Reducing higher inductive types to quotients

Floris van Doorn

Carnegie Mellon University

May 20, 2016

Motivation

Goal: Reduce complicated higher inductive types to simpler ones.

Analogue: In Extensional Type Theory, we can reduce all inductive types to W-types and Σ-types.

Question: What makes a higher inductive type complicated?

- It is recursive (*n*-truncation, localization, spectrification)
- It has higher-dimensional path-constructors (torus, Eilenberg-MacLane spaces $K(G, 1)$)
- Higher inductive-inductive types (Cauchy reals)

Motivation

Goal: Reduce complicated higher inductive types to simpler ones.

Analogue: In Extensional Type Theory, we can reduce all inductive types to W-types and Σ-types.

Question: What makes a higher inductive type complicated?

- It is recursive (n-truncation, localization, spectrification)
- It has higher-dimensional path-constructors (torus, Eilenberg-MacLane spaces $K(G, 1)$)
- Higher inductive-inductive types (Cauchy reals)

Quotients

Question: What higher inductive types do we start with?

Given $A : \mathcal{U}$ and $R : A \to A \to \mathcal{U}$ the quotient is:

`HIT` $\mathit{quotient}_A(R) :=$

- $q : A \to \mathit{quotient}_A(R)$
- $\Pi(x, y : A),\ R(x, y) \to q(x) = q(y)$

This is the homotopy-coequalizer of the projections

$$\Sigma(x, y : A),\ R(x, y) \underset{\pi_2}{\overset{\pi_1}{\rightrightarrows}} A$$

Quotients

Using quotients we can define other simple HITs:

- The pushout of $C \xleftarrow{g} A \xrightarrow{f} B$ is the quotient of $B + C$ under the relation R, defined as a inductively

  $\texttt{inductive}\ R : (B + C) \to (B + C) \to \mathcal{U} :=$

  - $\Pi(a : A),\ R(\mathsf{inl}(f(a)), \mathsf{inr}(g(a)))$

Quotients

Using quotients we can define other simple HITs:

- The pushout of $C \overset{g}{\leftarrow} A \overset{f}{\rightarrow} B$ is the quotient of $B + C$ under the relation R, defined as a inductively

  $\texttt{inductive}\ R : (B + C) \to (B + C) \to \mathcal{U} :=$
  - $\Pi(a : A),\ R(\mathrm{inl}(f(a)), \mathrm{inr}(g(a)))$
- The colimit of $A_0 \overset{f_0}{\rightarrow} A_1 \overset{f_1}{\rightarrow} \cdots$ is the quotient of $\Sigma(n : \mathbb{N}),\ A(n)$ under the relation S, defined inductively

  $\texttt{inductive}\ S : (\Sigma(n : \mathbb{N}),\ A(n)) \to (\Sigma(n : \mathbb{N}),\ A(n)) \to \mathcal{U} :=$
  - $\Pi(n : \mathbb{N})(a : A_n),\ S((n, a), (n + 1, f_n(a)))$

Quotients

Using quotients we can define other simple HITs:

- The pushout of $C \xleftarrow{g} A \xrightarrow{f} B$ is the quotient of $B + C$ under the relation R, defined as a inductively

  inductive $R : (B + C) \to (B + C) \to \mathcal{U} :=$
  - $\Pi(a : A),\ R(\mathrm{inl}(f(a)), \mathrm{inr}(g(a)))$
- The colimit of $A_0 \xrightarrow{f_0} A_1 \xrightarrow{f_1} \cdots$ is the quotient of $\Sigma(n : \mathbb{N}),\ A(n)$ under the relation S, defined inductively

  inductive $S : (\Sigma(n : \mathbb{N}),\ A(n)) \to (\Sigma(n : \mathbb{N}),\ A(n)) \to \mathcal{U} :=$
  - $\Pi(n : \mathbb{N})(a : A_n),\ S((n, a), (n + 1, f_n(a)))$
- This also gives suspensions, spheres, wedge product, join, smash product, cofibers, ...

Quotients

We can construct more HITs from quotients.

Today I will talk about the construction of

- The propositional truncation
- HITs with 2-constructors (torus, groupoid quotient, Eilenberg-MacLane spaces $K(G, 1)$, reduced suspension, reflexive quotient)

Work in progress:

- Define ω-compact localizations (which includes all n-truncations) using quotients (Egbert Rijke will talk about this in the afternoon).

