}}$ for all morphisms $F \in \mathbf{CwF}_\Sigma(\mathcal{C}, \mathcal{D})$. If Γ is such a family, a *uniform family of types* over Γ is a family of types $A = (A_{\mathcal{C}})$ with $A_{\mathcal{C}}$ a type over $\Gamma_{\mathcal{C}}$ and $F(A_{\mathcal{C}}) = A_{\mathcal{D}}$ for all morphisms $F \in \mathbf{CwF}_\Sigma(\mathcal{C}, \mathcal{D})$. Finally, given Γ and A, a *uniform family of terms* is a family of terms $a = (a_{\mathcal{C}})$ with $a_{\mathcal{C}} \in \mathrm{Tm}_{\mathcal{C}}(\Gamma_{\mathcal{C}}, A_{\mathcal{C}})$ such that $F(a_{\mathcal{C}}) = a_{\mathcal{D}}$ for all morphisms $F \in \mathbf{CwF}_\Sigma(\mathcal{C}, \mathcal{D})$.

Remark 1. Uniform families appear in Freyd's proof of the adjoint functor theorem (Freyd, 1964), in Reynolds' (1984) and Reynolds and Plotkin's construction (Reynolds and Plotkin, 1993) of an initial algebra for an endofunctor from an impredicative encoding of an inductive type, and in Awodey et al. (2018) construction of an impredicative encoding of a higher inductive type. The common idea in these works is to first construct a weakly initial object and then the initial object is obtained by taking uniform families.

Definition 4. *We define the notion of a presentation Σ and the category $\mathbf{CwF}_\Sigma$ of cwfs with a Σ-structure and cwf-morphisms that preserve Σ-structure. The definition is by induction on the length of Σ. We have the following base case:*

> **The empty presentation.** *The only presentation of length zero is the empty one $\emptyset$. We let $\mathbf{CwF}_\emptyset = \mathbf{CwF}$.*

Assume now that we have defined a presentation Σ of length n and the associated category $\mathbf{CwF}_\Sigma$. Then we define Σ' of length $n + 1$ and the associated category $\mathbf{CwF}_{\Sigma'}$, where Σ' is obtained from Σ by adding a new sort symbol, or a new operator symbol, or a new equation, as follows.

> **Adding a sort symbol.** *Let $\Gamma = (\Gamma_\mathcal{C})$ be a uniform family of contexts indexed by $\mathcal{C} \in \mathbf{CwF}_\Sigma$. Then we can extend Σ with a new sort symbol S relative to Γ, to obtain the generalized algebraic theory $\Sigma' = (\Sigma, (\Gamma, S))$. The objects of $\mathbf{CwF}_{\Sigma'}$ are pairs $(\mathcal{C}, S_\mathcal{C})$, where $\mathcal{C}$ is an object of $\mathbf{CwF}_\Sigma$ and $S_\mathcal{C} \in \mathrm{Ty}_\mathcal{C}(\Gamma_\mathcal{C})$. A morphism in $\mathbf{CwF}_{\Sigma'}((\mathcal{C}, S_\mathcal{C}), (\mathcal{D}, S_\mathcal{D}))$ is a morphism $F \in \mathbf{CwF}_\Sigma(\mathcal{C}, \mathcal{D})$ such that $F(S_\mathcal{C}) = S_\mathcal{D}$.*
>
> **Adding an operator symbol.** *If Γ is a uniform family of contexts and A a uniform family of types over Γ, then we can extend Σ with a new operator symbol f relative to Γ and A, to obtain the generalized algebraic theory $\Sigma' = (\Sigma, (\Gamma, A, f))$. An object of $\mathbf{CwF}_{\Sigma'}$ is a pair $(\mathcal{C}, f_\mathcal{C})$ where $\mathcal{C}$ is an object in $\mathbf{CwF}_\Sigma$ and $f_\mathcal{C} \in \mathrm{Tm}_\mathcal{C}(\Gamma_\mathcal{C}, A_\mathcal{C})$. A morphism in $\mathbf{CwF}_{\Sigma'}((\mathcal{C}, f_\mathcal{C}), (\mathcal{D}, f_\mathcal{D}))$ is a morphism $F \in \mathbf{CwF}_\Sigma(\mathcal{C}, \mathcal{D})$ such that $F(f_\mathcal{C}) = f_\mathcal{D}$.*
>
> **Adding an equation.** *If Γ is a uniform family of contexts, A is a uniform family of types over Γ and a, a' are uniform families of terms in A, then we can extend Σ with a new equation $a = a'$ relative to Γ and A, to obtain the generalized algebraic theory $\Sigma' = (\Sigma, (\Gamma, A, a, a'))$. In this case, $\mathbf{CwF}_{\Sigma'}$ is a full subcategory of $\mathbf{CwF}_\Sigma$. An object $\mathcal{C}$ in $\mathbf{CwF}_{\Sigma'}$ is an object $\mathcal{C}$ of $\mathbf{CwF}_\Sigma$ such that $a_\mathcal{C} = a'_\mathcal{C}$.*

This definition is *syntax independent*. In the next section, we show how to *syntactically* construct an *initial object* $\mathcal{T}_\Sigma$ in $\mathbf{CwF}_\Sigma$ (for an arbitrary presentation Σ) in terms of grammars and inference rules. A context in $\mathcal{T}_\Sigma$ will be an equivalence class $[\Gamma]$ of raw contexts, and similarly for substitutions, types, and terms.

We refer to Section 5.1 where we show a simple instance of this definition: a presentation Σ of internal monoids and its associated category of models $\mathbf{CwF}_\Sigma$ of cwfs with an internal monoid. We also show how to construct the initial cwf $\mathcal{T}_\Sigma$ with an internal monoid.

Remark 2. There is a bijective correspondence between contexts in $\mathcal{T}_\Sigma$ and uniform families of contexts indexed by $\mathcal{C} \in \mathbf{CwF}_\Sigma$. To each context $[\Gamma]$ in $\mathcal{T}_\Sigma$, we associate the uniform family $(\llbracket[\Gamma]\rrbracket_\mathcal{C})$ where $\llbracket - \rrbracket_\mathcal{C}$ is the unique morphism from $\mathcal{T}_\Sigma$ to $\mathcal{C}$. To each family $(\Gamma_\mathcal{C})$ indexed by $\mathcal{C} \in \mathbf{CwF}_\Sigma$ we associate the context $\Gamma_{\mathcal{T}_\Sigma}$ in $\mathcal{T}_\Sigma$. Moreover, $\llbracket[\Gamma]\rrbracket_{\mathcal{T}_\Sigma} = [\Gamma]$ since $\mathcal{T}_\Sigma$ is initial, and $\llbracket\Gamma_{\mathcal{T}_\Sigma}\rrbracket_\mathcal{C} = \Gamma_\mathcal{C}$ because of uniformity and since $\llbracket - \rrbracket_\mathcal{C}$ is a morphism in $\mathbf{CwF}_\Sigma$. For similar reasons, there are bijective correspondences between types and terms in $\mathcal{T}_\Sigma$ and uniform families of types and terms.

Remark 3. Note that in general a presentation Σ is a large set, since uniform families over $\mathbf{CwF}_\Sigma$ are. However, because of the bijective correspondence these uniform families can be replaced by contexts, types, and terms in $\mathcal{T}_\Sigma$. This replacement will turn the semantic Σ into a small syntactic version.

