Eilenberg-MacLane spaces in Homotopy Type Theory

Floris van Doorn

Carnegie Mellon University

March 22, 2017

j.w.w. Ulrik Buchholtz (TU Darmstadt) and Egbert Rijke (CMU)

Models of Type Theory

There are models of *type theory* in various abstract frameworks for *homotopy theory*.

Examples:

- Quillen model categories [Awodey, Warren, 2009];
- Simplicial sets [Streicher, 2011];
- Cubical sets [Bezem, Coquand, Huber, 2014];
- ... and many more.

Synthetic Homotopy Theory

This leads to a new program, *Synthetic Homotopy Theory*:

Study types in type theory as spaces in homotopy theory.

This gives a more general and constructive treatment of homotopy theory which is easier to verify formally in a computer proof assistant.

- The main theorem in this talk has been fully formalized.

I work in *Homotopy Type Theory* (HoTT): dependent type theory with *univalence* and *higher inductive types* [Homotopy Type Theory, 2013].

As motivating example I will concentrate on *Eilenberg-MacLane spaces*.

Homotopy Type Theory

Homotopy Type Theory combines Type Theory with Homotopy Theory.

	Type Theory	**Logic**	**Homotopy Theory**
A	Type	Formula	Space*
$a : A$	Term/Element	Proof	Point
$A \times B$	Product Type	Conjunction	Binary Product sp.
$A \to B$	Function Type	Implication	Mapping space
$P : A \to \mathrm{Type}$	Dependent Type	Predicate	Fibration
$\Sigma(x : A).\ P(x)$	Sigma Type	Ex. Quantifier	Total space
$\Pi(x : A).\ P(x)$	Dep. Fn. Type	Un. Quantifier	Product space
$a =_A b$	Identity Type	Equality	Path space

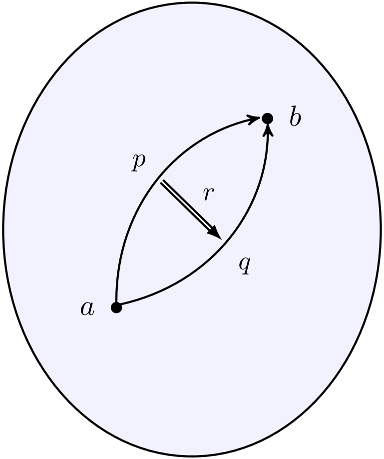
I will use these notions interchangably.

Types as spaces

A type A can have

- points $a, b : A$
- paths $p, q : a = b$
- paths between paths $r : p = q$

$$\vdots$$



Identity Type

Different ways to think about the identity type:

- **Type theory**: The identity type is generated by reflexivity: $\mathrm{refl}_a : a =_A a$.
- **Logic**: Equality is the least (free) reflexive relation.
- **Homotopy theory**: The path space with one point fixed is contractible.

(This does not mean every proof of equality is reflexivity)

Path Induction

This is made precise by *path induction*:

- If $C : \Pi(x : A).\ a = x \to \mathrm{Type}$,
- to prove/construct an element of $\Pi(x : A).\Pi(p : a = x).\ C(x, p)$
- it is sufficient to prove/construct an element of $C(a, \mathrm{refl}_a)$

Example Symmetry of equality (invertibility of paths)

$$\Pi(A : \mathrm{Type}).\ \Pi(a\ b : A).\ a = b \to b = a.$$

Proof. Suppose A is a type and $a : A$. We need to prove $\Pi(b : A).\ a = b \to b = a$.
We apply path induction, in which case we need to prove $a = a$, which is true by refl_a.

Identity Type (2)

We can also look at the identity type in a type-oriented way:

$$\begin{array}{rcll} (a,b) =_{A\times B} (a',b') & \text{is} & (a =_A a') \times (b =_B b') & \\ f = g & \text{is} & \Pi x.\ f(x) = g(x) & \text{(function extensionality)} \\ A =_{\mathrm{Type}} B & \text{is} & A \simeq B & \text{(univalence, Voevodsky)} \end{array}$$

This is done in *cubical type theory*.