Propositional Truncation

The Propositional Truncation $\|-\|$ as a HIT:

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

- $|-| : A \to \|A\|$
- $\varepsilon : \Pi(x, y : \|A\|),\ x = y$

Propositional Truncation

The Propositional Truncation $\|-\|$ as a HIT:

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

- $|-| : A \to \|A\|$
- $\varepsilon : \Pi(x, y : \|A\|),\ x = y$

ε is a recursive: the domain of the recursor is the type $\|A\|$ being constructed.

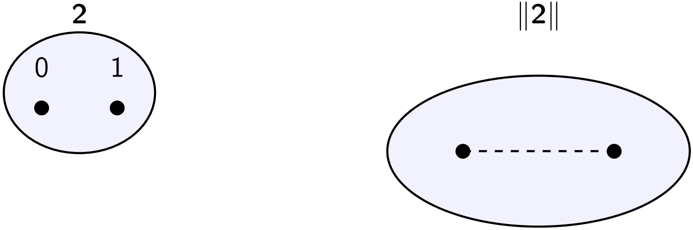
Propositional Truncation

The Propositional Truncation $\|-\|$ as a HIT:

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

- $|-| : A \to \|A\|$
- $\varepsilon : \Pi(x, y : \|A\|),\ x = y$

ε is a recursive: the domain of the recursor is the type $\|A\|$ being constructed.



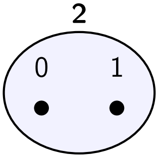
Propositional Truncation

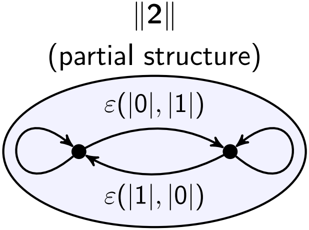
The Propositional Truncation $\|-\|$ as a HIT:

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

- $|-| : A \to \|A\|$
- $\varepsilon : \Pi(x, y : \|A\|),\ x = y$

ε is a recursive: the domain of the recursor is the type $\|A\|$ being constructed.





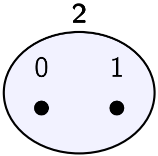
Propositional Truncation

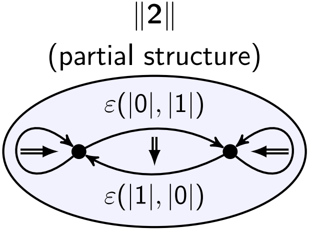
The Propositional Truncation $\|-\|$ as a HIT:

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

- $|-| : A \to \|A\|$
- $\varepsilon : \Pi(x, y : \|A\|),\ x = y$

ε is a recursive: the domain of the recursor is the type $\|A\|$ being constructed.





Propositional Truncation

Proof that $\varepsilon(|0|, |1|) = \varepsilon(|1|, |0|)^{-1}$.

Proof that $\varepsilon(|0|,|1|) = \varepsilon(|1|,|0|)^{-1}$.

We prove the following statement:

$$\Pi(x : \|\mathbf{2}\|)(p : |0| = x),\ p = \varepsilon(|0|,|0|)^{-1} \cdot \varepsilon(|0|,x).$$

Propositional Truncation

Proof that $\varepsilon(|0|, |1|) = \varepsilon(|1|, |0|)^{-1}$.

We prove the following statement:

$$\Pi(x : \|\mathbf{2}\|)(p : |0| = x),\ \ p = \varepsilon(|0|, |0|)^{-1} \cdot \varepsilon(|0|, x).$$

We can prove this by path induction on p. Then we need to prove

$$\mathsf{refl}_{|0|} = \varepsilon(|0|, |0|)^{-1} \cdot \varepsilon(|0|, |0|),$$

which is an ∞-groupoid law.

Propositional Truncation

Proof that $\varepsilon(|0|, |1|) = \varepsilon(|1|, |0|)^{-1}$.

We prove the following statement:

$$\Pi(x : \|\mathbf{2}\|)(p : |0| = x),\ \ p = \varepsilon(|0|, |0|)^{-1} \cdot \varepsilon(|0|, x).$$

We can prove this by path induction on p. Then we need to prove

$$\mathsf{refl}_{|0|} = \varepsilon(|0|, |0|)^{-1} \cdot \varepsilon(|0|, |0|),$$

which is an ∞-groupoid law.

