Cohomology in Cubical Type Theory and Agda

Axel Ljungström

Table of Contents

Introduction

- Cohomology is a powerful tool for characterising topological spaces
- A fair bit of cohomology theory been done in Homotopy Type Theory (HoTT):
  - Licata & Finster (2014)
  - Cavallo (2015)
  - Brunerie (2016)
  - Buchholtz & Favonia (2018)
  - van Doorn (2018)
- I have been working on integer cohomology in Cubical Type Theory (CuTT)

Introduction

- In CuTT, univalence has computational content, an thus we should be able to carry out computations relating to cohomology (e.g. in Cubical Agda)
- Problem with previous work: Either too general or not optimised enough for this to be feasible

Introduction

- In CuTT, univalence has computational content, an thus we should be able to carry out computations relating to cohomology (e.g. in Cubical Agda)
- Problem with previous work: Either too general or not optimised enough for this to be feasible
- In this talk, I will show how one can do integer cohomology in CuTT fairly directly, without having to rely on
  - The Mayer-Vietrois sequence
  - The Freudenthal suspension theorem
  - Theory about connected types and functions

Preliminaries – Paths

- CuTT comes with a primitive interval type I
- This type has two constructors, i_0 and i_1. These are to be though of as 0 and 1 on the unit interval.
- Given a two points $x, y : A$, the type of paths from x to y is denoted by $x \equiv y$

Preliminaries – Paths

- CuTT comes with a primitive interval type I
- This type has two constructors, i_0 and i_1. These are to be though of as 0 and 1 on the unit interval.
- Given a two points $x, y : A$, the type of paths from x to y is denoted by $x \equiv y$
- We construct an element of this type by providing a function $f : I \to A$ such that $f(i_0) := x$ and $f(i_1) := y$
- Here $:=$ denotes definitional equality

Preliminaries – Paths

- CuTT comes with a primitive interval type I
- This type has two constructors, i_0 and i_1. These are to be though of as 0 and 1 on the unit interval.
- Given a two points $x, y : A$, the type of paths from x to y is denoted by $x \equiv y$
- We construct an element of this type by providing a function $f : I \to A$ such that $f(i_0) \coloneqq x$ and $f(i_1) \coloneqq y$
- Here $\coloneqq$ denotes definitional equality
- If A itself is a type depending on I, so that e.g. $x : A(i_0)$ and $y : A(i_1)$ and f is a dependent function $(i : I) \to A(i)$, we say that we have a dependent path from x to y over A

Preliminaries – Function application/cong

- I will use cong for the construction usually referred to as ap in e.g. the HoTT book. Recall, this is just the function

$$\begin{aligned} \mathsf{cong}_f &: x \equiv y \to f(x) \equiv f(y) \\ \mathsf{cong}_f(p) &:= \lambda\, i \,.\, f(p(i)) \end{aligned}$$

  where $f : A \to B$ and $x, y : A$.

Preliminaries – Function application/cong

- I will use cong for the construction usually referred to as ap in e.g. the HoTT book. Recall, this is just the function

$$\begin{aligned} \mathrm{cong}_f &: x \equiv y \to f(x) \equiv f(y) \\ \mathrm{cong}_f(p) &:= \lambda\, i\,.\, f(p(i)) \end{aligned}$$

  where $f : A \to B$ and $x, y : A$.
- Given a binary function $g : A \to B \to C$, with $x, y : A$ and $z, w : B$, there is a binary version of cong which will be used often.

$$\begin{aligned} \mathrm{cong}^2_g &: x \equiv y \to z \equiv w \to g(x, z) \equiv g(y, w) \\ \mathrm{cong}^2_g(p, q) &:= \lambda\, i\,.\, g(p(i), q(i)) \end{aligned}$$

Preliminaries – Suspensions

Definition 1

The suspension of a type A, denoted $\mathsf{Susp}(A)$, is a HIT with the following constructors

- Two points $\mathsf{north}, \mathsf{south} : \mathsf{Susp}(A)$
- For every $x : A$, a path $\mathsf{merid}(x) : \mathsf{north} \equiv \mathsf{south}$

Definition 2

- We define the n-sphere by induction on $n \geq 0$.
- For $n = 0$, we let $\mathbb{S}^0 :\equiv \text{Bool}$.
- For $n = 1$, it is the HIT with the following constructors.
  - A point $\text{base} : \mathbb{S}^1$
  - A loop $\text{loop} : \text{base} \equiv \text{base}$
- For $n > 1$, we define it by

$$\mathbb{S}^n :\equiv \text{Susp}(\mathbb{S}^{n-1})$$

Preliminaries – Loop spaces

Definition 3

Given a pointed type $(A, *_A)$ and an integer $n \geq 1$, we define its n:th loop space $\Omega^n A$ by induction on n.

- $\Omega^1 A :\equiv (*_A \equiv *_A)$. This is itself a pointed type, pointed by refl_{*_A}.
- $\Omega^{n+1} A :\equiv \Omega^1(\Omega^n A)$.

For $n = 1$, we drop the superscript and simply write ΩA.

Preliminaries – Truncations

Definition 4

Given a type A and an integer $n \geq -1$, its n-truncation, denoted by $\|A\|_n$, is a HIT with constructors

- For $x : A$, a term $|x| : \|A\|_n$
- For every function $f : \mathbb{S}^{n+1} \to \|A\|_n$, a term $\mathrm{hub}_f : \|A\|_n$
- For every function $f : \mathbb{S}^{n+1} \to \|A\|_n$ and point $x : \mathbb{S}^{n+1}$, a path $\mathrm{spoke}_f(x) : f(x) \equiv \mathrm{hub}_f$.

$\|-\|_n$ turns a type into an n-type. That is, a type such that all loop spaces of dimension $\geq n$ are contractible.

Cohomology

- Cohomology in HoTT is defined using Eilenberg-MacLane spaces.

Definition 5

Given an abelian group G, its n:th Eilenberg-MacLane space is the (unique) space $K_n(G)$ satisfying:

- $\Omega^n(K_n(G)) \simeq G$
- $\|K_n(G)\|_{n-1}$ is trivial – that is, $K_n(G)$ is $(n-1)$-*connected*.

Cohomology

- Classically, we may identify the n:th cohomology group (with coefficients in a group G) of a CW-complex A with the space

$$\langle A \to K_n(G) \rangle$$

  That is, the space of homotopy classes of functions from A to $K_n(G)$
- In HoTT, we take this as our definition of cohomology
  - The classical definition of (singular) cohomology uses notions that are not homotopy invariant, so we have no choice

Definition 6 (Cohomology with coefficients in G)

Given a type A, its n:th cohomology group with coefficients in G is defined by

$$H^n(A, G) :\equiv \|A \to K_n(G)\|_0$$

Table of Contents

Integer cohomology

- In this talk, we consider the case when $G = \mathbb{Z}$.

Definition 7

Given a type A, its n:th cohomology group with coefficients in $\mathbb{Z}$ is defined by

$$H^n(A) :\equiv \|A \to K_n\|_0$$

where K_n is the family of types defined by

$$K_0 :\equiv \mathbb{Z}$$
$$K_{n+1} :\equiv \left\|\mathbb{S}^{n+1}\right\|_{n+1}$$

Integer cohomology

- We define the base point, $0_k : K_n$ by

$$
\begin{aligned}
0_k &:= 0 && \text{if } n = 0 \\
0_k &:= |\ast_{\mathbb{S}^n}| && \text{if } n \geq 1
\end{aligned}
$$

Connectedness of K_n

- We need to show that our spaces K_n satisfy the axioms of Eilenberg-MacLane spaces. The connectivity criterion is immediate by definition of n-truncations.

