A type system with native homotopy universes

Robin Adams
Andrew Polonsky

The HoTT Workshop
Porto, 25.06.2016

Syntax

$$
\begin{aligned}
t, A, e ::=\ & \mathbf{1} \mid \mathsf{Prop} \mid \mathsf{Set} \mid \mathsf{Grpd} \mid *_3 \\
& \mid P \leftrightarrow Q \mid A \Leftrightarrow B \mid G \Lleftarrow\!\!\Rrightarrow H \mid \mathsf{Grpd} \simeq_3 \mathsf{Grpd} \mid a \sim_e b \\
& \mid \mathtt{tt} \mid x \mid (\Pi x{:}A)B \mid (\Sigma x{:}A)B \mid \lambda x{:}A.t \mid st \mid (s, t) \mid \pi_1 t \mid \pi_2 t \\
& \mid *^* \mid \Pi^*[x, x', x^*]{:}A^*.B^* \mid \Sigma^*[x, x', x^*]{:}A^*.B^* \mid \simeq^* A^* B^* \\
& \mid \mathsf{r}(t) \mid e^+(s) \mid e^-(t) \mid e^=(s, t) \mid \overrightarrow{e}(s) \mid \overleftarrow{e}(t)
\end{aligned}
$$

Syntax

$$
\begin{aligned}
t, A, e ::= \;& \mathbf{1} \mid \mathsf{Prop} \mid \mathsf{Set} \mid \mathsf{Grpd} \mid *_3 \\
& \mid P \leftrightarrow Q \mid A \Leftrightarrow B \mid G \Lleftarrow\!\!\!\Rrightarrow H \mid \mathsf{Grpd} \simeq_3 \mathsf{Grpd} \mid a \sim_e b \\
& \mid \mathtt{tt} \mid x \mid (\Pi x{:}A)B \mid (\Sigma x{:}A)B \mid \lambda x{:}A.t \mid st \mid (s,t) \mid \pi_1 t \mid \pi_2 t \\
& \mid *^* \mid \Pi^*[x, x', x^*]{:}A^*.B^* \mid \Sigma^*[x, x', x^*]{:}A^*.B^* \mid \simeq^* A^* B^* \\
& \mid \mathsf{r}(t) \mid e^+(s) \mid e^-(t) \mid e^=(s,t) \mid \overrightarrow{e}(s) \mid \overleftarrow{e}(t)
\end{aligned}
$$

$$\boxed{\mathsf{r}(A) \quad : \quad A \simeq A}$$

$$\boxed{a \simeq_A a' \quad := \quad a \sim_{\mathsf{r}(A)} a'}$$

Equality = Path substitution + transport

$$\boxed{\dfrac{x : A \vdash B(x) : * \qquad a^* : a \simeq_A a'}{B\,(\!(a^*)\!) : B(a) \simeq B(a')}}$$

$$\boxed{\dfrac{x : A \vdash b : B \qquad a^* : a \simeq_A a'}{b\,(\!(a^*)\!) : b(a) \sim_{B(\!(a^*)\!)} b(a')}}$$

Equality = Path substitution + transport

$$\boxed{\dfrac{x : A \vdash B(x) : * \qquad a^* : a \simeq_A a'}{B\left(\!\left(a^*\right)\!\right) : B(a) \simeq B(a')}} \qquad \boxed{\dfrac{x : A \vdash b : B \qquad a^* : a \simeq_A a'}{b\left(\!\left(a^*\right)\!\right) : b(a) \sim_{B(\!(a^*)\!)} b(a')}}$$

- Let $B(x) := a \simeq_A x$. Given $\alpha : a \simeq_A a'$, $\alpha' : a' \simeq_A a''$, we have

$$\begin{aligned} B\left(\!\left(\alpha'\right)\!\right) \quad &: \quad (a \simeq_A a') \simeq (a \simeq_A a'') \\ \alpha \circ \alpha' := B\left(\!\left(\alpha'\right)\!\right)^+ (\alpha) \quad &: \quad a \simeq_A a'' \end{aligned}$$

Equality = Path substitution + transport

$$\boxed{\dfrac{x : A \vdash B(x) : * \qquad a^* : a \simeq_A a'}{B\left(\!\left(a^*\right)\!\right) : B(a) \simeq B(a')}} \qquad \boxed{\dfrac{x : A \vdash b : B \qquad a^* : a \simeq_A a'}{b\left(\!\left(a^*\right)\!\right) : b(a) \sim_{B\left(\!\left(a^*\right)\!\right)} b(a')}}$$