Now $\varepsilon(|0|, |1|)$ and $\varepsilon(|1|, |0|)^{-1}$ are both equal to the same path, hence equal to each other.

One-step truncation

We define the *one-step truncation* $\{A\}$, which is the following HIT.

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

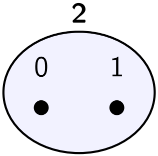
- $f : A \to \{A\}$
- $e : \Pi(x, y : A),\ f(x) = f(y)$

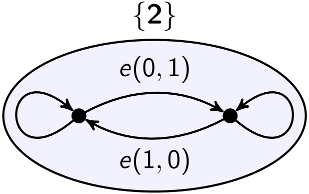
One-step truncation

We define the *one-step truncation* $\{A\}$, which is the following HIT.

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

- $f : A \to \{A\}$
- $e : \Pi(x, y : A),\ f(x) = f(y)$





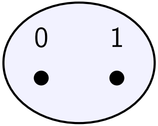
One-step truncation

We define the *one-step truncation* $\{A\}$, which is the following HIT.

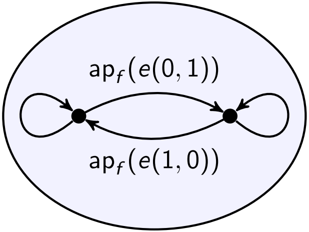
$\texttt{HIT}\ \{A\} :=$

- $f : A \to \{A\}$
- $e : \Pi(x, y : A),\ f(x) = f(y)$

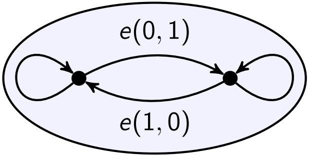
2



$\{\{\mathbf{2}\}\}$
(partial structure)



$\{\mathbf{2}\}$



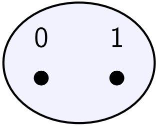
One-step truncation

We define the *one-step truncation* $\{A\}$, which is the following HIT.

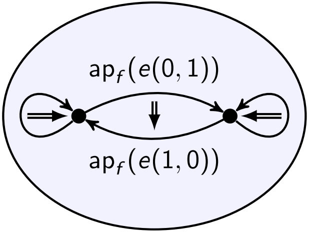
`HIT` $\{A\} :=$

- $f : A \to \{A\}$
- $e : \Pi(x, y : A),\ f(x) = f(y)$

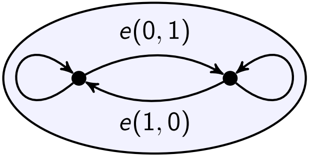
$\mathbf{2}$



$\{\{\mathbf{2}\}\}$
(partial structure)



$\{\mathbf{2}\}$



One-step truncation

Proof that $\mathsf{ap}_f(e(0,1)) = (\mathsf{ap}_f(e(1,0)))^{-1}$.

[This equality lives in type $f(f(0)) =_{\{\{\mathbf{2}\}\}} f(f(1))$]

One-step truncation

Proof that $\mathrm{ap}_f(e(0,1)) = (\mathrm{ap}_f(e(1,0)))^{-1}$.

[This equality lives in type $f(f(0)) =_{\{\{\mathbf{2}\}\}} f(f(1))$]

Definition. $g : A \to B$ is *weakly constant* if $\Pi(x, y : A),\ g(x) = g(y)$

One-step truncation

Proof that $\mathrm{ap}_f(e(0,1)) = (\mathrm{ap}_f(e(1,0)))^{-1}$.

[This equality lives in type $f(f(0)) =_{\{\{\mathbf{2}\}\}} f(f(1))$]

Definition. $g : A \to B$ is *weakly constant* if $\Pi(x, y : A),\ g(x) = g(y)$
Lemma. If $g : A \to B$ is weakly constant, then for any $x, y : A$ the function $\mathrm{ap}_g : x = y \to g(x) = g(y)$ is weakly constant.

Proof that $\mathrm{ap}_f(e(0,1)) = (\mathrm{ap}_f(e(1,0)))^{-1}$.

