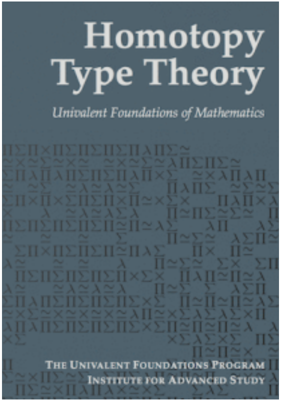
Homotopy type theory: A revolution in the foundations of mathematics?

David Corfield

University of Kent

7 March 2017

We live in interesting times!



A new foundational language for mathematics has just appeared.

Led by Vladimir Voevodsky, the production of this book required logicians, computer scientists (European style) and mathematicians to gather in Princeton for a year.

Amongst the mathematicians, homotopy theorists and category theorists in particular.

Graeme Segal on *The Ubiquity of Homotopy*

Much of mathematics is about discovering robust kinds of structure which organize and illuminate large areas of the subject. Perhaps the most basic organizing concept of our thought is space. It leads us to the homotopy category, which captures many of our geometric intuitions but also arises unexpectedly in contexts far from ordinary spaces. Still more is this true of the 'stable homotopy' category, which sits midway between geometry and algebra.

The theme of my lectures is the strangeness and the ubiquity of the homotopy and stable homotopy categories, and how they give us new ideas of what a space is, and why manifolds and spaces with algebraic structure play such a special role.

Computational trinitarianism

Constructive logic

Programming languages

Category theory

"The central dogma of computational trinitarianism holds that Logic, Languages, and Categories are but three manifestations of one divine notion of computation. There is no preferred route to enlightenment: each aspect provides insights that comprise the experience of computation in our lives." (Robert Harper)

Constructive logic Programming languages

Category theory

Each of these corners comes with a very intricate history.

It's important then to consider overlaps and differences.

Constructivism in maths

- Dutch philosophy involving denial of excluded middle, and double negation elimination.
- Often thought to have been defeated in maths at least back in the 1920s. (‘Like asking a boxer to tie one hand behind his back’, David Hilbert)
- However, it refused to go away.
- Computer science loves it.

Bad proof of 'Someone in this room is youngest'

- Assume not, so that for everyone in this nonempty room there is a younger person.
- Then (without loss of generality) start with me.
- By assumption, I have someone younger than myself, choose one such person. Then that person has someone younger again, and so on.
- There is an infinite chain of people which grows ever younger. Since there are only finitely many people, this sequence must have at least one person appear repeatedly, and certainly at least twice. But then that someone is younger than themselves. Contradiction.
- So it can't possibly be that no-one is the youngest. So someone must be the youngest.

This is needlessly non-constructive. Compare this with a constructive method to name the youngest.

- Place us in any order.
- Take the first person and compare their age with the next person. Repeat, replacing if and when you find someone younger.
- Repeat until you reach the end of the line.

Curry-Howard correspondence

The **Curry-Howard** correspondence associates proofs of propositions to programs carrying out tasks.

A constructive proof is like an algorithm.

A proof of A *implies* B corresponds to a recipe which transforms an input of type A to an output of type B.

Constructivity

- Unless otherwise specified, HoTT adopts a constructive outlook.
- But there's no difficulty in adding in classical principles if these are required .
- However, then one loses computational benefits, interpretation in wider range of settings, etc.

Type theory: Montague

Montague (from Church) had a sparse type theory with individuals, e, and truth values t, then properties $(e \to t)$. This then allowed quantifiers $(e \to t) \to t$, and so on.

But if the 'Battle of Hastings' and 'Vlad the Impaler' are two terms of type e, it seems as though we ought to be able to ask of them whether or not they are identical.

We are marginally better off for not being able to ask whether Julius Caesar is equal to 5.

But a richer type theory will allow us to represent the distinction between things and events. Philosophers have been led to take events as a basic ontological category for many reasons, not least because we can refer to them: 'Did you see that?'

Type theory: Martin-Löf

We don't ask of an event what its colour is, but where it took place, its duration and so on.

In the type theory we consider here, we won't ask of a person and an event whether they are the same. We will only ask of two terms whether they're equal *if they belong to the same type*.

