Identity types

Samuel Mimram

2025

École polytechnique

Identity types

Recall that in previous lesson we have seen **identity types**.

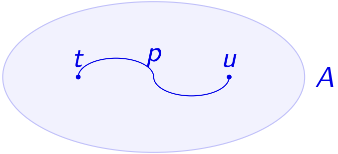
For $t, u : A$, we have a type

$$t = u$$

of proofs that t is the same as u (equalities/identities/paths).

On a semantic point of view,

- A corresponds to a *space*,
- t and u correspond to *points* in the space A,
- $p : t = u$ corresponds to a *path* from t to u in A:



Identity types

Type former:

$$- = - : (A : \mathcal{U}) \to A \to A \to \mathcal{U}$$

Constructor:

$$\mathsf{refl} : (x : A) \to x = x$$

Eliminator:

$$\begin{aligned} \mathsf{J} : (A : \mathcal{U}) \to (x : A) \to (P : (y : A) \to x = y \to \mathcal{U}) \to \\ P\,x\,(\mathsf{refl}\ x) \to \\ (y : A) \to (p : x = y) \to P\,y\,p \end{aligned}$$

Computation:

$$\mathsf{J}\ A\,x\,P\,r\,x\,(\mathsf{refl}\ x) \triangleq r$$

Identity types: J

We have

$$\begin{aligned} \mathsf{J} : (A : \mathcal{U}) \to (x : A) \to (P : (y : A) \to x = y \to \mathcal{U}) \to \\ P\,x\,(\mathsf{refl}\ x) \to \\ (y : A) \to (p : x = y) \to P\,y\,p \end{aligned}$$

For instance, we can prove that equality is symmetric, i.e. the property

$$\begin{aligned} P : (y : A) \to x = y \to \mathcal{U} \\ y \qquad\qquad p \qquad \mapsto y = x \end{aligned}$$

for given $A : \mathcal{U}$ and $x : A$ by

$$\mathsf{sym} \quad \hat{=} \quad \mathsf{J}\ A\,x\,P\ \mathsf{refl}$$

This corresponds to a pattern matching:

```
sym : {A : Type} {x y : A} → x ≡ y → y ≡ x
sym refl = refl
```

Identity types: J

Similarly, we can prove that sym is involutive, i.e. the property

$$\begin{aligned} P : (y : A) &\to x = y &&\to \mathcal{U} \\ y \quad & \qquad p && \mapsto \text{sym}\,(\text{sym}\ p) = p \end{aligned}$$

for given $A : \mathcal{U}$ and $x : A$ by

$$\text{sym} \quad \hat{=} \quad \mathsf{J}\ A\, x\, P\ \text{refl}$$

namely, by the computation rule, we have

$$\text{sym}\,(\text{sym}\ \text{refl}) \mathrel{\hat{=}} \text{refl}$$

UIP

Since identity types are introduced with refl only, do we get more than reflexivity?

This can be formulated as

- **uniqueness of reflexivity proofs**

$$URP : (x : A) \to (p : x = x) \to (p = \mathsf{refl})$$

- **uniqueness of identity proofs**

$$UIP : (x\,y : A) \to (p\,q : x = y) \to (p = q)$$

- K

$$K : (P : (x = x) \to \mathcal{U}) \to P\ \mathsf{refl} \to (p : x = x) \to P\ p$$

All are equivalent (see lab).

Can they be proved?

At first, it seems that the question was settled by the uniqueness rule.

Uniqueness rule

The uniqueness rule for booleans states that a function

$$f : (b : \mathsf{Bool}) \to A(b)$$

is entirely determined by the two values

$$f\ \mathsf{false} : A(\mathsf{false}) \qquad\qquad f\ \mathsf{true} : A(\mathsf{true})$$

Similarly, we expect that a function

$$f : (y : A) \to (p : x = y) \to B(y, p)$$

is entirely determined by

$$f\ x\ \mathsf{refl} : B(x, \mathsf{refl})$$

Uniqueness rule

We consider the following **uniqueness rule**: given $x : A$ and

$$f : (y : A) \to (p : x = y) \to B(y, p)$$

we have

$$f \quad \hat{=} \quad J\, A\, x\, (f\; x\; \mathsf{refl})$$

This rule was actually present in Martin-Löf's original type system [MLS84].

It implies the equality reflection rule, which is problematic in some ways.