[This equality lives in type $f(f(0)) =_{\{\{\mathbf{2}\}\}} f(f(1))$]

Definition. $g : A \to B$ is *weakly constant* if $\Pi(x, y : A),\ g(x) = g(y)$
Lemma. If $g : A \to B$ is weakly constant, then for any $x, y : A$ the function $\mathrm{ap}_g : x = y \to g(x) = g(y)$ is weakly constant.

Proof. Fix $x : A$ and let q be the proof that g is weakly constant. We first prove:

$$\Pi(z : A)(p : x = z),\ \mathrm{ap}_g(p) = q(x,x)^{-1} \cdot q(x,z).$$

Proof that $\mathrm{ap}_f(e(0,1)) = (\mathrm{ap}_f(e(1,0)))^{-1}$.

[This equality lives in type $f(f(0)) =_{\{\{\mathbf{2}\}\}} f(f(1))$]

Definition. $g : A \to B$ is *weakly constant* if $\Pi(x, y : A),\ g(x) = g(y)$
Lemma. If $g : A \to B$ is weakly constant, then for any $x, y : A$ the function $\mathrm{ap}_g : x = y \to g(x) = g(y)$ is weakly constant.

Proof. Fix $x : A$ and let q be the proof that g is weakly constant. We first prove:

$$\Pi(z : A)(p : x = z),\ \mathrm{ap}_g(p) = q(x,x)^{-1} \cdot q(x,z).$$

This again follows by path induction.
Now for any $p, q : x = y$ both $\mathrm{ap}_g(p)$ and $\mathrm{ap}_g(q)$ are both equal to $q(x,x)^{-1} \cdot q(x,y)$. □

One-step truncation

Proof that $\mathrm{ap}_f(e(0,1)) = (\mathrm{ap}_f(e(1,0)))^{-1}$.

[This equality lives in type $f(f(0)) =_{\{\{\mathbf{2}\}\}} f(f(1))$]

Definition. $g : A \to B$ is *weakly constant* if $\Pi(x, y : A),\ g(x) = g(y)$
Lemma. If $g : A \to B$ is weakly constant, then for any $x, y : A$ the function $\mathrm{ap}_g : x = y \to g(x) = g(y)$ is weakly constant.

Proof. Fix $x : A$ and let q be the proof that g is weakly constant. We first prove:

$$\Pi(z : A)(p : x = z),\ \mathrm{ap}_g(p) = q(x,x)^{-1} \cdot q(x,z).$$

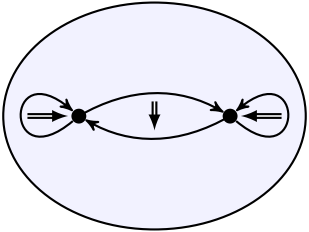
This again follows by path induction.
Now for any $p, q : x = y$ both $\mathrm{ap}_g(p)$ and $\mathrm{ap}_g(q)$ are both equal to $q(x,x)^{-1} \cdot q(x,y)$. □

Since e proves that f is weakly-constant, we have

$$\mathrm{ap}_f(e(0,1)) = \mathrm{ap}_f(e(1,0)^{-1}) = (\mathrm{ap}_f(e(1,0)))^{-1}.$$

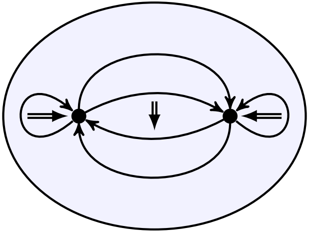
One-step truncation

$\{\{\mathbf{2}\}\}$
(partial structure)



One-step truncation

$\{\{\mathbf{2}\}\}$
(partial structure)



However, we have $e(f(0), f(1)) \neq \mathrm{ap}_f(e(0,1))$.
Lemma. If $p : a =_A b$, then the paths $\mathrm{ap}_f(p)$ and $e(a, b)$ in $\{A\}$ are provably different, i.e. $\mathrm{ap}_f(p) \neq e(a, b)$.
Proof sketch. We can define a map $\{A\} \to S^1$ sending all point constructors to base and all path constructors $e(a, b)$ to loop.

Construction of the Propositional truncation

We define $\{A\}_\infty$ as the colimit of this diagram:

$$A \xrightarrow{f} \{A\} \xrightarrow{f} \{\{A\}\} \xrightarrow{f} \{\{\{A\}\}\} \xrightarrow{f} \cdots \tag{1}$$

Theorem

$\{A\}_\infty$ is the propositional truncation of A.

