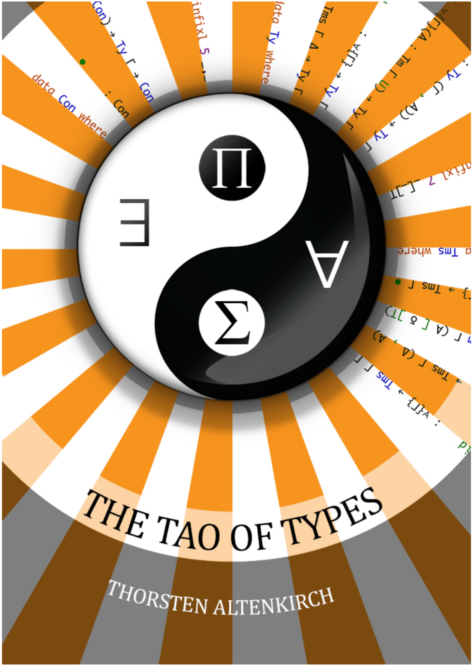
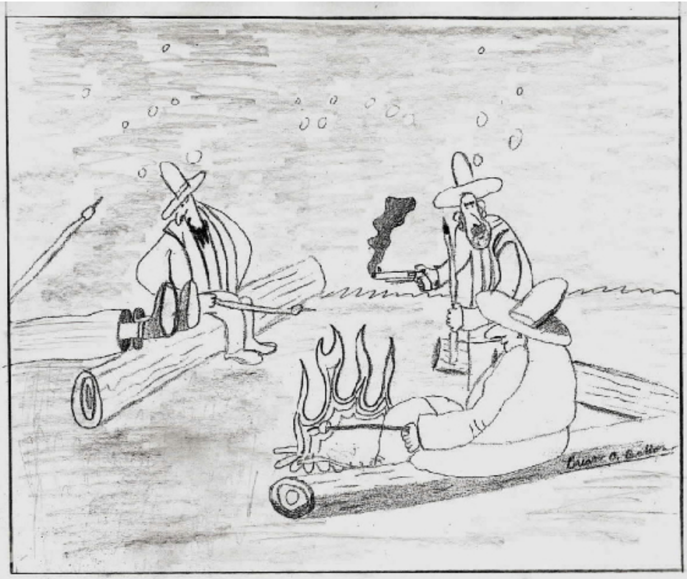
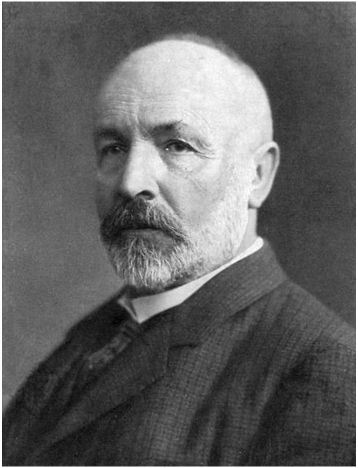

THE TAO OF TYPES
THORSTEN ALTENKIRCH
Π
∃
∀
Σ



You guys are both my witnesses... He insinuated that ZFC set theory is superior to Type Theory!

HISTORY

Sets



Georg Cantor (1845 - 1918)

Cantor's sets

By a **set** *we are to understand any collection into a whole of definite and separate objects of our intuition or our thought.*

Examples of sets

The empty set	$\{\}$
The set of natural numbers	$\mathbb{N} = \{0, 1, 2, \dots\}$
The set of integers	$\mathbb{Z} = \{\dots, -1, 0, 1, \dots\}$
The set of real numbers, e.g.	$\pi = 3.1415926 \dots \in \mathbb{R}$
The set of the previous examples	$\{\{\}, \mathbb{N}, \mathbb{Z}, \mathbb{R}\}$
A wierd set	$\{3, \mathbb{R}, \mathrm{you}\}$
The set of all sets (including itself)	$V \in V$

Counting elements (cardinality)

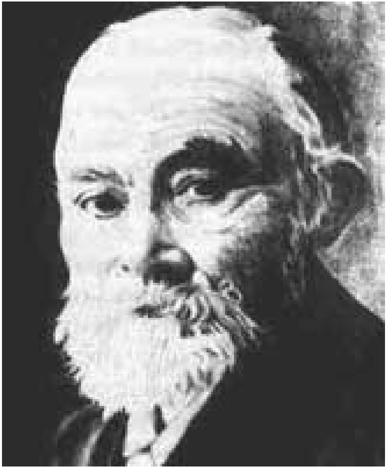
- When do two sets have the same number of elements?
- Methods of the shepherds who can't count.
- The natural numbers ($\mathbb{N}$) and the integers ($\mathbb{Z}$) have the same number of elements

0	1	2	3	4	...
0	-1	1	-2	2	...