Connectedness of K_n

- We need to show that our spaces K_n satisfy the axioms of Eilenberg-MacLane spaces. The connectivity criterion is immediate by definition of n-truncations.

Proposition 8

$\mathbb{S}^n$ is $(n-1)$-connected for each $n \geq 1$.

Connectedness of K_n

- We need to show that our spaces K_n satisfy the axioms of Eilenberg-MacLane spaces. The connectivity criterion is immediate by definition of n-truncations.

Proposition 8

$\mathbb{S}^n$ is $(n-1)$-connected for each $n \geq 1$.

Proof.

- We want to show that $\|\mathbb{S}^n\|_{n-1}$ is contractible.
- The definition of $\|A\|_{n-1}$ tells us that every map from $\mathbb{S}^n$ into $\|A\|_{n-1}$ is constant. In particular, this means that the constructor $|-|\colon \mathbb{S}^n \to \|\mathbb{S}^n\|_{n-1}$ is constant.
- Thus, $\|\mathbb{S}^n\|_{n-1}$ is contractible.

$\square$

Connectedness of K_n

Corollary 9

K_n is $(n-1)$-connected for each $n \geq 1$.

Proof.

We have

$$\begin{aligned} \|K_n\|_{n-1} &:\equiv \|\|\mathbb{S}^n\|_n\|_{n-1} \\ &\equiv \|\mathbb{S}^n\|_{n-1} \end{aligned}$$

which is contractible by Proposition 8. $\square$

The second criterion

- We still need to show that $\Omega^n K_n \simeq \mathbb{Z}$
- We have $\Omega K_1 \simeq \Omega \mathbb{S}^1 \simeq \mathbb{Z}$, so the missing component is a proof that $\Omega K_{n+1} \simeq K_n$

The second criterion

- We still need to show that $\Omega^n K_n \simeq \mathbb{Z}$
- We have $\Omega K_1 \simeq \Omega \mathbb{S}^1 \simeq \mathbb{Z}$, so the missing component is a proof that $\Omega K_{n+1} \simeq K_n$
- In e.g. Brunerie (2016) the equivalence $\Omega K_{n+1} \simeq K_n$ is used to define the group structure on $H^n(A)$
  - K_n inherits a group structure from (path composition in) ΩK_{n+1}
  - $A \to K_n$ inherits this group structure by pointwise addition

The second criterion

- In this talk, we do things the other way around

The second criterion

- In this talk, we do things the other way around
  - First, define a commutative addition operation $+_k : K_n \times K_n \to K_n$.

The second criterion

- In this talk, we do things the other way around
  - First, define a commutative addition operation $+_k : K_n \times K_n \to K_n$.
  - Then, use this to get $\Omega K_{n+1} \simeq K_n$ "for free".

The second criterion

- In this talk, we do things the other way around
  - First, define a commutative addition operation $+_k : K_n \times K_n \to K_n$.
  - Then, use this to get $\Omega K_{n+1} \simeq K_n$ "for free".
- In order to define $+_k$, we give a more direct proof of a special case of the *Wedge Connectivity Lemma* (Lemma 8.6.2 in the HoTT book).

Wedge connectivity lemma for spheres

Lemma 10

Let $n, m \geq 1$ *and suppose we have a fibration* $P : \mathbb{S}^n \times \mathbb{S}^m \to (n+m-2)-$Type *together with functions*

$$f_l : (x : \mathbb{S}^n) \to P(x, *)$$
$$f_r : (y : \mathbb{S}^m) \to P(*, y)$$

and a path

$$p : f_l(*) \equiv f_r(*)$$

Wedge connectivity lemma for spheres

Then there is a function

$$f : ((x, y) : \mathbb{S}^n \times \mathbb{S}^m) \to P(x, y)$$

with homotopies

$$\mathsf{left} : (x : \mathbb{S}^n) \to f(x, *) \equiv f_l(x) \tag{1}$$
$$\mathsf{right} : (y : \mathbb{S}^m) \to f(*, y) \equiv f_r(y) \tag{2}$$

such that

$$p \equiv \mathsf{left}(*)^{-1} \cdot \mathsf{right}(*)$$

Furthermore, either left *or* right *will hold definitonally.*

Proof Sketch

- The proof proceeds by induction on n and m

Proof Sketch

- The proof proceeds by induction on n and m
- The case when $n = m = 1$, the function f is easily defined by pattern matching on $\mathbb{S}^1$, sending

$$\begin{aligned}
(x, \text{base}) &\mapsto f_l(x) \\
(\text{base}, \text{loop}(i)) &\mapsto (p \cdot' \text{cong}_{f_r}(\text{loop}) \cdot' p^{-1})(i) \\
(\text{loop}(i), \text{loop}(j)) &\mapsto Q(i, j)
\end{aligned}$$

  where $\cdot'$ is dependent path composition and Q is given for free, using the fact that P is 0-truncated

Proof Sketch

- The proof proceeds by induction on n and m
- The case when $n = m = 1$, the function f is easily defined by pattern matching on $\mathbb{S}^1$, sending

$$
\begin{aligned}
(x, \mathrm{base}) &\mapsto f_l(x) \\
(\mathrm{base}, \mathrm{loop}(i)) &\mapsto (p \cdot' \mathrm{cong}_{f_r}(\mathrm{loop}) \cdot' p^{-1})(i) \\
(\mathrm{loop}(i), \mathrm{loop}(j)) &\mapsto Q(i, j)
\end{aligned}
$$

  where $\cdot'$ is dependent path composition and Q is given for free, using the fact that P is 0-truncated
- The idea for larger values of n and m is to repeat the above pattern and construct the path corresponding to the loop-cases above (which will now be $\mathrm{merid}(a)$) by applying the induction hypothesis

Group structure on K_n

- Using this lemma, we get an immediate definition of $+_k : K_n \times K_n \to K_n$ when $n \geq 2$:
- By truncation elimination, it suffices to provide a map $\mathbb{S}^n \times \mathbb{S}^n \to K_n$
- K_n is an n-type, and thus by wedge connectivity, we only need to define maps

$$f_l, f_r : \mathbb{S}^n \to K_n$$

and prove that they agree on $*_{\mathbb{S}^n}$

Group structure on K_n

- In both cases, we simply choose the inclusion map $|-| \colon \mathbb{S}^n \to K_n$. The fact that these agree on $*$ now holds trivially.
- This certainly looks like a naive choice of maps, but it provably agrees with with the addition on K_n defined in Brunerie (2016)
  - In fact, any two h-structures on K_n must be equal

Group structure on K_n

- Wedge connectivity breaks down for the special case $n = 1$. Fortunately, the addition here is easy to define explicitly by induction on $\mathbb{S}^1$:

Group structure on K_n

- Wedge connectivity breaks down for the special case $n = 1$. Fortunately, the addition here is easy to define explicitly by induction on $\mathbb{S}^1$:

$$|\,x\,| +_k |\,\mathrm{base}\,| := |\,x\,|$$

Group structure on K_n

- Wedge connectivity breaks down for the special case $n = 1$. Fortunately, the addition here is easy to define explicitly by induction on $\mathbb{S}^1$:

$$
\begin{aligned}
|\,x\,| +_k |\,\mathsf{base}\,| &:= |\,x\,| \\
|\,\mathsf{base}\,| +_k |\,\mathsf{loop}(j)\,| &:= |\,\mathsf{loop}(j)\,|
\end{aligned}
$$