This isn't something to be determined. Terms always come typed

- $A : Type, a, b : A$ then $(\vdash)$ $Id_A(a, b) : Type$

I can't even pose the question of the identity of two terms of different types.

Dependent types

Types depend on other types,

- $m : \mathit{Months} \vdash \mathit{Days}(m) : \mathit{Type}$;
- $t : \mathit{Teams} \vdash \mathit{Players}(t) : \mathit{Type}$.
- $n : \mathbb{N} \vdash \mathit{SquareMatrices}(n) : \mathit{Type}$.

Propositions (as types) do too:

- $t : \mathit{Team} \vdash \mathit{Plays\ In\ London}(t) : \mathit{Prop}$.
- $n : \mathbb{N} \vdash \mathit{prime}(n) : \mathit{Prop}$.

A very important idea in maths is the notion of a 'moduli space':

I have some space, set,..., and I want to know about certain structures to place on it. Often I can find an objects such that maps to that object correspond to a certain kind of structure on my set/space.

An easy example of this is to equip a set with a designated subset. The moduli space for such equippings is a two-element set.

One can think of the subset fibred above the set with fibres with 0 or 1 element.

If I ask you to raise your hand if your birth year is even, we can see this realised.

$$\begin{array}{ccc} \textit{Person born in even year} & & \\ \downarrow & & \\ \textit{Person} & \to & \mathbf{2} \end{array}$$

One can think of the subset fibred above the set with fibres with 0 or 1 element.

If I ask you to raise your hand if your birth year is even, we can see this realised.

$$\begin{array}{ccc} \textit{Person born in even year} & \rightarrow & \mathbf{1} \\ \downarrow & & \downarrow \\ \textit{Person} & \rightarrow & \mathbf{2} \end{array}$$

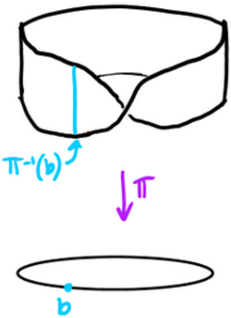
Similarly, dependent types are presented by maps, $T : A \to Type$ and can be represented as a downward arrow, e.g.:

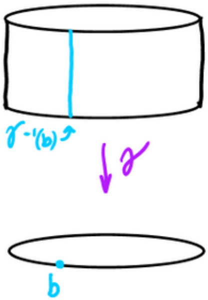
$$\begin{array}{ccc} Players & & \\ \downarrow & & \\ Teams & \to & Type \end{array}$$

Two important constructions, mathematically and in physics are *the total space* and *the space of sections*. This is coded in the type theory as dependent sum and dependent product.

$$\begin{array}{ccc} Players & & \\ \downarrow\uparrow & & \\ Teams & \to & Type \end{array}$$

So with dependent types it's helpful to have in mind the imagery of spaces fibred over other spaces:





- *Dependent sum* is the total space.
- *Dependent product* is the collection of sections.

In terms of physics, a field is a section of a bundle over space-time.

$$\begin{array}{ccc} \textit{Principle bundle on spacetime} & & \\ \downarrow\uparrow & & \\ \textit{Spacetime} & \rightarrow & \textit{Moduli space} \end{array}$$

In the case of the birth year property,

- dependent sum picks up the 'even year born people',
- dependent product would pick up, were there such a thing, a proof that each of us is born in an even year.

There will only be such a thing if everyone happens to be born in an even year.

Analogies between logic and arithmetic

If we assign the values 1 to True and 0 to False, then forming the conjunction ("and") of two propositions, the resulting truth value is formed very much as a product of numbers chosen from $\{0, 1\}$ is formed:

- Unless both values are 1, the product will be 0.
- Unless both truth values are True, the truth value of the conjunction will be False.

It is natural then to wonder if the disjunction ("or") of two propositions corresponds to addition. Here things don't appear to work out precisely. In the case of 'True or True', we seem to be dealing with an addition capped at 1.

Implication

- $(A \wedge B) \to C$ is True if and only if $A \to (B \to C)$ is True.
- $c^{(a \times b)} = (c^b)^a$

A proof of an implication is a mapping of proofs. Very much the approach of Martin-Löf and Dummett.