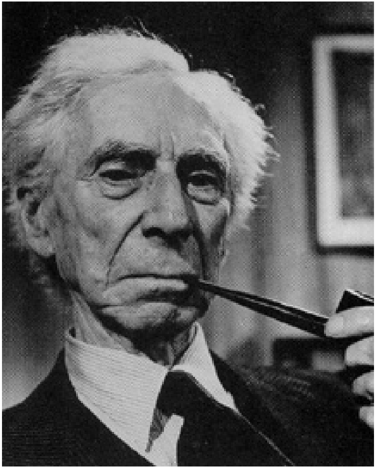
- Cantor showed that there is no such assignment between the natural numbers ($\mathbb{N}$) and the real numbers ($\mathbb{R}$).
- There are more real numbers than natural numbers!

Ordering sets

- The natural numbers can be ordered:
  $0 < 1 < 2 < 3 < \dots$
  - All the elements are related one way or the other.
  - The order is transitive e.g. from $1 < 2$ and $2 < 3$ we can conclude $1 < 3$.
  - We can go down only finitely many steps.
- Cantor proved that every set can be ordered this way. (Wellordering theorem)
- But what is a ordering of the Reals?
  Note that $0.1 > 0.01 > 0.001 > \dots$.
- Cantor's proof was highly controversial.
  Kronecker: *I don't know what predominates in Cantor's theory – philosophy or theology, but I am sure that there is no mathematics there*



Gottlob Frege (1848 - 1925)



Bertrand Russell (1872-1970)

Foundations!

- Frege wrote a book: *Die Begriffsschrift* to make the foundation of Mathematics precise.
- He developed an early version of *predicate logic*
  Example: *In every non-empty pub there is a person such that if this person drinks, then everybody drinks.*

$$(\exists x.\mathrm{True}) \to \exists x.\mathrm{Drinks}(x) \to \forall y.\mathrm{Drinks}(y)$$

  This is aways true (a tautology).

Russell's paradox

- Russell wrote a letter to Frege pointing out a problem with Frege's account of sets (in german!)
- We can construct a set of all sets which do not contain themselves.

$$\{X \mid X \notin X\}$$

If this set contains itself then it doesn't.
But if it doesn't contain itself, then it does.
???

c/o Conor McBride

Axioms

Ernst Zermelo (1871-1953)

Abraham Fraenkel (1891-1965)

- Zermelo fixed the theory of sets by formulating some axioms (using predicate logic).
- These axioms allow the construction of useful sets but rule out bad sets like Russell's.
- Zermelo's sets don't need urelements (like numbers) - everything can be constructed from sets.
- Fraenkel added an important axiom called *replacement*
- Another axiom is the *axiom of choice* which is needed to prove Cantor's wellordering theorem.
- Mathematicians usually admit if they use choice. (maybe they are embarrassed?).

1. Axiom of extensionality.
$$\forall x, y.(\forall z.z \in x \iff z \in y) \to x = y$$
2. Axiom scheme of limited comprehension.(*)
$$\forall z.\exists x.\forall y.(y \in x) \iff (y \in z \wedge \Phi(y))$$
3. Axiom of pairing
$$\forall x, y.\exists z.x \in z \wedge y \in z$$
4. Axiom of union
$$\forall x.\exists y.(\forall z.z \in x.\forall w.w \in z \to w \in y)$$
5. Axiom of infinity
$$\exists x.\emptyset \in x \wedge \forall y.y \in x \to \{y, \{y\}\} \in x$$
6. Powerset axiom
$$\forall x.\exists y.\forall z.z \subseteq x \to z \in y$$
7. Axiom scheme of replacement (*)
$$(\forall x, y, z.\Psi(x, y) \wedge \Psi(x, z) \to y = z) \to \forall u.\exists w.\forall v.v \in w \iff (\exists r.r \in u \wedge \Psi(r, v))$$
8. Axiom of regularity
$$\forall x.(\exists y.y \in x) \to \exists z.z \in x \wedge x \cap z = \emptyset$$
9. Axiom of choice
$$\forall x.(\emptyset \notin x \wedge \forall y, z.y \in x \wedge z \in x \wedge y \cap z \neq \emptyset \to y = z) \to \forall y.y \in x \to \exists! v.v \in x \wedge v \in y$$

Natural numbers in set theory

- Von Neumann's definition:

0	1	2	3	...
$\{\}$	$\{0\}$ $=\{\{\}\}$	$\{0,1\}$ $=\{\{\},\{\{\}\}\}$	$\{0,1,2\}$ $=\{\{\},\{\{\}\},\{\{\},\{\{\}\}\}\}$	...

