Categorical structures in type theory, in (univalent) type theory

B. Ahrens
about j. w. w. P. LeF. Lumsdaine, V. Voevodsky

2016–05–20

Goal

Compare various categorical structures arising in study of type theory:

- categories with families
- (split) type categories
- categories with display maps

Remarks:

- In general, we will compare structures on a fixed base category.
  E.g., we consider $\mathsf{FamStr}(\mathcal{C})$ the family structures on fixed category $\mathcal{C}$, etc.

What are those categorical structures?

- gadgets are sometimes used to interpret syntax
- additional structure on those gadgets is often considered (Pi, Id,...)

- This talk is not about syntax nor about interpretation of syntax.
- In this talk additional structure is not considered.

For all this, see Lumsdaine's talks.

How to compare?

- in set theory and traditional ML type theory, comparison on the level of categories—i.e., up to **isomorphism**
- in **univalent** type theory, can meaningfully compare on level of objects—i.e., up to **identity**

Refined goal: compare in (univalent) type theory and formalize in `UniMath`.

The type theory we work in—UniMath

- Martin-Löf type theory
- identity types
- sigma types
- dependent function types
- natural numbers, booleans
- disjoint sum types (could be replaced by sigma)
- propositional truncation (via impredicative encoding)
- $\mathcal{U} : \mathcal{U}$
- univalence axiom for $\mathcal{U}$

For the purpose of formalization, the fragment of the Coq proof assistant that constitutes the `UniMath` language

Outline

1. Review of categories in univalent type theory

2. Categorical structures in type theory
   - Families and split type structures
   - Families and Rezk completion

Outline

Categories in univalent mathematics

A category is

- a type $O : \mathcal{U}$ of objects
- a dependent type $A : O \times O \to \mathcal{S}et$ of arrows
- $\mathsf{id} : \prod_{(a:O)} A(a,a)$
- $(\circ) : \prod_{(a,b,c:O)} A(a,b) \times A(b,c) \to A(a,c)$
- axioms postulating identities of arrows

Categories in univalent mathematics

A **univalent** category is

- a type $O : \mathcal{U}$ of objects
- a dependent type $A : O \times O \to \mathcal{S}et$ of arrows
- $\mathsf{id} : \prod_{(a:O)} A(a,a)$
- $(\circ) : \prod_{(a,b,c:O)} A(a,b) \times A(b,c) \to A(a,c)$
- axioms postulating identities of arrows
- such that the natural map

$$\mathsf{idtoiso} : \prod_{a,b:O} (a = b) \to \mathsf{iso}(a,b)$$

is an equivalence for any a, b

Sameness for univalent categories

$$\mathcal{C} = \mathcal{D}$$
$$\mathcal{C} \cong \mathcal{D}$$
$$\mathcal{C} \simeq \mathcal{D}$$

Theorem

For ***univalent*** *categories* $\mathcal{C}$ *and* $\mathcal{D}$*, these are equivalent as types.*

Consequence

Every property of univalent categories definable in UF is invariant under equivalence.

Examples of univalent categories

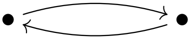
- $\mathcal{S}et$ (discrete types)
- Groups, rings, ... (Structure Identity Principle)
- Functor category $[\mathcal{C}, \mathcal{D}]$, if $\mathcal{D}$ is univalent
- Full subcategories of univalent categories

More examples of univalent categories

- A preorder is univalent iff it is antisymmetric
- If X is of h-level 3, i.e., 1-truncated, then there is a univalent category with X as objects and $\hom(x, y) := (x = y)$
- If $\mathcal{C}$ is univalent, then the category of cones of shape $F : \mathcal{J} \to \mathcal{C}$ is
  - $\rightsquigarrow$ limits (limiting cones) in a univalent category are unique **up to identity**

Non-univalent categories

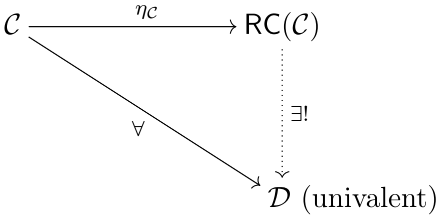
- Any "chaotic" category $\mathcal{C}$ with $\mathcal{C}(x, y) := 1$, for $\mathcal{C}_0$ not a prop



- Any chaotic category $\mathcal{C}$ with an object $c : \mathcal{C}_0$ is **equivalent** to the terminal category $\mathbf{1}$
  - $\rightsquigarrow$ a category can be equivalent to a univalent one without being univalent itself

Rezk completion

To any category $\mathcal{C}$, associate a univalent one, its "Rezk completion",



Intuitively, obtain $\mathsf{RC}(\mathcal{C})_0$ by adding to $\mathcal{C}_0$ as many identities as needed