Remark 4. Cartmell's notion of generalized algebraic theory (Cartmell, 1978, 1986) also makes it possible to stipulate equations between type expressions. However, none of our examples makes use of this extra generality. In particular, in Section 5, we present the generalized algebraic theory of cwfs with extra structure for N, Π and a first universe U_0 without needing type equations. The reason is that equations between types become equations between terms in our rendering of dependent type theory as a generalized algebraic theory. See Remark 7 in Section 5.6 for more explanation.

Like Cartmell, we could consider generalized algebraic theories with type equations, but we prefer not to make such equations part of our notion.

4. The Construction of an Initial Object in $\mathbf{CwF}_\Sigma$

In Section 3, we gave a *syntax independent specification* of a generalized algebraic theory as the initial object of the category $\mathbf{CwF}_\Sigma$ of models of a (semantic) presentation Σ. Now we show our main theorem: the *syntactic construction* of such an initial object $\mathcal{T}_\Sigma$. This construction is done in several steps. We first define the "raw" syntactic expressions. Then we define four families of partial equivalence relations (pers) over those raw expressions, corresponding to the four equality judgments. The term model $\mathcal{T}_\Sigma$ is obtained by quotienting with these pers.

The following theorem can be viewed as a generalization of Birkhoff's completeness theorem for equational logic (Birkhoff, 1935).

Theorem 1. The category $\mathbf{CwF}_\Sigma$ has an initial object $\mathcal{T}_\Sigma$, for every presentation Σ of a generalized algebraic theory.

The construction of $\mathcal{T}_\Sigma$ will be by induction on the construction of Σ. It is based on construction of initial cwfs in Castellan et al. (2015, 2017) and we refer the reader to those papers for more details. Here we only provide a sketch and focus on how to extend the construction to $\mathcal{T}_\Sigma$.

For each Σ we will define the following.

(1) A grammar for the *raw syntax*, that is, raw contexts in $\mathtt{Ctx}_\Sigma$, raw substitutions in $\mathtt{Sub}_\Sigma$, raw types in $\mathtt{Ty}_\Sigma$, and raw terms in $\mathtt{Tm}_\Sigma$.

(2) A system of inference rules that generate four families of partial equivalence relations (pers) by a mutual inductive definition:

$$\Gamma = \Gamma' \vdash_\Sigma \qquad \Gamma \vdash_\Sigma A = A' \qquad \Delta \vdash_\Sigma \gamma = \gamma' : \Gamma \qquad \Gamma \vdash_\Sigma a = a' : A$$

where $\Gamma, \Gamma' \in \mathtt{Ctx}_\Sigma$, $\gamma, \gamma' \in \mathtt{Sub}_\Sigma$, $A, A' \in \mathtt{Ty}_\Sigma$, and $a, a' \in \mathtt{Tm}_\Sigma$. These pers define the valid equality judgments of a variable-free version of dependent type theory with explicit substitutions based on the cwf-combinators. The ordinary judgments will be defined as the reflexive instances of these equality judgments. For example, $\Gamma \vdash_\Sigma$, meaning that Γ is a valid context, is defined as the reflexive instance $\Gamma = \Gamma \vdash_\Sigma$.

(3) A cwf $\mathcal{T}_\Sigma$ is then constructed from the equivalence classes of derivable judgments. For example, the contexts in $\mathcal{T}_\Sigma$ are equivalence classes $[\Gamma]$, such that $\Gamma \vdash_\Sigma$. We will show that $\mathcal{T}_\Sigma$ is a cwf with a Σ-structure, that is, an object of $\mathbf{CwF}_\Sigma$.

(4) A $\mathbf{CwF}_\Sigma$-morphism $[\![-]\!] : \mathcal{T}_\Sigma \to \mathcal{C}$ for every $\mathcal{C} \in \mathbf{CwF}_\Sigma$. This is the *interpretation morphism*. This morphism is a partial function defined by induction on the raw syntax, that (whenever it is defined) maps raw contexts to contexts in $\mathcal{C}$, raw substitutions to substitutions in $\mathcal{C}$, raw types to types in $\mathcal{C}$, and raw terms to terms in $\mathcal{C}$. We show that these partial functions preserve the partial equivalence relations so that we can define the interpretation morphism on the equivalence classes. Finally we show that it indeed is a $\mathbf{CwF}_\Sigma$-morphism and the unique such into $\mathcal{C}$.

We begin with the construction for the base case: the **empty presentation** ∅.

(1) We start with the *raw syntax* for the initial pure cwf $\mathcal{T}_{\emptyset}$. It is specified by the following grammar for raw contexts, raw substitutions, raw types, and raw terms.

$$\begin{aligned} \Gamma \in \mathtt{Ctx}_{\emptyset} &::= 1 \mid \Gamma.A \\ \gamma \in \mathtt{Sub}_{\emptyset} &::= \gamma \circ \gamma \mid \mathrm{id}_{\Gamma} \mid \langle\rangle_{\Gamma} \mid \mathrm{p}_A \mid \langle \gamma, a \rangle_A \\ A \in \mathtt{Ty}_{\emptyset} &::= A[\gamma] \\ a \in \mathtt{Tm}_{\emptyset} &::= a[\gamma] \mid \mathrm{q}_A \end{aligned}$$

This grammar generates a language of *cwf-combinators*.

(2) The system of inference rules for $\mathcal{T}_{\emptyset}$ is displayed in Castellan et al. (2015, 2017). It is a system of *general rules*, rules for dependent type theory which come before we introduce any sort symbols and operator symbols and equations (or any rules for the type formers of intuitionistic type theory). We do not have room here to display them, but note that they can be divided into four groups:
- the per rules, amounting to symmetry and transitivity for the four forms of equality judgments;
- preservation rules for judgments, amounting to substitution of equals for equals (an example of such a rule is the *type equality rule*);
- congruence rules for operators expressing that the cwf-combinators preserve equality;
- conversion rules for the cwf-combinators.

(3) Note that the initial cwf $\mathcal{T}_{\emptyset}$ is trivial: its category of contexts contains only a terminal object (the empty context), and there are no types and terms. Nevertheless, the grammar and inference rules used in its definition form a starting point. The grammar for raw types and raw terms will be extended each time we add a new sort symbol or operator symbol, respectively. For each such new symbol and each new equation, we will add a new inference rule. As a consequence, we will generate nontrivial $\mathcal{T}_{\Sigma}$.

(4) The definition of the interpretation morphism $[\![-]\!] : \mathcal{T}_{\emptyset} \to \mathcal{C}$ and its proof of uniqueness are routine and can be found in Castellan et al. (2015, 2017).

Assume now for the induction step that we have defined the grammar, the inference rules, $\mathcal{T}_{\Sigma}$ and the interpretation morphism $[\![-]\!] : \mathcal{T}_{\Sigma} \to \mathcal{C}$ in $\mathbf{CwF}_{\Sigma}$. Let Σ' be Σ extended by a new sort symbol, a new operator symbol, or a new equation. We shall now explain how to define $\mathcal{T}_{\Sigma'}$.

Adding a sort symbol. If $\Gamma \vdash_{\Sigma}$, then we can introduce a new sort symbol S in the context Γ representing the sequence of types of the arguments of S.

