Propositions as types & Identity types
Homotopy Type Theory Seminar, Summer '17

Lukas Braun

FAU Erlangen-Nürnberg

May 24, 2017

Propositions as types

- Propositions can be expressed as types.
- Inhabitants of a type are *evidence* or *witnesses* of its truth.
- A proposition is the collection of possible witnesses of its truth.

Propositions as types

English	Type Theory
True	$\mathbf{1}$
False	$\mathbf{0}$
A and B	$A \times B$
A or B	$A + B$
If A then B	$A \to B$
A if and only if B	$(A \to B) \times (B \to A)$
Not A	$A \to \mathbf{0}$

Constructive logic

- **No** elimination of double negation:

$$\neg\neg A \to A \equiv ((A \to 0) \to 0) \to A$$

- **No** excluded middle:

$$A + \neg A$$

Predicates

English	Type Theory
For all $x : A$, $P(x)$ holds	$\prod_{(x:A)} P(x)$
There exists $x : A$ such that $P(x)$	$\sum_{(x:A)} P(x)$

Predicates

English	Type Theory
For all $x : A$, $P(x)$ holds	$\prod_{(x:A)} P(x)$
There exists $x : A$ such that $P(x)$	$\sum_{(x:A)} P(x)$

Example: *If for all $x : A$, $P(x)$ and $Q(x)$ then (for all $x : A$, $P(x)$) and (for all $x : A$, $Q(x)$)* as a type:

$$(\prod_{(x:A)} P(x) \times Q(x)) \to (\prod_{(x:A)} P(x)) \times (\prod_{(x:A)} Q(x)).$$

Σ-Types as subtypes

- Read $\sum_{(x:A)} P(x)$ as "the type of all elements $x : A$ such that $P(x)$".

Σ-Types as subtypes

- Read $\sum_{(x:A)} P(x)$ as "the type of all elements $x : A$ such that $P(x)$".
- Example:

$$\text{Semigroup} :\equiv \sum_{(A:\mathcal{U})} \sum_{(m:A\to A\to A)} \prod_{(x,y,z:A)} m(x, m(y, z)) = m(m(x, y), z)$$

Higher order logic

- Propositions over predicates

$$\prod_{P:A\to\mathcal{U}} P(a) \to P(b)$$

Higher order logic

- Propositions over predicates

$$\prod_{P:A\to\mathcal{U}} P(a) \to P(b)$$

- Universes:

$$\Big(\prod_{P:A\to\mathcal{U}_i} P(a) \to P(b)\Big) : \mathcal{U}_{i+1}.$$

Identity types

- We write $\mathrm{Id}_A(a, b)$ or $a =_A b$ for the *type* representing the proposition of equality between $a, b : A$.
- Note this is different from judgemental equality $a \equiv b$.

Identity types

- We write $\mathrm{Id}_A(a, b)$ or $a =_A b$ for the *type* representing the proposition of equality between $a, b : A$.
- Note this is different from judgemental equality $a \equiv b$.
- Formally:

$$\frac{\Gamma \vdash A : \mathcal{U}_i \qquad \Gamma \vdash a : A \qquad \Gamma \vdash b : A}{\Gamma \vdash a =_A b : \mathcal{U}_i} =\text{-form}$$

Identity types

- We write $\mathrm{Id}_A(a, b)$ or $a =_A b$ for the *type* representing the proposition of equality between $a, b : A$.
- Note this is different from judgemental equality $a \equiv b$.
- Formally:

$$\frac{\Gamma \vdash A : \mathcal{U}_i \qquad \Gamma \vdash a : A \qquad \Gamma \vdash b : A}{\Gamma \vdash a =_A b : \mathcal{U}_i} \text{=-form}$$

- In the homotopical interpretation, inhabitants of $a =_A b$ are paths between a and b in the space A.

Introducing identities

- Elements of $a = b$ are introducted by knowing that a and b are the same, aka reflexivity:

$$\mathsf{refl} : \prod_{a:A} (a =_A a)$$

Introducing identities

- Elements of $a = b$ are introducted by knowing that a and b are the same, aka reflexivity:

$$\mathsf{refl} : \prod_{a:A} (a =_A a)$$

- Formally:

$$\frac{\Gamma \vdash A : \mathcal{U}_i \qquad \Gamma \vdash a : A}{\Gamma \vdash \mathsf{refl}_a : a =_A a} = \text{-intro}$$

Indiscernability of identicals

For every family

$$C : A \to \mathcal{U}$$

there is a function

$$f : \prod_{(x,y:A)} \prod_{(p:x=_A y)} C(x) \to C(y)$$

such that

$$f(x, x, \mathsf{refl}_x) :\equiv \mathsf{id}_{C(x)}.$$

Path induction

Given a family

$$C : \prod_{x,y:A} (x =_A y) \to \mathcal{U}$$

and a function

$$c : \prod_{x:A} C(x, x, \mathsf{refl}_x),$$

there is a function

$$f : \prod_{(x,y:A)} \prod_{(p:x=_A y)} C(x, y, p)$$

such that

$$f(x, x, \mathsf{refl}_x) :\equiv c(x).$$

Path induction formally

$$\frac{\begin{array}{c}\Gamma, x{:}A, y{:}A, p{:}x =_A y \vdash C : \mathcal{U}_i \\ \Gamma, z{:}A \vdash c : C[z, z, \mathsf{refl}_z/x, y, p] \\ \Gamma \vdash a : A \qquad \Gamma \vdash b : A \qquad \Gamma \vdash p' : a =_A b\end{array}}{\Gamma \vdash \mathsf{ind}_{=_A}(x.y.p.C, z.c, a, b, p') : C[a, b, p'/x, y, p]} \; \text{=-elim}$$

Path induction as a function

$$\mathrm{ind}_{=_A} : \prod_{(C:\prod_{(x,y:A)}(x=_A y)\to\mathcal{U})} \left(\prod_{(x:A)} C(x,x,\mathsf{refl}_x)\right) \to \prod_{(x,y:A)} \prod_{(p:x=_A y)} C(x,y,p)$$

with

$$\mathrm{ind}_{=_A}(C,c,x,x,\mathsf{refl}_x) :\equiv c(x).$$

Based path induction

Fix an element $a : A$, and suppose given a family

$$C : \prod_{x:A} (a =_A x) \to \mathcal{U}$$

and an element

$$c : C(a, \mathsf{refl}_a).$$

Then we obtain a function

$$f : \prod_{(x:A)} \prod_{(p:a=x)} C(x, p)$$

such that

$$f(a, \mathsf{refl}_a) :\equiv c.$$

Interpretation

- Path induction can be said to assert that reflexivity is the only path.
- But the homotopy interpretation says there are many different paths between elements x and y!

Interpretation

- Path induction can be said to assert that reflexivity is the only path.
- But the homotopy interpretation says there are many different paths between elements x and y!
- Solution: It is the *family* of types $(x =_A y)$, for varying elements $x, y : A$, that is inductively defined by elements of the form refl_x.
- Meaning, "the space of paths starting at some point is contractible to the constant loop at that point".