Group structure on K_n

- Wedge connectivity breaks down for the special case $n = 1$. Fortunately, the addition here is easy to define explicitly by induction on $\mathbb{S}^1$:

$$
\begin{aligned}
|\,x\,| +_k |\,\mathsf{base}\,| &:= |\,x\,| \\
|\,\mathsf{base}\,| +_k |\,\mathsf{loop}(j)\,| &:= |\,\mathsf{loop}(j)\,| \\
|\,\mathsf{loop}(i)\,| +_k |\,\mathsf{loop}(j)\,| &:= P(i,j)
\end{aligned}
$$

  where P is a suitably chosen filler of the square

$$
\begin{tikzcd}
{} \arrow[r, "\mathsf{loop}"] & {} \\
{} \arrow[u, "\mathsf{loop}"] \arrow[r, "\mathsf{loop}"'] & {} \arrow[u, "\mathsf{loop}"']
\end{tikzcd}
$$

Group structure on K_n

- For $n = 0$, $+_k$ is just regular integer addition
- With this addition, the group laws are very easy to prove using wedge connectivity and definitional equalities
- For instance, we can prove that $+_k$ is commutative directly and without any reference to Eckmann-Hilton

Proposition 11

For any $n \geq 0$ and $x, y : K_n$, we have that

$$x +_k y \equiv y +_k x$$

Proof

The case when $n = 0$ is trivial, so let us assume that $n \geq 1$.

- The goal type is an $(n-1)$-type. Hence, we may apply the wedge-connectivity lemma.

Proof (contd.)

- We have, due to reductions:

$$\begin{aligned} \mathsf{refl} &: |x| +_k 0_k \equiv |x| \\ \mathsf{right}(x) &: 0_k +_k |x| \equiv |x| \end{aligned}$$

- It remains to show that these two paths are equal at $*$. This holds by definition.

$\square$

Commutativity

- The remaining group laws are proved in a similar manner
- Subtraction is easily constructed using the fact that $+_k$ is an equivalence (since it is just the identity at 0_k)
- Before we continue, we look at some other consequences of this new operation

Lemma 12

For any $p, q : \Omega K_n$, we have $p \cdot q \equiv \mathrm{cong}^2_{+_k}(p, q)$

Proof.

Since $p, q : 0_k \equiv 0_k$ and $0_k +_k 0_k \equiv 0_k$ holds definitionally, we have, by the right- and left-unit laws that

$$p \equiv \mathrm{cong}_{\lambda x \,.\, x +_k 0_k}(p)$$
$$q \equiv \mathrm{cong}_{\lambda y \,.\, 0_k +_k y}(q)$$

By functoriality of cong^2, we have that

$$\mathrm{cong}_{\lambda x \,.\, x +_k 0_k}(p) \cdot \mathrm{cong}_{\lambda y \,.\, 0_k +_k y}(q) \equiv \mathrm{cong}^2_{+_k}(p, q)$$

and we are done. $\square$

Lemma 13

For any $p, q : \Omega K_n$, *we have that* $\mathrm{cong}^2_{+_k}(p, q) \equiv \mathrm{cong}^2_{+_k}(q, p)$

Proof.

Proved like Lemma 12, abusing definitional equalities and commutativity of $+_k$. $\square$

Corollary 14

ΩK_n *is commutative w.r.t. path composition.*

$\Omega K_{n+1} \simeq K_n$

- Our new addition also simplifies the proof of $\Omega K_{n+1} \simeq K_n$ significantly.

Theorem 15

For any $n \geq 1$, we have $\Omega K_{n+1} \simeq K_n$.

Proof

The proof is by the encode-decode method. It boils down to showing that the map $\sigma : K_n \to \Omega K_{n+1}$ (defined below) is an equivalence.

$$\sigma(|\,a\,|) :\equiv \mathrm{cong}_{\lambda\, x\,.\,|x|}(\mathrm{merid}(a) \cdot \mathrm{merid}(*)^{-1})$$

$\Omega K_{n+1} \simeq K_n$

- The proof is by the encode-decode method. We begin by defining the code fibration $\mathrm{Code} : K_{n+1} \to n\text{-Type}$. Since n-Type is an $(n+1)$-type, we may do this by truncation elimination.

$$\begin{aligned}
\mathrm{Code}(|\,\mathrm{north}\,|) &:= K_n \\
\mathrm{Code}(|\,\mathrm{south}\,|) &:= K_n \\
\mathrm{Code}(|\,(\mathrm{merid}(a))(i)\,|) &:= (\mathrm{ua}(\lambda\, y\,.|\,a\,| +_k y))(i)
\end{aligned}$$

using that $\lambda\, y\,.\, x +_k y$ is an equivalence for any $x : K_n$.

Cohomology group structure

- The encode function $\mathrm{encode}_x : 0_k \equiv x \to \mathrm{Code}(x)$ is as usual defined by

$$\mathrm{encode}_x(p) := \mathrm{transport}^{\lambda\, i\, .\, \mathrm{Code}(p(i))}(0_k)$$

- The decode function $\mathrm{decode}_x : \mathrm{Code}(x) \to 0_k \equiv x$ is slightly more involved, but by the group laws of K_n things turn out to be pretty straight-forward. We do it by inducting on x so that

$$\mathrm{decode}_{|\mathrm{north}|}(|\, a\,|) := \sigma(|\, a\,|)$$
$$\mathrm{decode}_{|\mathrm{south}|}(|\, a\,|) := \mathrm{cong}_{\lambda\, x\, .\, |x|}(\mathrm{merid}(a))$$

$\Omega K_{n+1} \simeq K_n$

- For the case of $\mathrm{decode}_{|(\mathrm{merid}(b))(i)|}$, the problem essentially reduces to proving that σ is morphism in the sense that it maps addition in K_n to path composition.
- This is easily proved using wedge connectivity.
- Proving that these maps cancel over the base point follows by path induction and some simple algebra over K_n.

$\square$

Cohomology group structure

The group structre on cohomology groups is immediately inherited from the structure on K_n. Given a type A, we define

$$
\begin{aligned}
0_h &: H^n(A) \\
+_h &: H^n(A) \to H^n(A) \to H^n(A) \\
-_h &: H^n(A) \to H^n(A)
\end{aligned}
$$

by

$$
\begin{aligned}
0_h &:= |\, \lambda x \,.\, 0_k \,| \\
|\, f \,| +_h |\, g \,| &:= |\, \lambda x \,.\, f(x) +_k g(x) \,| \\
-_h |\, f \,| &:= |\, \lambda x \,.\, -_k f(x) \,|
\end{aligned}
$$

Table of Contents

Group characterisations

- We are now ready to compute some cohomology groups
  - $H^n(\mathbb{S}^n)$
  - $H^n(A \vee B)$
  - $H^2(\mathbb{K}^2)$
- It turns out that all of these groups can be characterised by means of direct synthetic proofs, rather than by Mayer-Vietoris.
- Note: We only establish type equivalences and not group equivalences.

Proposition 16

For $n \geq 0$, we have $H^n(\mathbb{S}^n) \simeq \mathbb{Z}$.

Proof

We proceed by induction on n. The base case is easy:

$$H^1(\mathbb{S}^1) :\equiv \left\| \mathbb{S}^1 \to K_1 \right\|_0$$

Proposition 16

For $n \geq 0$, we have $H^n(\mathbb{S}^n) \simeq \mathbb{Z}$.

Proof

We proceed by induction on n. The base case is easy:

$$\begin{aligned} H^1(\mathbb{S}^1) &:= \left\| \mathbb{S}^1 \to K_1 \right\|_0 \\ &\simeq \left\| \sum_{x:K_1} x \equiv x \right\|_0 \end{aligned}$$

Proposition 16

For $n \geq 0$, we have $H^n(\mathbb{S}^n) \simeq \mathbb{Z}$.

