The FOLDS theory associated to a contextual category

Chaitanya Leena Subramaniam
(joint w.i.p. with Michael Shulman)

University of San Diego

HoTT 2023

22 May 2023

Dependent types and (higher) algebra

HoTT (i.e. "space theory") should be to higher algebra what set theory is to ordinary algebra.

Dependent types are fundamental to HoTT (but not to set theory).

$$x, y{:}A \vdash \mathrm{Id}_A(x, y) \text{ type}$$

So dependent types must play a rôle in (universal) higher algebra.

Dependently typed algebraic theories

A well-known approach to doing algebra with dependent types are Cartmell's **generalised algebraic theories** [Car78].

A GAT consists of:

- type declarations:

$$\Gamma \vdash A \text{ type}$$

- function symbols:

$$\Gamma \vdash f : A$$

- and equational axioms

$$\Gamma \vdash A = B \text{ type} \qquad\qquad \Gamma \vdash t = u : A$$

The GAT of categories

Many algebraic structures are described by GATs, e.g. **categories**.

$$\text{Types:} \quad \begin{cases} \vdash O \quad \text{type} \\ x{:}O, y{:}O \vdash A(x,y) \quad \text{type} \end{cases}$$

The GAT of categories.

The GAT of categories

Many algebraic structures are described by GATs, e.g. **categories**.

$$\text{Types:} \quad \begin{cases} \vdash O \quad \text{type} \\ x{:}O, y{:}O \vdash A(x,y) \quad \text{type} \end{cases}$$

$$\text{Function symbols:} \quad \begin{cases} x{:}O \vdash \mathrm{i}(x) : A(x,x) \\ x,y,z{:}O, f{:}A(x,y), g{:}A(y,z) \vdash \mathrm{c}(g,f) : A(x,z) \end{cases}$$

The GAT of categories.

The GAT of categories

Many algebraic structures are described by GATs, e.g. **categories**.

$$\text{Types:} \quad \begin{cases} \vdash O \ \ \text{type} \\ x{:}O, y{:}O \vdash A(x,y) \ \ \text{type} \end{cases}$$

$$\text{Function symbols:} \quad \begin{cases} x{:}O \vdash \mathrm{i}(x) : A(x,x) \\ x,y,z{:}O, f{:}A(x,y), g{:}A(y,z) \vdash \mathrm{c}(g,f) : A(x,z) \end{cases}$$

$$\text{Axioms:} \quad \begin{cases} x,y{:}O, f{:}A(x,y) \vdash \mathrm{c}(\mathrm{i}(y),f) = f : A(x,y) \\ x,y{:}O, f{:}A(x,y) \vdash \mathrm{c}(f,\mathrm{i}(x)) = f : A(x,y) \\ \\ \\ \end{cases}$$

The GAT of categories.

The GAT of categories

Many algebraic structures are described by GATs, e.g. **categories**.

$$\text{Types:} \quad \begin{cases} \vdash O \quad \text{type} \\ x{:}O, y{:}O \vdash A(x,y) \quad \text{type} \end{cases}$$

$$\text{Function symbols:} \quad \begin{cases} x{:}O \vdash \mathrm{i}(x) : A(x,x) \\ x,y,z{:}O, f{:}A(x,y), g{:}A(y,z) \vdash \mathrm{c}(g,f) : A(x,z) \end{cases}$$

$$\text{Axioms:} \quad \begin{cases} x,y{:}O, f{:}A(x,y) \vdash \mathrm{c}(\mathrm{i}(y),f) = f : A(x,y) \\ x,y{:}O, f{:}A(x,y) \vdash \mathrm{c}(f,\mathrm{i}(x)) = f : A(x,y) \\ x,y,z,w{:}O, f{:}A(x,y), g{:}A(y,z), h{:}A(z,w) \\ \qquad\qquad \vdash \mathrm{c}(h,\mathrm{c}(g,f)) = \mathrm{c}(\mathrm{c}(h,g),f) : A(x,w) \end{cases}$$

The GAT of categories.

