RDP 2015, Jun. 30, 2015, WCMCS, Warsaw.

From syntax to semantics of Dependent Type Theories - Formalized

by Vladimir Voevodsky
from the Institute for Advanced Study in Princeton, NJ.

I will be speaking about a project that I have been working on since 2005.

The goal of this project is to make mathematicians to trust dependent type theories.

It started with a smaller goal of making myself to understand and to trust such theories. I had largely achieved this smaller goal at the end of 2009.

Then I was surrounded by a group of logisticians who trusted the connection between type theories and set theory on faith and who tried to convince me and others that mathematicians should do likewise.

It is only less than a year ago and due to the steadfast refusal of the mathematical community to do so that I was able to disentangle myself from them and to continue the work interrupted in 2009.

I will be speaking today only about type theories that can be called type theories of "Martin-Lof genus" i.e. about the type theories whose only semantically meaningful elements are sentences of four kinds:

Γ ; - meaning that Γ is a valid context,

$\Gamma; t : T$ - meaning that t is an object expression of type T in the context Γ,

$\Gamma; T=T'$ - meaning that in the context Γ the type expressions T and T' are substitutionally equal,

$\Gamma; t=t' : T$ - meaning that in the context Γ, t and t' are object expressions of type T that are substitutionally equal in T,

and where Γ is of the form $x_1:T_1,\ldots x_n:T_n$

As a proof assistant system I am using UniMath. UniMath is a library of mathematics formalized in Coq that has two unusual features.

One is that it uses only a very small subset of Coq that on the level of type theory roughly corresponds with the original Martin-Lof type theory (MLTT71).

Another one is that it represents mathematics in type theory using the univalent approach to formalization that is based on the concept of h-level and on the intuition provided by "the standard simplicial set model."

Most of the material in UniMath does not use any axioms or parameters other than the function extensionality and sometimes propositional extensionality and "resizing rule" for types of h-level 1.

Univalent approach to formalization allows one to formalize easily a large portion of mathematics especially in the area that is centered on what one may call modern algebra.This includes both the standard algebraic hierarchy (groups, rings etc.) and the algebraic objects of 'higher dimension'' such as categories, comprehension categories etc.

It also includes the theory of ordered algebraic structures and the theory of some essentially algebraic structures such as C-systems (contextual categories) that are important for the theory of type theories.

However, when my mathematical colleagues try to learn UniMath they encounter a problem, the same problem that I encountered back in 2004/5, which is that there is no convincing mathematical theory of type systems.

When I started learning this field the only book that I could find *that was written from a mathematical perspective* and that explained the type theory reaching up to the level close to the Martin-Lof type theories, was the book "Categorical logic and type theory" by B. Jacobs. This book has 780 pages and the most basic notions of dependent type theory appear there only close to the end somewhere after the "Advanced fibered category theory" section.

As far as I know it is still the only book that attempts a mathematical approach to Martin-Lof type theories.

After 10 years of active work in this area I came to the opinion that the reason that there is no book that I could refer a mathematician to is not the lack of effort or will on the part of the authors but the fact that there ***is no*** rigorous mathematical theory of type theories that would reach up to the level of the Martin-Lof type theories and that would connect them to the mesh of constructions that is modern mathematics.

There is a theory that logicians developed that is based on concepts such as meaning explanation or strong normalization that provides consistency proofs for some sample type theories but it leads to the isolation of type theory from the rest of mathematics rather than to their integration.

The problem starts at the level of syntax.

The first reaction to the problem of formalizing the syntax in this audience is likely to be as follows:

“But certainly this at least has already been investigated by many different teams and many good solutions have been found!”

This is only partially the case. In fact, I was able to find only one solution that connects general syntax with bound variables to modern mathematics. It was found by A. Hirschowitz and M. Maggesi around 2007 and outlined in a number of papers that consider as their common topic modules over monads and linear morphisms of such modules.

This solution has not yet become widely known not only among mathematicians but even among computer scientists so please let me spend some time on describing it.

The key issue here is to explain syntax and computation based on syntax, i.e., rewriting to mathematicians in such a way that they could connect this explanation to the rest of their mathematical knowledge.

This has been successfully achieved for the syntax that does not involve bound variables. The mathematical theory of such syntax and of the rewriting in it is known as ***universal algebra***. An average mathematician knows what is an abstract algebraic operation with n variables. He or she knows about important computation algorithms for systems with such operations and that these algorithms are the basis of the symbolic algebra systems.

Equally importantly, he or she knows that systems of universal algebra such as groups or rings are connected through multiple constructions to each other and to other areas of mathematics such as geometry, topology and analysis.

The following constructions allow one to do the same for the systems of expressions and rewriting rules that one obtains from operations that bound variables such as application, λ-abstraction and β-reduction.

Let us say that a logical signature Σ is a list of lists of natural numbers. The positions of Σ will correspond to operations of the signature and the value $(n_1,n_2,\ldots,n_d)$ at a given position to the arity of the operation where d is the number of arguments and n_i is the number of variables that are bound at the i-th argument. For example, the arity of λ in untyped λ-calculus is (1) and the arity of application is (0,0). Similarly, the arity of the "there exists" quantifier in single sorted logic is (1) and the arity of dependent product in Martin-Lof type theory is (0,1).

For any logical signature Σ and a set X let $R_{\Sigma}(X)$ be the set of expressions built using operations from Σ with free variables from the set X and considered modulo α-equivalence. As was observed many times the function that maps X to $R_{\Sigma}(X)$ can be given a structure of a monad on the category of sets which reflects the possibility of capture-free substitution for expressions modulo α-equivalence.