Corollary

A function in $\|A\| \to B$ is the same as a cocone over (1), for any type B.

2-HITs

We want to construct HITs with 2-path constructors.
Examples:
`HIT` $T^2 :=$

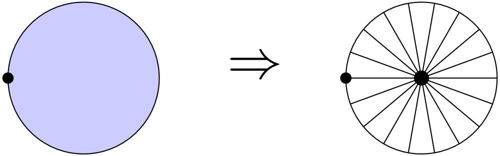
- $b : T^2$
- $\ell_1, \ell_2 : b = b$
- $s : \ell_1 \cdot \ell_2 = \ell_2 \cdot \ell_1$

`HIT` $K(G,1) :=$

- $\star : K(G,1)$
- $p : G \to \star = \star$
- $m : \Pi(g, h : G),\ p(gh) = p(g) \cdot p(h)$
- $K(G,1)$ is 1-truncated

Hubs and spokes

In the book, 2-HITs are reduced to 1-HITs using the hubs and spokes method.



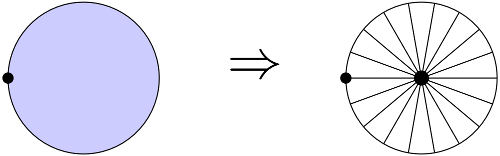
The torus becomes:

`HIT` $T^2 :=$

- $b : T^2$
- $\ell_1, \ell_2 : b = b$
- $h : T^2$
- $s : \Pi(x : S^1), \underbrace{\qquad\qquad\qquad\qquad}_{S^1 \to T^2} x = h$

Hubs and spokes

In the book, 2-HITs are reduced to 1-HITs using the hubs and spokes method.



The torus becomes:

`HIT` $T^2 :=$

- $b : T^2$
- $\ell_1, \ell_2 : b = b$
- $h : T^2$
- $s : \Pi(x : S^1),\ \underbrace{\text{circle.rec } b\ (\ell_1 \cdot \ell_2 \cdot \ell_1^{-1} \cdot \ell_2^{-1})}_{S^1 \to T^2}\ x = h$

Hubs and spokes

Why does hubs and spokes work?
For $a_0 : A$ and $p : a_0 = a_0$ we have

$$\big(\Sigma(h : A),\ \Pi(x : S^1),\ \text{circle.rec}\ a_0\ p\ x = h\big) \simeq (p = 1).$$

Proof. Computing with equivalences:

Hubs and spokes

Why does hubs and spokes work?
For $a_0 : A$ and $p : a_0 = a_0$ we have

$$\big(\Sigma(h : A),\ \Pi(x : S^1),\ \text{circle.rec}\ a_0\ p\ x = h\big) \simeq (p = 1).$$

Proof. Computing with equivalences:

$$\Sigma(h : A),\ \Pi(x : S^1),\ \text{circle.rec}\ a_0\ p\ x = h$$

Hubs and spokes

Why does hubs and spokes work?
For $a_0 : A$ and $p : a_0 = a_0$ we have

$$\big(\Sigma(h : A),\ \Pi(x : S^1),\ \text{circle.rec } a_0\ p\ x = h\big) \simeq (p = 1).$$

Proof. Computing with equivalences:

$$\begin{aligned} &\Sigma(h : A),\ \Pi(x : S^1),\ \text{circle.rec } a_0\ p\ x = h \\ \simeq\ &\Sigma(h : A),\ \Sigma(q : a_0 = h),\ q =^{\lambda x.\text{circle.rec } a_0\ p\ x = h}_{\text{loop}} q \end{aligned}$$

Hubs and spokes

Why does hubs and spokes work?
For $a_0 : A$ and $p : a_0 = a_0$ we have

$$\big(\Sigma(h : A),\ \Pi(x : S^1),\ \text{circle.rec } a_0\ p\ x = h\big) \simeq (p = 1).$$

Proof. Computing with equivalences:

$$\begin{aligned} &\Sigma(h : A),\ \Pi(x : S^1),\ \text{circle.rec } a_0\ p\ x = h \\ \simeq\ &\Sigma(h : A),\ \Sigma(q : a_0 = h),\ q =^{\lambda x.\text{circle.rec } a_0\ p\ x = h}_{\text{loop}} q \\ \simeq\ &1 =^{\lambda x.\text{circle.rec } a_0\ p\ x = a_0}_{\text{loop}} 1 \end{aligned}$$