The GAT of categories

Many algebraic structures are described by GATs, e.g. **categories**.

$$\text{Types:} \quad \begin{cases} \vdash O \quad \text{type} \\ x{:}O, y{:}O \vdash A(x,y) \quad \text{type} \end{cases}$$

$$\text{Function symbols:} \quad \begin{cases} x{:}O \vdash \mathrm{i}(x) : A(x,x) \\ x,y,z{:}O, f{:}A(x,y), g{:}A(y,z) \vdash \mathrm{c}(g,f) : A(x,z) \end{cases}$$

$$\text{Axioms:} \quad \begin{cases} x,y{:}O, f{:}A(x,y) \vdash \mathrm{c}(\mathrm{i}(y),f) = f : A(x,y) \\ x,y{:}O, f{:}A(x,y) \vdash \mathrm{c}(f,\mathrm{i}(x)) = f : A(x,y) \\ x,y,z,w{:}O, f{:}A(x,y), g{:}A(y,z), h{:}A(z,w) \\ \qquad\qquad \vdash \mathrm{c}(h,\mathrm{c}(g,f)) = \mathrm{c}(\mathrm{c}(h,g),f) : A(x,w) \end{cases}$$

The GAT of categories.

(Also GATs of n-categories, ω-categories, coloured operads, semisimplicial sets, simplicial sets, opetopic sets, ...)

Models of GATs in sets

Every GAT has a l.f.p. 1-category of its Set-models.

The category of Set-models of the GAT of categories is the 1-category Cat of (small) categories.

The only notion of equivalence of Set-models of the GAT of categories is **isomorphism** of categories.

But the correct notion of equivalence should be **equivalence** of categories.

Models of GATs in types

There isn't a good definition of models in HoTT of an arbitrary GAT (that's the goal of this talk!).

Models of GATs in types

There isn't a good definition of models in HoTT of an arbitrary GAT (that's the goal of this talk!).

Defining a model of the GAT of categories in HoTT is not automatic.

A $(1$-)**category** C in HoTT is given by:

1. a type C_O (of objects) and a type family (of arrows) $C : C_O \times C_O \to \mathsf{U}$,

Models of GATs in types

There isn't a good definition of models in HoTT of an arbitrary GAT (that's the goal of this talk!).

Defining a model of the GAT of categories in HoTT is not automatic.

A (1-)**category** C in HoTT is given by:

1. a type C_O (of objects) and a type family (of arrows) $C\colon C_O \times C_O \to \mathsf{U}$,
2. operations for identity arrows and composition of arrows:

$$1_{(-)} : \prod_{a : C_O} C(a,a) \qquad - \circ - : \prod_{a,b,c : C_O} C(b,c) \to C(a,b) \to C(a,c),$$

Models of GATs in types

There isn't a good definition of models in HoTT of an arbitrary GAT (that's the goal of this talk!).

Defining a model of the GAT of categories in HoTT is not automatic.
A (1-)**category** C in HoTT is given by:

1. a type C_O (of objects) and a type family (of arrows) $C\colon C_O \times C_O \to \mathsf{U}$,
2. operations for identity arrows and composition of arrows:

$$1_{(-)} : \prod_{a:C_O} C(a,a) \qquad - \circ - : \prod_{a,b,c:C_O} C(b,c) \to C(a,b) \to C(a,c),$$

3. for every $a, b{:}C_O$ and $f{:}C(a,b)$, paths $1_b \circ f = f$ and $f \circ 1_a = f$,

Models of GATs in types

There isn't a good definition of models in HoTT of an arbitrary GAT (that's the goal of this talk!).