Truncated Types

Some types are *truncated*, which means there are all higher paths are trivial.

A type A is *contractible* ((-2)-type) if it has exactly one element, if

$$\Sigma(x : A).\ \Pi(y : A).\ x = y.$$

A type A is a *proposition* ((-1)-type) if it has at most one element, if

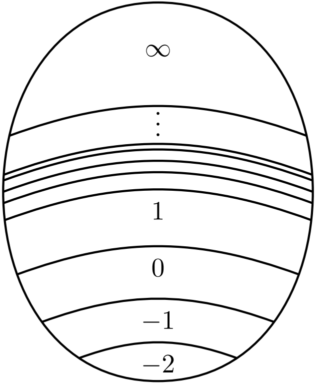
$$\Pi(x\ y : A).\ x = y.$$

In either of the above cases, all (higher) paths in A are trivial.

A is a *set* (0-type) if for all $x\ y : A$ the type $x = y$ is a proposition.

A is an $(n+1)$-type if for all $x\ y : A$ the type $x = y$ is an n-type.

Truncated Types



Given A, we can form the n-*truncation* $\|A\|_n$.

$\|A\|_n$ is the "best approximation" of A which is n-truncated.

If X is n-truncated, we get the following universal property:

$$\begin{tikzcd} A \arrow[d, "|-|_n"'] \arrow[dr, "\forall"] & \\ \|A\|_n \arrow[r, dashed, "\exists!"'] & X \end{tikzcd}$$

Higher Inductive Types

In Type Theory there are *inductive types*, in which you specify its points.

Examples. $\mathbb{N}$ is generated by 0 and succ
$A + B$ is generated by either $a : A$ or $b : B$
$a =_A (-)$ is generated by $\mathrm{refl}_a : a =_A a$

In homotopy theory we can build cell complexes inductively.

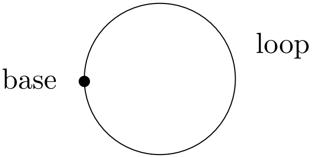
In HoTT we can combine these into *higher inductive types* [Shulman, Lumsdaine, 2012].

The circle

Example. The circle $\mathbb{S}^1$

$\texttt{HIT}\ \mathbb{S}^1 :=$

- $\mathrm{base} : \mathbb{S}^1$
- $\mathrm{loop} : \mathrm{base} = \mathrm{base}$



Using univalence, we can prove $\mathrm{loop} \neq \mathrm{refl}$.

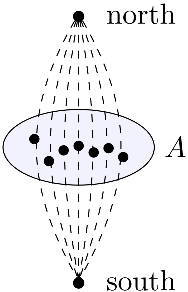
Recursion Principle. To define $f : \mathbb{S}^1 \to A$ we need to define $a : A$ and $p : a = a$.

The suspension

Example. The suspension ΣA

$\texttt{HIT}\ \Sigma A :=$

- $\mathrm{north}, \mathrm{south} : \Sigma A$
- $\mathrm{merid} : A \to (\mathrm{north} = \mathrm{south})$



Remark. $\mathbb{S}^1 \simeq \Sigma \mathbf{2}$

Definition. We can now define the n-spheres by $\mathbb{S}^{n+1} := \Sigma \mathbb{S}^n$ and $\mathbb{S}^0 := \mathbf{2}$

Homotopy Groups

In algebraic topology, we look for algebraic invariants of spaces, like the *homotopy groups*.

Traditionally: $\pi_n(A, a_0) = \{f : \mathbb{S}^n \to A \mid f \text{ preserves basepoints}\}/\sim$.

In HoTT $\pi_n(A, a_0) = \|\mathbb{S}^n \to^* A\|_0$ where we use $\to^*$ for basepoint preserving maps.

Alternative characterization: $\pi_n(A, a_0) = \|\Omega^n(A, a_0)\|_0$ where $\Omega(A, a_0) = (a_0 = a_0, \mathrm{refl}_{a_0})$.

These are groups for $n \geq 1$ (abelian for $n \geq 2$).

