HLF 2016, Sep. 22, 2016, Heidelberg.

UniMath

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

Part 1. Univalent foundations

Today we face a problem that involves two difficult to satisfy conditions.

On the one hand we have to find a way for computer assisted verification of mathematical proofs.

This is necessary, first of all, because we have to stop the dissolution of the concept of proof in mathematics.

On the other hand we have to preserve the intimate connection between mathematics and the world of human intuition.

This connection is what moves mathematics forward and what we often experience as the beauty of mathematics.

The Univalent Foundations (UF) is, a yet imperfect, solution to this problem.

In their original form, the UF combined three components:

- the view of mathematics as the study of structures on sets and their higher analogs,

- the idea that the higher analogs of sets are reflected in the set-based mathematics as homotopy types,

- the idea that one can formalize our intuition about structures on these higher analogs using the Martin-Lof Type Theory (MLTT) extended with the Law of Excluded Middle for propositions (LEM) , the Axiom of Choice for sets (AC), the Univalence Axiom (UA) and the Resizing Rules (RR).

The main new concepts that were since added to these are the following:

- the understanding that a lot of mathematics can be formalized in the MLTT without the LEM and the AC and that excluding these two axioms one obtains foundations for a ***new form of constructive mathematics***,
- the understanding that classical mathematics appears as a subset of this new constructive mathematics,
- the understanding that the MLTT extended with the UA is an imperfect formalization system for this constructive mathematics and that it should be possible to integrate the UA into the MLTT obtaining a new type theory with better computational properties.

What does it mean for a formalization system to be constructive?

Some expressions in type theory are said to be in normal form. Any expression can be automatically and deterministically "normalized", that is, an equivalent expression in normal form can be computed.

In type theory there are type expressions and element expressions. If "T" is a type (expression) and "o" is an element (expression) one writes "o:T" if the type of "o" is "T".

In most type systems there is the type of natural numbers. In the UniMath it is written as "nat".

There is the zero element "O:nat" and the successor function "S" from "nat" to "nat" that intuitively corresponds to the function that takes "n" to "1+n".

A constructive system satisfies the ***canonicity property*** for "nat", which asserts that the normal form of any expression "o:nat" has the form "S(S(....(SO)..))".

By counting how many "S" there is in the normal form one obtains an actual natural number from any element expression of type "nat".

This is a tremendously strong property.

Consider the example: a set “X:hSet” is defined to be finite if there exists an isomorphism between it and the standard finite set “stn n”. Here “n” is an expression of type “nat”. It is well defined and one obtains a function “fincard” from finite sets to “nat” called the cardinality - the number of elements of the set.

Now suppose that I have proved, constructively, that “X” is finite. Then

“(fincard X):nat”

is defined. By normalizing “fincard X” I obtain an actual natural number.

If I had a constructive proof of ***Faltings’s Theorem,*** stating that the number of rational points on a curve of genus >1 is finite, I could find the actual number of points on any curve of genus >1.

We don't know whether such a proof exists. It is a very interesting and hard problem.

The reason that the MLTT+UA is an imperfect system for constructive formalization is that while MLTT itself has the canonicity property MLTT+UA does not.

Therefore, formalizing the proof of Faltings's Theorem in the UniMath, which is based on MLTT+UA, would not immediately give us an algorithm to compute the number of rational points on a curve of genus >1.

This is where a new type theory that integrates the UA into the MLTT in such a way as to preserve the canonicity would help.

The search for such a type theory became one of the main driving forces in the development of the UF.

Today several groups are working on the construction and implementation in a proof assistant of candidate type theories.

The ***cubical type theory*** and the prototype proof assistant ***cubicaltt*** created by the group of Thierry Coquand with the help of many researchers from different parts of the world is at the most advanced stage of development today.

A proof in the UniMath easily translates into a proof in the cubilatt.

The new form of the UF that emerges can be seen as combining the following components:

- the view of mathematics as the study of structures on sets and their higher analogs,
- the view of mathematics as constructive with the classical mathematics being a subset consisting of the results that require LEM and/or AC among their assumptions,
- the idea that the higher analogs of sets are reflected in the set-based mathematics as constructive homotopy types - objects of the new constructive homotopy theory that can so far be formulated only in terms of cubical sets,
- the idea that one can formalize our intuition about structures on these higher analogs using Cubical Type Theory (CTT).

In addition to the understanding that to obtain a formal system for the new constructive mathematics the UA needs to be integrated into the MLTT constructively, several more things are felt as lacking in the MLTT+UA:

- higher inductive types,
- resizing rules,
- a possible strict extensional equality combined with the "fibrancy discipline",
- as yet unknown mechanism to construct the types of structures that involve infinite hierarchies of coherence conditions.

Surprisingly, it might be easier to add these features to the CTT than to the MLTT. The work in these directions is ongoing.

Part 2. The UniMath library

In the development of the UniMath library we attempt to do something that might be compared with the effort by the Bourbaki group to write a systematic exposition of mathematics based on the set theory and the view of mathematics as studying structures on sets.

The effort by Bourbaki stalled at some point around the middle of the 20th century, in part, because it was very complicated to describe the emerging category-theoretic constructions in set-theoretic terms.

One may however ask, is there any mathematical innovation in what we are doing? Is there a discovery of the unknown in the work on the UniMath?

We have already seen how well-known problems in fields such as arithmetic algebraic geometry can be related to the search for a new foundation of constructive mathematics and for building proofs in the UniMath.

Here is a different example.

Some years ago, at the IAS, I had a conversation at lunch with Armand Borel. I mentioned how I like Bourbaki "Algebra" and how it helped me to become a mathematician.

I then mentioned that some places there were really dense. For example, said I, the description of the tensor product was hard to follow.

Of course, said Borel, ***we have invented tensor product to get a systematic exposition of multi-linear maps***.

It was new research, this is why it was not very smoothly written.

I was amazed.

It is hard to imagine today's mathematics without the concept of the tensor product. It would never occurred to me that it was invented by Bourbaki with the only purpose to obtain a more systematic exposition of multi-linear maps of vector spaces!

This example shows how a major innovation can emerge from the work on systematization of knowledge.

Finally, a few words to those mathematicians who will decide to understand UniMath and maybe to contribute to it.

The UniMath library is being created using the proof assistant Coq. It is freely available on GitHub.

The language of Coq is a very substantial extension of the MLTT and UniMath uses a very small subset of the full Coq language that approximately corresponds to the original MLTT.

The first file in the UniMath after the *Basics/preamble.v* is *Basics/PartA*.v.

The first line in *Basics/PartA.v* after the preamble section is as follows:

```
Definition fromempty  : Π X : UU , empty -> X.
```

It should be understood as a declaration of intent to define a constant called ***fromempty*** whose type is described by the expression that is written to the right of the colon.

Following this line there is a paragraph that starts with ***Proof.*** and ends with ***Defined.*** where the constant is actually defined using the little sub-programs of Coq called tactics which help to build complex expressions of the underlying type theory language in simple steps.

A mathematician who wants to understand UniMath should expect a very non-linear learning curve:

- In the lectures that I gave in Oxford and in the similar lectures in the Hebrew University it took me the whole first lecture to explain what that first line and the following it paragraph really mean.
- In the next lecture I was able to explain the next few hundred lines of PartA.
- By the fourth lecture in Oxford, the video of which can be found on my website, I was explaining the invariant formalization of fibration sequences.

I hope that was able to show how important Univalent Foundations are and how important is the work on libraries such as UniMath.

Thank you!