Construction of the Rezk completion

- $\mathsf{RC}(\mathcal{C})$ is the full image subcategory of the Yoneda embedding $\mathcal{C} \to [\mathcal{C}^{\mathrm{op}}, \mathcal{S}et]$
- $\eta_{\mathcal{C}} : \mathcal{C} \to \mathsf{RC}(\mathcal{C})$ is fully faithful and essentially surjective
- precomposition with a ff. and es. functor is ff. and es.
- a ff. and es. functor is an equivalence if source category is univalent
- the object map of an equivalence of univalent categories is an equivalence of types

About the Rezk completion

- Name inspired by
  Charles Rezk, *A model for the homotopy theory of homotopy theory*
  categories $\sim$ truncated Segal spaces
  univalent categories $\sim$ complete truncated Segal spaces
  in Voevodsky's simplicial set model

- More details in Ahrens, Kapulkin, Shulman, *Univalent categories and the Rezk completion*
  `http://arxiv.org/abs/1303.0584`

Outline

Rezk completion and categorical structures of type theory

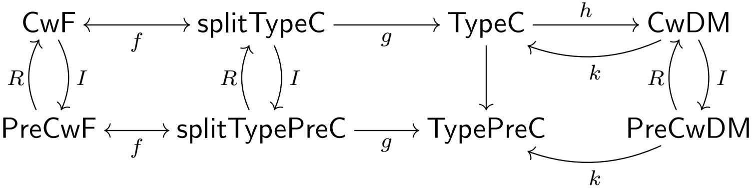
Questions

1. Does a given structure on $\mathcal{C}$, e.g., a CwF-structure, induce a structure on $\mathsf{RC}(\mathcal{C})$?
2. When can we construct a map

$$s_1(\mathcal{C}) \to s_2(\mathcal{C})$$

turning one kind of structure into another kind?
[Answer: may depend on $\mathcal{C}$ being univalent!]

Goal



- These are maps of types
- f, g, h, k do not change the base category
- g is injective (forgets splitness)
- f is an equivalence

Categories with Families

A **families structure** on a category $\mathcal{C}$ is

- presheaves $\mathsf{Ty}, \mathsf{Tm} : \mathcal{C}^{\mathrm{op}} \to \mathsf{Set}$;
- a natural transformation $\mathsf{p} : \mathsf{Tm} \to \mathsf{Ty}$;
- for any $\Gamma : \mathcal{C}$ and $A : \mathsf{Ty}(\Gamma)$, an object $\Gamma.A$ and a map $\pi_A : \mathcal{C}(\Gamma.A, \Gamma)$
- for any $\Gamma : \mathcal{C}$ and $A : \mathsf{Ty}(\Gamma)$, a map Q_A making a pullback square

$$\begin{array}{ccc}
\mathsf{Yo}(\Gamma.A) & \xrightarrow{\mathsf{Q}_A} & \mathsf{Tm} \\
{\scriptstyle \mathsf{Yo}(\pi_A)}\downarrow & & \downarrow{\scriptstyle \mathsf{p}} \\
\mathsf{Yo}(\Gamma) & \xrightarrow[\mathsf{v}(A)]{} & \mathsf{Ty}
\end{array}$$

Type of such structures is called $\mathsf{FamStr}(\mathcal{C})$

Split type categories

A **split type structure** on a category $\mathcal{C}$ is

- a presheaf $\mathsf{Ty} : \mathcal{C}^{\mathrm{op}} \to \mathsf{Set}$
- for any $\Gamma : \mathcal{C}_0$ and $A : \mathsf{Ty}(\Gamma)$, an object $\Gamma.A$ and $\pi_A : \mathcal{C}(\Gamma.A, \Gamma)$
- for any $\Gamma, \Gamma' : \mathcal{C}_0$ and $f : \mathcal{C}(\Gamma', \Gamma)$ and $A : \mathsf{Ty}(\Gamma)$, a morphism $\mathsf{q}_{(f,A)} : \mathcal{C}(\Gamma.A, \Gamma)$ such that

$$\begin{array}{ccc}
\Gamma'.A[f] & \xrightarrow{\mathsf{q}_{(f,A)}} & \Gamma.A \\
{\scriptstyle \pi_{A[f]}}\downarrow & & \downarrow{\scriptstyle \pi_A} \\
\Gamma' & \xrightarrow[f]{} & \Gamma
\end{array}$$

  is a pullback
- axioms about functoriality of q

Split type categories

"Functoriality" of q:

- $\mathsf{q}_{(1,A)} = \mathsf{idtoiso}(\mathsf{ap}_{x \mapsto \Gamma.x}(p))$
- $\mathsf{q}_{(f' \cdot f, A)} = \mathsf{idtoiso}(\mathsf{ap}_{x \mapsto \Gamma''.x}(p')) \cdot \mathsf{q}_{(f', A[f])} \cdot \mathsf{q}_{(f,A)}$

where $p : A[1] = A$ and $p' : A[f' \cdot f] = A[f][f']$

- Type of such structures is called $\mathsf{sTypeStr}(\mathcal{C})$
- Called CwA in Lumsdaine's talk

Constructions

Construction 1

An equivalence

$$\mathsf{sTypeStr}(\mathcal{C}) \simeq \mathsf{FamStr}(\mathcal{C})$$

Here, $\simeq$ is equivalence of both types and categories

Construction 2

A function

$$\mathsf{FamStr}(\mathcal{C}) \to \mathsf{FamStr}(\mathsf{RC}(\mathcal{C}))$$

Outline

1. Review of categories in univalent type theory

2. Categorical structures in type theory
   - Families and split type structures
   - Families and Rezk completion

Goal

To construct an equivalence of types

$$\mathsf{FamStr}(\mathcal{C}) \simeq \mathsf{sTypeStr}(\mathcal{C})$$

Sketch of the equivalence

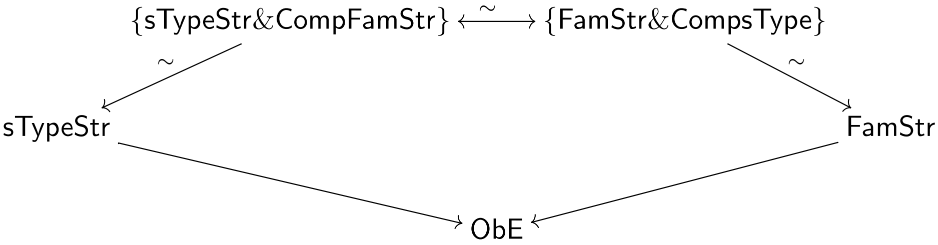
$$\begin{array}{ccc}
\{\mathsf{sTypeStr}\&\mathsf{CompFamStr}\} & \stackrel{\sim}{\longleftrightarrow} & \{\mathsf{FamStr}\&\mathsf{CompsTypeStr}\} \\
\sim\downarrow & & \downarrow\sim \\
\mathsf{sTypeStr} & & \mathsf{FamStr}
\end{array}$$

- vertical maps are first projections—we show that fibers are contractible
- horizontal map is rearranging of components
- need univalence to show that left projection is equivalence
- need function ext. for right projection being an equivalence

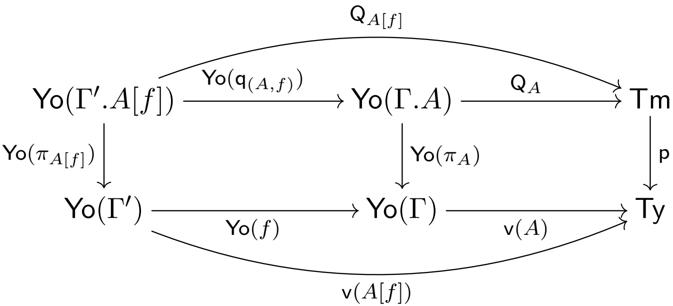
Some details

Fiberwise over common elements of $\mathsf{FamStr}(\mathcal{C})$ and $\mathsf{sTypeStr}(\mathcal{C})$:
An **object extension structure** on a category $\mathcal{C}$ is given by

- a functor $\mathsf{Ty} : \mathcal{C}^{\mathrm{op}} \to \mathsf{Set}$
- for any $\Gamma : \mathcal{C}_0$ and $A : \mathsf{Ty}(\Gamma)$, an object $\Gamma.A$ a morphism $\pi_A : \mathcal{C}(\Gamma.A, \Gamma)$



Compatibility between families and split type structure



Compatibility is commutativity of upper triangle

$$\mathsf{Q}_{A[f]} = \mathsf{Yo}(\mathsf{q}(A, f)) \cdot \mathsf{Q}_A$$

Categorical equivalence

Also construct a categorical equivalence

$$\mathsf{FamStr}(\mathcal{C}) \simeq \mathsf{sTypeStr}(\mathcal{C})$$

- does not require univalence axiom
- makes use of notion of **displayed categories** to work fiberwise over a fixed category of object extension structures

Type-theoretic and categorical equivalence

Lemma

If $F : \mathcal{C} \simeq \mathcal{D}$ is an equivalence and $\mathcal{C}$ and $\mathcal{D}$ are univalent, then $F_0 : \mathcal{C}_0 \simeq \mathcal{D}_0$.