Connectedness and truncatedness

If X is n-truncated then $\pi_k(X) = 0$ for all $k > n$.

The converse is not true in general.

Definition. A type A is *n-connected* if $\|A\|_n$ is contractible.

Remark. (-1)-connected: merely inhabited;
0-connected: path-connected;
1-connected: simply connected.

X is n-connected if and only if $\pi_k(X) = 0$ for all $k \leq n$.

If X is n-connected, then ΣX is $(n+1)$-connected.

Thus the n-sphere $\mathbb{S}^n$ is $(n-1)$-connected.

Homotopy Groups of spheres

	$\mathbb{S}^0$	$\mathbb{S}^1$	$\mathbb{S}^2$	$\mathbb{S}^3$	$\mathbb{S}^4$	$\mathbb{S}^5$	$\mathbb{S}^6$	$\mathbb{S}^7$	$\mathbb{S}^8$
π_1	0	$\mathbb{Z}$	0	0	0	0	0	0	0
π_2	0	0	$\mathbb{Z}$	0	0	0	0	0	0
π_3	0	0	$\mathbb{Z}$	$\mathbb{Z}$	0	0	0	0	0
π_4	0	0	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}$	0	0	0	0
π_5	0	0	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}$	0	0	0
π_6	0	0	$\mathbb{Z}_{12}$	$\mathbb{Z}_{12}$	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}$	0	0
π_7	0	0	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}\times\mathbb{Z}_{12}$	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}$	0
π_8	0	0	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}_2^2$	$\mathbb{Z}_{24}$	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}$
π_9	0	0	$\mathbb{Z}_3$	$\mathbb{Z}_3$	$\mathbb{Z}_2^2$	$\mathbb{Z}_2$	$\mathbb{Z}_{24}$	$\mathbb{Z}_2$	$\mathbb{Z}_2$
π_{10}	0	0	$\mathbb{Z}_{15}$	$\mathbb{Z}_{15}$	$\mathbb{Z}_{24}\times\mathbb{Z}_3$	$\mathbb{Z}_2$	0	$\mathbb{Z}_{24}$	$\mathbb{Z}_2$
π_{11}	0	0	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}_{15}$	$\mathbb{Z}_2$	$\mathbb{Z}$	0	$\mathbb{Z}_{24}$
π_{12}	0	0	$\mathbb{Z}_2^2$	$\mathbb{Z}_2^2$	$\mathbb{Z}_2$	$\mathbb{Z}_{30}$	$\mathbb{Z}_2$	0	0
π_{13}	0	0	$\mathbb{Z}_{12}\times\mathbb{Z}_2$	$\mathbb{Z}_{12}\times\mathbb{Z}_2$	$\mathbb{Z}_2^3$	$\mathbb{Z}_2$	$\mathbb{Z}_{60}$	$\mathbb{Z}_2$	0

Homotopy Groups of spheres

	$\mathbb{S}^0$	$\mathbb{S}^1$	$\mathbb{S}^2$	$\mathbb{S}^3$	$\mathbb{S}^4$	$\mathbb{S}^5$	$\mathbb{S}^6$	$\mathbb{S}^7$	$\mathbb{S}^8$
π_1	0	$\mathbb{Z}$	0	0	0	0	0	0	0
π_2	0	0	$\mathbb{Z}$	0	0	0	0	0	0
π_3	0	0	$\mathbb{Z}$	$\mathbb{Z}$	0	0	0	0	0
π_4	0	0	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}$	0	0	0	0
π_5	0	0	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}$	0	0	0
π_6	0	0	$\mathbb{Z}_{12}$	$\mathbb{Z}_{12}$	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}$	0	0
π_7	0	0	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}\times\mathbb{Z}_{12}$	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}$	0
π_8	0	0	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}_2^2$	$\mathbb{Z}_{24}$	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}$
π_9	0	0	$\mathbb{Z}_3$	$\mathbb{Z}_3$	$\mathbb{Z}_2^2$	$\mathbb{Z}_2$	$\mathbb{Z}_{24}$	$\mathbb{Z}_2$	$\mathbb{Z}_2$
π_{10}	0	0	$\mathbb{Z}_{15}$	$\mathbb{Z}_{15}$	$\mathbb{Z}_{24}\times\mathbb{Z}_3$	$\mathbb{Z}_2$	0	$\mathbb{Z}_{24}$	$\mathbb{Z}_2$
π_{11}	0	0	$\mathbb{Z}_2$	$\mathbb{Z}_2$	$\mathbb{Z}_{15}$	$\mathbb{Z}_2$	$\mathbb{Z}$	0	$\mathbb{Z}_{24}$
π_{12}	0	0	$\mathbb{Z}_2^2$	$\mathbb{Z}_2^2$	$\mathbb{Z}_2$	$\mathbb{Z}_{30}$	$\mathbb{Z}_2$	0	0
π_{13}	0	0	$\mathbb{Z}_{12}\times\mathbb{Z}_2$	$\mathbb{Z}_{12}\times\mathbb{Z}_2$	$\mathbb{Z}_2^3$	$\mathbb{Z}_2$	$\mathbb{Z}_{60}$	$\mathbb{Z}_2$	0