1 We add a new production for raw types

$$A ::= S$$

to the productions for $\mathcal{T}_{\Sigma}$.

2 We add the inference rule

$$\Gamma \vdash_{\Sigma'} S$$

to the inference rules for $\mathcal{T}_{\Sigma}$.

3 We define $S_{\mathcal{T}_{\Sigma'}} = [S]$ and $\Sigma' = (\mathcal{T}_{\Sigma}, [S])$.

4 We extend the definition of the interpretation morphism $[\![-]\!]$ to an interpretation morphism $[\![-]\!]' : \mathcal{T}_{\Sigma'} \to \mathcal{C}$ by

$$[\![[S]]\!]' = S_{\mathcal{C}}$$

It follows that this is a morphism in $\mathbf{CwF}_{\Sigma'}$ and that it is unique.

Adding an operator symbol. If $\Gamma \vdash_\Sigma A$, then we can introduce a new operator symbol f, where the context Γ represents the sequence of types of the arguments and A is the type of the result.

1 We add a new production for raw terms

$$a ::= f$$

to the productions for $\mathcal{T}_\Sigma$.

2 We add the inference rule

$$\Gamma \vdash_{\Sigma'} f : A$$

to the inference rules for $\mathcal{T}_\Sigma$.

3 We define $f_{\mathcal{T}_{\Sigma'}} = [f]$ and $\mathcal{T}_{\Sigma'} = (\mathcal{T}_\Sigma, [f])$.

4 We extend the definition of the interpretation morphism $[\![-]\!]$ to an interpretation morphism $[\![-]\!]' : \mathcal{T}_{\Sigma'} \to \mathcal{C}$ by

$$[\![[f]]\!]' = f_\mathcal{C}$$

It follows that this is a morphism in $\mathbf{CwF}_{\Sigma'}$ and that it is unique.

Adding an equation. If $\Gamma \vdash_\Sigma a : A$ and $\Gamma \vdash_\Sigma a' : A$, we can introduce a new equation $a = a'$.

1 $\mathcal{T}_{\Sigma'}$ has the same productions as $\mathcal{T}_\Sigma$.

2 We add the inference rule

$$\Gamma \vdash_{\Sigma'} a = a' : A$$

to the inference rules for $\mathcal{T}_\Sigma$.

3 $\mathcal{T}_{\Sigma'}$ is based on the same raw syntax as $\mathcal{T}_\Sigma$, but the equivalence relation has changed. To show that $\mathcal{T}_{\Sigma'} \in \mathbf{CwF}_{\Sigma'}$ we just need to show that $[a] = [a']$ but this follows from the inference rule $\Gamma \vdash_{\Sigma'} a = a' : A$.

4 In order to define $[\![-]\!]'$ we first define the partial function on the raw syntax to be identical to the partial function on the raw syntax for $[\![-]\!]$. We then prove that this partial function preserves the extended partial equivalence relation and define $[\![-]\!]'$ on the new equivalence classes. It follows that $[\![-]\!]'$ is unique.

This concludes the proof of the theorem.

5. Examples of Generalized Algebraic Theories

We will now display the sort symbols, operator symbols, and equations for the generalized algebraic theories of monoids, categories, and cwfs. We will then show how to add operator symbols and equations when extending cwfs with Π-types, natural numbers N, and a universe closed under Π and N. The models of these theories (the objects of $\mathbf{CwF}_\Sigma$ for the respective Σ) are small cwfs with internal monoids, internal categories, and internal cwfs (possibly with extra structure for internal Π, N, and universes), respectively.

We begin by using the recipe in Definition 4 to construct the presentation of monoids and its associated category of models, that is, of cwfs with an internal monoid. We then follow the recipe in Theorem 1 and construct the initial cwf with an internal monoid.

For ease of readability, we will only present the sort symbols, operator symbols, and equations in the remaining examples by using an informal notation with named variables, rather than the formal notations using cwf-combinators employed in Definitions 3 and 1.

Our final example is the generalized algebraic theory of contextual cwfs, a variant of Cartmell's contextual categories. The contexts in such contextual cwfs come with a length n. We sketch how this can be axiomatized as a generalized algebraic theory with countably many sort symbols ctx_n, sub_n, ty_n, tm_n for an external natural number n (and similarly for the operator symbols and equations). We also indicate how our framework can be extended to cover such generalized algebraic theories.

5.1 Internal monoids

The one-sorted algebraic theory of monoids has two operator symbols, e for identity and $*$ for composition, and associativity and identity laws as equations. As any other one-sorted algebraic theory, the theory of monoids yields a generalized algebraic theory. In ordinary notation with variables it might be rendered as follows, where M is the only sort:

$$\begin{aligned} &\vdash \mathrm{M}\\ &\vdash e : \mathrm{M}\\ x, y : \mathrm{M} &\vdash *(x, y) : \mathrm{M}\\ y : \mathrm{M} &\vdash *(\mathrm{e}, y) = y : \mathrm{M}\\ x : \mathrm{M} &\vdash *(x, \mathrm{e}) = x : \mathrm{M}\\ x, y, z : \mathrm{M} &\vdash *(*(x, y), z) = *(x, *(y, z)) : \mathrm{M} \end{aligned}$$

We now show how the corresponding official (in the sense of Definition 4) presentation of monoids Σ and its associated category of models $\mathbf{CwF}_\Sigma$ are constructed step-wise.

As always, we begin with the empty presentation $\emptyset$ and its category of models $\mathbf{CwF}_\emptyset = \mathbf{CwF}$.

Adding the sort symbol M**.** Each cwf $\mathcal{C}$ has a chosen empty context (terminal object) $1_\mathcal{C}$. Since cwf-morphisms preserve empty contexts on the nose, $1 = (1_\mathcal{C})$ is a uniform family of contexts in $\mathbf{CwF}_\emptyset$. Hence, we can introduce a new sort symbol M in the empty context. The resulting presentation is

$$\Sigma_1 = (\emptyset, (1, \mathrm{M}))$$

The objects of $\mathbf{CwF}_{\Sigma_1}$ are pairs $(\mathcal{C}, \mathrm{M}_\mathcal{C})$, where $\mathcal{C}$ is a cwf and $\mathrm{M}_\mathcal{C} \in \mathrm{Ty}_\mathcal{C}(1_\mathcal{C})$.

Adding the operator symbol for the identity. Since, morphisms in $\mathbf{CwF}_{\Sigma_1}$ preserve both empty contexts $1_\mathcal{C}$ and types $\mathrm{M}_\mathcal{C}$ on the nose, we have a uniform family of contexts $1 = (1_\mathcal{C})$ and a uniform family of types $\mathrm{M} = (\mathrm{M}_\mathcal{C})$ in $\mathbf{CwF}_{\Sigma_1}$. Hence, we can introduce a new operator symbol e (the identity of the monoid). The resulting presentation is

$$\Sigma_2 = (\Sigma_1, (1, \mathrm{M}, e))$$

The objects of $\mathbf{CwF}_{\Sigma_2}$ are triples $(\mathcal{C}, \mathrm{M}_\mathcal{C}, e_\mathcal{C})$, where $\mathcal{C}$ is a cwf, $\mathrm{M}_\mathcal{C} \in \mathrm{Ty}_\mathcal{C}(1_\mathcal{C})$ and $e_\mathcal{C} \in \mathrm{Tm}_\mathcal{C}(1_\mathcal{C}, \mathrm{M}_\mathcal{C})$.