Equality reflection rule

The **equality reflection rule** states that from $a = b$ we can deduce $a \doteq b$.

The resulting type theory is called **extensional type theory**.

Lemma ([Str93, Theorem 1.1])
The uniqueness rule $t \doteq J\, A\, x\, (t\, x\, \mathsf{refl}) : (y : A) \to (p : x = y) \to B(y, p)$ *implies equality reflection.*

Proof.
Suppose given $p : x = y$. Taking $B \doteq A$ we have, with $t \doteq \lambda y.\lambda p.x$,

$$\lambda y.\lambda p.x \doteq \mathsf{J}\ A\, x\, x$$

and with $t \doteq \lambda y.\lambda p.y$,

$$\lambda y.\lambda p.y \doteq \mathsf{J}\ A\, x\, x$$

Thus,

$$x \doteq (\lambda y.\lambda p.x)\, y\, p \doteq (\lambda y.\lambda p.y)\, y\, p \doteq y$$

□

Equality reflection rule

Lemma
The equality reflection rule

$$\frac{\Gamma \vdash p : x = y}{\Gamma \vdash x \triangleq y}$$

implies UIP.

Proof.
Consider the type

$$(x\, y : A) \to (p : x = y) \to (p = \mathsf{refl})$$

Note that this is well-typed because of the equality reflection rule!

By J, in order to prove this, it is enough to prove it for $x \triangleq y$ and $p \triangleq \mathsf{refl}$.

This can be done by refl. □

Equality reflection rule

It seems that the issue is settled but...

Proposition ([Hof95, Theorem 3.2.1])
The equality reflection rule

$$\frac{\Gamma \vdash p : x = y}{\Gamma \vdash x \triangleq y}$$

makes typechecking undecidable.

Because of this, the uniqueness rule for identities is almost never considered.

The groupoid model

The question of whether UIP holds remained opened for some time.

The question was settled by Hofmann and Streicher [HS98] who constructed a model in which UIP is not validated, as we now sketch.

Don't think *we cannot prove what we want*, but rather *we have more models*!

The groupoid model

In the set semantics of logic we interpret:

- a type A as a set $[\![A]\!]$
- a term $t : A$ as an element of $[\![t]\!] \in [\![A]\!]$

More generally, we interpret

- a dependent type $x : A \vdash B(x)$ as a function $[\![B]\!] : [\![A]\!] \to \mathsf{Set}$,
- a dependent term $x : A \vdash t : B(x)$ as a function $[\![t]\!] : (x : [\![A]\!]) \to [\![B]\!](x)$

For instance, we interpret

$$x : A, y : A \vdash x = y$$

as the function

$$[\![A]\!] \times [\![A]\!] \to \mathsf{Set}$$
$$(\tilde{x}, \tilde{y}) \mapsto \begin{cases} \{\star\} & \text{if } \tilde{x} = \tilde{y} \\ \emptyset & \text{otherwise} \end{cases}$$

The groupoid model

In the model of Hoffman and Streicher,
we interpret types as **groupoids** instead of sets.

The groupoid model

A **category** $\mathcal{C}$ consists of

- a set $\mathcal{C}_0$ of objects,
- a set $\mathcal{C}(A, B)$ of morphisms for every objects A and B,
- composition operations $\circ : \mathcal{C}(B, C) \to \mathcal{C}(A, B) \to \mathcal{C}(A, C)$
- identities $\mathrm{id}_A \in \mathcal{C}(A, B)$

such that composition is associative: for $f : A \to B$, $g : B \to C$ and $h : C \to D$,

$$h \circ (g \circ f) = (h \circ g) \circ f$$

and identities are neutral elements: for $f : A \to B$,

$$\mathrm{id}_B \circ f = f = f \circ \mathrm{id}_A$$

The groupoid model

In a category $\mathcal{C}$, a morphism $f : A \to B$ is **invertible** when there exists a morphism $g : B \to A$ such that $g \circ f = \mathrm{id}_A$ and $f \circ g = \mathrm{id}_B$.

A category is a **groupoid** when every morphism is invertible.

For instance,

- the category Bij of sets and bijections,
- the category $\mathbb{Z}$ with one object, $\mathbb{Z}$ as morphisms and composition given by addition,
- ...

The groupoid model

We interpret

- a type A as a groupoid $[\![A]\!]$
- a term $t : A$ as an object $[\![t]\!] \in [\![A]\!]$.