- Let $B(x) := a \simeq_A x$. Given $\alpha : a \simeq_A a'$, $\alpha' : a' \simeq_A a''$, we have

$$\begin{aligned} B\left(\!\left(\alpha'\right)\!\right) &\quad : \quad (a \simeq_A a') \simeq (a \simeq_A a'') \\ \alpha \circ \alpha' := B\left(\!\left(\alpha'\right)\!\right)^+ (\alpha) &\quad : \quad a \simeq_A a'' \end{aligned}$$

- Let $B(x) := x \simeq_A a$. Given $\alpha : a \simeq_A a'$, we have

$$\begin{aligned} B\left(\!\left(\alpha\right)\!\right) &\quad : \quad (a \simeq_A a) \simeq (a' \simeq_A a) \\ \alpha^{-1} := B\left(\!\left(\alpha\right)\!\right)^+ (\mathrm{r}(a)) &\quad : \quad (a' \simeq_A a) \end{aligned}$$

Equality = Path substitution + transport

$$\boxed{\dfrac{x : A \vdash B(x) : * \qquad a^* : a \simeq_A a'}{B\,(\!(a^*)\!) : B(a) \simeq B(a')}} \qquad \boxed{\dfrac{x : A \vdash b : B \qquad a^* : a \simeq_A a'}{b\,(\!(a^*)\!) : b(a) \sim_{B(\!(a^*)\!)} b(a')}}$$

- Let $B(x) := a \simeq_A x$. Given $\alpha : a \simeq_A a'$, $\alpha' : a' \simeq_A a''$, we have

$$\begin{aligned} B\,(\!(\alpha')\!) &\quad : \quad (a \simeq_A a') \simeq (a \simeq_A a'') \\ \alpha \circ \alpha' := B\,(\!(\alpha')\!)^+\,(\alpha) &\quad : \quad a \simeq_A a'' \end{aligned}$$

- Let $B(x) := x \simeq_A a$. Given $\alpha : a \simeq_A a'$, we have

$$\begin{aligned} B\,(\!(\alpha)\!) &\quad : \quad (a \simeq_A a) \simeq (a' \simeq_A a) \\ \alpha^{-1} := B\,(\!(\alpha)\!)^+\,(\mathrm{r}(a)) &\quad : \quad (a' \simeq_A a) \end{aligned}$$

- Let $b : B$, with $x \notin \mathrm{FV}(b, B)$. Given $\alpha : a \simeq_A a'$, we have

$$\begin{aligned} \mathrm{r}(B) := B\,(\!(\alpha)\!) = B\,(\!(\,)\!) &\quad : \quad B \simeq B \\ \mathrm{r}(b) := b\,(\!(\alpha)\!) = b\,(\!(\,)\!) &\quad : \quad b \sim_{B(\!(\,)\!)} b \end{aligned}$$

Plan

- Background
- The system
- Formalization

Plan

- Background
- The system
- Formalization

$\lambda_{\simeq}$

$$\begin{aligned} A, t, e ::=\ & * \mid x \mid \Pi x{:}A.B \mid \Sigma x{:}A.B \\ & \mid \lambda x{:}A.t \mid st \mid (s, t) \mid \pi_1 t \mid \pi_2 t \end{aligned}$$

$\lambda_{\simeq}$

$$
\begin{aligned}
A, t, e ::= {} & * \mid x \mid \Pi x{:}A.B \mid \Sigma x{:}A.B \mid A \simeq B \mid a \sim_e b \\
& \mid \lambda x{:}A.t \mid st \mid (s, t) \mid \pi_1 t \mid \pi_2 t \\
& \mid *^* \mid \Pi^*[x, x', x^*]{:}A^*.B^* \mid \Sigma^*[x, x', x^*]{:}A^*.B^* \mid \simeq^* A^* B^*
\end{aligned}
$$

$\lambda{\simeq}$

$$A, t, e ::= * \mid x \mid \Pi x{:}A.B \mid \Sigma x{:}A.B \mid A \simeq B \mid a \sim_e b$$
$$\mid \lambda x{:}A.t \mid st \mid (s,t) \mid \pi_1 t \mid \pi_2 t$$
$$\mid *^* \mid \Pi^*[x, x', x^*]{:}A^*.B^* \mid \Sigma^*[x, x', x^*]{:}A^*.B^* \mid \simeq^* A^* B^*$$

$$\frac{A : * \qquad B : *}{A \simeq B : *}\ (\simeq\text{-Form}) \qquad\qquad \frac{e : A \simeq B}{\sim e : A \to B \to *}\ (\simeq\text{-Elim})$$