Proof

We proceed by induction on n. The base case is easy:

$$
\begin{aligned}
H^1(\mathbb{S}^1) &:= \left\| \mathbb{S}^1 \to K_1 \right\|_0 \\
&\simeq \left\| \sum_{x : K_1} x \equiv x \right\|_0 \\
&\simeq \| K_1 \times \Omega K_1 \|_0 && \text{(base change)}
\end{aligned}
$$

Proposition 16

For $n \geq 0$, we have $H^n(\mathbb{S}^n) \simeq \mathbb{Z}$.

Proof

We proceed by induction on n. The base case is easy:

$$\begin{aligned} H^1(\mathbb{S}^1) &:= \left\|\mathbb{S}^1 \to K_1\right\|_0 \\ &\simeq \left\| \sum_{x:K_1} x \equiv x \right\|_0 \\ &\simeq \|K_1 \times \Omega K_1\|_0 && \text{(base change)} \\ &\simeq \|K_1\|_0 \times \|\Omega K_1\|_0 \end{aligned}$$

Proposition 16

For $n \geq 0$, we have $H^n(\mathbb{S}^n) \simeq \mathbb{Z}$.

Proof

We proceed by induction on n. The base case is easy:

$$
\begin{aligned}
H^1(\mathbb{S}^1) &:= \left\|\mathbb{S}^1 \to K_1\right\|_0 \\
&\simeq \left\| \sum_{x:K_1} x \equiv x \right\|_0 \\
&\simeq \|K_1 \times \Omega K_1\|_0 && \text{(base change)} \\
&\simeq \|K_1\|_0 \times \|\Omega K_1\|_0 \\
&\simeq \mathbb{1} \times \mathbb{Z}
\end{aligned}
$$

Proposition 16

For $n \geq 0$, we have $H^n(\mathbb{S}^n) \simeq \mathbb{Z}$.

Proof

We proceed by induction on n. The base case is easy:

$$\begin{aligned}
H^1(\mathbb{S}^1) &:= \left\|\mathbb{S}^1 \to K_1\right\|_0 \\
&\simeq \left\| \sum_{x:K_1} x \equiv x \right\|_0 \\
&\simeq \left\|K_1 \times \Omega K_1\right\|_0 && \text{(base change)} \\
&\simeq \left\|K_1\right\|_0 \times \left\|\Omega K_1\right\|_0 \\
&\simeq \mathbb{1} \times \mathbb{Z} \\
&\simeq \mathbb{Z}
\end{aligned}$$

Proof (contd.)

For the inductive step we have the following

$$H^{n+1}(\mathbb{S}^{n+1}) :\equiv \|\mathsf{Susp}(\mathbb{S}^n) \to K_{n+1}\|_0$$

$\square$

Proof (contd.)

For the inductive step we have the following

$$
\begin{aligned}
H^{n+1}(\mathbb{S}^{n+1}) &:= \|\mathsf{Susp}(\mathbb{S}^n) \to K_{n+1}\|_0 \\
&\simeq \left\| \sum_{x,y:K_{n+1}} (\mathbb{S}^n \to x \equiv y) \right\|_0
\end{aligned}
$$

□

Proof (contd.)

For the inductive step we have the following

$$
\begin{aligned}
H^{n+1}(\mathbb{S}^{n+1}) &:= \|\mathrm{Susp}(\mathbb{S}^n) \to K_{n+1}\|_0 \\
&\simeq \left\| \sum_{x,y:K_{n+1}} (\mathbb{S}^n \to x \equiv y) \right\|_0 \\
&\simeq \| \, K_{n+1} \times K_{n+1} \times (\mathbb{S}^n \to \Omega K_{n+1}) \, \|_0
\end{aligned}
$$

$\square$

Proof (contd.)

For the inductive step we have the following

$$
\begin{aligned}
H^{n+1}(\mathbb{S}^{n+1}) &:= \|\mathrm{Susp}(\mathbb{S}^n) \to K_{n+1}\|_0 \\
&\simeq \left\| \sum_{x,y:K_{n+1}} (\mathbb{S}^n \to x \equiv y) \right\|_0 \\
&\simeq \| \, K_{n+1} \times K_{n+1} \times (\mathbb{S}^n \to \Omega K_{n+1}) \, \|_0 \\
&\simeq \| \, \mathbb{S}^n \to \Omega K_{n+1} \, \|_0
\end{aligned}
$$

□

Proof (contd.)

For the inductive step we have the following

$$
\begin{aligned}
H^{n+1}(\mathbb{S}^{n+1}) &:\equiv \|\mathrm{Susp}(\mathbb{S}^n) \to K_{n+1}\|_0 \\
&\simeq \left\| \sum_{x,y:K_{n+1}} (\mathbb{S}^n \to x \equiv y) \right\|_0 \\
&\simeq \| \, K_{n+1} \times K_{n+1} \times (\mathbb{S}^n \to \Omega K_{n+1}) \, \|_0 \\
&\simeq \| \, \mathbb{S}^n \to \Omega K_{n+1} \, \|_0 \\
&\simeq \| \, \mathbb{S}^n \to K_n \, \|_0
\end{aligned}
$$

$\square$

Proof (contd.)

For the inductive step we have the following

$$
\begin{aligned}
H^{n+1}(\mathbb{S}^{n+1}) &:\equiv \|\mathrm{Susp}(\mathbb{S}^n) \to K_{n+1}\|_0 \\
&\simeq \left\| \sum_{x,y:K_{n+1}} (\mathbb{S}^n \to x \equiv y) \right\|_0 \\
&\simeq \| \, K_{n+1} \times K_{n+1} \times (\mathbb{S}^n \to \Omega K_{n+1}) \, \|_0 \\
&\simeq \| \, \mathbb{S}^n \to \Omega K_{n+1} \, \|_0 \\
&\simeq \| \, \mathbb{S}^n \to K_n \, \|_0 \\
&:\equiv H^n(\mathbb{S}^n) \\
&\simeq \mathbb{Z}
\end{aligned}
$$

□

Wedge sums

Definition 17

Given two pointed types $(A, *_A)$ and $(B, *_B)$, their wedge sum $A \vee B$ is defined as a HIT with the following constructors

- Two inclusion functions $\mathsf{inl} : A \to A \vee B$ and $\mathsf{inr} : B \to A \vee B$.
- A path $\mathsf{push} : \mathsf{inl}(*_A) \equiv \mathsf{inr}(*_B)$

Wedge sums

Definition 17

Given two pointed types $(A, *_A)$ and $(B, *_B)$, their wedge sum $A \vee B$ is defined as a HIT with the following constructors

- Two inclusion functions $\mathsf{inl} : A \to A \vee B$ and $\mathsf{inr} : B \to A \vee B$.
- A path $\mathsf{push} : \mathsf{inl}(*_A) \equiv \mathsf{inr}(*_B)$

Proposition 18

For any pointed types $(A, *_A)$ and $(B, *_B)$, we have $H^n(A \vee B) \simeq H^n(A) \times H^n(B)$ for $n \geq 1$.

Wedge sums

Definition 17

Given two pointed types $(A, *_A)$ and $(B, *_B)$, their wedge sum $A \vee B$ is defined as a HIT with the following constructors

- Two inclusion functions $\mathsf{inl} : A \to A \vee B$ and $\mathsf{inr} : B \to A \vee B$.
- A path $\mathsf{push} : \mathsf{inl}(*_A) \equiv \mathsf{inr}(*_B)$

Proposition 18

For any pointed types $(A, *_A)$ and $(B, *_B)$, we have $H^n(A \vee B) \simeq H^n(A) \times H^n(B)$ for $n \geq 1$.

