

Univalent Foundation and Constructive Mathematics

Thierry Coquand

Oberwolfach, November 18, 2014

Univalent Foundations

Voevodsky's program to express mathematics in

type theory

instead of

set theory

Foundation of mathematics

This program relies on 2 points

(1) description of mathematics as analysis of *structures on ∞ -groupoids*

(2) *dependent* type theory provides a suitable language and system of notations to express structures on ∞ -groupoids

Description of mathematical object

First level: algebraic structure, ordered structure

E.g. groups, rings, lattices

Set with operations and/or relations satisfying some properties

Uniqueness up to isomorphisms

It is the level considered by Bourbaki in his *theory of structures*

Description of mathematical object

The next level is usually described as the level of *categories*

Actually the next level is the level of *groupoid with structures*

(A category will be like a poset at the level of groupoids)

The notion of isomorphism becomes at this level the notion of *equivalences*

Description of mathematical object

At the next level we have structures on *2-groupoids*

And so on, *n*-groupoids and then ∞ -groupoids

«the intuition appeared that ∞ -groupoids should constitute particularly adequate models for homotopy types, the *n*-groupoids corresponding to truncated homotopy types (with $\pi_i = 0$ for $i > n$)» (Grothendieck, Sketch of a program)

The notion of *homotopy type* generalizes the notion of *set*

The notion of (homotopical) equivalence generalizes the notion of bijection

Description of mathematical object

This description of mathematical objects gets a remarkably simple formal representation in *dependent type theory*

The notion of ∞ -groupoid becomes there a primitive notion

The notions of set, groupoid, 2-groupoid, \dots are derived notions

Set theory and type theory

1908 Zermelo *Untersuchungen über die Grundlagen der Mengenlehre*

1908 Russell *Mathematical Logic as Based on the Theory of Types*

«Simple» type theory

1940 Church *A Formulation of the Simple Theory of Types*

Extremely simple and natural

A type *bool* as a type of «propositions»

A type *I* for «individuals»

Function type $A \rightarrow B$

Natural semantics of *types as sets*

Functions in simple type theory

In set theory, a function is a *functional graph*

In type theory, a function is given by an *explicit definition*

If $t : B$, we can introduce f of type $A \rightarrow B$ by the definition

$$f(x) = t$$

$f(a)$ «reduces» to $(a/x)t$ if a is of type A

Functions in simple type theory

We have two notions of function

-*functional graph*

-*function explicitly defined* by a term

What is the connection between these two notions?

Church introduces a special operation $\iota x.P(x)$ and the «axiom of description»

If $\exists!x : A.P(x)$ then $P(\iota x.P(x))$

Functions in simple type theory

We can then define a function from a functional graph

$$\forall x. \exists! y. R(x, y) \rightarrow \exists f. \forall x. R(x, f(x))$$

by taking $f(x) = \iota y. R(x, y)$

By contrast, Hilbert's operation $\epsilon x. P(x)$ (also used by Bourbaki) satisfies

if $\exists x : A. P(x)$ then $P(\epsilon x. P(x))$

To use $\exists! x : A. \varphi$ presupposes a notion of equality on the type A

Rules of equality

Equality can be specified by the following purely logical rules

(1) $a =_A a$

(2) if $a_0 =_A a_1$ and $P(a_0)$ then $P(a_1)$

Equality in mathematics

The first axiom of set theory is the axiom of *extensionality* stating that two sets are equal if they have the same element

In Church's system we have two form of the axiom of extensionality

(1) two equivalent propositions are equal

$$(P \equiv Q) \rightarrow P =_{bool} Q$$

(2) two pointwise equal functions are equal

$$(\forall x : A. f(x) =_B g(x)) \rightarrow f =_{A \rightarrow B} g$$

The axiom of univalence will be a generalization of (1)

Dependent types

The basic notion is the one of *family of types* $B(x)$, $x : A$

We describe directly some *primitive* operations

$(\Pi x : A)B(x)$ f where $f(x) = b$

$(\Sigma x : A)B(x)$ (a, b)

$A + B$ $i(a), j(b)$

which are *derived* operations in set theory

Dependent types

Logical operations are reduced to constructions on types by the following dictionary

$$A \wedge B \qquad A \times B = (\Sigma x : A)B$$

$$A \vee B \qquad A + B$$

$$A \rightarrow B \qquad A \rightarrow B = (\Pi x : A)B$$

$$(\forall x : A)B(x) \qquad (\Pi x : A)B(x)$$

$$(\exists x : A)B(x) \qquad (\Sigma x : A)B(x)$$

Dependent types

de Bruijn (1967) notices that this approach is suitable for representation of mathematical proofs on a computer (AUTOMATH)

Proving a proposition is reduced to building an element of a given type

« This reminds me of the very interesting language AUTOMATH, invented by Dijkstra's colleague (and next-door neighbor) N. G. de Bruijn. AUTOMATH is not a programming language, it is a language for expressing proofs of mathematical theorems. The interesting thing is that AUTOMATH works entirely by type declarations, without any need for traditional logic! I urge you to spend a couple of days looking at AUTOMATH, since it is the epitome of the concept of type. »