$\lambda\simeq$

$$A, t, e ::= * \mid x \mid \Pi x{:}A.B \mid \Sigma x{:}A.B \mid A \simeq B \mid a \sim_e b$$
$$\mid \lambda x{:}A.t \mid st \mid (s, t) \mid \pi_1 t \mid \pi_2 t$$
$$\mid *^* \mid \Pi^*[x, x', x^*]{:}A^*.B^* \mid \Sigma^*[x, x', x^*]{:}A^*.B^* \mid \simeq^* A^* B^*$$

$$\frac{A : * \qquad B : *}{A \simeq B : *}\ (\simeq\text{-Form}) \qquad\qquad \frac{e : A \simeq B}{\sim e : A \to B \to *}\ (\simeq\text{-Elim})$$

Given $A^* : A \simeq A'$, $B^* : B \simeq B'$, one can construct:

$\lambda\simeq$

$$A, t, e ::= * \mid x \mid \Pi x{:}A.B \mid \Sigma x{:}A.B \mid A \simeq B \mid a \sim_e b$$
$$\mid \lambda x{:}A.t \mid st \mid (s,t) \mid \pi_1 t \mid \pi_2 t$$
$$\mid *^* \mid \Pi^*[x, x', x^*]{:}A^*.B^* \mid \Sigma^*[x, x', x^*]{:}A^*.B^* \mid \simeq^* A^* B^*$$

$$\frac{A : * \qquad B : *}{A \simeq B : *}\ (\simeq\text{-Form}) \qquad\qquad \frac{e : A \simeq B}{\sim e : A \to B \to *}\ (\simeq\text{-Elim})$$

Given $A^* : A \simeq A'$, $B^* : B \simeq B'$, one can construct:

- $\times^* A^* B^* \ : (A \times B) \simeq (A' \times B')$

$\lambda{\simeq}$

$$
\begin{aligned}
A, t, e ::= {} & * \mid x \mid \Pi x{:}A.B \mid \Sigma x{:}A.B \mid A \simeq B \mid a \sim_e b \\
& \mid \lambda x{:}A.t \mid st \mid (s, t) \mid \pi_1 t \mid \pi_2 t \\
& \mid *^* \mid \Pi^*[x, x', x^*]{:}A^*.B^* \mid \Sigma^*[x, x', x^*]{:}A^*.B^* \mid \simeq^* A^* B^*
\end{aligned}
$$

$$
\frac{A : * \qquad B : *}{A \simeq B : *}\ (\simeq\text{-Form}) \qquad\qquad \frac{e : A \simeq B}{\sim e : A \to B \to *}\ (\simeq\text{-Elim})
$$

Given $A^* : A \simeq A'$, $B^* : B \simeq B'$, one can construct:

- $\times^* A^* B^* : (A \times B) \simeq (A' \times B')$
- $\to^* A^* B^* : (A \to B) \simeq (A' \to B')$

$\lambda\simeq$

$$A, t, e ::= \ * \ \mid x \mid \Pi x{:}A.B \mid \Sigma x{:}A.B \mid A \simeq B \mid a \sim_e b$$
$$\mid \lambda x{:}A.t \mid st \mid (s, t) \mid \pi_1 t \mid \pi_2 t$$
$$\mid *^* \mid \Pi^*[x, x', x^*]{:}A^*.B^* \mid \Sigma^*[x, x', x^*]{:}A^*.B^* \mid \simeq^* A^* B^*$$

$$\frac{A : * \qquad B : *}{A \simeq B : *}\ (\simeq\text{-Form}) \qquad\qquad \frac{e : A \simeq B}{\sim e : A \to B \to *}\ (\simeq\text{-Elim})$$

Given $A^* : A \simeq A'$, $B^* : B \simeq B'$, one can construct:

- $\times^* A^* B^* : (A \times B) \simeq (A' \times B')$
- $\to^* A^* B^* : (A \to B) \simeq (A' \to B')$
- $\simeq^* A^* B^* : (A \simeq B) \simeq (A' \simeq B')$