Proof

- We want to show that

$$\|A \vee B \to K_n\|_0 \simeq \|A \to K_n\|_0 \times \|B \to K_n\|_0$$

Proof (contd.)

- We pick the naive candidate for the left-to-right map F: simply forget about the additional data given by push

Proof (contd.)

- We pick the naive candidate for the left-to-right map F: simply forget about the additional data given by push
- That is, we define $F : \|A \vee B \to K_n\|_0 \to \|A \to K_n\|_0 \times \|B \to K_n\|_0$ by

$$F(|\, f \,|) :\equiv (|\, f \circ \mathsf{inl} \,|, |\, f \circ \mathsf{inr} \,|)$$

Proof (contd.)

- We pick the naive candidate for the left-to-right map F: simply forget about the additional data given by push
- That is, we define $F : \|A \vee B \to K_n\|_0 \to \|A \to K_n\|_0 \times \|B \to K_n\|_0$ by

$$F(|f|) :\equiv (|f \circ \mathsf{inl}|, |f \circ \mathsf{inr}|)$$

- The inverse F^{-1} is somewhat less obvious. Problem : There is no guarantee that two maps $f : A \to K_n$ and $g : B \to K_n$ satisfy $f(*_A) \equiv g(*_B)$.

Proof (contd.)

- We pick the naive candidate for the left-to-right map F: simply forget about the additional data given by push
- That is, we define $F : \|A \vee B \to K_n\|_0 \to \|A \to K_n\|_0 \times \|B \to K_n\|_0$ by

$$F(|\,f\,|) :\equiv (|\,f \circ \mathsf{inl}\,|, |\,f \circ \mathsf{inr}\,|)$$

- The inverse F^{-1} is somewhat less obvious. Problem : There is no guarantee that two maps $f : A \to K_n$ and $g : B \to K_n$ satisfy $f(*_A) \equiv g(*_B)$.
- Solution: perform a suitable base change

Proof (contd.)

- We define $F^{-1} : \|A \to K_n\|_0 \times \|B \to K_n\|_0 \to \|A \vee B \to K_n\|_0$ by

$$F^{-1}(|\, f \,|, |\, g \,|) :\equiv |\, \phi_{f,g} \,|$$

where $\phi_{f,g} : A \vee B \to K_n$ is defined inductively by

Proof (contd.)

- We define $F^{-1} : \|A \to K_n\|_0 \times \|B \to K_n\|_0 \to \|A \vee B \to K_n\|_0$ by

$$F^{-1}(|\, f \,|, |\, g \,|) :=|\, \phi_{f,g} \,|$$

where $\phi_{f,g} : A \vee B \to K_n$ is defined inductively by

$$\phi_{f,g}(\mathrm{inl}(x)) := f(x) +_k g(*_B)$$

Proof (contd.)

- We define $F^{-1} : \|A \to K_n\|_0 \times \|B \to K_n\|_0 \to \|A \vee B \to K_n\|_0$ by

$$F^{-1}(|f|, |g|) :=|\phi_{f,g}|$$

where $\phi_{f,g} : A \vee B \to K_n$ is defined inductively by

$$\phi_{f,g}(\mathrm{inl}(x)) := f(x) +_k g(*_B)$$
$$\phi_{f,g}(\mathrm{inr}(x)) := f(*_A) +_k g(x)$$

Proof (contd.)

- We define $F^{-1} : \|A \to K_n\|_0 \times \|B \to K_n\|_0 \to \|A \vee B \to K_n\|_0$ by

$$F^{-1}(|f|, |g|) := |\phi_{f,g}|$$

where $\phi_{f,g} : A \vee B \to K_n$ is defined inductively by

$$\phi_{f,g}(\mathsf{inl}(x)) := f(x) +_k g(*_B)$$
$$\phi_{f,g}(\mathsf{inr}(x)) := f(*_A) +_k g(x)$$
$$\phi_{f,g}(\mathsf{push}(i)) := f(*_A) +_k g(*_B)$$

Proof (contd.)

- We define $F^{-1} : \|A \to K_n\|_0 \times \|B \to K_n\|_0 \to \|A \vee B \to K_n\|_0$ by

$$F^{-1}(|f|, |g|) :=|\phi_{f,g}|$$

where $\phi_{f,g} : A \vee B \to K_n$ is defined inductively by

$$\phi_{f,g}(\mathrm{inl}(x)) := f(x) +_k g(*_B)$$
$$\phi_{f,g}(\mathrm{inr}(x)) := f(*_A) +_k g(x)$$
$$\phi_{f,g}(\mathrm{push}(i)) := f(*_A) +_k g(*_B)$$

- The fact that $F(F^{-1}(x)) \equiv x$ is easy

Proof (contd.)

- We want to show that

$$F^{-1}(F(\,|f|)) :\equiv |\,\phi_{(f\circ\mathsf{inl}),(f\circ\mathsf{inr})}\,| \equiv |\,f\,|$$

  for any $f : A \vee B \to K_n$
- Since this is a proposition and K_n is 0-connected, we may assume that we have a path

$$p : f(\mathsf{inl}(*_A)) \equiv 0_k$$

- Under this assumption, we prove that $f(x) \equiv \phi_{(f\circ\mathsf{inl}),(f\circ\mathsf{inr})}(x)$ for $x : A \vee B$ by induction on x.

Proof

- For $\mathsf{inl}(x)$, we give the homotopy
  $P_l : (x : A) \to f(\mathsf{inl}(x)) \equiv f(\mathsf{inl}(x)) +_k f(\mathsf{inr}(*_B))$

Proof

- For $\mathsf{inl}(x)$, we give the homotopy $P_l : (x : A) \to f(\mathsf{inl}(x)) \equiv f(\mathsf{inl}(x)) +_k f(\mathsf{inr}(*_B))$ defined by

$$
\begin{aligned}
f(\mathsf{inl}(x)) &\rightsquigarrow f(\mathsf{inl}(x)) +_k 0_k \\
&\rightsquigarrow f(\mathsf{inl}(x)) +_k f(\mathsf{inl}(*_A)) \\
&\rightsquigarrow f(\mathsf{inl}(x)) +_k f(\mathsf{inr}(*_B))
\end{aligned}
$$

Proof

- For $\mathsf{inl}(x)$, we give the homotopy $P_l : (x : A) \to f(\mathsf{inl}(x)) \equiv f(\mathsf{inl}(x)) +_k f(\mathsf{inr}(*_B))$ defined by

$$
\begin{aligned}
f(\mathsf{inl}(x)) &\overset{\mathsf{rUnit}}{\rightsquigarrow} f(\mathsf{inl}(x)) +_k 0_k \\
&\rightsquigarrow f(\mathsf{inl}(x)) +_k f(\mathsf{inl}(*_A)) \\
&\rightsquigarrow f(\mathsf{inl}(x)) +_k f(\mathsf{inr}(*_B))
\end{aligned}
$$

Proof

- For $\mathsf{inl}(x)$, we give the homotopy $P_l : (x : A) \to f(\mathsf{inl}(x)) \equiv f(\mathsf{inl}(x)) +_k f(\mathsf{inr}(*_B))$ defined by

$$
\begin{aligned}
f(\mathsf{inl}(x)) &\overset{\mathsf{rUnit}}{\rightsquigarrow} f(\mathsf{inl}(x)) +_k 0_k \\
&\overset{\mathsf{cong}_{f(\mathsf{inl}(x))+_k\_}(p)}{\rightsquigarrow} f(\mathsf{inl}(x)) +_k f(\mathsf{inl}(*_A)) \\
&\rightsquigarrow f(\mathsf{inl}(x)) +_k f(\mathsf{inr}(*_B))
\end{aligned}
$$

