The Logic of ∞-Category Theory

Denis-Charles Cisinski

Universität Regensburg

Greifswald, May 15, 2024

Aim of Logic

Logic serves at least two purposes in Mathematics

1. foundations for mathematics
2. subfield of mathematics, with its own methods and applications (e.g. model theory)

Logic has a rather special flavour:

very abstract and conceptual as well as extremely close to practice (Computer Science)

Foundations of Mathematics

- at least include Cantor's set theory (naive set theory)
- possibly include ∞-category theory (Grothendieck, Joyal, Simpson, Rezk, Lurie, Toën, Vezzosi, Clausen, Scholze...)
- ideally allow an effective and complete formalization of all Mathematics

Claim:

∞-categories are not sophisticated constructions but a **primitive concept**

Possible applications of formalization of ∞-category theory

- teaching – we can teach modern techniques in derived algebraic geometry and algebraic topology rigorously without explaining how to construct ∞-categories within set theory; we only have to explain how to reason with them
- a formalized proof is more robust: it can be checked by a computer (e.g. who can claim to have checked the proof of the Bloch-Kato conjecture by Rost and Voevodsky in full?)
- a formalized proof can have new semantic interpretation, hence give new results – if we can formalize non-trivial theorems of modern algebraic geometry (e.g. the proof of the Bloch-Kato conjecture), there is a good chance that we will get new interesting results
- if ∞-category is a primitive concept in the logic underlying the formalization process, all the constructions and results using ∞-category theory will be much easier to formalize; e.g. most of modern (derived) algebraic geometry and a significant part of logic itself!
- if ∞-category is a primitive concept, then many logics becomes primitives: we have a logic of logic that has all the expressive power of modern ∞-category theory; this will change the face of logic and of computer science.

ZFC

Need: formal language

Set Theory of Zermelo and Fraenkel (1908–1922, von Neumann's axiom of foundation 1925)

Uses very few symbols: $=, \in$

Very efficient: formalizes all Mathematics, avoids paradoxes

Type Theory

Alternative, mainly introduced by Bertrand Russell:

Type Theory

- hygiene to avoid paradoxes (1903, Principles of Mathematics)
- building principle (1908, Mathematical Logic as Based on the Theory of Types)

Monumental work together with Alfred North Whitehead: Principia Mathematica (1910, 1912, 1913)

λ-calculus

Type Theory has never been popular among mathematicians, but for logicians and computer scientists, this is another story.

Alonzo Church revisited Type Theory in several ways

- simplify Russell's axiomatic (1932–1933, A Set of Postulates for the Foundations of Logic),
- introduce λ-calculus, a kind of type theory, yielding the modern notion of **calculable** function; Church's 1941 monograph, The Calculi of λ-conversion, was instrumental in the development of semantics for programming languages. λ-calculus is a major topic in theoretical Computer Science.

Dependent Type Theory

John Cartmell introduced Generalized Algebraic Theories (1978–1986), a very general notion of **dependent type theories**, and their associated **contextual categories**

This means the following:

a dependent type theory essentially is the language of a given category (and choices of pullbacks)

This is a very dynamical moment for type theory in logic and computer sciences, with many fundamental contributions of Curry, Howard, de Bruijn, Tait, Scott, Girard.

During the same period (1972–1990), the swedish philosopher Per Martin-Löf also studied dependent type theory and made several deep contributions to the theory, in particular, concerning the notion of **identity**.

Declarations

In a dependent type theory there are

contexts Γ, Δ,...

types A, B, C,... X, Y, Z,...

terms a, b, c,... x, y, z,...

Judgements

The principle of type theory is that we want to talk about terms, and terms are always given in a given type. E.g.

$a\colon A$ means "a is a term of type A"

There are **judgements**. They can be of three kinds

1. $\Gamma \vdash A$ Type means "in context Γ, there is a type A"
2. $\Gamma \vdash x\colon A$ means "in context Γ, there is the term x of type A"
3. $\Gamma \vdash x \equiv_A y$ means "in context Γ, there are terms x and y of type A and they are definitionally equal"

Basic constructors

Examples:

- terms of $A \times B$ are pairs (a, b) with $a\colon A$ and $b\colon B$
- there is a type $A \to B$ whose terms
$$f\colon A \to B$$
are rules assigning a term $f(a)\colon B$ for each term $a\colon A$
- there is an identity
$$\mathrm{Id}_A\colon A \to A$$
assigning the term $a\colon A$ to each $a\colon A$
- there is a composition law
$$(A \to B) \times (B \to C) \to (A \to C) \quad , \qquad (f, g) \mapsto g \circ f$$
where $g \circ f$ assigns to each term $a\colon A$ the term $g(f(a))\colon C$