- recover equivalence of types $\mathsf{FamStr}(\mathcal{C}) \simeq \mathsf{sTypeStr}(\mathcal{C})$ by showing that $\mathsf{FamStr}(\mathcal{C})$ and $\mathsf{sTypeStr}(\mathcal{C})$ are univalent categories
- proving that categories $\mathsf{FamStr}(\mathcal{C})$ and $\mathsf{sTypeStr}(\mathcal{C})$ are univalent requires univalence axiom

Outline

1. Review of categories in univalent type theory

2. Categorical structures in type theory
   - Families and split type structures
   - Families and Rezk completion

Goal

To construct a function

$$\mathsf{FamStr}(\mathcal{C}) \to \mathsf{FamStr}(\mathsf{RC}(\mathcal{C}))$$

- can be done directly via universal property of $\mathsf{RC}(\mathcal{C})$
- is cumbersome

Generalize ...

Universe relative to a functor

A **universe structure** relative to a functor $J : \mathcal{C} \to \mathcal{D}$ is given by

- objects U and $\tilde{\mathsf{U}}$ in $\mathcal{D}$ and a morphism $\mathsf{p} : \tilde{\mathsf{U}} \to \mathsf{U}$;
- for any object X in $\mathcal{C}$ and any morphism $f : J(X) \to \mathsf{U}$,
  - an object X_f in $\mathcal{C}$
  - a morphism $\mathsf{p}_f : X_f \to X$
  - a morphism $\mathsf{q}_f : J(X_f) \to \tilde{\mathsf{U}}$ such that

$$\begin{array}{ccc} J(X_f) & \xrightarrow{\mathsf{q}_f} & \tilde{\mathsf{U}} \\ {\scriptstyle J(\mathsf{p}_f)}\downarrow & & \downarrow{\scriptstyle \mathsf{p}} \\ J(X) & \xrightarrow[f]{} & \mathsf{U} \end{array}$$

  is a pullback square.

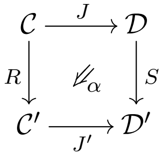
Relative universe on Yoneda, identity functors

- Denote $\mathsf{Yo} : \mathcal{C} \to [\mathcal{C}^{\mathrm{op}}, \mathcal{S}et]$ the Yoneda functor. Then

$$\mathsf{RelUniv}(\mathsf{Yo}) \simeq \mathsf{FamStr}(\mathcal{C})$$

- $J = \mathrm{Id}(\mathcal{C}) \quad \rightsquigarrow \quad$ universe category (Voevodsky, *A C-system defined by a universe category*)

Transfer of relative universe structures

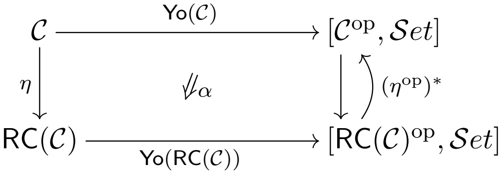


- $\mathcal{C}', \mathcal{D}'$ are univalent
- R is essentially surjective,
- S is fully faithful and preserves pullback squares
- J' is fully faithful
- α is iso

Construction

$$\mathsf{RelUniv}(J) \to \mathsf{RelUniv}(J')$$

Rezk completion and families



Construction

$$\mathsf{FamStr}(\mathcal{C}) \simeq \mathsf{RelUniv}(\mathsf{Yo}(\mathcal{C})) \to \mathsf{RelUniv}(\mathsf{Yo}(\mathsf{RC}(\mathcal{C}))) \simeq \mathsf{FamStr}(\mathsf{RC}(\mathcal{C}))$$

Some remarks

Formalization:

- ca. 4000loc, some of it general purpose stuff
- not public yet

Future work:

- universal property of completion of a category with families
- connection to C-systems (formalized by Voevodsky)
- consider additional structure (Pi, Sigma, Id,...)

Some remarks

Formalization:

- ca. 4000loc, some of it general purpose stuff
- not public yet

Future work:

- universal property of completion of a category with families
- connection to C-systems (formalized by Voevodsky)
- consider additional structure (Pi, Sigma, Id,...)

Thanks for your attention.

References

- First occurrence of univalent categories: Hofmann & Streicher, *The groupoid interpretation of type theory*
- Structure Identity Principle: Coquand & Danielsson, *Isomorphism is equality*
- (Complete) segal spaces: Rezk, *A model for the homotopy theory of homotopy theory*
- Categories with families: Fiore, *Discrete Generalised Polynomial Functors*, Talk at ICALP 2012 (Appendix) `http://www.cl.cam.ac.uk/~mpf23/talks/ICALP2012.pdf`
- (Split) type categories: van den Berg & Garner, *Topological and simplicial models*
  also Pitts, *Categorical Logic*
- Universe categories: Voevodsky, *A C-system defined by a universe category*