Proof

- For $\mathsf{inl}(x)$, we give the homotopy $P_l : (x : A) \to f(\mathsf{inl}(x)) \equiv f(\mathsf{inl}(x)) +_k f(\mathsf{inr}(*_B))$ defined by

$$
\begin{aligned}
f(\mathsf{inl}(x)) &\xrightarrow{\ \mathsf{rUnit}\ } f(\mathsf{inl}(x)) +_k 0_k \\
&\xrightarrow{\ \mathsf{cong}_{f(\mathsf{inl}(x))+_k\_}(p)\ } f(\mathsf{inl}(x)) +_k f(\mathsf{inl}(*_A)) \\
&\xrightarrow{\ \mathsf{cong}_{f(\mathsf{inl}(x))+_k\_}(\mathsf{push})\ } f(\mathsf{inl}(x)) +_k f(\mathsf{inr}(*_B))
\end{aligned}
$$

Proof

- For $\mathrm{inl}(x)$, we give the homotopy $P_l : (x : A) \to f(\mathrm{inl}(x)) \equiv f(\mathrm{inl}(x)) +_k f(\mathrm{inr}(*_B))$ defined by

$$
\begin{aligned}
f(\mathrm{inl}(x)) &\xrightarrow{\mathrm{rUnit}} f(\mathrm{inl}(x)) +_k 0_k \\
&\xrightarrow{\mathrm{cong}_{f(\mathrm{inl}(x))+_k \_}(p)} f(\mathrm{inl}(x)) +_k f(\mathrm{inl}(*_A)) \\
&\xrightarrow{\mathrm{cong}_{f(\mathrm{inl}(x))+_k \_}(\mathrm{push})} f(\mathrm{inl}(x)) +_k f(\mathrm{inr}(*_B))
\end{aligned}
$$

- For $\mathrm{inr}(x)$, we give the homotopy $P_r : (x : B) \to f(\mathrm{inr}(x)) \equiv f(\mathrm{inl}(*_A)) +_k f(\mathrm{inr}(x))$

Proof

- For $\mathsf{inl}(x)$, we give the homotopy $P_l : (x : A) \to f(\mathsf{inl}(x)) \equiv f(\mathsf{inl}(x)) +_k f(\mathsf{inr}(*_B))$ defined by

$$
\begin{aligned}
f(\mathsf{inl}(x)) &\xrightarrow{\mathsf{rUnit}} f(\mathsf{inl}(x)) +_k 0_k \\
&\xrightarrow{\mathsf{cong}_{f(\mathsf{inl}(x))+_k\_}(p)} f(\mathsf{inl}(x)) +_k f(\mathsf{inl}(*_A)) \\
&\xrightarrow{\mathsf{cong}_{f(\mathsf{inl}(x))+_k\_}(\mathsf{push})} f(\mathsf{inl}(x)) +_k f(\mathsf{inr}(*_B))
\end{aligned}
$$

- For $\mathsf{inr}(x)$, we give the homotopy $P_r : (x : B) \to f(\mathsf{inr}(x)) \equiv f(\mathsf{inl}(*_A)) +_k f(\mathsf{inr}(x))$ defined by

$$
\begin{aligned}
f(\mathsf{inr}(x)) &\rightsquigarrow 0_k +_k f(\mathsf{inr}(x)) \\
&\rightsquigarrow f(\mathsf{inl}(*_A)) +_k f(\mathsf{inr}(x))
\end{aligned}
$$

Proof

- For $\mathsf{inl}(x)$, we give the homotopy $P_l : (x : A) \to f(\mathsf{inl}(x)) \equiv f(\mathsf{inl}(x)) +_k f(\mathsf{inr}(*_B))$ defined by

$$
\begin{aligned}
f(\mathsf{inl}(x)) &\overset{\mathsf{rUnit}}{\rightsquigarrow} f(\mathsf{inl}(x)) +_k 0_k \\
&\overset{\mathsf{cong}_{f(\mathsf{inl}(x))+_k\_}(p)}{\rightsquigarrow} f(\mathsf{inl}(x)) +_k f(\mathsf{inl}(*_A)) \\
&\overset{\mathsf{cong}_{f(\mathsf{inl}(x))+_k\_}(\mathsf{push})}{\rightsquigarrow} f(\mathsf{inl}(x)) +_k f(\mathsf{inr}(*_B))
\end{aligned}
$$

- For $\mathsf{inr}(x)$, we give the homotopy $P_r : (x : B) \to f(\mathsf{inr}(x)) \equiv f(\mathsf{inl}(*_A)) +_k f(\mathsf{inr}(x))$ defined by

$$
\begin{aligned}
f(\mathsf{inr}(x)) &\overset{\mathsf{lUnit}}{\rightsquigarrow} 0_k +_k f(\mathsf{inr}(x)) \\
&\rightsquigarrow f(\mathsf{inl}(*_A)) +_k f(\mathsf{inr}(x))
\end{aligned}
$$

Proof

- For $\mathsf{inl}(x)$, we give the homotopy $P_l : (x : A) \to f(\mathsf{inl}(x)) \equiv f(\mathsf{inl}(x)) +_k f(\mathsf{inr}(*_B))$ defined by

$$
\begin{aligned}
f(\mathsf{inl}(x)) &\xrightarrow{\mathsf{rUnit}} f(\mathsf{inl}(x)) +_k 0_k \\
&\xrightarrow{\mathsf{cong}_{f(\mathsf{inl}(x)) +_k \_}(p)} f(\mathsf{inl}(x)) +_k f(\mathsf{inl}(*_A)) \\
&\xrightarrow{\mathsf{cong}_{f(\mathsf{inl}(x)) +_k \_}(\mathsf{push})} f(\mathsf{inl}(x)) +_k f(\mathsf{inr}(*_B))
\end{aligned}
$$

- For $\mathsf{inr}(x)$, we give the homotopy $P_r : (x : B) \to f(\mathsf{inr}(x)) \equiv f(\mathsf{inl}(*_A)) +_k f(\mathsf{inr}(x))$ defined by

$$
\begin{aligned}
f(\mathsf{inr}(x)) &\xrightarrow{\mathsf{lUnit}} 0_k +_k f(\mathsf{inr}(x)) \\
&\xrightarrow{\mathsf{cong}_{\_ +_k f(\mathsf{inl}(x))}(p)} f(\mathsf{inl}(*_A)) +_k f(\mathsf{inr}(x))
\end{aligned}
$$

Proof (contd.)

- For $f(\mathrm{push}(i))$ we need to fill the following square

$$
\begin{tikzcd}
\cdot \arrow[r, "\mathrm{refl}"] & \cdot \\
\cdot \arrow[u, "P_l(*_A)"] \arrow[r, "\mathrm{cong}_f(\mathrm{push})"'] & \cdot \arrow[u, "P_r(*_B)"']
\end{tikzcd}
$$

Proof (contd.)

- For $f(\mathrm{push}(i))$ we need to fill the following square

$$\begin{tikzcd} {} \arrow[r, "\mathrm{refl}"] & {} \\ {} \arrow[u, "P_l(*_A)"] \arrow[r, "\mathrm{cong}_f(\mathrm{push})"'] & {} \arrow[u, "P_r(*_B)"'] \end{tikzcd}$$

- We may substitute 0_k for $f(\mathrm{inl}(*_A))$ and $f(\mathrm{inr}(*_B))$ (using the paths p and push)

Proof (contd.)