For a functor F on Sets let F' be the functor that sends X to F(X+pt) where pt is a chosen one point set. Let $F^{(n)}$ be the object defined by iterating the F' construction n times.

If R is a monad and M is a left module over R then, as Hirschowitz and Maggesi observed, M' can be given a structure of a left R-module in a natural way.

Their next observation is that an operation with arity $(n_1, n_2, \ldots, n_d)$ in $\mathbf{\Sigma}$ defines an operation

$$R^{(n_1)} \times \ldots \times R^{(n_d)} \to R$$

on $R = R_\Sigma$ and that this operation is a homomorphism of left R-modules. Moreover, and this is the key point,

R_Σ is the universal object among monads with such operations.

This provides a ***mathematical*** definition of the sets $R_\Sigma(X)$ that enables one to start explaining other constructions based on these sets to mathematicians.

The next step is to explain to mathematicians what are the inference rules.

This is at the moment the least developed part of the theory. Basically we only have examples.

But we can say that eventually we will define, for any logical signature **Σ**, a set of possible systems of inference rules **InfR**(**Σ**).

A type theory **T** of Martin-Lof genus will then be determined by a logical signature **Σ** and an element **S** in **InfR**(**Σ**):

$$\mathbf{T}=\mathbf{T}(\mathbf{\Sigma}, \mathbf{S})$$

From these data we should be able to construct in a mathematically acceptable way the sets of sentences of four Martin-Lof kinds derivable in this type theory.

Once we have the subsets of derivable sentences the theory gets again on a firmer footing.The key organizing concept here is the concept of a C-system or contextual category.

Two key mathematical constructions allow us to associate a C-system to a type theory once the raw syntax (logical signature) and the sets of derivable judgements are known.

In fact it is convenient to reflect the separation between types and terms on the syntactic level by having two signatures $\Sigma\tau$ and Σo where the first one contains the type-forming operations and the second one the object-forming ones. From such a pair one can construct a monad $R=R_{\Sigma o}$ and a module $LM=LM_{\Sigma\tau,\Sigma o}$ that contains expressions that can be obtained by substituting term expressions instead of variables into type-forming operations.

Any such pair of a monad R and a left module LM over R all on the category of sets defines a C-system CC(R,LM) as is described in my paper "C-system of a module over a monad on sets" that can be found on the arXiv.

All of the possible sentences of the first two Martin-Lof kinds that can be stated using the expressions generated by the signatures $\Sigma\tau$ and Σo form the B-sets B and Bt of the C-system CC(R,LM).

The sentences derivable by means of the system of inference rules S are subsets B(S) and Bt(S) of B and Bt and the sets of derivable sentences of the third and the fourth kind form subsets in the sets Beq and Bteq also defined in terms of R and LM and furthermore they define relations Eq(S) and Eqt(S) on B(S) and Bt(S).

In another paper (“Subsystems and regular quotients of C-systems”) I have worked out the exact conditions that the sets of derivable sentences have to satisfy in order for the sets B(S)/Eq(S) and Bt(S)/Eqt(S) to correspond again to a C-system that is then a sub-quotient (a quotient of a sub-object) of CC(R,LM).

We can denote this C-system by CC(Στ,Σο,S). This is an an essentially algebraic structure and its connections with mathematical structures of other classes such as categories with families, comprehension categories, type categories and display map categories is the subject of the joint project that we have embarked on with Benedikt Ahrens and Peter Lumsdaine.

The language of the univalent foundations makes the relationship between these various structures much more precise and easy to formalize.This part of the story will be addressed in the talk by Benedikt Ahrens.

The other side of the picture also involves a part that is poorly understood in general. One expects that any **Στ**,**Σo** and any system of inference rules formulated using **Στ**,**Σo**, i.e., any element S of **InfR**(**Στ**,**Σo**) should define a system of essentially algebraic operations and relations on C-systems. Let us denote this system of operations by Opr(S) and let us call the C-systems with operations from Opr(S), the C(S)-systems.

Then one expects the following fundamental **initiality conjecture** to hold:

Conjecture: The C-system CC(**Στ**,**Σo**,S) can be given a structure of a C(S)-system that makes it initial among C(S)-systems, i.e., for any C(S)-system CC there exists unique C(S)-homomorphism

$$CC(\boldsymbol{\Sigma\tau},\boldsymbol{\Sigma o},S) \to CC$$

As you can see at the moment we do not even known how to formulate this this conjecture precisely.

What makes it very difficult is that it appears, from the consideration of one or two particular cases, to be “purely technical” in the sense that the path to the formulation and solution of it is known and does not promise anything other than boring technical work on the way.

Another reason why it is so difficult is that the objects, such as **InfR**(**Σ**τ,**Σo**) that are involved in its are very complex. Until recently it was probably unreasonable to expect to have this conjecture seriously considered but now with the available formalization tools we can hope to have it properly formulated and proved.

The importance of the initiality conjecture lies in the fact that it provides us with the **only** known way to prove that type theories such as the Martin-Lof type theories, Calculus of Constructions or Calculus of Inductive Constructions remain consistent after one adds to them axioms such ah **the axiom of excluded middle or the axiom of choice.**

Therefore, any formalization of classical mathematics using proof assistants such as Coq relies implicitly on the initiality conjecture that remains not proved.

Moreover, this applies also to the use of the other semi-constructive axioms such as the propositional extensionality, functional extensionality, univalence axiom and probably Markov principle.