Curry-Howard Correspondence

The Curry-Howard correspondence is the principle that

propositions are nothing else than types

We say that "A is true" if there exists a term $a\colon A$. In other words:

terms are proofs (in Computer Science: programs)

For instance $A \to B$ is the statement that B follows from A: indeed, if there is a term

$$f\colon A \to B$$

then, for each term $a\colon A$, there is a term $f(a)\colon B$

Identity Type, after Martin-Löf

In type theory, there is a way to avoid the notion of identity to be taken for granted: we can build the theory so that it remains as **agnostic** as possible when it comes to determine what that means to say that two terms are the same.
Given a type A, for each terms a and b in A, there is

$$a = b \text{ Type}$$

A term

$$\sigma : a = b$$

is thus a proof that a and b are equal.

Identity Type, after Martin-Löf

There are rules saying that

- there is a canonical proof of reflexivity

$$\mathrm{refl}_a \colon a = a$$

- there is a canonical proof of symmetry

$$\mathrm{sym} \colon (a = b) \to (b = a)$$

- there is a canonical proof of transitivity

$$\mathrm{trans} \colon (a = b) \times (b = c) \to (a = c)$$

Equivalences

An **equivalence** is a map

$$f\colon A \to B$$

such that there are maps

$$g\colon B \to A \qquad \text{and} \qquad h\colon B \to A$$

such that

$$f \circ g = \mathrm{Id}_B \qquad \text{and} \qquad h \circ f = \mathrm{Id}_A$$

are true.

There is a type "A is equivalent to B"

$$A \cong B$$

whose terms are tuples (f, g, h, σ, τ) with $f\colon A \to B$, both g and h terms of type $B \to A$, and $\sigma\colon f \circ g = \mathrm{Id}_B$ and $\tau\colon h \circ f = \mathrm{Id}_A$

Sets

A typical interpretation of types is to see types as sets. In this case, if a is a term of A, we think of them as elements of the set A, and we add the axiom that the type $a = a$ is equivalent to the point. Given a big enough regular cardinal α, let us write Set for the set of subsets of α of cardinal $< \alpha$.We define a set to be small if it is in bijection with a set that is an element of Set. We can now declare small sets as terms

$$X : \mathrm{Set}$$

Drawback: if X and Y are small sets, then we have two ways to identify them.
As types:

$X \cong Y$ whose terms are bijections from X to Y

As terms of Set:

$X = Y$ whose terms are proofs that X and Y have the same elements

Univalence

We can interpret types as “homotopy types of spaces”, more precisely, as Kan complexes – we shall call them spaces.

We can think of terms as points. For two terms a and b in a space X, the type $x = y$ is the space of paths from a to b in X. This validates Martin-Löf's rules on identity types. The associated notion of equivalence between types is the notion of **homotopy equivalence**.

During the years 2007–2009, Voevodsky introduced the univalence axiom: the existence of a type of small types U, allowing to interpret (small) types as terms of U, so that the following holds

(Voevodsky's Univalence Principle)

For any (small) types X and Y, we have a canonical equivalence

$$u\colon (X = Y) \cong (X \cong Y)$$

Homotopy Type Theory

In other words, it does not change anything to work with types or with terms of the universe. In order to get its full strength, this principle must hold true in any context. The fact that univalence principle holds in any context for the dependent type theory defined by the homotopy theory of Kan complexes is a theorem of Voevodsky (2012). This has been generalized by Mike Shulman (arXiv:1904.07004): any ∞-topos in the sense of Lurie has a model that validates the axioms of Martin-Löf dependent type theory as well as Voevodsky's univalence principle (Homotopy Type Theory).

Homotopy Type Theory is characterized by the following properties:

1. Types are terms of a universe
2. The rule of identification of types is given by the identity type of the universe, and there is no other way to compare types between each other

Structure vs Properties

One drawback left in homotopy type theory: the universe is a only a type, i.e. the mere collection of all types, as opposed to the **category of types**. The operation of associating $A \to B$ to each pair (A, B) is not build in the universe.

The nature of the universe does not reflect the logic of types by itself.

Remedy: think of types as categories rather than groupoids or homotopy types. This is the purpose of **Synthetic Category Theory**.

Main change: classically, when type theories are designed to axiomatize some form of theory of sets or of homotopy types, all universal truths are stated in every context by default. In synthetic category theory, universal truths will be stated in groupoidal contexts only (i.e. "objectwise").

Types as categories

From now on, we decide to call types **(synthetic) categories**. There are two features we should have for this to make sense.