D. Knuth (1973, letter to Hoare)

Dependent types

This is the approach followed for the formalization of Feit-Thompson's theorem

Voevodsky's program precises this representation by characterizing which types correspond to mathematical propositions

Universes

A universe is a type the element of which are types, and which is closed by the operations

$$(\Pi x : A)B(x)$$

$$(\Sigma x : A)B(x)$$

$$A + B$$

Russell's paradox does not apply directly since one *cannot* express $X : X$ as a *type*

However, Girard (1971) shows how to represent Burali-Forti paradox if one introduces a type of all types

Univers

Martin-Löf (1973), following Grothendieck, introduces of hierarchy of universe

$$U_0 : U_1 : U_2 : \dots$$

Each universe U_n is closed by the operations

$$(\Pi x : A)B(x)$$

$$(\Sigma x : A)B(x)$$

$$A + B$$

Universes and dependent sums

We can formally represent the notion of structure

$$(\Sigma X : U_0)((X \times X \rightarrow X) \times X)$$

collection of types with a binary operation and a constant

$$(X \times X \rightarrow X) \times X \text{ family of types for } X : U_0$$

This kind of representation is used by Girard for expressing Burali-Forti paradox

New laws for equality

Martin-Löf introduces (1973) a primitive notion of equality in dependent type theory

The « proposition » expressing the equality of a_0 and a_1 of type A is represented by a family of type $\text{Eq}_A(a_0, a_1)$

Since $\text{Eq}_A(a_0, a_1)$ is itself a type, one can iterate this construction

$$\text{Eq}_{\text{Eq}_A(a_0, a_1)}(p, q)$$

This is the core of the connection with ∞ -groupoid

New laws for equality

What are the rules of equality?

- (1) Any element is equal to itself $1_a : \text{Eq}_A(a, a)$
- (2) $C(a)$ implies $C(x)$ if we have $p : \text{Eq}_A(a, x)$

New laws for equality

The *new* law discovered by Martin-Löf (1973) can be expressed as the fact that in the type

$$(\Sigma x : A) \mathbf{Eq}_A(a, x)$$

which contains the special element

$$(a, 1_a) : (\Sigma x : A) \mathbf{Eq}_A(a, x)$$

any element (x, ω) is actually *equal* to this special element $(a, 1_a)$

New laws for equality

It follows from these laws that any type has a ∞ -groupoid structure

For instance, composition corresponds to transitivity of equality

The fact that equality is symmetric corresponds to the inverse operation

Hoffman-Streicher (1993)

S. Awodey, M. Warren (2009), P. Lumsdaine (2010), B. van den Berg, R. Garner

New laws for equality

These laws were discovered in 1973

Should equality be extensional?

Actually, how to express the extensionality axioms in this context?

An answer to this question is given by Voevodsky (2009)

Stratification

A type A is a *proposition*

$$(\prod x_0 : A)(\prod x_1 : A)\mathbf{Eq}_A(x_0, x_1)$$

A type A is a *set*

$$(\prod x_0 : A)(\prod x_1 : A)\mathbf{prop}(\mathbf{Eq}_A(x_0, x_1))$$

A type A is a *groupoid*

$$(\prod x_0 : A)(\prod x_1 : A)\mathbf{set}(\mathbf{Eq}_A(x_0, x_1))$$

Stratification

The notions of *propositions*, *sets*, *groupoids* have now acquired a precise meaning

They will be used with this meaning in the rest of this talk

Type theory appears as a generalization of set theory

Equivalence

Voevodsky gives a simple and uniform definition of the notion of *equivalence* for $f : A \rightarrow B$

If A and B are *sets* we get back the notion of *bijection* between sets

If A and B are *propositions* we get back the notion of *logical equivalence* between propositions

If A and B are *groupoids* we get back the notion of *categorical equivalence* between groupoids

Equivalence

If $f : A \rightarrow B$ the *fiber* of f at $b : B$ is the type

$$F(b) = (\Sigma x : A) \mathbf{Eq}_B(b, f(x))$$

f is an *equivalence* if this fiber is *contractible* for each b

$$(\Pi b : B)(F(b) \times \mathbf{prop}(F(b)))$$

$$A \simeq B \text{ is defined to be } (\Sigma f : A \rightarrow B) \mathbf{Equiv}(f)$$

For instance, the identity function is an equivalence using the new law of equality discovered by Martin-Löf and hence we have $A \simeq A$

The axiom of univalence

The *axiom of univalence* states roughly that if $f : A \rightarrow B$ is an equality then A and B are equal

More precisely, since $A \simeq A$ we have a map $\mathbf{Eq}_U(A, B) \rightarrow A \simeq B$

the canonical map $\mathbf{Eq}_U(A, B) \rightarrow A \simeq B$ is an equivalence

This generalizes Church's axiom of extensionality for *propositions*

Voevodsky has shown that this axiom implies *function extensionality*

The axiom of univalence

$$\text{Eq}_U(A \times B, B \times A)$$

$$\text{Eq}_U(A \times (B \times C), (A \times B) \times C)$$

Any property satisfied by $A \times B$ that can be expressed in type theory is also satisfied by $B \times A$