Defining a model of the GAT of categories in HoTT is not automatic.
A (1-)**category** C in HoTT is given by:

1. a type C_O (of objects) and a type family (of arrows) $C : C_O \times C_O \to \mathsf{U}$,
2. operations for identity arrows and composition of arrows:

$$1_{(-)} : \prod_{a:C_O} C(a,a) \qquad - \circ - : \prod_{a,b,c:C_O} C(b,c) \to C(a,b) \to C(a,c),$$

3. for every $a,b{:}C_O$ and $f{:}C(a,b)$, paths $1_b \circ f = f$ and $f \circ 1_a = f$,
4. for every $a,b,c,d{:}C_O$, $f{:}C(a,b)$, $g{:}C(b,c)$ and $h{:}C(c,d)$, a path $h \circ (g \circ f) = (h \circ g) \circ f$,

Univalence principle for categories

5. such that for every $a, b{:}C_O$, we have $\mathsf{isSet}(C(a,b))$, and

Univalence principle for categories

5. such that for every $a, b{:}C_O$, we have $\mathsf{isSet}(C(a,b))$, and
6. such that for every $a, b{:}C_O$, the map $\mathsf{idtoiso}_{a,b} \colon (a = b) \to (a \cong b)$ taking paths to invertible arrows is an equivalence ("Rezk completeness").

Univalence principle for categories

5. such that for every $a, b{:}C_O$, we have $\mathsf{isSet}(C(a,b))$, and
6. such that for every $a, b{:}C_O$, the map $\mathsf{idtoiso}_{a,b} \colon (a = b) \to (a \cong b)$ taking paths to invertible arrows is an equivalence ("Rezk completeness").

In HoTT, the type Cat of categories satisfies a **univalence principle**: for any $C, D{:}\mathsf{Cat}$, the function $(C =_{\mathsf{Cat}} D) \to (C \simeq D)$ is an equivalence ($C \simeq D$ is the type of equivalences of categories between C and D).

Univalence principle for categories

5. such that for every $a, b{:}C_O$, we have isSet$(C(a,b))$, and
6. such that for every $a, b{:}C_O$, the map $\mathsf{idtoiso}_{a,b}\colon (a = b) \to (a \cong b)$ taking paths to invertible arrows is an equivalence ("Rezk completeness").

In HoTT, the type Cat of categories satisfies a **univalence principle**: for any $C, D{:}\mathsf{Cat}$, the function $(C =_{\mathsf{Cat}} D) \to (C \simeq D)$ is an equivalence ($C \simeq D$ is the type of equivalences of categories between C and D).

Conditions (5) and (6) are ad hoc, but are needed for important "facts" about categories, like "a fully faithful and (merely) essentially surjective functor is an equivalence".

Univalence for models of GATs in types

Problem

Given a GAT $\mathbb{T}$, determine

1. the type $\mathsf{Mod}_{\mathbb{T}}$ of $\mathbb{T}$-models in HoTT,
2. for any $A, B{:}\mathsf{Mod}_{\mathbb{T}}$, the type $A \simeq B$ of equivalences of $\mathbb{T}$-models,
3. so that $\mathsf{Mod}_{\mathbb{T}}$ satisfies the univalence principle: for any $A, B{:}\mathsf{Mod}_{\mathbb{T}}$, the map $(A = B) \to (A \simeq B)$ is an equivalence.

Obviously, nothing silly like $(A \simeq B) \stackrel{\mathsf{def}}{=} (A = B)$.

FOLDS theories and FOLDS-equivalence

Makkai's **first-order logic with dependent sorts** (FOLDS) [Mak95] is a language designed specifically to obtain a weaker notion of equivalence than isomorphism of Set-models of a FOLDS theory.

A FOLDS theory consists of:

- a type signature L of type declarations, some of which are *propositions*:

$$\Gamma \vdash A \text{ type} \qquad \Gamma \vdash R \text{ prop}$$

- and (first-order) axioms over L:

$$\Gamma \vdash P \qquad (P \text{ is some first-order sentence over } \Gamma)$$

FOLDS is purely *relational* — a dependently typed operation must be encoded as a functional relation.