- Zermelo's definition:

0	1	2	3	...
$\{\}$	$\{0\}$ $=\{\{\}\}$	$\{1\}$ $=\{\{\{\}\}\}$	$\{2\}$ $=\{\{\{\{\}\}\}\}$	...

- The axiom of infinity states that the natural numbers form a set.
- Usually we use von-Neumann's definition.
- To show that Zermelo's numbers are a set we need the axiom of replacement.
- We can distinguish the two definitions:
  $1 \subseteq 2$ is true for von Neumann's numbers
  but not for Zermelo's numbers.

Intuition

L. E. J. Brouwer (1881-1953)

David Hilbert (1862 -1943)

Intuitionism

- Brouwer criticised that many mathematical proofs are not based on intuitively valid reasoning.
- In particular he rejected the *principle of the excluded middle* : every proposition is either true or false

$$P \vee \neg P$$

- Platonism
  The world of ideas is like the real world.
- He had an argument with Hilbert
  Grundlagenstreit
- Brouwer's *intuitionistic logic* is constructive:
  If we can show that an object exists we can also compute it.

The negative translation

- Later logicians realised that excluded middle can be justified intuitionistically.
- We need to translate P *or* Q ($P \vee Q$) as
  it cannot be that both P and Q are false ($\neg(\neg P \wedge \neg Q)$)
- And *there is an object satisfying* R ($\exists x.R(x)$) as
  it cannot be that there isn't an object satisfying R ($\neg(\forall x.\neg R(x))$
- *A classical mathematician is somebody who cannot say anything positive.*

The Axiom of Choice

- However, we cannot use the negative translation to justify the axiom of choice :
  If for every left sock there is a matching right sock then there is an assignment of left socks to matching right socks.

$$(\forall x : L.\exists y : R.M(x, y)) \to \exists f : L \to R.\forall x : L.M(x, f(x))$$

- The axiom of choice is needed for example to prove Cantor's wellordering theorem.

Types

Per Martin-Löf (1942)

Type Theory

- Martin-Löf developed Type Theory as
  a foundation of intuitionistic Mathematics.
- What is the difference between sets and types?
  E.g. what is the difference between $3 \in \mathbb{N}$ and $3 : \mathbb{N}$?
- In Set Theory an element can belong to many different sets.
- In Type Theory an element always belongs to the same type.
- In Set Theory membership is a proposition (dynamic).
- In Type Theory membership is a judgement (static).

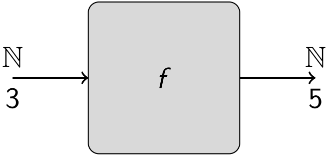
$$\frac{\text{Haskell}}{\text{Python}} = \frac{\text{Static typing}}{\text{Dynamic typing}} = \frac{\text{Type Theory}}{\text{Set Theory}}$$

Example : Function sets

- Let $f(x) = 2 + x$ be the function on natural numbers that doubles its input
- In set theory such a function is a set of pairs $f = \{(0,2),(1,3),(2,5),(3,6),\dots\}$ such that for every natural number x there is exactly one number y such that $(x,y) \in f$.
  $\forall x \in \mathbb{N}.\exists! y : \mathbb{N}.(x,y) \in f$
- Functions in set theory don't need to be computable.

Example : Function types

- In Type Theory functions are a primitive concept. We can visualize them as a black box:



- The laws of functions are the laws of λ-calculus, e.g. if $f(x) = 2 + x$ then $f(3) = 2 + 3$ (β-law).
- Every function is computable.
- *A function that doesn't function, shouldn't be called a function!*
- In predicate logic relations are a primitive concept but in Type Theory we use functions to represent them, a relation between A and B is a function $A \to B \to \mathrm{Prop}$.