Similarly, these arithmetic quantities measure the cardinalities of sets of functions.

- We can explain this analogy via **intensional** type theory which adds the twist that an identity is not just a proposition but a type in itself $Id_A(a, b)$.
- Propositions are then taken as a certain kind of type (sometimes called 'mere propositions' when proof irrelevance assumed).

Dependent sum	Dependent product
$\sum_{x:A} B(x)$ is the collection of pairs (a, b) with $a : A$ and $b : B(a)$	$\prod_{x:A} B(x)$, is the collection of functions, f, such that $f(a) : B(a)$
When A is a set and $B(x)$ is a constant set B: The product of the sets.	When A is a set and $B(x)$ is a constant set B: The set of functions from A to B.
When A is a proposition and $B(x)$ is a constant proposition, B: The conjunction of A and B.	When A is a proposition and $B(x)$ is a constant proposition, B: The implication $A \to B$.

The bottom line is that homotopy type theory for the lower levels of the hierarchy encapsulates:

- Propositional logic
- (Typed) predicate logic
- Structural set theory

It is a structural theory *par excellence*. It seems impossible to say anything more by speaking of ‘the structure of A’ or ‘places in the structure of A’.

n-type hierarchy

Intensional type theory allows for more interesting identity structures on types:

...	...
2	2-groupoid
1	groupoid
0	set
-1	mere proposition
-2	

Forming identity types, $id_A(a, b)$, lowers the level.

Higher inductive types are used to construct interesting n-types such as the 2-sphere.

Category theory

- Formulated in the 1940s, it looks for common constructions throughout mathematics.
- Entities are gathered together in categories with some relevant kind of mapping between them.
- The nature of an entity in a category is determined by the patterns of arrows in and out of it.
- Some categories are especially 'nice' and support a 'logic' of a certain strength.
- Toposes are extremely nice, and support an (extensional) type theory.

Category theory

- ∞-toposes are needed in modern geometry (Lurie).
- *Homotopy Type Theory* corresponds to their internal language.
- HoTT = Intensional Martin-Löf type theory + Higher inductive types + Univalence axiom
- Still a work in progress.

Lawvere on quantifiers

For $\mathbf{H}$ is a topos (or ∞-topos) $f : X \to Y$ an arrow in $\mathbf{H}$, then base change induces between over-toposes:

$$(\sum_f \dashv f^* \dashv \prod_f) : \mathbf{H}/X \overset{\overset{f_!}{\to}}{\underset{f_*}{\overset{\overset{f^*}{\leftarrow}}{\to}}} \mathbf{H}/Y$$

Lawvere on quantifiers

Take a mapping

$$Owner : Dog \to Person,$$

then any property of people can be transported over to a property of dogs, e.g.,

$$Being\ French \mapsto Being\ owned\ by\ a\ French\ person.$$

We shouldn't expect every property of dogs will occur in this fashion.

In other words, we can't necessarily invert this mapping to send, say, 'Pug' to a property of People.

Lawvere on quantifiers

We can try...

Pug $\mapsto$ *Owning some pug* $\mapsto$???

Lawvere on quantifiers

But then

$$Pug \mapsto Owning\ some\ pug \mapsto Owned\ by\ someone\ who\ owns\ a\ pug.$$

However, people may own more than one breed of dog.

Lawvere on quantifiers

How about

$$Pug \mapsto Owning\ only\ pugs \mapsto ???$$

Lawvere on quantifiers

But this leads to

$$\textit{Pug} \mapsto \textit{Owning only pugs} \mapsto \textit{Owned by someone owning only pugs}$$

But again, not all pugs are owned by single breed owners.

Lawvere on quantifiers

In some sense, these are the best approximations to an inverse (left and right adjoints). They correspond to the type theorist's *dependent sum* and *dependent product*.

Were we to take the terminal map so as to group all dogs together ($Dog \to \mathbf{1}$), then the attempts at inverses would send a property such as 'Pug' to familiar things:

'Some dog is a pug' and 'All dogs are pugs'.

What if we take a map *Worlds* $\to \mathbf{1}$?