- For $f(\mathrm{push}(i))$ we need to fill the following square

$$\begin{tikzcd} \cdot \arrow[r, "\mathrm{refl}"] & \cdot \\ \cdot \arrow[u, "P_l(*_A)"] \arrow[r, "\mathrm{cong}_f(\mathrm{push})"'] & \cdot \arrow[u, "P_r(*_B)"'] \end{tikzcd}$$

- We may substitute 0_k for $f(\mathrm{inl}(*_A))$ and $f(\mathrm{inr}(*_B))$ (using the paths p and push)
- This will give some nice reductions in $P_l(*_A)$ and $P_r(*_B)$

Proof.

- Exploiting definitional equalities, this substitution reduces the problem to filling the following square

$$
\begin{tikzcd}
\cdot \arrow[r, "\mathsf{refl}"] & \cdot \\
\cdot \arrow[u, "\mathsf{cong}_f(\mathsf{push})"] \arrow[r, "\mathsf{cong}_f(\mathsf{push})"'] & \cdot \arrow[u, "\mathsf{refl}"']
\end{tikzcd}
$$

- Trivial

□

The Klein bottle

Definition 19

The Klein bottle, $\mathbb{K}^2$, is a HIT defined by:

- A point base : $\mathbb{K}^2$
- Two paths ℓ_1, ℓ_2 : base $\equiv$ base
- A filler $\square$ of the following square

$$\begin{tikzcd} {} \arrow[r, "\ell_1"] & {} \\ {} \arrow[u, "\ell_2^{-1}"] \arrow[r, "\ell_1"'] & {} \arrow[u, "\ell_2"'] \end{tikzcd}$$

Alternatively, we may interpret $\square$ as a path $\ell_2 \cdot \ell_1 \cdot \ell_2 \equiv \ell_1$.
For our purposes, this interpretation is more useful.

The Klein bottle

- For any type A, we may characterise the function space $(\mathbb{K}^2 \to A)$ by the nested Σ-type

$$\sum_{x:A} \sum_{p,q:x\equiv x} (p \cdot q \cdot p \equiv q)$$

- When path composition over A is commutative, we have that

$$\begin{aligned} & (p \cdot q \cdot p \equiv q) \\ \simeq\ & (p^2 \cdot q \equiv q) \\ \simeq\ & (p^2 \equiv \mathsf{refl}) \end{aligned}$$

The Klein bottle

- Thus, in particular we have that

$$(\mathbb{K}^2 \to K_n) \simeq \sum_{x:K_n} \sum_{p,q:x \equiv x} (p^2 \equiv \mathsf{refl})$$

- This is the key-component in the proof of the following proposition.

Proposition 20

$H^2(\mathbb{K}^2) \simeq \mathbb{Z}/2\mathbb{Z}$

Proof

- We begin by rewriting $H^2(\mathbb{K}^2)$ in accordance with the previous discussion.

$$H^2(\mathbb{K}^2) :\equiv \left\|\mathbb{K}^2 \to K_2\right\|_0$$
$$\simeq \left\|\sum_{x:K_2} \sum_{p,q:x\equiv x} (p^2 \equiv \mathsf{refl})\right\|_0$$

Proof

- We begin by rewriting $H^2(\mathbb{K}^2)$ in accordance with the previous discussion.

$$H^2(\mathbb{K}^2) :\equiv \left\|\mathbb{K}^2 \to K_2\right\|_0$$
$$\simeq \left\|\sum_{x:K_2} \sum_{p,q:x\equiv x} (p^2 \equiv \mathsf{refl})\right\|_0$$

- For connectedness-reasons, this can be simplified to

$$\left\|\sum_{p:\Omega K_2} (p^2 \equiv \mathsf{refl})\right\|_0$$

The Klein bottle

- Using the equivalence $\sigma : K_n \simeq \Omega K_{n+1}$ and the fact it is a morphism with respect to path composition and $+_k$, the above type is just

$$\left\| \sum_{x:K_1} x +_k x \equiv 0_k \right\|_0$$

The Klein bottle

- Using the equivalence $\sigma : K_n \simeq \Omega K_{n+1}$ and the fact it is a morphism with respect to path composition and $+_k$, the above type is just

$$\left\| \sum_{x:K_1} x +_k x \equiv 0_k \right\|_0$$

- An element $|(x, p)|$ of this type falls into two categories:

The Klein bottle

- Using the equivalence $\sigma : K_n \simeq \Omega K_{n+1}$ and the fact it is a morphism with respect to path composition and $+_k$, the above type is just

$$\left\| \sum_{x : K_1} x +_k x \equiv 0_k \right\|_0$$

- An element $|(x, p)|$ of this type falls into two categories:
  - When $x := 0_k$, p is just a loop in ΩK_1 (since $0_k +_k 0_k := 0_k$). Furthermore, p is equal to loop^k for some $k : \mathbb{Z}$.

The Klein bottle

- Using the equivalence $\sigma : K_n \simeq \Omega K_{n+1}$ and the fact it is a morphism with respect to path composition and $+_k$, the above type is just

$$\left\| \sum_{x:K_1} x +_k x \equiv 0_k \right\|_0$$

- An element $|(x, p)|$ of this type falls into two categories:
  - When $x :\equiv 0_k$, p is just a loop in ΩK_1 (since $0_k +_k 0_k :\equiv 0_k$). Furthermore, p is equal to loop^k for some $k : \mathbb{Z}$.
  - When $x :\equiv \mathsf{loop}(i)$, p is essentially a path $\mathsf{loop} \cdot \mathsf{loop} \equiv \mathsf{refl}$, since $\mathsf{cong}^2_{+_k}(\mathsf{loop}, \mathsf{loop}) :\equiv \mathsf{loop} \cdot \mathsf{loop}$.

The Klein bottle

- It suffices to show that for any element $\alpha : \left\|\sum_{x:K_1} x +_k x \equiv 0_k\right\|_0$ on the form $|0_k, \mathsf{loop}^k|$ we have that
    - $\alpha \equiv |0_k, \mathsf{refl}|$ if k is even
    - $\alpha \equiv |0_k, \mathsf{loop}|$ if k is odd

The Klein bottle

- It suffices to show that for any element $\alpha : \left\|\sum_{x:K_1} x +_k x \equiv 0_k\right\|_0$ on the form $|0_k, \mathsf{loop}^k|$ we have that
  - $\alpha \equiv |0_k, \mathsf{refl}|$ if k is even
  - $\alpha \equiv |0_k, \mathsf{loop}|$ if k is odd
- Idea: induct on k. Let us assume $k \geq 0$
  - $k < 0$ is handled in an entirely symmetric manner

The Klein bottle

- It suffices to show that for any element $\alpha : \left\| \sum_{x:K_1} x +_k x \equiv 0_k \right\|_0$ on the form $|0_k, \mathsf{loop}^k|$ we have that
  - $\alpha \equiv |0_k, \mathsf{refl}|$ if k is even
  - $\alpha \equiv |0_k, \mathsf{loop}|$ if k is odd
- Idea: induct on k. Let us assume $k \geq 0$
  - $k < 0$ is handled in an entirely symmetric manner
- When $k = 0, 1$, the lemma is trivial

The Klein bottle

- For the base case is $k = 2$. We want to show that $|\, 0_k, \mathsf{refl} \,| \equiv |\, 0_k, \mathsf{loop}^2 \,|$. Let us ignore the truncation.

The Klein bottle

- For the base case is $k = 2$. We want to show that $|\, 0_k, \mathsf{refl} \,| \equiv |\, 0_k, \mathsf{loop}^2 \,|$. Let us ignore the truncation.
- We first have to prove that the first components agree – that is, we need a path $0_k \equiv 0_k$. Rather than choosing the tempting refl, we choose $\mathsf{loop} : 0_k \equiv 0_k$.