Adding the operator symbol for composition. Again using that cwf-morphisms preserve all cwf-structure and $\mathrm{M}_\mathcal{C}$, we deduce that we have a uniform family of contexts 1.M.M[p] and a uniform family of types M[p][p] in $\mathbf{CwF}_{\Sigma_2}$. Thus we can add a binary operator symbol $*$. The resulting presentation is

$$\Sigma_3 = (\Sigma_2, (1.\mathrm{M}.\mathrm{M}[\mathrm{p}], \mathrm{M}[\mathrm{p}][\mathrm{p}], *))$$

The objects of $\mathbf{CwF}_{\Sigma_3}$ are quadruples $(\mathcal{C}, \mathrm{M}_\mathcal{C}, e_\mathcal{C}, *_\mathcal{C})$, where $\mathcal{C}$ is a cwf, $\mathrm{M}_\mathcal{C} \in \mathrm{Ty}_\mathcal{C}(1_\mathcal{C})$, $e_\mathcal{C} \in \mathrm{Tm}_\mathcal{C}(1_\mathcal{C}, \mathrm{M}_\mathcal{C})$, and $*_\mathcal{C} \in \mathrm{Tm}_\mathcal{C}((1.\mathrm{M}.\mathrm{M}[\mathrm{p}])_\mathcal{C}, (\mathrm{M}[\mathrm{p}][\mathrm{p}])_\mathcal{C})$.

Adding the left identity law. Furthermore, we extend the presentation with the equations stating that e is a left identity as follows:

$$\Sigma_4 = (\Sigma_3, (1.\mathrm{M}, \mathrm{M}[\mathrm{p}], *[\langle\langle\langle\rangle, \mathrm{e}[\langle\rangle]\rangle, \mathrm{q}\rangle], \mathrm{q}))$$

The uniform family of contexts 1.M expresses that the equation has one variable of type M, the uniform family of types M[p] expresses that the two sides of the equation have type M, and the uniform families of terms $*[\langle\langle\langle\rangle, \mathrm{e}[\langle\rangle]\rangle, \mathrm{q}\rangle]$ and q express the two sides of the equation. $\mathbf{CwF}_{\Sigma_4}$ is the full subcategory of $\mathbf{CwF}_{\Sigma_3}$ with objects $\mathcal{C}$ such that $(*[\langle\langle\langle\rangle, \mathrm{e}[\langle\rangle]\rangle, \mathrm{q}\rangle])_{\mathcal{C}} = \mathrm{q}_{\mathcal{C}}$.

Adding the right identity and the associativity laws. Finally, we add the right identity equation and the associativity equation to get the presentations Σ_5 and Σ_6. We omit the details.

We define the presentation of monoids to be $\Sigma = \Sigma_6$. The category $\mathbf{CwF}_\Sigma$ is the category of V-small cwfs with an *internal monoid*. This is a cwf-version of the notion of internal monoid which can be defined in any category with finite products. As we mentioned in Section 2, $\mathbf{Set}_{\mathrm{V}'}$ is an example of a V-small cwf, for a Grothendieck universe $\mathrm{V}' \in \mathrm{V}$. An internal monoid in $\mathbf{Set}_{\mathrm{V}'}$ is a monoid in the usual sense with a V'-small carrier set.

We now sketch the construction of the initial object $\mathcal{T}_\Sigma$ of $\mathbf{CwF}_\Sigma$ following the recipe for introducing sort symbols, operators symbols, and equations in Section 1. (We omit the index Σ in $\vdash_\Sigma$.)

Adding the sort symbol M. First, we have $1 \vdash$ for the empty presentation, so we can add a production for the sort symbol M and the inference rule:

$$1 \vdash \mathrm{M}$$

For later use we infer $1.\mathrm{M} \vdash$ and, using $\mathrm{p} : 1.\mathrm{M} \to 1$, $1.\mathrm{M} \vdash \mathrm{M}[\mathrm{p}]$, so $1.\mathrm{M}.\mathrm{M}[\mathrm{p}] \vdash$. (Here and in the following we sometimes drop indices on raw expressions and for example write p instead of the official p_M, as prescribed by the grammar for the raw syntax.)

Adding the operator symbol for identity. We then add a production for the operator symbol e and the inference rule:

$$1 \vdash \mathrm{e} : \mathrm{M}$$

Again for later use we infer $1.\mathrm{M} \vdash \mathrm{e}[\mathrm{p}] : \mathrm{M}[\mathrm{p}]$. Note that here $\mathrm{p} = \langle\rangle$, the empty substitution $1.\mathrm{M} \to 1$, since there is only one substitution $1.\mathrm{M} \to 1$.

Adding the operator symbol for composition. We then add a production for the binary operator symbol $*$. Using another $\mathrm{p} : 1.\mathrm{M}.\mathrm{M}[\mathrm{p}] \to 1.\mathrm{M}$ (note the different type), we can derive $1.\mathrm{M}.\mathrm{M}[\mathrm{p}] \vdash \mathrm{M}[\mathrm{p}][\mathrm{p}]$, so we can add the inference rule

$$1.\mathrm{M}.\mathrm{M}[\mathrm{p}] \vdash * : \mathrm{M}[\mathrm{p}][\mathrm{p}]$$

Note that we project M on the right twice, reflecting that $*$ is binary.

Adding the left identity law. We can derive $1.\mathrm{M} \vdash \mathrm{q} : \mathrm{M}[\mathrm{p}]$. With some effort, using previous inferences, we can derive $1.\mathrm{M} \vdash *[\langle\langle\langle\rangle, \mathrm{e}[\langle\rangle]\rangle, \mathrm{q}\rangle] : \mathrm{M}[\mathrm{p}]$. Hence we can add the inference rule for the equation (e is a left identity):

$$1.\mathrm{M} \vdash *[\langle\langle\langle\rangle, \mathrm{e}[\langle\rangle]\rangle, \mathrm{q}\rangle] = \mathrm{q} : \mathrm{M}[\mathrm{p}]$$

Adding the right identity and the associativity laws. We omit the details.

The resulting initial object $\mathcal{T}_\Sigma = \mathcal{T}_{\Sigma_6}$ is generated by a system of grammar and inference rules for dependent type theory with an internal monoid. In this theory, we can prove statements such as

$$\Gamma \vdash a : \mathrm{M}$$

stating that a is a well-formed monoid expression in the context Γ and

$$\Gamma \vdash a = a' : \mathrm{M}$$

stating that $a = a'$ is a derivable equation between monoid expressions in the context Γ. Note that both contexts and monoid expressions use cwf-combinators and are variable-free.

Of course, using dependent type theory for reasoning about monoid expressions is overkill; monoids form a single-sorted algebraic theory in the usual sense. The remaining examples will use dependent types in an essential way. However, for reasons of readability we will from now on only use ordinary notation with named variables. Hopefully, it is clear from the above how to formally construct the corresponding official presentations, categories of models, and initial models using cwf-combinators. For example, these constructions for the theory of internal categories are similar to the constructions for the theory of internal monoids.

