Equivalence and higher groupoids

Gé Waayers & Nick Hendriks

30-11-2022

Intro

- Equivalence
  - Homotopy
  - Quasi-inverse
  - Equivalence, isequiv
- Higher groupoids
  - Unit type
  - Cartesian product types

Intro to homotopy

- We have seen $x =_A y$ as paths between elements of type A.
- Now "identification" between functions and between types.
- Traditional two functions are equal if they take equal values on all inputs.
- Under the homotopical interpretation, this dependent function type, $\prod_{(x:A)}(f(x) = g(x))$, consists of continuous paths and may be regarded as the type of homotopies.

Homotopy

Definition

Let $f, g : \prod_{(x:A)} P(x)$ be two sections of a family $P : A \to \mathcal{U}$. A homotopy from f to g is a dependent function of type

$$(f \sim g) :\equiv \prod_{x:A} (f(x) = g(x)).$$

- Note that a homotopy is not the same as identification $(f = g)$, but they are "equivalent".

Homotopy is an equivalence relation

Lemma

Homotopy is an equivalence relation on each dependent function functions type $\prod_{(x:A)} P(x)$*. That is, we have elements of the types*

$$\prod_{f:\prod_{(x:A)} P(x)} (f \sim f)$$

$$\prod_{f,g:\prod_{(x:A)} P(x)} (f \sim g) \to (g \sim f)$$

$$\prod_{f,g,h:\prod_{(x:A)} P(x)} (f \sim g) \to (g \sim h) \to (f \sim h)$$

Different perspectives

- From a traditional perspective a function $f : A \to B$ is an isomorphism if there is a function $g : B \to A$ such that $f \circ g \sim \mathrm{id}_B$ and $g \circ f \sim \mathrm{id}_A$.
- From a homotopical perspective this should be called a homotopy equivalence.
- From a categorical perspective this should be called an equivalence of (higher) groupoids.

Quasi-inverse

- The corresponding type

$$\sum_{g:B\to A} ((f \circ g \sim \mathrm{id}_B) \times (g \circ f \sim \mathrm{id}_A)) \tag{1}$$

is poorly behaved. For instance, for a single function $f : A \to B$ there may be multiple unequal inhabitants.

Definition

For a function $f : A \to B$, a quasi-inverse of f is a triple (g, α, β) consisting of a function $g : B \to A$ and homotopies $\alpha : f \circ g \sim \mathrm{id}_B$ and $\beta : g \circ f \sim \mathrm{id}_A$.
The type seen in (1) is the type of the quasi-inverses of f, denoted by $\mathrm{qinv}(f)$.

Example identity function

Example

The identity function $\mathrm{id}_A : A \to A$ has a quasi-inverse given by id_A itself, together with homotopies defined by $\alpha(y) :\equiv \mathsf{refl}_y$ and $\beta(x) :\equiv \mathsf{refl}_x$

Equivalence

- Isomorphism and bijection are both only used when A and B behave like sets.
- Equivalence is reserved for an improved notion isequiv.

Definition

The notion isequiv has the following properties:

(i) For each function $f : A \to B$ there is a function $\mathrm{qinv}(f) \to \mathrm{isequiv}(f)$.

(ii) Similarly, for each f we have $\mathrm{isequiv}(f) \to \mathrm{qinv}(f)$.

(iii) For any two inhabitants $e_1, e_2 : \mathrm{isequiv}(f)$ we have $e_1 = e_2$

We write $(A \simeq B)$ for the type of equivalences from A to B, i.e. the type

$$(A \simeq B) :\equiv \sum_{f:A\to B} isequiv(f).$$

Example of isequiv type

Example

$$isequiv(f) :\equiv \Big(\sum_{g:B\to A}(f \circ g \sim \mathrm{id}_B)\Big) \times \Big(\sum_{h:B\to A}(h \circ f \sim \mathrm{id}_A)\Big)$$

Take home message

There is a well-behaved type which we can pronounce as "f is an equivalence", and we can prove f to be an equivalence by exhibiting a quasi-inverse to it. This is in practice the most common way to prove that a function is an equivalence.

Type equivalence is an equivalence relation

Lemma

Type equivalence is an equivalence relation on $\mathcal{U}$. More specifically:

(i) *For any A the identity type id_A is an equivalence, hence $A \simeq A$.*

(ii) *For any $f : A \simeq B$, we have an equivalence $f^{-1} : B \simeq A$.*

(iii) *For any $f : A \simeq B$ and $g : B \simeq C$, we have $g \circ f : A \simeq C$.*

The unit type

Definition

The unit type $\mathbf{1} : \mathcal{U}$ is the nullary product type, which only has one inhabitant $\star : \mathbf{1}$.