- categories have objects
- categories have morphisms

New notations:

$$\mathrm{Fun}(A, B) = (A \to B)$$

Objects in a type

We introduce formally a new operator that associates to each type C its **core** $C^{\simeq}$ together with the rule:

$$\text{any term } c\colon C^{\simeq} \text{ determines a term } c\colon C$$

We call **objects of** C the terms of $C^{\simeq}$. We call **groupoids** or **anima** those types C such that the canonical map $\imath\colon C^{\simeq} \to C$ is an equivalence. We postulate that $C^{\simeq}$ always is a groupoid.

Observe that (with Δ^0 the terminal type):

$$C^{\simeq} \cong \mathrm{Fun}(\Delta^0, C)^{\simeq}$$

We have a notion of collection of functors:

$$\mathrm{Map}(A, B) = \mathrm{Fun}(A, B)^{\simeq}$$

Morphisms in a type

We introduce a constant type Δ^1 that represents the idea of morphisms: a **morphism in** C is an object of $\Delta^1 \to C$.

A morphism has a domain and a codomain. We thus postulate that there are two morphisms

$$0\colon \Delta^0 \to \Delta^1 \qquad \text{and} \qquad 1\colon \Delta^0 \to \Delta^1$$

Given a morphism $f\colon \Delta^1 \to C$, we denote by $x = f(0)$ and $y = f(1)$, we obtain

$$f\colon x \to y$$

in C. We write

$$C(x, y)$$

for the type of morphisms of C with domain x and codomain y.
Given $x \in C$ there is

$$\mathrm{id}_x\colon \Delta^1 \to C$$

defined as the composition of $!\colon \Delta^1 \to \Delta^0$ with $x\colon \Delta^0 \to C$.

Segal condition

We postulate that Δ^1 has enough structure to encode composition of morphisms. We assume given an object Δ^2 such that, for any $i, j \in \{0,1,2\}$ any non-decreasing map

$$f\colon \{0,\dots,i\} \to \{0,\dots,j\} \quad \text{induces} \quad f\colon \Delta^i \to \Delta^j$$

(together with suitable universal properties turning all this into a property of Δ^1).We assume:

Given sequence of terms of the form

$$f\colon x \to y \qquad g\colon y' \to z \qquad \sigma\colon y = y' \qquad \text{in a type } C$$

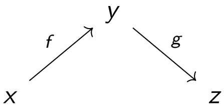
there is a unique sequence of terms

$$t\colon \Delta^2 \to C \qquad \tau_2\colon t \circ \delta_2 = f \qquad \tau_0\colon t \circ \delta_0 = g \quad \text{(together with a compatibility with } \sigma)$$

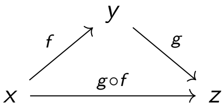
with $\delta_j\colon \Delta^1 \to \Delta^2$ the unique map whose image does not contain j

Composition law

This means that there is a canonical law that associates to each diagram in C of the form



a diagram in C of the form



Hence a composition law

$$- \circ - \colon C(y,z) \times C(x,y) \to C(x,z)$$

Homotopy Hypothesis

We can now say when a morphism $f\colon x \to y$ in a type C is an **isomorphism**: this simply means that there are morphisms

$$g\colon y \to x \qquad \text{and} \qquad h\colon y \to x$$

as well as proofs that

$$g \circ f = \mathrm{id}_x \qquad \text{and} \qquad f \circ h = \mathrm{id}_y$$

hold true.

A type is a groupoid if and only if all its morphisms are isomorphisms.

Univalence revisited

In category theory, the way to compare objects is through the notion of isomorphism. Therefore, we postulate:

For any objects x and y in a type C, there is a canonical equivalence

$$(x = y) \cong C^{\simeq}(x, y)$$

(here "canonical" means that it is completely determined by the property that it sends refl_x to id_x).

Isofibrations

An **isofibration** is a map $p\colon X \to B$ with the following property.

- there is a dependent type: a rule that assigns to each term $b\colon B$ a type $X(b)$ (in particular, if $b = c$ holds, so does $X(b) \cong X(c)$).
- a term $x\colon X(b)$ is the same thing as a term $x\colon X$ such that $p(x) = b$ holds

Example

the projection $p\colon A \times B \to A$ assigning $a\colon A$ to each pair $(a, b)\colon A \times B$ is an isofibration.

Example

We postulate that $(ev_0, ev_1)\colon \mathrm{Fun}(\Delta^1, C) \to C \times C$ is an isofibration.