Propositions as types

- In Set Theory we are using predicate logic to express the axioms.
- Type Theory doesn't require predicate logic, it can define it.
- To every mathematical proposition, we associate a type which corresponds to evidence that the proposition holds.
- For example: evidence of a proposition of the form
  If P then Q ($P \to Q$)
  is the type of functions from evidence of P to evidence of Q ($P \to Q$).
- This works for all of predicate logic.
- The logic is intuitionistic, in particular we cannot prove excluded middle ($P \vee \neg P$).

Structuralism

- In Set Theory we can distinguish different encodings of the natural numbers (e.g. von-Neumann vs Zermelo).
- In Type Theory we can also define the natural numbers in different ways, e.g.:

  ala Peano: Natural numbers are constructed from $0 : \mathbb{N}$ and $\mathrm{suc} : \mathbb{N} \to \mathbb{N}$.

  binary: Natural numbers are sequences of binary digits $0, 1$ with no leading 0s.
- However, in Type Theory we cannot differentiate these two types.
- In general we have a structural equivalence principle.

Geometry

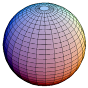
Vladimir Voevodsky (1966-2017)

Martin Hofmann (1965 - 2018)

Homotopy Lambda Calculus?

- In 2010 Vladimir Voevodsky sent an email to the Coq mailing list with a draft paper about a *Homotopy Lambda Calculus* ?
- I couldn't make much sense out of it.
- I thought what crackpot is this?
- I googled Voevodsky and realised that he has got a Fields medal!
- Ok, not a crackpot.
- But what is *homotopy theory* ?

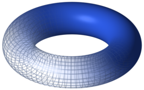
wikipedia: homotopy groups



A sphere

$$\pi_1(T) = \mathbb{Z}^2,$$

because the universal cover of the torus is the Euclidean plane $\mathbb{R}^2$, mapping to the torus $T \cong \mathbb{R}^2/\mathbb{Z}^2$. Here the quotient is in the category of topological spaces, rather than groups or rings. On the other hand, the sphere S^2 satisfies:



A torus

$$\pi_1(S^2) = 0,$$

because every loop can be contracted to a constant map (see homotopy groups of spheres for this and more complicated examples of homotopy groups).

Hence the torus is not homeomorphic to the sphere.

Definition

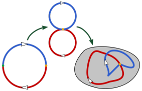
In the n-sphere S^n we choose a base point a. For a space X with base point b, we define $\pi_n(X)$ to be the set of homotopy classes of maps

$$f : S^n \to X$$

that map the base point a to the base point b. In particular, the equivalence classes are given by homotopies that are constant on the basepoint of the sphere. Equivalently, we can define $\pi_n(X)$ to be the group of homotopy classes of maps $g: [0,1]^n \to X$ from the n-cube to X that take the boundary of the n-cube to b.

For $n \geq 1$, the homotopy classes form a group. To define the group operation, recall that in the fundamental group, the product $f * g$ of two loops $f, g : [0,1] \to X$ is defined by setting

$$f * g = \begin{cases} f(2t) & t \in \left[0, \frac{1}{2}\right] \\ g(2t-1) & t \in \left[\frac{1}{2}, 1\right] \end{cases}$$



Composition in the fundamental group

The idea of composition in the fundamental group is that of traveling the first path and the second in succession, or, equivalently, setting their two domains together. The concept of composition that we want for the n-th homotopy group is the same, except that now the domains that we stick together are cubes, and we must glue them along a face. We therefore define the sum of maps $f, g : [0,1]^n \to X$ by the formula

$$(f+g)(t_1, t_2, \ldots, t_n) = \begin{cases} f(2t_1, t_2, \ldots, t_n) & t_1 \in \left[0, \frac{1}{2}\right] \\ g(2t_1 - 1, t_2, \ldots, t_n) & t_1 \in \left[\frac{1}{2}, 1\right] \end{cases}$$

For the corresponding definition in terms of spheres, define the sum $f + g$ of maps $f, g: S^n \to X$ to be Ψ composed with h, where Ψ is the map from S^n to the wedge sum of two n-spheres that collapses the equator and h is the map

Groupoids?

- Voevodsky was suggesting that we could view types as geometric objects (in higher dimensions).
- I sent a reply to Voevodsky, suggesting there may be a relation to Hofmann and Streicher's groupoid interpretation of type theory.
- Voevodsky replied: *You are right but actually I am using infinity groupoids*.