5.2 Internal categories

The generalized algebraic theory of categories was one of Cartmell's motivating examples. It has the following sort symbols, operator symbols, and equations.

Sort symbols:

$$\begin{gathered} \vdash \mathrm{obj} \\ \Delta, \Gamma : \mathrm{obj} \vdash \mathrm{hom}(\Delta, \Gamma) \end{gathered}$$

Operator symbols:

$$\begin{aligned} \Gamma : \mathrm{obj} &\vdash \mathrm{id}_\Gamma : \mathrm{hom}(\Gamma, \Gamma) \\ \Xi, \Delta, \Gamma : \mathrm{obj}, \gamma : \mathrm{hom}(\Delta, \Gamma), \delta : \mathrm{hom}(\Xi, \Delta) &\vdash \gamma \circ \delta : \mathrm{hom}(\Xi, \Gamma) \end{aligned}$$

Equations:

$$\begin{aligned} \Delta, \Gamma : \mathrm{obj}, \gamma : \mathrm{hom}(\Delta, \Gamma) &\vdash \mathrm{id}_\Gamma \circ \gamma = \gamma : \mathrm{hom}(\Delta, \Gamma) \\ \Delta, \Gamma : \mathrm{obj}, \gamma : \mathrm{hom}(\Delta, \Gamma) &\vdash \gamma \circ \mathrm{id}_\Delta = \gamma : \mathrm{hom}(\Delta, \Gamma) \\ \Theta, \Xi, \Delta, \Gamma : \mathrm{obj}, \gamma : \mathrm{hom}(\Delta, \Gamma), \delta : \mathrm{hom}(\Xi, \Delta), \xi : \mathrm{hom}(\Theta, \Xi) &\vdash (\gamma \circ \delta) \circ \xi = \gamma \circ (\delta \circ \xi) : \mathrm{hom}(\Theta, \Gamma) \end{aligned}$$

Note that composition is officially an operator symbol with five arguments. In the official notation, we should write $\gamma \circ_{\Xi,\Delta,\Gamma} \delta$, but we suppress the context arguments Ξ, Δ, Γ. We will do so for some other operations too.

The rendering of the generalized algebraic theory of categories in cwf-combinator language and the proof that it indeed yields a presentation are similar to what they were for the generalized algebraic theory of monoids. The inference rules for the two sort symbols in cwf-combinator language are

$$\begin{gathered} 1 \vdash \mathrm{obj} \\ 1.\mathrm{obj}.\mathrm{obj}[\mathrm{p}] \vdash \mathrm{hom} \end{gathered}$$

and the operator symbols for identity

$$1.\mathrm{obj} \vdash \mathrm{e} : \mathrm{hom}[\langle \mathrm{id}, \mathrm{q} \rangle]$$

We omit the verbose cwf-renderings of the operator symbol for composition and the equations.

A small cwf with extra structure for the generalized algebraic theory of categories is a small cwf with an *internal category*. This is a cwf-based analogue of the usual notion of internal category in a category with finite limits. As shown by Martin Hofmann (1994, 1996), every category with finite limits yields a category with attributes, and hence a cwf. However, not every cwf has finite limits. To achieve this we need more structure. As shown by Clairambault and Dybjer (2011, 2014), the 2-category of categories with finite limits is biequivalent to the 2-category of democratic cwfs that support Σ-types and extensional identity types.

An internal category in the V-small cwf $\mathbf{Set}_{\mathrm{V}'}$ is a category in the usual sense with a V'-small set of objects. For example, if $\mathrm{V}'' \in \mathrm{V}'$ is yet another Grothendieck universe, then $\mathbf{Set}_{\mathrm{V}''}$ is an internal category in $\mathbf{Set}_{\mathrm{V}'}$.

5.3 Internal cwfs

The generalized algebraic theory of cwfs is obtained by extending the generalized algebraic theory of categories with new sort symbols, operator symbols, and equations for a family valued functor, a terminal object, and context comprehension. We here rename the sort obj of objects of the category of contexts to ctx. To emphasize the difference between the internal notion of cwf and the external notions (introduced in Section 2), our notation for sort symbols in the generalized algebraic theory of internal cwfs use lower case letters for the sort symbols. For example, we write ty and tm in contrast to the upper case letters for the external notions Ty and Tm. We will however overload notation for operator symbols, and for example use ∘ both for composition in a cwf and for the operator symbol in the generalized algebraic theory of cwfs.

5.3.1 The extension with a family valued functor.

Sort symbols:

$$\Gamma : \mathrm{ctx} \vdash \mathrm{ty}(\Gamma)$$

$$\Gamma : \mathrm{ctx}, A : \mathrm{ty}(\Gamma) \vdash \mathrm{tm}(\Gamma, A)$$

Operator symbols:

$$\Gamma, \Delta : \mathrm{ctx}, A : \mathrm{ty}(\Gamma), \gamma : \mathrm{hom}(\Delta, \Gamma) \vdash A[\gamma] : \mathrm{ty}(\Delta)$$

$$\Gamma, \Delta : \mathrm{ctx}, A : \mathrm{ty}(\Gamma), \gamma : \mathrm{hom}(\Delta, \Gamma), a : \mathrm{tm}(\Gamma, A) \vdash a[\gamma] : \mathrm{tm}(\Delta, A[\gamma])$$

Equations:

$$\Gamma : \mathrm{ctx}, A : \mathrm{ty}(\Gamma) \vdash A[\mathrm{id}_\Gamma] = A : \mathrm{ty}(\Gamma)$$

$$\Gamma : \mathrm{ctx}, A : \mathrm{ty}(\Gamma), a : \mathrm{tm}(\Gamma, A) \vdash a[\mathrm{id}_\Gamma] = a : \mathrm{tm}(\Gamma, A)$$

$$\Xi, \Delta, \Gamma : \mathrm{ctx}, \delta : \mathrm{hom}(\Xi, \Delta), \gamma : \mathrm{hom}(\Delta, \Gamma), A : \mathrm{ty}(\Gamma) \vdash A[\gamma \circ \delta] = A[\gamma][\delta] : \mathrm{ty}(\Xi)$$

$$\Xi, \Delta, \Gamma : \mathrm{ctx}, \delta : \mathrm{hom}(\Xi, \Delta), \gamma : \mathrm{hom}(\Delta, \Gamma), A : \mathrm{ty}(\Gamma), a : \mathrm{tm}(\Gamma, A) \vdash a[\gamma \circ \delta] = a[\gamma][\delta] : \mathrm{tm}(\Xi, A[\gamma \circ \delta])$$

5.3.2 The extension with a terminal object.

No new sorts are required.

Operator symbols:

$$\vdash 1 : \mathrm{ctx}$$

$$\Gamma : \mathrm{ctx} \vdash \langle\rangle_\Gamma : \mathrm{hom}(\Gamma, 1)$$

Equations:

$$\vdash \mathrm{id}_1 = \langle\rangle_1 : \mathrm{hom}(1, 1)$$

$$\Gamma, \Delta : \mathrm{ctx}, \gamma : \mathrm{hom}(\Delta, \Gamma) \vdash \langle\rangle_\Gamma \circ \gamma = \langle\rangle_\Delta : \mathrm{hom}(\Delta, 1)$$

(The latter two equations are better for term rewriting than the obvious single one expressing the uniqueness of $\langle\rangle_\Gamma$.)