We begin to see the modal logician's *possibly* (in some world) and *necessarily* (in all worlds) appear.

What if we take a map $Worlds \to \mathbf{1}$?

We begin to see the modal logician's *possibly* (in some world) and *necessarily* (in all worlds) appear.

Things work out well if we form the (co)monads of dependent sum (product) followed by base change, so that possibly P and necessarily P are dependent on the type $Worlds$.

Such composites will be adjoint to each other, expressing their 'opposition'.

['Reader monad' and 'write comonad' are other two composites.]

These constructions applied to our pug case are:

$Pug \mapsto Owning\ some\ pug \mapsto Owned\ by\ someone\ who\ owns\ a\ pug.$

$Pug \mapsto Owning\ only\ pugs \mapsto Owned\ by\ someone\ owning\ only\ pugs$

We have equivalents of

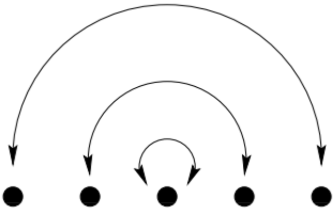
- $P \to \bigcirc P$ and $\bigcirc \bigcirc P \to \bigcirc P$
- $\Box P \to P$ and $\Box P \to \Box\Box P$

A context of symmetries

Since types need not just be sets, we should see what happens when we work in a simple non-set context such as $* : \mathbf{B}G$, for a group G.

A type in this context is something equipped with an action by G.

For example, a set of 5 objects acted on by the group of order 2:



Then for the unique map $\mathbf{B}G \to \mathbf{1}$

- Dependent sum is the quotient (action groupoid, orbits);
- Dependent product is the fixed points of the action.

Consider with Black (1952), a universe empty apart from two identical spheres. If I cannot describe a differentiating property, how many spheres are there: 0, 1 or 2?

For the inclusion of a subgroup, $H \to G$, we recover restricted, induced and coinduced representations.

In full generality, we can work with any map between ∞-groups.

Towards physics

- All very well, but we need to recreate differential topology and geometry. We have something like the 'total space' and 'space of sections' constructions, but we need spatial cohesion and smoothness.
- To try to do this in plain HoTT would commit the same mistake as to adopt set theoretic 'in principle' foundations.
- We need a tailored way to express spatial cohesion and smoothness.
- Fortunately, Urs Schreiber and Mike Shulman have developed Lawvere's ideas on cohesion to do just this, see dcct.

Variants of HoTT

There are ways to expand HoTT:

- Plain HoTT
- Cohesive HoTT
- Directed HoTT
- Linear HoTT (as in the 'linear' of 'linear logic')

'Modalities' can be introduced to allow for spaces to be constructed by gluing together their parts.

Synthetic theories in mathematics and physics

> *The idea of synthetic spaces can be summarized as follows: if all objects in mathematics come naturally with spatial structure, then it is perverse to insist on defining them first in terms of bare sets, as is the official foundational position of most mathematicians, and only later equipping them with spatial structure. (Mike Shulman)*

We can use a variant of type theory to create sophisticated new mathematics, e.g., to guide the next twisted differential cohomology needed by the string theorist, from the twisted K-theory that describes the B-field-twisted Yang-Mills fields over D-branes in type II string theory to twisted equivariant elliptic cohomology in heterotic string theory, and beyond. (Urs Schreiber)

Physics and HoTT

First, pre-quantum geometry ... is naturally axiomatized in cohesive homotopy type theory; second, quantization (geometric quantization and path integral quantization, in fact we find a subtle mix of both) is naturally axiomatized in linear homotopy-type theory.

In fact we find that linear homotopy type theory provides an improved quantum logic that, contrary to the common perception of traditional quantum logic, indeed serves as a powerful tool for reasoning about what is just as commonly perceived as the more subtle aspects of quantum theory, such as the path integral, quantum anomalies, holography, motivic structure. (Urs Schreiber)

Lawvere's cohesion

Consider a chain of adjunctions between a category of spaces and the category of sets. If we take the former to be topological spaces, then

- One basic mapping takes such a space and gives its underlying set of points. All the cohesive 'glue' has been removed.
- There are two ways to generate a space from a set: one is to form the space with the *discrete* topology, where no point sticks to another.
- The other is to form the space with the *codiscrete* topology, where the points are all glued together into a single blob so that no part is separable,in the sense that any map into it is continuous.
- Finally, we need a second map from spaces to sets, one which 'reinforces' the glue by reducing each connected part to an element of a set, the connected components functor, π_0.