More generally, we interpret a type $\Gamma \vdash A$ as a functor $[\![\Gamma]\!] \to \mathsf{Gpd}$.

In particular, we interpret $x : A, y : A \vdash x = y$ as the functor

$$[\![A]\!] \times [\![A]\!] \to \mathsf{Gpd}$$

which to objects $x, y : [\![A]\!]$ associates groupoid whose objects are morphisms in $[\![A]\!](x, y)$ and only trivial morphisms.

By taking $[\![A]\!]$ a groupoid such as Bij, we see that our model does not validate UIP!

(note: we are leaving out lots of details, see [HS98])

The space model

We have a more general semantics here were we interpret

- a type A as a **space** $[\![A]\!]$,
- a term $t : A$ as a point $[\![t]\!] : A$.

More generally, we interpret

- a type $\Gamma \vdash A$ as a **fibration** $[\![\Gamma \vdash A]\!] : [\![A]\!] \to [\![\Gamma]\!]$,
- a term $\Gamma \vdash t : A$ as a section $[\![t]\!] : [\![\Gamma]\!] \to [\![A]\!]$ of the fibration $[\![\Gamma \vdash A]\!]$.

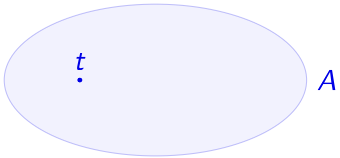
Setting this up precisely is quite subtle [Rie24]:

- MLTT with identity types can be modeled in **model categories** [GG08, AW09],
- Voevodsky constructed a model of HoTT in **simplicial sets** [KL21],
- Shulman extended this to ∞-**toposes** [Shu19].

The space model

In the model, we interpret

- a type A as a **space** (think: topological space),
- a term $t : A$ as a **point** of A.



The space model: paths

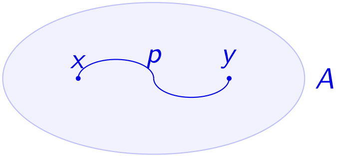
A **path** p in a space A is a continuous map

$$p : \mathsf{I} \to A$$

with $\mathsf{I} = [0, 1]$. The points $p(0)$ and $p(1)$ are its *source* and *target*.

Given $x, y : A$, we interpret the type $x = y : A$ as the space of paths from x to y in A.

We thus interpret a term $p : x = y$ as a path from x to y.



In particular, we interpret $\mathsf{refl}_x : x = x$ as the constant path $p : \mathsf{I} \to A$ with $p(t) \mathrel{\hat{=}} x$.

The space model: paths between paths

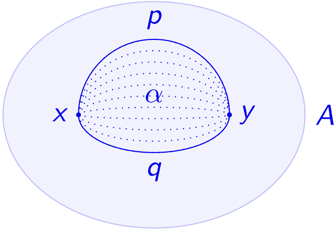
Given points $x, y : A$ and paths $p, q : x = y$, we can consider the type $p = q$.

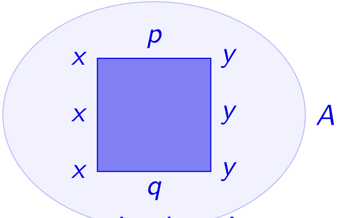
Its points are **homotopies** between p and q, i.e. continuous deformations from p to q, i.e. maps

$$\alpha : \mathsf{I} \to \mathsf{I} \to A$$

such that

$$\alpha\, 0 \triangleq p \qquad \alpha\, 1 \triangleq q \qquad \alpha\, t\, 0 \triangleq x \qquad \alpha\, t\, 1 \triangleq y$$





$\alpha : I \times I \to A$

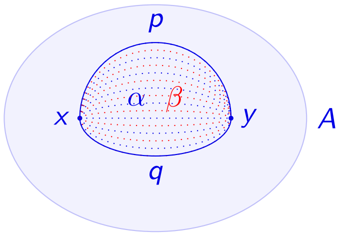
The space model: paths between paths between paths

Given $x, y : A$, $p, q : x = y$, $\alpha, \beta : p = q$, a path $\Psi : \alpha = \beta$ is a homotopy between homotopies, i.e. a map

$$I \to I \to I \to A$$

such that

$$\Psi\, 0 \triangleq \alpha \qquad \Psi\, 1 \triangleq \beta \qquad \Psi\, t\, 0 \triangleq p \qquad \Psi\, t\, 1 \triangleq q$$



and so on...