Any map is a composition of an equivalence followed by an isofibration

cocartesian morphisms

Let $p\colon X \to B$ be an isofibration. Given a morphism

$$v\colon b_0 \to b_1 \qquad \text{in } B$$

a lift of v in X is a morphism

$$u\colon x_0 \to x_1 \qquad \text{in } X$$

such that $p(u) = v$ holds. We say that u is **locally p-cocartesian** if for any other lift of v of the form

$$u'\colon x_0 \to x_1' \qquad \text{in } X$$

there is a unique morphism $w\colon x_1 \to x_1'$ such that $p(w) = \mathrm{id}_{b_1}$ and $w \circ v = v'$ both hold.

cocartesian fibrations

Let $p\colon X \to B$ be an isofibration. We say that p is a **cocartesian fibration** if the following two conditions hold:

1. for any $v\colon b_0 \to b_1$ in B and any object x_0 in X such that $p(x_0) = b_0$ holds, there is a locally p-cocartesian lift of v of the form $u\colon x_0 \to x_1$ in X
2. locally p-cocartesian morphisms are stable under composition in X

Example

The first projection $p\colon A \times B \to A$ is a cocartesian fibration

Example

$\mathrm{ev}_1\colon \mathrm{Fun}(\Delta^1, C) \to C$ is a cocartesian fibration

The meaning of cocartesian fibrations is that these are the isofibrations $p\colon X \to B$ such that each morphism $v\colon b_0 \to b_1$ in B induces a canonical functor $v_!\colon X(b_0) \to X(b_1)$

Universal properties

We postulate that any object-wise construction defined by a universal property is functorial:

For any cocartesian fibration $f : X \to B$, if each fiber X_b has a terminal object s_b for each object b of B, then f has a fully faithful right adjoint $s : B \to X$ such that $s(b) = s_b$ holds object-wise.

Straightening/Unstraightening

We postulate that there are universes: a type Cat equipped with a cocartesian fibration

$$\pi_{univ} \colon \mathrm{Cat}_\bullet \to \mathrm{Cat}$$

with the following meaning.
We think of the objects of Cat as small categories. Each object c gives rise to a type C through the following pullback.

$$\begin{array}{ccc} C & \longrightarrow & \mathrm{Cat}_\bullet \\ \downarrow & & \downarrow{\scriptstyle \pi_{univ}} \\ \Delta^0 & \xrightarrow{c} & \mathrm{Cat} \end{array}$$

If $c_i \colon \mathrm{Cat}^{\simeq}$ corresponds to type C_i, $i = 0, 1$, we want an equivalence

$$\mathrm{Cat}(c_0, c_1) \cong \mathrm{Fun}(C_0, C_1)^{\simeq}$$

More generally, we ask for the following property:

- Any cocartesian fibration with small fibers is a pullback of π_{univ}.
- For any pullbacks $\begin{array}{ccc} X_i & \longrightarrow & \mathrm{Cat}_\bullet \\ {\scriptstyle p_i}\downarrow & & \downarrow{\scriptstyle \pi_{univ}} \\ B & \xrightarrow{f_i} & \mathrm{Cat} \end{array}$ with $i = 0, 1$, there is a canonical equivalence

$$\mathrm{Fun}(B, \mathrm{Cat})(f_0, f_1) \cong \mathrm{coCart}_B(X_0, X_1)$$

where $\mathrm{coCart}_B(X_0, X_1)$ is the groupoid whose objects are functors $f : X_0 \to X_1$ over B that send p_0-cocartesian morphisms to p_1-cocartesian morphisms.

Cat is a universe whose **properties** reflects faithfully **all** the features of synthetic category theory: it is a semantic interpretation of synthetic category theory itself

Semantic interpretation

Synthetic category theory is at least as sound as ZFC:

Theorem (D.-C. C. & K. Nguyen)

The theory of ∞-categories, as developed by Joyal and Lurie, is a semantic interpretation of synthetic category theory.

Perspectives

- Yoneda Lemma
- Existence of limits and colimits in the universe Cat
- pointwise Kan extensions in (co)complete categories
- presentable categories
- K-theory (what it means to count)
- topoi
- operads, higher algebra
- spectral/derived algebraic geometry
- motivic sheaves, étale cohomology, six operations
- ...

References

joint work with Bastiaan Cnossen, Hoang Kim Nguyen and Tashi Walde

- E. Riehl, M. Shulman *A type theory for synthetic ∞-categories*, Higher Structures **1** (2017), 116–193
- D.-C. Cisinski, *Higher Categories and Homotopical Algebra*, Cambridge studies in advanced mathematics, vol. 180, Cambridge University Press, 2019.
- H. K. Nguyen, *Covariant & Contravariant Homotopy Theories*, arXiv:1908.06879.
- H. K. Nguyen, *A note on coCartesian fibrations*, arXiv:2210.07753
- D.-C. Cisinski and H. K. Nguyen, *The universal coCartesian fibration*, arXiv:2210.08945
- B. Cnossen, *Formalization of Higher Categories*, based on a lecture course by D.-C. Cisinski. Available at https://sites.google.com/view/bastiaan-cnossen
- Series of papers on ∞-category theory in higher topoi by L. Martini and S. Wolf