The FOLDS theory of categories

Sorts: $\begin{cases} \vdash O \quad \text{type} \\ x{:}O, y{:}O \vdash A(x,y) \quad \text{type} \\ \\ \end{cases}$

The FOLDS theory of categories

$$\text{Sorts:} \quad \begin{cases} \vdash O \quad \text{type} \\ x{:}O, y{:}O \vdash A(x,y) \quad \text{type} \\ x{:}O, f{:}A(x,x) \vdash I(f) \ \text{prop.} \\ x,y,z{:}O, f{:}A(x,y), g{:}A(y,z), h{:}A(x,z) \vdash T(f,g,h) \quad \text{prop.} \end{cases}$$

The FOLDS theory of categories.

The FOLDS theory of categories

$$\text{Sorts:} \quad \begin{cases} \vdash O \quad \text{type} \\ x{:}O, y{:}O \vdash A(x,y) \quad \text{type} \\ x{:}O, f{:}A(x,x) \vdash I(f) \ \text{prop.} \\ x,y,z{:}O, f{:}A(x,y), g{:}A(y,z), h{:}A(x,z) \vdash T(f,g,h) \quad \text{prop.} \end{cases}$$

$$\text{Axioms:} \quad \begin{cases} x{:}O \vdash \exists! i{:}A(x,x).I(i) \\ x,y,z{:}O, f{:}A(x,y), g{:}A(y,z) \vdash \exists! h{:}A(x,z).T(f,g,h) \\ \\ \\ \\ \end{cases}$$

The FOLDS theory of categories.

The FOLDS theory of categories

$$\text{Sorts:} \quad \begin{cases} \vdash O \quad \text{type} \\ x{:}O, y{:}O \vdash A(x,y) \quad \text{type} \\ x{:}O, f{:}A(x,x) \vdash I(f) \ \text{prop.} \\ x,y,z{:}O, f{:}A(x,y), g{:}A(y,z), h{:}A(x,z) \vdash T(f,g,h) \quad \text{prop.} \end{cases}$$

$$\text{Axioms:} \quad \begin{cases} x{:}O \vdash \exists! i{:}A(x,x).I(i) \\ x,y,z{:}O, f{:}A(x,y), g{:}A(y,z) \vdash \exists! h{:}A(x,z).T(f,g,h) \\ x,y{:}O, i{:}A(x,x), f{:}A(x,y), I(i) \vdash T(i,f,f) \\ x,y{:}O, i{:}A(y,y), f{:}A(x,y), I(i) \vdash T(f,i,f) \\ x,y,z,w{:}O, f{:}A(x,y), g{:}A(y,z), h{:}A(z,w), j{:}A(x,z), k{:}A(y,w), l{:}A(x,w), \\ \qquad T(f,g,j), T(g,h,k), T(j,h,l) \vdash T(f,k,l) \end{cases}$$

The FOLDS theory of categories.

FOLDS-equivalence

FOLDS type signatures correspond to *direct* categories whose slice categories are finite. (A direct category is a small category D equipped with an identity-reflecting functor $D \to \omega$.)

Given a type signature L, an L-**structure** is a presheaf $X \colon L^{op} \to \mathrm{Set}$ such that for every ($\Gamma \vdash R$ prop) in L, the function $X_R \to \widehat{L}(\Gamma, X)$ is injective.

Makkai defines a notion of FOLDS-equivalence between L-structures.

Models of FOLDS theories in sets

Given a FOLDS theory $\mathbf{T}$ over L, its category of Set-models is the full subcategory of $\widehat{L}$ on the L-structures that satisfy the axioms of $\mathbf{T}$.

The category of models of the FOLDS theory of categories is the 1-category Cat.

FOLDS-equivalence between models of the FOLDS theory of categories coincides with equivalence of categories.