The space model: path types

Let's have a look at some path types.

- In 1, we have

$$(\star = \star) = 1$$

- In $\mathbb{N}$, we have

$$(m = n) = \begin{cases} 1 & \text{if } m \triangleq n \\ 0 & \text{otherwise} \end{cases}$$

- In $\mathbf{I}$

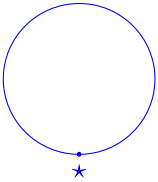
we have

$$(x = y) = \bullet \simeq 1$$

The space model: path types

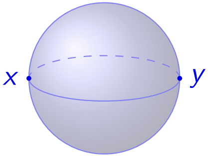
Let's have a look at some path types.

- In $\mathbb{S}^1$, we have



$$(\star = \star) \simeq \mathbb{Z}$$

- In $\mathbb{S}^2$, we have



$$(x = y) = \; ? \neq \mathbb{S}^1$$

The space model: J

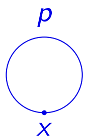
The axiom

$$\mathsf{J} : (A : \mathcal{U}) \to (x : A) \to (P : (y : A) \to x = y \to \mathcal{U}) \to$$
$$P\,x\,(\mathsf{refl}\ x) \to$$
$$(y : A) \to (p : x = y) \to P\,y\,p$$

states that in order to prove a property P on a path p, we can always suppose that p is refl *provided that y is a generic point*.

For instance, suppose that we want to prove UIP:

$$(x : A) \to (p : x = x) \to p = \mathsf{refl}$$



which is URP. We cannot use J anymore!

The space model: J

In Agda, we can do this by using pattern matching. With

```
{-# OPTIONS --without-K #-}
UIP : {A : Type} {x y : A} (p q : x ≡ y) → p ≡ q
UIP p refl = ?
```

we have to prove

```
error: [SplitError.UnificationStuck]
I'm not sure if there should be a case for the constructor refl,
because I get stuck when trying to solve the following unification
problems (inferred index ≟ expected index):
  x₁ ≟ x₁
Possible reason why unification failed:
  Cannot eliminate reflexive equation x₁ = x₁ of type A₁ because K
  has been disabled.
```

[AW09] Steve Awodey and Michael A Warren.
Homotopy theoretic models of identity types.
In *Mathematical proceedings of the cambridge philosophical society*, volume 146, pages 45–55. Cambridge University Press, 2009.
arXiv:0709.0248, doi:10.1017/S0305004108001783.

[GG08] Nicola Gambino and Richard Garner.
The identity type weak factorisation system.
Theoretical computer science, 409(1):94–109, 2008.
arXiv:0803.4349, doi:10.1016/j.tcs.2008.08.030.

[Hof95] Martin Hofmann.
***Extensional concepts in intensional type theory*.**
PhD thesis, University of Edinburgh, 1995.
https://era.ed.ac.uk/handle/1842/399.

[HS98] Martin Hofmann and Thomas Streicher.
The groupoid interpretation of type theory.
Twenty-five years of constructive type theory (Venice, 1995), 36:83–111, 1998.
https://ncatlab.org/nlab/files/HofmannStreicherGroupoidInterpretation.pdf.

[KL21] Krzysztof Kapulkin and Peter LeFanu Lumsdaine.
The simplicial model of univalent foundations (after Voevodsky).
Journal of the European Mathematical Society, 23(6):2071–2126, 2021.
`arXiv:1211.2851`, `doi:10.4171/JEMS/1050`.

[MLS84] Per Martin-Löf and Giovanni Sambin.
***Intuitionistic type theory*, volume 9.**
Bibliopolis Naples, 1984.
`https://archive-pml.github.io/martin-lof/pdfs/Bibliopolis-Book-retypeset-1984.pdf`.

[Rie24] Emily Riehl.
On the ∞-topos semantics of homotopy type theory.
Bulletin of the London Mathematical Society, 56(2):461–517, 2024.
`arXiv:2212.06937`, `doi:10.1112/blms.12997`.

[Shu19] Michael Shulman.
All $(\infty, 1)$-toposes have strict univalent universes.
Preprint, 2019.
`arXiv:1904.07004`.

[Str93] Thomas Streicher.
Investigations into intensional type theory, 1993.
Habilitiation Thesis, Ludwig Maximilian Universität.