This is not the case in set theory

$$(1, -1) \in \mathbb{N} \times \mathbb{Z} \qquad (1, -1) \notin \mathbb{Z} \times \mathbb{N}$$

The axiom of univalence

This also entails

- two isomorphic sets are equal
- two isomorphic algebraic structures are equal
- two (categorically) equivalent groupoid are equal
- two equivalent categories are equal

The equality of a and b entails that any property of a is also a property of b

Algebraic structures

An algebraic structure is an element of a type of the form

$$(\Sigma X : U_0) \text{set}(X) \times T(X)$$

sets with operations and properties

Semantics

It is natural to represent a type as a *homotopy type*

D. Kan *A Combinatorial Definition of Homotopy Groups*, 1958

A type is interpreted as a Kan simplicial set

A family of type $B(x)$, $x : A$ is interpreted as a *Kan fibration*

The type $\mathbf{Eq}_A(a_0, a_1)$ becomes the space of *paths* joining a_0 and a_1

This model satisfies the axiom of univalence (Voevodsky, 2009)

Semantics

What happens to the new law for equality discovered by Martin-Löf in this interpretation?

Any element of $(\Sigma x : A)\text{Eq}_A(a, x)$ is equal to $(a, 1_a)$

It expresses the fact that the total space of the fibration defined by the space of paths having a given origin is *contractible*

This is exactly this fact which was the starting point of the loop-space method in algebraic topology (J.P. Serre)

Semantics

«Indeed, to apply Leray's theory I needed to construct fibre spaces which did not exist if one used the standard definition. Namely, for every space X , I needed a fibre space E with base X and with trivial homotopy (for instance contractible). But how to get such a space? One night in 1950, on the train bringing me back from our summer vacation, I saw it in a flash: just take for E the space of paths on X (with fixed origin a), the projection $E \rightarrow X$ being the evaluation map: path \rightarrow extremity of the path. The fibre is then the loop space of (X, a) . I had no doubt: this was it! ... It is strange that such a simple construction had so many consequences.»

Transport de structures

Soit $\mathbf{Grp}(A)$ le type qui donne une structure de groupe sur A

$$\mathbf{Grp}(A) = (\Sigma f : A \rightarrow A \rightarrow A)(\Sigma a : A) \dots$$

The collection of all groups is $(\Sigma X : U_0)\mathbf{set}(X) \times \mathbf{Grp}(X)$

This type is a groupoid

Transport of structures

If A and B are two isomorphic sets we have a proof of

$$\text{Eq}_U(A, B)$$

by the axiom of univalence and hence a proof of

$$\text{Grp}(A) \rightarrow \text{Grp}(B)$$

This expresses the notion of *transport of structure* (Bourbaki) along the given isomorphism between A and B

Differences with set theory

Any property is transportable

No need of «critères de transportabilité» as in set theory

«Only practice can teach us in what measure the identification of two sets, with or without additional structures, presents more advantage than inconvenient. It is necessary in any case, when applying it, that we are not lead to describe non transportable relations.» Bourbaki, Théorie des Ensembles, Chapitre 4, Structures (1957)

$0 \in A$ is a non transportable property of a group A

«to be solvable» is a transportable property

Differences with set theory

The collection of all groups/rings/posets form a *groupoid*

U_0 is *not* a set (at least a groupoid)

U_1 is *not* a groupoid (at least a **2**-groupoid)

Complexity of equality of a type versus set theoretic «size»

Posets and categories

In this approach

the notion of groupoid is more fundamental than the notion of category

A groupoid is defined as a type satisfying a property

Posets and categories

A *preorder* is a set A with a relation $R(x, y)$ satisfying

$$(\Pi x : A)(\Pi y : A)\text{prop}(R(x, y))$$

which is reflexive and transitive

A *poset* is a preorder such that the canonical implication

$$\text{Eq}_A(x, y) \rightarrow R(x, y) \times R(y, x)$$

is a logical equivalence

Posets and categories

A *category* is a *groupoid* A with a relation $\mathbf{Hom}(x, y)$ satisfying

$$(\prod x : A)(\prod y : A)\mathbf{set}(\mathbf{Hom}(x, y))$$

This family of sets is «transitive» (associative composition operation) and «reflexive» (we have a neutral element)

This corresponds to the notion of *preorder*

Posets and categories

One can define $\mathbf{Iso}(x, y)$ which is a *set* and show $\mathbf{Iso}(x, x)$

This defines a canonical map

$$\mathbf{Eq}_A(x, y) \rightarrow \mathbf{Iso}(x, y)$$

For being a *category* we require that this map is an equivalence (bijection) between the sets $\mathbf{Eq}(x, y)$ and $\mathbf{Iso}(x, y)$

The axiom of univalence implies that the groupoid of rings, for instance, has a categorical structure