The unit type

Theorem

For any $x, y : \mathbf{1}$*, we have* $(x = y) \simeq \mathbf{1}$.

The unit type

Theorem

For any $x, y : \mathbf{1}$, *we have* $(x = y) \simeq \mathbf{1}$.

Proof.

We define a function $(x = y) \to \mathbf{1}$ by mapping everything to $\star$. If $x, y : \mathbf{1}$, by induction, we may assume $x \equiv \star \equiv y$. We now have $\mathsf{refl}_\star : x = y$ which gives us a constant function $\mathbf{1} \to (x = y)$.

The unit type

Theorem

For any $x, y : \mathbf{1}$*, we have* $(x = y) \simeq \mathbf{1}$*.*

Proof.

We define a function $(x = y) \to \mathbf{1}$ by mapping everything to $\star$. If $x, y : \mathbf{1}$, by induction, we may assume $x \equiv \star \equiv y$. We now have $\mathsf{refl}_\star : x = y$ which gives us a constant function $\mathbf{1} \to (x = y)$. Now let $u : \mathbf{1}$ be an element, then by induction, we may assume $u \equiv \star$, but that is exactly the image of the composite $\mathbf{1} \to (x = y) \to \mathbf{1}$.

The unit type

Theorem

For any $x, y : \mathbf{1}$*, we have* $(x = y) \simeq \mathbf{1}$*.*

Proof.

We define a function $(x = y) \to \mathbf{1}$ by mapping everything to $\star$. If $x, y : \mathbf{1}$, by induction, we may assume $x \equiv \star \equiv y$. We now have $\mathsf{refl}_\star : x = y$ which gives us a constant function $\mathbf{1} \to (x = y)$.
Now let $u : \mathbf{1}$ be an element, then by induction, we may assume $u \equiv \star$, but that is exactly the image of the composite $\mathbf{1} \to (x = y) \to \mathbf{1}$.
The other way around, let $p : x = y$. By path induction, we may assume $x \equiv y$ and p is refl_x. Then, by induction, we may assume $x \equiv \star$, so that the composite $(x = y) \to \mathbf{1} \to (x = y)$ takes p to refl_x, which is p. $\square$

Higher groupoids

- We have seen that *all* types in homotopy type theory behave like higher groupoids.

Example

Suppose $A : \mathcal{U}$ and that $x, y : A$. There exists an identity type $x =_A y$. Suppose elements $p : x =_A y$ and $q : x =_A y$. Then there also exists an identity type $p =_{x =_A y} q$. The latter is an identity type of indentity types. If we view identity types as paths, then $p =_{x =_A y} q$ can be viewed as an homotopy.

Type formers

- Given types $A, B : \mathcal{U}$, there exist many ways to form new types. For example:
- Cartesian product
- Disjoint union
- Dependent product
- Σ-types
- Unit type
- We will now look at the explicit behaviour of the groupoid structure on the cartesian product.

Cartesian product types

- Given types A and B we have the cartesian product type $A \times B$.
- Consider elements $x, y : A \times B$ and a path $p : x =_{A \times B} y$.
- We can extract paths $\mathrm{pr}_1(p) : \mathrm{pr}_1(x) =_A \mathrm{pr}_1(y)$ and $\mathrm{pr}_2(p) : \mathrm{pr}_2(x) =_B \mathrm{pr}_2(y)$.
- This yields a function:

$$x =_{A \times B} y \to (\mathrm{pr}_1(x) =_A \mathrm{pr}_1(y)) \times (\mathrm{pr}_2(x) =_B \mathrm{pr}_2(y)).$$

Theorem

For any x and y, the above function is an equivalence.

Cartesian product types

Theorem

For any x and y, the following function is an equivalence:

$$x =_{A\times B} y \to (\mathrm{pr}_1(x) =_A \mathrm{pr}_1(y)) \times (\mathrm{pr}_2(x) =_B \mathrm{pr}_2(y)).$$

Proof.