Eilenberg MacLane spaces

Question. Can we construct spaces with simple homotopy groups?

In classical homotopy theory, the *Eilenberg MacLane space* $K(G,n)$ is the *unique* space such that

$$\pi_n(K(G,n)) = \begin{cases} G & \text{if } n = k \\ 0 & \text{if } n \neq k. \end{cases}$$

We have already seen one example $K(\mathbb{Z},1) = \mathbb{S}^1$.

Eilenberg-MacLane spaces classify homology and cohomology.

These can be constructed in HoTT [Licata, Finster, 2014].

We write $\mathrm{Type}_*^{=n}$ for the universe of pointed $(n-1)$-connected n-truncated types. We want to construct $K(G,n) : \mathrm{Type}_*^{=n}$.

Eilenberg MacLane space $K(G,1)$

For $n = 1$, suppose G is a group.

`HIT` $\widetilde{K}(G,1) :=$

- $\star : \widetilde{K}(G,1)$
- $\mathrm{pth} : G \to (\star = \star)$
- $\mathrm{pth\text{-}mul} : \Pi(g\ h : G).\ \mathrm{pth}(gh) = \mathrm{pth}(g) \cdot \mathrm{pth}(h)$

$\widetilde{K}(G,1)$ is not quite an Eilenberg-MacLane space; it has nontrivial higher structure.

$$K(G,1) := \|\widetilde{K}(G,1)\|_1$$

$K(G,1)$ is 0-connected, 1-truncated (it lives in $\mathrm{Type}_*^{=1}$) and one can show that $\pi_1(K(G,1)) = G$.

Eilenberg MacLane space $K(G,n)$

Suppose $n \geq 1$. We want to construct $K(G,n+1)$ out of $K(G,n)$. This only works if G is abelian.

Definition. $K(G,n+1) :\equiv \|\Sigma K(G,n)\|_{n+1}$

Now $K(G,n+1)$ is indeed n-connected and $(n+1)$-truncated (it lives in $\mathrm{Type}_*^{=n+1}$). We can show that $\Omega K(G,n+1) = K(G,n)$ (if G is abelian). Hence

$$\begin{aligned}
\Omega^{n+1} K(G,n+1) &= \Omega^n \Omega K(G,n+1) \\
&= \Omega^n K(G,n) \\
&= G
\end{aligned}$$

So $K(G,n)$ has the right homotopy groups.

Main result

Theorem. Any $X : \mathrm{Type}_*^{=n}$ is equivalent to $K(\pi_n(X), n)$.
Moreover, $K(-, n)$, interpreted as a functor from $\mathrm{AbGrp} \to \mathrm{Type}_*^{=n}$ is an equivalence of categories for $n \geq 2$.
For $n = 1$ it is an equivalence of categories $\mathrm{Grp} \to \mathrm{Type}_*^{=1}$.

This means that not only every $X : \mathrm{Type}_*^{=n}$ is an Eilenberg-MacLane space, but also any map $f : X \to Y$ is given by the action of a unique group homomorphism on Eilenberg MacLane spaces.