5.3.3 The extension with context comprehension.

No new sorts are required.

Operator symbols:

$$\Gamma : \mathrm{ctx}, A : \mathrm{ty}(\Gamma) \vdash \Gamma.A : \mathrm{ctx}$$

$$\Gamma, \Delta : \mathrm{ctx}, A : \mathrm{ty}(\Gamma), \gamma : \mathrm{hom}(\Delta, \Gamma), a : \mathrm{tm}(\Delta, A[\gamma]) \vdash \langle \gamma, a\rangle : \mathrm{hom}(\Delta, \Gamma.A)$$

$$\Gamma : \mathrm{ctx}, A : \mathrm{ty}(\Gamma) \vdash \mathrm{p} : \mathrm{hom}(\Gamma.A, \Gamma)$$

$$\Gamma : \mathrm{ctx}, A : \mathrm{ty}(\Gamma) \vdash \mathrm{q} : \mathrm{tm}(\Gamma.A, A[\mathrm{p}])$$

Equations:

$$
\begin{aligned}
\Gamma, \Delta : \mathrm{ctx}, A : \mathrm{ty}(\Gamma), \gamma : \mathrm{hom}(\Delta, \Gamma), a : \mathrm{tm}(\Delta, A[\gamma]) &\vdash \mathrm{p} \circ \langle \gamma, a \rangle = \gamma : \mathrm{hom}(\Delta, \Gamma) \\
\Gamma, \Delta : \mathrm{ctx}, A : \mathrm{ty}(\Gamma), \gamma : \mathrm{hom}(\Delta, \Gamma), a : \mathrm{tm}(\Delta, A[\gamma]) &\vdash \mathrm{q}[\langle \gamma, a \rangle] = a : \mathrm{tm}(\Delta, A[\gamma]) \\
\Gamma, \Delta, \Xi : \mathrm{ctx}, A : \mathrm{ty}(\Gamma), \gamma : \mathrm{hom}(\Delta, \Gamma), a : \mathrm{tm}(\Delta, A[\gamma]), \delta : \mathrm{hom}(\Xi, \Delta) &\vdash \langle \gamma, a \rangle \circ \delta = \langle \gamma \circ \delta, a[\delta] \rangle : \mathrm{hom}(\Xi, \Gamma.A) \\
\Gamma : \mathrm{ctx}, A : \mathrm{ty}(\Gamma) &\vdash \mathrm{id}_{\Gamma.A} = \langle \mathrm{p}, \mathrm{q} \rangle : \mathrm{hom}(\Gamma.A, \Gamma.A)
\end{aligned}
$$

(If $\mathrm{p} \circ \delta = \gamma$ and $\mathrm{q}[\delta] = a$, we get $\langle \gamma, a \rangle = \langle \mathrm{p} \circ \delta, \mathrm{q}[\delta] \rangle = \langle \mathrm{p}, \mathrm{q} \rangle \circ \delta = \delta$, the uniqueness requirement of the universal property. However, the equation for surjective pairing is not left-linear and with a variable on one side, which is not good for rewriting.)

If Σ is the presentation of the generalized algebraic theory of cwfs, then $\mathbf{CwF}_\Sigma$ is the category of V-small cwfs with an internal cwf. Again we assume two more Grothendieck universes V' and V'', where $\mathrm{V}'' \in \mathrm{V}' \in \mathrm{V}$. Then the V-small cwf $\mathbf{Set}_{\mathrm{V}'}$ has an internal cwf of V''-small sets obtained by interpreting the sort of objects ctx as the V'-small set V'' and the sorts of types $\mathrm{ty}(\Gamma)$ also as V''.

5.4 Internal cwfs with Π-types

We add three operator symbols in addition to the operator symbols for cwfs in Sections 5.2 and 5.3:

$$
\begin{aligned}
\Gamma : \mathrm{ctx}, A : \mathrm{ty}(\Gamma), B : \mathrm{ty}(\Gamma.A) &\vdash \Pi(A, B) : \mathrm{ty}(\Gamma) \\
\Gamma : \mathrm{ctx}, A : \mathrm{ty}(\Gamma), B : \mathrm{ty}(\Gamma.A), b : \mathrm{tm}(\Gamma.A, B) &\vdash \lambda(b) : \mathrm{tm}(\Gamma, \Pi(A, B)) \\
\Gamma : \mathrm{ctx}, A : \mathrm{ty}(\Gamma), B : \mathrm{ty}(\Gamma.A), c : \mathrm{tm}(\Gamma, \Pi(A, B)), a : \mathrm{tm}(\Gamma, A) &\vdash \mathsf{app}(c, a) : \mathrm{tm}(\Gamma, B[\langle \mathrm{id}, a \rangle])
\end{aligned}
$$

(again omitting some of the official arguments) and equations for β, η (also omitting the context and type of the equality judgment)

$$
\begin{aligned}
\mathsf{app}(\lambda(b), a) &= b[\langle \mathrm{id}, a \rangle] \\
\lambda(\mathsf{app}(c[\mathrm{p}], \mathrm{q})) &= c
\end{aligned}
$$

and commutation with respect to substitution:

$$
\begin{aligned}
\Pi(A, B)[\gamma] &= \Pi(A[\gamma], B[\gamma^+]) \\
\lambda(b)[\gamma] &= \lambda(b[\gamma^+]) \\
\mathsf{app}(c, a)[\gamma] &= \mathsf{app}(c[\gamma], a[\gamma])
\end{aligned}
$$

where $\gamma^+ = \langle \gamma \circ \mathrm{p}, \mathrm{q} \rangle$.

Remark 5. Cartmell (1986) defines a generalized algebraic theory for Σ-types as follows. If we start with any generalized algebraic theory with a sort symbol A in the empty context and a sort symbol B in context $x : A$, then we can extend it with a new sort symbol ΣB in the empty context and operator symbols and equations for the two projections and the pairing. In a similar way, we could extend any such generalized algebraic theory with a new sort symbol ΠB in the empty context and operator symbols for λ and app and the equations for β and η. The reader should be aware of the difference between the resulting generalized algebraic theories and our generalized algebraic theory for cwfs with Π-types.

Remark 6. Furthermore, just as we can extend the internal notion of cwf with Σ- and Π-types we can extend the external notion of cwf defined in Section 2 with structures for Σ- and Π-types (Castellan et al., 2015, 2017).

5.5 Internal cwfs with Π and N

Furthermore, we add the operator symbol

$$\Gamma : \mathrm{ctx} \vdash \mathsf{N}_\Gamma : \mathrm{ty}(\Gamma)$$

We also add operator symbols for $0, \mathrm{s}, \mathrm{R}$ and the equations for R and for commutativity with substitution, but omit the details. Note that the type of the primitive recursion operator R relies on the presentation of Π-types.