Uniqueness of Equality proofs

- When I was doing my PhD we asked the question: *Can we show in Type Theory that there is at most one proof of an equality?*
- So for example there is only one proof that $3 = 3$, called refl.
- There is no proof that $2 = 3$.
- And we can never construct more than one proof.
- But can we show in Type Theory that there is at most one proof?
- Many people tried but nobody succeeded.
- But how can we show that it is impossible?

The groupoid model

- Martin came up with an idea: we can interpret types as *groupoids*.
  Groupoids are a generalisation of
  equivalence relations (like having the same size)
  and groups (like the integers with addition and minus).
- This interpretation satisfies all the laws of Martin-Löf's Type Theory.
- But there are groupoids which have more then one element.
- Hence it is impossible that we can prove uniqueness.

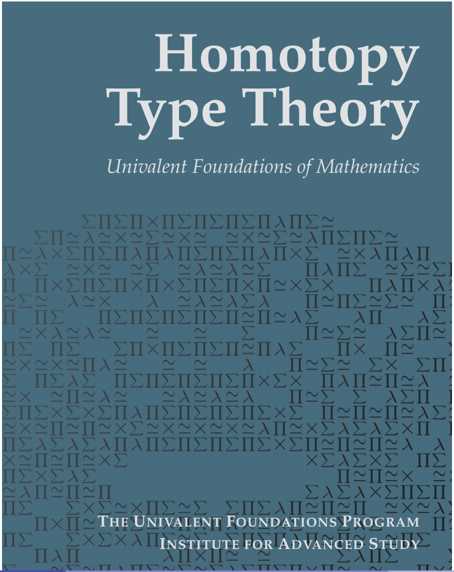
Univalence

- One of the most interesting features of Voevodsky's proposal was the *univalence principle*.
- It says that equality of types is equivalent to equivalence.
- E.g. if we view 3 as a type with 3 elements then there are $3! = 6$ ways to prove that $3 = 3$.
- And we can show that the Peano numbers and the binary numbers are actually equal.

Martin's proposal

- Many years earlier: I thought that the consequence of Martin's construction was that we need to add an additional principle to capture uniqueness of equality proofs.
- But Martin said: there are some interesting principles that hold in the groupoid model.
- For example, we can interpret equality of types as isomorphism (i.e. equivalence).
- I said: *That seems very exotic*.
- *And anyway then we should do infinity groupoids . . .*

The HoTT book



Institute for Advanced Study, Princeton

Special year on Homotopy Type Theory (2013)

Computer

Implementations

- Type Theory is a programming language!
- and a logic
- and an alternative to set theory.
- There are a number of implementations:
  NuPRL, Coq, Agda, Lean, Idris, ...
- My favorite is Agda.
- It now has an experimental implementation of HoTT: cubical agda.

Take home

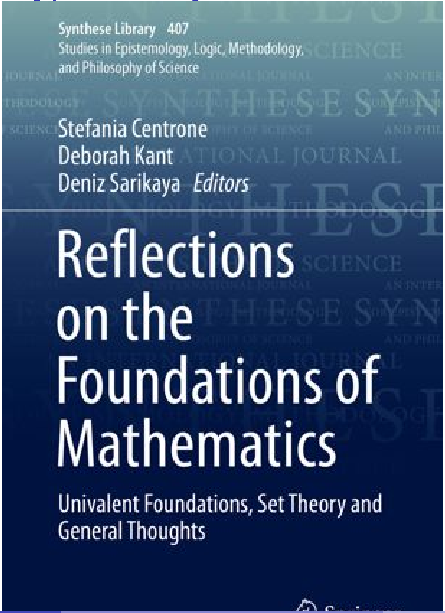
The new Archimedian principle

Give me the right language,

and we can solve any problem!

Ads

Chapter: Naïve Type Theory



Programs, Proofs and Types (COMP4074)

- 20 credit Spring module
  for 3rd and 4th year Undergraduates and MSc.
- Introduces Agda and Type Theory.
- Comments from SET/SEM:
  Very enjoyable module.

  Hard. As it should be, considering it is a master module. Loved it!

  Really good lecturer. Very enthusiastic about the subject, and he makes the sessions very interesting with lots of jokes and humour.
- 70% of students got a first in 2018/19.
- Module average: 72
- No fails.

The Python book (with Isaac Triguero)