Models of FOLDS theories in types

For a (suitable) FOLDS theory $\mathbf{T}$ over a type signature L, there are definitions in type theory [ANST21] of L-structures, **univalent** L-structures, FOLDS-equivalence, and (univalent) $\mathbf{T}$-models, such that:

Theorem ([ANST21])

The type $\mathsf{Mod}_{\mathbf{T}}$ *of univalent* $\mathbf{T}$*-models satisfies a univalence principle: for any* $A, B{:}\mathsf{Mod}_{\mathbf{T}}$*, the map* $(A = B) \to (A \simeq B)$ *is an equivalence (where* $A \simeq B$ *is the type of FOLDS-equivalences between* A *and* B*).*

Drawback: FOLDS is purely relational, but the syntax of HoTT allows for dependently typed operations/function symbols.

"Can the theory of univalence be extended from our functorial signatures to a wider class of Generalized Algebraic Theories? In particular, can we deal directly with theories that include functions, perhaps by finding a uniform way to encode their graphs as relations? (We thank Steve Awodey for raising this question to us.)"

— [ANST21]

Presentations of GATs

Let $\langle\!\langle \Gamma_0 \rangle\!\rangle$ be the empty (initial) GAT.

For all $n > 0$,

- let $\langle\!\langle \Gamma_n \rangle\!\rangle$ be the GAT generated by the type declarations

$$\begin{aligned} &\vdash A_1 \text{ type} \\ &\dots \\ x_1{:}A_1, \dots, x_{n-1}{:}A_{n-1}(x_1, \dots, x_{n-2}) &\vdash A_n(x_1, \dots, x_{n-1}) \text{ type,} \end{aligned}$$

- let $\langle\!\langle \Gamma_{n-1} \vdash a : A_n \rangle\!\rangle$ be $\langle\!\langle \Gamma_n \rangle\!\rangle$ extended with a function symbol

$$x_1{:}A_1, \dots, x_{n-1}{:}A_{n-1}(x_1, \dots, x_{n-2}) \vdash a : A_n(x_1, \dots, x_{n-1})$$

- and let $\langle\!\langle \Gamma_{n-1} \vdash a, b : A_n \rangle\!\rangle$ be $\langle\!\langle \Gamma_n \rangle\!\rangle$ extended with two function symbols a and b as above.

Let G be the set of the following morphisms of GATs:

- The inclusion $\langle\!\langle \Gamma_n \rangle\!\rangle \longrightarrow \langle\!\langle \Gamma_{n+1} \rangle\!\rangle$.
- The inclusion $\langle\!\langle \Gamma_n \rangle\!\rangle \longrightarrow \langle\!\langle \Gamma_{n-1} \vdash a : A_n \rangle\!\rangle$.
- The map $\langle\!\langle \Gamma_{n-1} \vdash a, b : A \rangle\!\rangle \longrightarrow \langle\!\langle \Gamma_{n-1} \vdash a : A \rangle\!\rangle$ equalising a and b.

A **presentation** of a GAT $\mathbb{T}$ is a G-cell complex (a G-tree) in the category of GATs, whose composite is $\langle\!\langle \Gamma_0 \rangle\!\rangle \to \mathbb{T}$.

Main result

Theorem (LS–Shulman)

Given any presentation of a GAT $\mathbb{T}$, there is a cell complex (a tree) of direct categories and sieve inclusions, such that

- *the composite direct category is a FOLDS type signature L,*
- *there is a FOLDS theory $\mathbf{T}$ over L whose category of Set-models is equivalent to the category of Set-models of $\mathbb{T}$.*

Remark: When applied to the GAT of categories, the previous construction results in the FOLDS theory of categories.

Corollary

In combination with the results of [ANST21], this provides a univalent type of models in HoTT of a suitably presented GAT.

Benedikt Ahrens, Paige Randall North, Michael Shulman, and Dimitris Tsementzis.
The univalence principle.
arXiv preprint arXiv:2102.06275, 2021.

J. W. Cartmell.
Generalised algebraic theories and contextual categories.
PhD thesis, University of Oxford, 1978.

Michael Makkai.
First Order Logic with Dependent Sorts, with Applications to Category Theory.
Preprint, 1995.
math.mcgill.ca/makkai/folds/foldsinpdf/FOLDS.pdf.