We have

$$(\pi_0 \dashv Disc \dashv U \dashv coDisc) : Top \to Set$$

These four functors form an adjoint chain, where any of the three compositions of two adjacent functors $(U \circ coDisc, U \circ Disc, \pi_0 \circ Disc)$ from the category of sets to itself is the identity, whereas, in the other direction, composing adjacent functors to produce endofunctors on Top $(coDisc \circ U, Disc \circ U, Disc \circ \pi_0)$ yields two idempotent monads and one idempotent comonad.

These correspond to the three adjoint modalities of the diagram:

$$\int \vdash \flat \vdash \sharp$$

To participate in such adjoint strings is demanding. By the time we find another (correctly related) layer, smoothness, or differential cohesion is expressed.

$$\Re \vdash \Im \vdash \&$$

In a context $\mathbf{H}$ of differential cohesion with $\Im$ the infinitesimal shape modality, then for any object $X \in \mathbf{H}$ the comonad

$$Jet_X := i^* i_*$$

for base change along the X-component of the unit of $\Im$

$$\mathbf{H}_{/X} \underset{i_*}{\overset{i^*}{\rightleftarrows}} \mathbf{H}_{/\Im(X)} ,$$

may be interpreted as sending any bundle over X to its jet bundle.

For which bundles is there a map from E to $Jet(E)$?

Those E which themselves have been pulled back from $\Im(\Sigma)$.

Marvan showed that these 'coalgebras' are solutions sets of PDEs.

Pulling back from $\Im(\Sigma)$ gives a way of comparing infinitesimally close fibres.

At last, we have a foundational language with the capacity to make serious contact with mainstream mathematics, including mathematical physics.

$$\begin{tikzcd}
8.2 & \text{11d Supergravity with M2/M5-branes} \\
8.1 & \text{Lorentzian supergeometry} \arrow[u, mapsto, "\text{unravel bouquet of Whitehead towers}"'] \\
6 & \text{Higher supergeometry} \arrow[u, mapsto, "\text{represent rheonomy and form universal cover}"'] \\
4.3 & \text{Solid homotopy theory} \arrow[u, mapsto, "\text{faithful model}"'] \\
4.2 & \text{Elastic homotopy theory} \arrow[u, mapsto, "\text{resolve finite/infinitesimal}"'] \\
4.1 & \text{Cohesive homotopy theory} \arrow[u, mapsto, "\text{resolve continuous/discrete}"'] \\
3 & \text{Homotopy topos theory} \arrow[u, mapsto, "\text{resolve initial opposition}"'] \\
2 & \text{Homotopy type theory} \arrow[u, mapsto, "\text{internal self-reflection (univalent universe)}"'] \\
 & \begin{array}{c}\text{Logical substance}\\ \text{type theory}\end{array} \arrow[u, mapsto, "\text{first law of thought: } A = A"']
\end{tikzcd}$$

$$
\begin{array}{rcccl}
 & \mathrm{id} & \dashv & \mathrm{id} & \\
 & \vee & & \vee & \\
\text{fermionic} & \rightrightarrows & \dashv & \rightsquigarrow & \text{bosonic} \\
 & \perp & & \perp & \\
\text{bosonic} & \rightsquigarrow & \dashv & \mathrm{Rh} & \text{rheonomic} \\
 & \vee & & \vee & \\
\text{reduced} & \Re & \dashv & \Im & \text{infinitesimal} \\
 & \perp & & \perp & \\
\text{infinitesimal} & \Im & \dashv & \& & \text{étale} \\
 & \vee & & \vee & \\
\text{cohesive} & \int & \dashv & \flat & \text{discrete} \\
 & \perp & & \perp & \\
\text{discrete} & \flat & \dashv & \sharp & \text{continuous} \\
 & \vee & & \vee & \\
 & \emptyset & \dashv & * &
\end{array}
$$