Special case: uniqueness of $K(G,1)$

As a special case we show: if $(X, x_0) : \mathrm{Type}_*^{=1}$ and we have a group isomorphism $e : G \simeq \pi_1(X)$ then $K(G,1) \simeq X$.

`HIT` $\widetilde{K}(G,1) :=$

- $\star : \widetilde{K}(G,1)$
- $\mathrm{pth} : G \to \star = \star$
- $\text{pth-mul} : \Pi(g\ h : G).\ \mathrm{pth}(gh) = \mathrm{pth}(g) \cdot \mathrm{pth}(h)$

$K(G,1) = \|\widetilde{K}(G,1)\|_1$

Recursion Principle. To define $f : K(G,1) \to A$ for a 1-type A we need $a : A$ and $p : G \to a = a$ such that $p(gh) = p(g) \cdot p(h)$.

We define a map $f : K(G,1) \to X$ by sending $\star$ to x_0, $\mathrm{pth}(g)$ to $e(g)$, viewed as element of ΩX, and $e(gh) = e(g)e(h)$ because e is a group homomorphism. Is f an equivalence?

Special case: uniqueness of $K(G, 1)$

f induces an isomorphism on $\pi_k(K(G, 1)) \to \pi_k(X)$ for all k (trivially for $k \neq 1$).
Such an f is called a *weak equivalence*.
For $k = 1$, we use that the following triangle commutes:

$$\begin{tikzcd}
\pi_1(K(G,1)) \arrow[rr, "\pi_1(f)"] \arrow[dr, equal] & & \pi_1(X) \arrow[dl, equal, "e"] \\
& G &
\end{tikzcd}$$

Theorem. (Whitehead) If $g : A \to B$ is a weak equivalence, A and B are n-types for some n, then f is an equivalence.

Hence $f : K(G, 1) \to X$ is an equivalence.

Lean

This result is formally proven in the proof assistant *Lean*.

Lean is a new open source proof assistant with support for HoTT, similar to Coq and Agda.

Lean implements dependent type theory with a hierarchy of (non-cumulative) universes and inductive types (à la Dybjer, with constructors and recursors).

The kernel is smaller and simpler than those of Coq and Agda.

Lean has two modes: a standard mode for classical and constructive reasoning and a HoTT mode for Homotopy Type Theory.

HoTT library

The Lean HoTT library contains an extensive collection of basic concepts, and the following results were formalized:

- The Freudenthal Suspension Theorem
- The Hopf fibration
- The long exact sequence of homotopy groups
- The Seifert-van Kampen theorem
- The adjunction between the smash product and pointed maps
- Eilenberg MacLane spaces

Currently I'm working in a group project to formalize spectral sequences in Lean.

Code snippets

```
definition KG1_map {G : Group} {X : Type*} (e : G → Ω X)
  (r : Πg h, e (g * h) = e g · e h) [is_conn 0 X] [is_trunc 1 X]
  : K G 1 → X :=
begin
  intro x, induction x using EM.elim,
  { exact Point X },
  { exact e g },
  { exact r g h }
end

definition Grp_equivalence : Grp ≃c cType*[1] :=
equivalence.mk EM1_cfunctor is_equivalence_EM1_cfunctor

definition AbGrp_equivalence (n : ℕ) : AbGrp ≃c cType*[n+2] :=
equivalence.mk (EM_cfunctor (n+2)) (is_equivalence_EM_cfunctor n)
```

Conclusion

Advantages of Synthetic homotopy theory:

- More general
  - There are multiple models of HoTT;
- The homotopy theoretic notions are primitives in type theory
  - We don't have to talk about topology, continuity, ...
- Novel ways of reasoning
  - Path induction, homotopy invariance;
- Constructive (but not anti-classical)
  - Has computational interpretation;
- Possible to verify formally in practice
  - Proof fully formalized in Lean.

Thank you

The Lean HoTT library is available at:

https://github.com/leanprover/lean2/blob/master/hott/hott.md