Hubs and spokes

Why does hubs and spokes work?
For $a_0 : A$ and $p : a_0 = a_0$ we have

$$\big(\Sigma(h : A),\ \Pi(x : S^1),\ \text{circle.rec}\ a_0\ p\ x = h\big) \simeq (p = 1).$$

Proof. Computing with equivalences:

$$\begin{aligned}
&\Sigma(h : A),\ \Pi(x : S^1),\ \text{circle.rec}\ a_0\ p\ x = h \\
\simeq\ &\Sigma(h : A),\ \Sigma(q : a_0 = h),\ q =^{\lambda x.\text{circle.rec}\ a_0\ p\ x = h}_{\text{loop}} q \\
\simeq\ &1 =^{\lambda x.\text{circle.rec}\ a_0\ p\ x = a_0}_{\text{loop}} 1 \\
\simeq\ &1 \cdot 1 = \text{ap}_{\text{circle.rec}\ a_0\ p}(\text{loop}) \cdot 1
\end{aligned}$$

Hubs and spokes

Why does hubs and spokes work?
For $a_0 : A$ and $p : a_0 = a_0$ we have

$$\big(\Sigma(h : A),\ \Pi(x : S^1),\ \text{circle.rec}\ a_0\ p\ x = h\big) \simeq (p = 1).$$

Proof. Computing with equivalences:

$$\begin{aligned}
&\Sigma(h : A),\ \Pi(x : S^1),\ \text{circle.rec}\ a_0\ p\ x = h \\
\simeq\ &\Sigma(h : A),\ \Sigma(q : a_0 = h),\ q =^{\lambda x.\text{circle.rec}\ a_0\ p\ x = h}_{\text{loop}} q \\
\simeq\ &1 =^{\lambda x.\text{circle.rec}\ a_0\ p\ x = a_0}_{\text{loop}} 1 \\
\simeq\ &1 \cdot 1 = \text{ap}_{\text{circle.rec}\ a_0\ p}(\text{loop}) \cdot 1 \\
\simeq\ &p = 1
\end{aligned}$$

Torus from quotients

Problem. The hubs-and-spokes-torus is *not* a quotient. The last path constructor refers to previous path constructors, and cannot be written down before you have ℓ_1 and ℓ_2 as paths in the torus.

$$s : \Pi(x : S^1),\ \text{circle.rec } b\ (\ell_1 \cdot \ell_2 \cdot \ell_1^{-1} \cdot \ell_2^{-1})\ x = h$$

Torus from quotients

Problem. The hubs-and-spokes-torus is *not* a quotient. The last path constructor refers to previous path constructors, and cannot be written down before you have ℓ_1 and ℓ_2 as paths in the torus.

$$s : \Pi(x : S^1),\ \text{circle.rec}\ b\ (\ell_1 \cdot \ell_2 \cdot \ell_1^{-1} \cdot \ell_2^{-1})\ x = h$$

Solution. Do the constructions in two stages. First define a HIT

`HIT pretorus` :=

- $\tilde{b}$: `pretorus`
- $\tilde{\ell}_1, \tilde{\ell}_2 : b = b$
- $\tilde{h}$: `pretorus`

This is a quotient. Now we can define $f : S^1 \to$ `pretorus` by

$$f(\text{base}) :\equiv \tilde{b}$$
$$\text{ap}_f(\text{loop}) := \tilde{\ell}_1 \cdot \tilde{\ell}_2 \cdot \tilde{\ell}_1^{-1} \cdot \tilde{\ell}_2^{-1}.$$

Torus from quotients

We then define the torus:

`HIT` $T^2 :=$

- $i : \mathtt{pretorus} \to T^2$
- $\sigma : \Pi(x : S^1),\ i(f(x)) = i(h)$

The constructors for T^2 are defined as:

$$\begin{aligned} b &:\equiv i(\tilde{b}) && : T^2 \\ \ell_i &:\equiv \mathsf{ap}_i(\tilde{\ell}_i) && : b = b \\ s &:\equiv ?? && : \ell_1 \cdot \ell_2 = \ell_2 \cdot \ell_1 \end{aligned}$$

Torus from quotients

We want to apply the equivalence

$$\big(\Sigma(h : A),\ \Pi(x : S^1),\ \text{circle.rec } a_0\ p\ x = h\big) \simeq (p = 1).$$