5.6 Internal cwfs with U_0 closed under Π and N

We add four more operator symbols

$$\begin{aligned} \Gamma : \mathrm{ctx} &\vdash (\mathsf{U}_0)_\Gamma : \mathrm{ty}(\Gamma) \\ \Gamma : \mathrm{ctx}, a : \mathrm{tm}(\Gamma, (\mathsf{U}_0)_\Gamma) &\vdash \mathrm{T}_0(a) : \mathrm{ty}(\Gamma) \\ \Gamma : \mathrm{ctx} &\vdash \mathsf{N}^0_\Gamma : \mathrm{tm}(\Gamma, (\mathsf{U}_0)_\Gamma) \\ \Gamma : \mathrm{ctx}, a : \mathrm{tm}(\Gamma, (\mathsf{U}_0)_\Gamma), b : \mathrm{tm}(\Gamma.\mathrm{T}_0(a), (\mathsf{U}_0)_\Gamma)) &\vdash \Pi^0(a, b) : \mathrm{tm}(\Gamma, (\mathsf{U}_0)_\Gamma) \end{aligned}$$

$(\mathsf{U}_0)_\Gamma$ is the universe (a type) relative to the context Γ; T_0 is the decoding operation mapping a term in the universe to the corresponding type; N^0 is the code for N in the universe, and Π^0 forms codes for Π-types in the universe. (Note that we have dropped the context argument of T_0 and Π^0.)

We add the decoding equations:

$$\begin{aligned} \mathrm{T}_0(\mathsf{N}^0_\Gamma) &= \mathsf{N}_\Gamma \\ \mathrm{T}_0(\Pi^0(a, b)) &= \Pi(\mathrm{T}_0(a), \mathrm{T}_0(b)) \end{aligned}$$

and the equations for preservation of substitution:

$$\begin{aligned} (\mathsf{U}_0)_\Gamma[\gamma] &= (\mathsf{U}_0)_\Delta \\ \mathrm{T}_0(a)[\gamma] &= \mathrm{T}_0(a[\gamma]) \\ \mathsf{N}^0_\Gamma[\gamma] &= \mathsf{N}^0_\Delta \\ \Pi^0(a, b)[\gamma] &= \Pi^0(a[\gamma], b[\gamma^+]) \end{aligned}$$

Remark 7. Note that all equations are between *terms* in the generalized algebraic theory of cwfs with extra structure for N, Π, and U_0; we do not need the extra generality of stipulating type equations as discussed in the introduction. For example, $\mathrm{T}_0(\mathsf{N}^0_\Gamma) = \mathsf{N}_\Gamma$ is an equation between *internal* types, that is, terms of type $\mathrm{ty}(\Gamma)$.

Remark 8. Also note that the generalized algebraic theory for the universe is inevitably *à la Tarski* in the sense that we distinguish between types and terms in a cwf and we must have an operation decoding a term into a type. However, Martin-Löf's distinction between *à la Russell* and *à la Tarski* (Martin-Löf, 1984) is a distinction between two different formulations of the raw syntax and inference rules of type theory.

5.7 A possible refinement to internal contextual cwfs

Our treatment can be adapted to some non finitely presented generalized algebraic theories. If we have an increasing sequence of presentations Σ_n, we can consider their union. For instance, we can describe a generalized algebraic theory of contextual cwfs (Castellan et al., 2021) (similar to Cartmell's contextual categories and Voevodsky's *C*-systems) by the following stratification of the theory of cwfs. We replace the sort ctx by a sequence of sorts $\mathrm{ctx}_0, \ \mathrm{ctx}_1, \ \ldots$, where ctx_n represents the sort of contexts of length n and a corresponding sequence of sorts $\mathrm{ty}_n(\Gamma)$ for Γ in ctx_n and $\mathrm{tm}_n(\Gamma, A)$ for A in $\mathrm{ty}_n(\Gamma)$. Context extension $\Gamma.A$ is now in ctx_{n+1} if A is in $\mathrm{ty}_n(\Gamma)$ and so on. We also add *destructors*: we have $\mathrm{ft}(\Gamma)$ in ctx_n and $\mathrm{st}(\Gamma)$ in $\mathrm{ty}_n(\mathrm{ft}(\Gamma))$ with $\Gamma = \mathrm{ft}(\Gamma).\mathrm{st}(\Gamma)$. Similarly, we have a stratification of the sort of substitutions $\mathrm{hom}_{n,m}\,(\Delta, \Gamma)$ for Δ in ctx_n and Γ in ctx_m. The resulting models are *internal contextual cwfs* in a cwf.

Remark 9. Generalized algebraic presentations of contextual categories (C-systems) have been suggested by Voevodsky (2014) and Cartmell (2018).

6. Related Work

Streicher (1991) defined *doctrines of constructions* (contextual categories with suitable extra structure) as a notion of model of the Calculus of Constructions. He also constructed a term model and remarked that it is an initial object in a category of doctrines of constructions. Recently, Brunerie et al (2019) presented a formalized proof in the Agda system that a formal system for Martin-Löf type theory forms an initial object in a category of contextual categories with extra structure for the type formers.

More generally, Voevodsky (2017) outlined a new vision of the theory of syntax and semantics of dependent type theories. In this vision formal systems for dependent type theory are proved to be initial in suitable categories of models (*the initiality conjecture*). The above-mentioned contributions by Streicher and Brunerie et al are two examples of such characterizations. However, Voevodsky's aim was to go further and characterize a whole class of type theories and prove a general initiality result for this class with the aim to form the basis for a general metatheory of dependent type theory. Our work can be viewed as a contribution to Voevodsky's programme, since we prove an initiality theorem for the whole class of finitely presented generalized algebraic theories. Another characterization of a general class of dependent type theories and their initial models has been proposed by Uemura (2019). Another related contribution is Palmgren and Vickers' 2007 construction of initial models of essentially algebraic theories.

Altenkirch and Kaposi (2016) gave several examples of *quotient inductive-inductive types (qiits)*. Their main example is a definition of dependent type theory with Π-types and a universe, as a simultaneous definition in the Agda system (agda) of the data types Ctx of contexts, $\mathrm{Sub}(\Delta, \Gamma)$ of substitutions, $\mathrm{Ty}(\Gamma)$, and $\mathrm{Tm}(\Gamma, A)$ of terms. Their definition is *inductive-inductive* (Forsberg and Setzer, 2010), since the index sets of Sub, Ty, and Tm are generated simultaneously, and as a consequence are not indexed inductive definitions in the usual sense where the index sets are fixed in advance. Furthermore, it is a quotient inductive-inductive type since they also have constructors for identity types, as in a *higher inductive type*. There is a close relationship between this qiit and our initial internal cwf with Π-types and a universe. Like our definition, their qiit-definition uses cwf-combinators. Moreover, our sort symbols correspond to their formation rules (data type constructors), our operator symbols correspond to their introduction rules (constructors), and our equations correspond to their propositional identities. However, a differerence is that our equations are judgmental equalities while theirs are propositional. As a consequence, they use transport maps when moving between identical types.

The notion of qiit is the latest in a series of generalizations of inductive type (inductive family, inductive-recursive type and family, inductive-inductive type, higher inductive type) extending Martin-Löf type theory. Kaposi et al. (2019) developed a general theory of qiits. This includes a notion of signature for a qiit, the notion of an algebra of such a signature, and a construction of initial algebras. In particular, they introduce a domain-specific type theory of signatures (and implement it in Agda), and define a signature for a qiit to be a context in this theory. It would be interesting to try to relate such signatures for qiits to the presentations of generalized algebraic theories in our paper, but this is beyond the scope of the present paper.