The Klein bottle

- For the base case is $k = 2$. We want to show that $|\, 0_k, \mathrm{refl}\,| \equiv |\, 0_k, \mathrm{loop}^2\,|$. Let us ignore the truncation.
- We first have to prove that the first components agree – that is, we need a path $0_k \equiv 0_k$. Rather than choosing the tempting refl, we choose $\mathrm{loop} : 0_k \equiv 0_k$.
- We now need to provide a dependent path from refl to loop^2 over the path $\lambda i \,.\, (\mathrm{loop}(i) +_k \mathrm{loop}(i) \equiv 0_k)$. But $\lambda i \,.\, \mathrm{loop}(i) +_k \mathrm{loop}(i)$ is definitionally equal to loop^2, so this becomes trivial.

The Klein bottle

- For the inductive step, the trick is defining a multiplication $\diamond$ on $\left\| \sum_{x:K_1} x +_k x \equiv 0_k \right\|_0$ satisfying

$$|\, 0_k, p \,| \diamond |\, 0_k, q \,| \equiv |\, 0_k, p \cdot q \,|$$

  for any $p, q : \Omega K_1$
- This is easily carried out using the wedge connectivity lemma

- We now have, for every $k \geq 0$

$$
\begin{aligned}
|\, 0_k, \mathsf{loop}^{k+2} \,| &\equiv |\, 0_k, \mathsf{loop}^{k} \,| \diamond |\, 0_k, \mathsf{loop}^{2} \,| \\
&\equiv |\, 0_k, \mathsf{loop}^{k} \,|
\end{aligned}
$$

and thus we are done by the inductive hypothesis

$\square$

The Klein bottle

- We now have, for every $k \geq 0$

$$\begin{aligned} |\, 0_k, \mathsf{loop}^{k+2} \,| &\equiv |\, 0_k, \mathsf{loop}^{k} \,| \diamond |\, 0_k, \mathsf{loop}^{2} \,| \\ &\equiv |\, 0_k, \mathsf{loop}^{k} \,| \end{aligned}$$

  and thus we are done by the inductive hypothesis
- This proves (modulo some handwaving) that there are precisely 2 elements in $\left\| \sum_{x:K_1} x +_k x \equiv 0_k \right\|_0$

$\square$

The Klein bottle

- We now have, for every $k \geq 0$

$$
\begin{aligned}
|\, 0_k, \mathsf{loop}^{k+2} \,| &\equiv |\, 0_k, \mathsf{loop}^{k} \,| \diamond |\, 0_k, \mathsf{loop}^{2} \,| \\
&\equiv |\, 0_k, \mathsf{loop}^{k} \,|
\end{aligned}
$$

  and thus we are done by the inductive hypothesis
- This proves (modulo some handwaving) that there are precisely 2 elements in $\left\| \sum_{x:K_1} x +_k x \equiv 0_k \right\|_0$
- Thus $H^2(\mathbb{K}^2) \simeq \left\| \sum_{x:K_1} x +_k x \equiv 0_k \right\|_0 \simeq \mathbb{Z}/2\mathbb{Z}$

$\square$

The Klein bottle

- We now have, for every $k \geq 0$

$$\begin{aligned} |\,0_k, \mathrm{loop}^{k+2}\,| &\equiv |\,0_k, \mathrm{loop}^{k}\,| \diamond |\,0_k, \mathrm{loop}^{2}\,| \\ &\equiv |\,0_k, \mathrm{loop}^{k}\,| \end{aligned}$$

  and thus we are done by the inductive hypothesis
- This proves (modulo some handwaving) that there are precisely 2 elements in $\left\| \sum_{x:K_1} x +_k x \equiv 0_k \right\|_0$
- Thus $H^2(\mathbb{K}^2) \simeq \left\| \sum_{x:K_1} x +_k x \equiv 0_k \right\|_0 \simeq \mathbb{Z}/2\mathbb{Z}$
- It is easy to show that this map can be turned into a homomorphism, using decidability of $\mathbb{Z}/2\mathbb{Z}$

$\square$

Computations

- Given a characterisation $f : H^n(A) \cong G$, I have run two tests in Cubical Agda:
- Test 1: check whether $f(f^{-1}(x))$ reduces to x (assuming G is a closed type)
- Test 2: check whether $f(f^{-1}(x) +_h f^{-1}(y))$ reduces to $x + y$

Computations

Group	Equiv.	Test 1	Test 2
$H^1(\mathbb{S}^1)$	$\mathbb{Z}$	Fast	Fast
$H^2(\mathbb{S}^2)$	$\mathbb{Z}$	Fast	?/very slow
$H^{n>2}(\mathbb{S}^n)$	$\mathbb{Z}$	?	?

Table: Spheres

Computations

Group	Equiv.	Test 1	Test 2
$H^1(\mathbb{S}^1 \vee \mathbb{S}^1 \vee \mathbb{S}^2)$	$\mathbb{Z} \times \mathbb{Z}$	Fast	Fast
$H^2(\mathbb{S}^1 \vee \mathbb{S}^1 \vee \mathbb{S}^2)$	$\mathbb{Z}$	Ok	?

Table: Wedge sums (of spheres)

Group	Equiv.	Test 1	Test 2
$H^1(\mathbb{T}^2)$	$\mathbb{Z} \times \mathbb{Z}$	Fast	Fast
$H^2(\mathbb{T}^2)$	$\mathbb{Z}$	Fast	?/very slow

Table: Torus

Computations

Group	Equiv.	Test 1	Test 2
$H^1(\mathbb{K}^2)$	$\mathbb{Z}$	Fast	Fast
$H^2(\mathbb{K}^2)$	$\mathbb{Z}/2\mathbb{Z}$	?	?

Table: Klein bottle

Conclusions

- Elementary cohomology theory can be done synthetically in CuTT
- With the right choice of the addition $+_k$ on Eilenberg-MacLane spaces, things turn out to be even easier, due to reductions
- The cubical primitives help make many proofs particularly short
- Path induction is only used in some very specific cases (and even then can often be replaced by more cubical proofs)

Conclusions

- Agda still struggles with computations for dimension $n \geq 2$
- This is perhaps not so surprising – at this stage, the computations are starting to look a lot like those of the Brunerie number
- In contrast to my previous efforts, however, there is a big improvement for $n = 1$ and $n = 2$.
- We get a bunch of new examples of "Brunerie numbers"

Future work

- Define the cup product and try to compute it for the cohomology rings of $\mathbb{T}^2$ and $\mathbb{S}^1 \vee \mathbb{S}^1 \vee \mathbb{S}^2$
- See what can be done about more general cohomology theories from a computational point of view

References I

Guillaume Brunerie. "On the homotopy groups of spheres in homotopy type theory.". In: (2016). URL: https://ezp.sub.su.se/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=edsarx&AN=edsarx.1606.05916&site=eds-live&scope=site.

Ulrik Buchholtz and Kuen-Bang Hou Favonia. "Cellular Cohomology in Homotopy Type Theory". In: *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science*. LICS '18. Oxford, United Kingdom: Association for Computing Machinery, 2018, pp. 521–529. ISBN: 9781450355834. DOI: 10.1145/3209108.3209188. URL: https://doi.org/10.1145/3209108.3209188.

Evan Cavallo. "Synthetic Cohomology in Homotopy Type Theory". In: (2015). Master's thesis.

References II

Floris Doorn. "On the Formalization of Higher Inductive Types and Synthetic Homotopy Theory". In: (Aug. 2018).

Daniel Licata and Eric Finster. "Eilenberg-MacLane spaces in homotopy type theory". In: (July 2014). DOI: 10.1145/2603088.2603153.