By induction we have $x \equiv (a, b)$ and $y \equiv (a', b')$ for some $a, a' : A$ and $b, b' : B$, so we want a function

$$(a =_A a') \times (b =_B b') \to ((a, b) =_{A\times B} (a', b')).$$

By induction we may assume there exist $p : a = a'$ and $q : b = b'$. By two path inductions we can then assume $a \equiv a'$, $b \equiv b'$ and that p and q are both reflexivity. We now have $(a, b) \equiv (a', b')$, so we can take the output to be reflexivity. □

Cartesian product types

- The inverse that we constructed is denoted by:

$$\mathsf{pair}^{=} : (\mathsf{pr}_1(x) = \mathsf{pr}_1(y)) \times (\mathsf{pr}_2(x) = \mathsf{pr}_2(y)) \to (x = y).$$

- We can view $\mathsf{pair}^{=}$ as an *introduction rule* for $x = y$.
- This is analogous to the pairing constructor of $A \times B$.
- Similarly we can introduce *elimination rules*:

$$\mathsf{ap}_{\mathsf{pr}_1} : (x = y) \to (\mathsf{pr}_1(x) = \mathsf{pr}_1(y));$$
$$\mathsf{ap}_{\mathsf{pr}_2} : (x = y) \to (\mathsf{pr}_2(x) = \mathsf{pr}_2(y)).$$

Cartesian product types

- Once again, consider :

$$\mathrm{pair}^{=} : (\mathrm{pr}_1(x) = \mathrm{pr}_1(y)) \times (\mathrm{pr}_2(x) = \mathrm{pr}_2(y)) \to (x = y).$$

- If we write $p : \mathrm{pr}_1\, x = \mathrm{pr}_1\, y$ and $q : \mathrm{pr}_2\, x = \mathrm{pr}_2\, y$ and $r : x =_{A\times B} y$, then we have

$$\begin{aligned} \mathrm{ap}_{\mathrm{pr}_1}(\mathrm{pair}^{=}(p,q)) &= p; \\ \mathrm{ap}_{\mathrm{pr}_2}(\mathrm{pair}^{=}(p,q)) &= q; \\ \mathrm{pair}^{=}(\mathrm{ap}_{\mathrm{pr}_1}(r), \mathrm{ap}_{\mathrm{pr}_2}(r)) &= r. \end{aligned}$$

Cartesian product types

- We can now deduce the following characterizations:

$$\begin{aligned} \mathsf{ap}_{\mathsf{pr}_i}(\mathsf{refl}_{(z:A\times B)}) &= \mathsf{refl}_{\mathsf{pr}_i\, z} \ \text{ for } i = 1, 2; \\ \mathsf{pair}^{=}(p^{-1}, q^{-1}) &= \mathsf{pair}^{=}(p, q)^{-1}; \\ \mathsf{pair}^{=}(p \cdot q, p' \cdot q') &= \mathsf{pair}^{=}(p, p') \cdot \mathsf{pair}^{=}(q, q'). \end{aligned}$$

- All of these can be derived using path induction.
- This turns $A \times B$ into a product groupoid.

Transport in product types

- Given two type families $A, B : Z \to \mathcal{U}$, we write $A \times B : Z \to \mathcal{U}$ for the type family defined by $(A \times B)(z) :\equiv A(z) \times B(z)$.
- If we let $p : z =_Z w$ and $x : A(z) \times B(z)$, then we can transport x along p to obtain an element of $A(w) \times B(w)$.

Theorem

In the above situation, we have

$$\mathrm{transport}^{A \times B}(p, x) =_{A(w) \times B(w)} (\mathrm{transport}^{A}(p, \mathrm{pr}_1\, x), \mathrm{transport}^{B}(p, \mathrm{pr}_2\, x))$$

Functoriality of ap under cartesian products

- Let $A, B, A', B' : \mathcal{U}$ be given types and let $g : A \to A'$ and $h : B \to B'$ be functions.
- We can define $f : A \times B \to A' \times B'$ by $f(x) :\equiv (g(\mathrm{pr}_1\, x), h(\mathrm{pr}_2\, x))$.

Theorem

In the above situation, given $x, y : A \times B$ and $p : \mathrm{pr}_1\, x = \mathrm{pr}_1\, y$ and $q : \mathrm{pr}_2\, x = \mathrm{pr}_2\, y$, we have:

$$f(\mathrm{pair}^{=}(p, q)) =_{(f(x)=f(y))} \mathrm{pair}^{=}(g(p), h(q)).$$

The unit type revisited

Definition

The unit type $\mathbf{1} : \mathcal{U}$ is the nullary product type, which only has one inhabitant $\star : \mathbf{1}$.

Theorem

For any $x, y : \mathbf{1}$, we have $(x = y) \simeq \mathbf{1}$.

- In a similar fashion, we can formulate equivalence in terms of introduction rules etc.