Acknowledgements. We are grateful to the anonymous referees for constructive criticism and pointers to related work. We would also like to thank Andrej Bauer, John Cartmell, and Christian Sattler for useful comments.

References

The Agda wiki. `http://appserv.cs.chalmers.se/users/ulfn/wiki/agda.php`.

Aczel, P. (1978). The type theoretic interpretation of constructive set theory. In: *Logic Colloquium 1977 (Proc. Conf., Wrocław, 1977)*, vol. 96. *Stud. Logic Foundations Math.* Amsterdam-New York, North-Holland, 55–66 .

Aczel, P. (1998). On relating type theories and set theories. In: Altenkirch, T., Reus, B. and Naraschewski, W. (eds.) *Types for Proofs and Programs*. Springer, 33–46.

Altenkirch, T. and Kaposi, A. (2016). Type theory in type theory using quotient inductive types. In: Bodík, R. and Majumdar, R. (eds.) *Proceedings of the 43rd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2016, St. Petersburg, FL, USA, January 20–22, 2016*. ACM, 18–29.

Awodey, S., Frey, J. and Speight, S. (2018). Impredicative encodings of (higher) inductive types. In: *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS 2018)*. New York, NY, USA, Association for Computing Machinery.

Birkhoff, G. (1935). On the structure of abstract algebras. In: *Proceeding of the Cambridge Philosophical Society*, vol. 31, 433–454.

Brunerie, G. (2019). A formalization of the initiality conjecture in Agda. Slides from a talk about joint work with Menno de Boer, Peter Lumsdaine, and Anders Mörtberg, at HoTT, Pittsburgh, CMU.

Cartmell, J. (1978). *Generalized Algebraic Theories and Contextual Categories*. D. Phil., Oxford University.

Cartmell, J. (1986). Generalized algebraic theories and contextual categories. *Annals of Pure and Applied Logic* **32** 209–243.

Cartmell, J. (2018). Generalised algebraic axiomatisations of contextual categories. `https://www.researchgate.net/publication/325763538_Generalised_Algebraic_Axiomatisations_of_Contextual_Categories`.

Castellan, S., Clairambault, P. and Dybjer, P. (2015). Undecidability of equality in the free locally cartesian closed category. In *13th International Conference on Typed Lambda Calculi and Applications, TLCA 2015, July 1–3, 2015, Warsaw, Poland*, 138–152.

Castellan, S., Clairambault, P. and Dybjer, P. (2017). Undecidability of equality in the free locally cartesian closed category (extended version). *Logical Methods in Computer Science* **13** (4).

Castellan, S., Clairambault, P. and Dybjer, P. (2021). Categories with families: Unityped, simply typed, and dependently typed. In: Casadio, C. and Scott, P. J. (eds.) *Joachim Lambek: The interplay of Mathematics, Logic, and Linguistics*, Outstanding Contributions to Logic, Editor-in-Chief: Sven-Ove Hansson. vol. 20. Springer.

Clairambault, P. and Dybjer, P. (2011). The biequivalence of locally cartesian closed categories and Martin-Löf type theories. In: *Typed Lambda Calculi and Applications - 10th International Conference, TLCA 2011, Novi Sad, Serbia, June 1–3, 2011. Proceedings*, 91–106.

Clairambault, P. and Dybjer, P. (2014). The biequivalence of locally cartesian closed categories and Martin-Löf type theories. *Mathematical Structures in Computer Science* **24** (6).

Curien, P., Garner, R. and Hofmann, M. (2014). Revisiting the categorical interpretation of dependent type theory. *Theoretical Computer Science* **546** 99–119.

Dybjer, P. (1996). Internal type theory. In: *TYPES 1995, Types for Proofs and Programs*, vol. 1158. Lecture Notes in Computer Science. Springer, 120–134.

Forsberg, F. N. and Setzer, A. (2010). Inductive-inductive definitions. In Dawar, A. and Veith, H., editors, *Computer Science Logic, 24th International Workshop, CSL 2010, 19th Annual Conference of the EACSL, Brno, Czech Republic, August 23–27, 2010. Proceedings*, vol. 6247. Lecture Notes in Computer Science. Springer, 454–468.

Freyd, P. (1964). *Abelian Categories*. New York, Harper and Row.

Hofmann, M. (1994). On the interpretation of type theory in locally cartesian closed categories. In: Pacholski, L. and Tiuryn, J. (eds.) *CSL*, vol. 933. Lecture Notes in Computer Science. Springer.

Hofmann, M. (1995). Conservativity of equality reflection over intensional type theory. In: Berardi, S. and Coppo, M. (eds.) *Types for Proofs and Programs, International Workshop TYPES 1995, Torino, Italy, June 5–8, 1995, Selected Papers*, vol. 1158. Lecture Notes in Computer Science. Springer, 153–164.

Hofmann, M. (1996). Syntax and semantics of dependent types. In: Pitts, A. and Dybjer, P. (eds.) *Semantics and Logics of Computation*. Cambridge University Press.

Hofmann, M. (1997). *Extensional Constructs in Intensional Type Theory*. CPHC/BCS distinguished dissertations. Springer.

Hofmann, M. and Streicher, T. (1994). A groupoid model refutes uniqueness of identity proofs. In: *LICS 1994*, 208–212. IEEE Press.

Kaposi, A., Kovács, A. and Altenkirch, T. (2019). Constructing quotient inductive-inductive types. *Proceedings of the ACM on Programming Languages*, **3** (POPL) 2:1–2:24.

Martin-Löf, P. (1984). *Intuitionistic Type Theory*. Bibliopolis.

Martin-Löf, P. (1992). Substitution calculus. Notes from a lecture given in Göteborg.

Palmgren, E. and Vickers, S. J. (2007). Partial Horn logic and cartesian categories. *Annals of Pure and Applied Logic* **145** (3) 314–353.

Reynolds, J. C. (1984). Polymorphism is not set-theoretic. In: Kahn, G., MacQueen, D. B., and Plotkin, G. D., editors, *Semantics of Data Types, International Symposium, Sophia-Antipolis, France, June 27–29, 1984, Proceedings*, vol. 173. Lecture Notes in Computer Science. Springer, 145–156.

Reynolds, J. C. and Plotkin, G. D. (1993). On functors expressible in the polymorphic typed lambda calculus. *Information and Computation* **105** (1) 1–29.

Streicher, T. (1991). *Semantics of Type Theory*. Birkhäuser.

Uemura, T. (2019). A general framework for the semantics of type theory. ArXiv, abs/1904.04097.

Voevodsky, V. (2014). Subsystems and regular quotients of C-systems. ArXiv, abs/1406.7413.

Voevodsky, V. (2017). Models, interpretations and the initiality conjectures. Notes from a lecture at the 2017 Logic Colloquium in Stockholm, special session on Category Theory and Type Theory in honor of Per Martin-Lf on his 75th birthday. https://www.math.ias.edu/Voevodsky/files/files-annotated/Dropbox/Unfinished_papers/Type_systems/Notes_on_Type_Systems/2017_LC_Martin-Lof_special_session/BSL_extended_abstract.pdf.

Cite this article: Bezem M, Coquand T, Dybjer P and Escardó M (2021). On generalized algebraic theories and categories with families. *Mathematical Structures in Computer Science.* https://doi.org/10.1017/S0960129521000268