Let $\tilde{p} = \tilde{\ell}_1 \cdot \tilde{\ell}_2 \cdot \tilde{\ell}_1^{-1} \cdot \tilde{\ell}_2^{-1}$.
For $x : S^1$ we have

$$\text{circle.rec } (i\ a_0)\ (\mathsf{ap}_i(\tilde{p}))\ x = i(\text{circle.rec } a_0\ \tilde{p}\ x) \stackrel{\sigma}{=} i(\tilde{h}).$$

From the equivalence we get

$$\mathsf{ap}_i(\tilde{p}) = 1.$$

This gives

$$\ell_1 \cdot \ell_2 \cdot \ell_1^{-1} \cdot \ell_2^{-1} \equiv \mathsf{ap}_i(\tilde{\ell}_1) \cdot \mathsf{ap}_i(\tilde{\ell}_2) \cdot \mathsf{ap}_i(\tilde{\ell}_1)^{-1} \cdot \mathsf{ap}_i(\tilde{\ell}_2)^{-1} = 1.$$

which gives

$$s : \ell_1 \cdot \ell_2 = \ell_2 \cdot \ell_1.$$

Remarks

- This is just the definition of the constructors! We still need to define the induction principle, and computation rules.
- It should be a little bit simpler in a cubical type theory. However, most steps do not become definitional even in cubical type theory.
- We don't prove the computation rule for the induction principle on 2-paths. However, this is not needed to characterize T^2 up to equivalence.
- We don't just define the torus, but a wide class of 2-HITs. The 2-path constructor must be an equality between
  - 1-path constructors;
  - A point constuctor $f : A \to X$ applied to a path in A;
  - reflexivity;
  - concatenations/inverses of such paths.

More 2-HITs

Given $A : \mathcal{U}$ and $R : A \to A \to \mathcal{U}$ we define *words* in R to be
`inductive` $\overline{R} : A \to A \to \mathcal{U} :=$

- $[-] : \Pi(a, a' : A),\ R(a, a') \to \overline{R}(a, a')$
- $\rho : \Pi(a, a' : A),\ a = a' \to \overline{R}(a, a')$
- $-^{-1} : \Pi(a, a' : A),\ \overline{R}(a, a') \to \overline{R}(a', a)$
- $- \cdot - : \Pi(a, a', a'' : A),\ \overline{R}(a, a') \to \overline{R}(a', a'') \to \overline{R}(a, a'')$

Then, if we have a map

$$p : \Pi(a, a' : A),\ R(a, a') \to i(a) = i(a')$$

we can extend it to a map

$$\overline{p} : \Pi(a, a' : A),\ \overline{R}(a, a') \to i(a) = i(a')$$

More 2-HITs

If we are also given a "relation over $\overline{R}$," i.e. a family

$$Q : \Pi(a, a' : A),\ \overline{R}(a, a') \to \overline{R}(a, a') \to \mathcal{U}$$

Then we define the following 2-HIT:

`HIT` two-quotient$(A, R, Q) : \mathcal{U} :=$

- $i : A \to \text{two-quotient}(A, R, Q)$
- $p : \Pi(a, a' : A),\ R(a, a') \to i(a) = i(a')$
- $r : \Pi(a, a' : A)(r, r' : \overline{R}(a, a')),\ Q(r, r') \to \overline{p}(r) = \overline{p}(r')$

More 2-HITs

Example: $K(G,1) :\equiv \|\text{two-quotient}(A,R,Q)\|_1$ with

$$A :\equiv 1$$
$$R(\star,\star) :\equiv G$$

and Q is an inductive family with 1 constructor, namely:

$$q : \Pi(g_1, g_2 : G),\ \ Q([g_1 * g_2], [g_1] \cdot [g_2])$$

`HIT` $K(G,1) :=$

- $b : K(G,1)$
- $p : G \to b = b$
- $m : \Pi(g,h : G),\ \ p(gh) = p(g) \cdot p(h)$
- $K(G,1)$ is 1-truncated

Conclusions

- We can reduce a wide class of HITs to quotients.
- Are there HITs which we cannot reduce to quotients?
  - I don't know
  - There are certainly HITs where I have no idea *how* to reduce them. (e.g. arbitrary localizations)

Thank you