$\lambda\simeq$

$$A, t, e ::= * \mid x \mid \Pi x{:}A.B \mid \Sigma x{:}A.B \mid A \simeq B \mid a \sim_e b$$
$$\mid \lambda x{:}A.t \mid st \mid (s, t) \mid \pi_1 t \mid \pi_2 t$$
$$\mid *^* \mid \Pi^*[x, x', x^*]{:}A^*.B^* \mid \Sigma^*[x, x', x^*]{:}A^*.B^* \mid \simeq^* A^* B^*$$

$$\frac{A : * \qquad B : *}{A \simeq B : *}\ (\simeq\text{-Form}) \qquad \frac{e : A \simeq B}{\sim e : A \to B \to *}\ (\simeq\text{-Elim})$$

Given $A^* : A \simeq A'$, $B^* : B \simeq B'$, one can construct:

- $\times^* A^* B^* : (A \times B) \simeq (A' \times B')$
- $\to^* A^* B^* : (A \to B) \simeq (A' \to B')$
- $\simeq^* A^* B^* : (A \simeq B) \simeq (A' \simeq B')$
- $*^* : * \simeq *$

Reduction rules

$$
\begin{array}{rcl}
(\lambda x : A.t)a & \longrightarrow & t[a/x] \\
\pi_i(s_1, s_2) & \longrightarrow & s_i \\
\\
A \sim_{*^*} B & \longrightarrow & A \simeq B \\
f \sim_{\Pi^*[x,x',x^*]:A^*.B^*(x,x',x^*)} f' & \longrightarrow & \Pi a{:}A \Pi a'{:}A' \Pi a^* : a \sim_{A^*} a'. \\
 & & \quad fx \sim_{B^*(a,a',a^*)} f'x' \\
p \sim_{\Sigma^*[x,x',x^*]:A^*.B^*(x,x',x^*)} p' & \longrightarrow & \Sigma a^* : \pi_1 p \sim_{A^*} \pi_1 p'. \\
 & & \quad \pi_2 p \sim_{B^*(\pi_1 p, \pi_1 p', a^*)} \pi_2 p' \\
e \sim_{\simeq^* A^* B^*} e' & \longrightarrow & \Pi a{:}A \Pi a'{:}A' \Pi a^* : a \sim_{A^*} a' \\
 & & \Pi b{:}B \Pi b'{:}B' \Pi b^* : b \sim_{B^*} b'. \\
 & & \quad (a \sim_e b) \simeq (a' \sim_{e'} b')
\end{array}
$$

Extensional equality of closed types

THEOREM. There is an operation $(\cdot)^* : \mathcal{T}er(\lambda\simeq) \to \mathcal{T}er(\lambda\simeq)$ such that

$$\Gamma \vdash M : A \implies \Gamma^* \vdash M^* : M \sim_{A^*} M'$$

Extensional equality of closed types

THEOREM. There is an operation $(\cdot)^* : \mathcal{T}er(\lambda\simeq) \to \mathcal{T}er(\lambda\simeq)$ such that

$$\Gamma \vdash M : A \implies \Gamma^* \vdash M^* : M \sim_{A^*} M'$$

In particular,

- For any closed type $\vdash A : *$, there exist

$$\begin{array}{lcl} \vdash A^* & : & A \simeq A \\ \simeq_A & := & \sim(A^*) \quad : \quad A \to A \to * \end{array}$$

Extensional equality of closed types

THEOREM. There is an operation $(\cdot)^* : \mathcal{Ter}(\lambda\simeq) \to \mathcal{Ter}(\lambda\simeq)$ such that

$$\Gamma \vdash M : A \implies \Gamma^* \vdash M^* : M \sim_{A^*} M'$$

In particular,

- For any closed type $\vdash A : *$, there exist

$$\begin{array}{lclcl} \vdash A^* & : & A \simeq A & & \\ \simeq_A & := & \sim(A^*) & : & A \to A \to * \end{array}$$

- For any closed term $\vdash a : A$, there exists

$$a^* \quad : \quad a \simeq_A a$$

Our goal

Restricting to a low dimension, work out a system for computing with equalities which is

- Simple and intuitive
- Amenable to formalization
- Feasible to scale to the next dimension, in principle

$\lambda{\simeq}_2$

$$
\begin{aligned}
t, A, e ::= \ & \mathbf{1} \mid \mathsf{Prop} \mid \mathsf{Set} \mid \mathsf{Grpd} \mid *_3 \\
& \mid P \leftrightarrow Q \mid A \Leftrightarrow B \mid G \Leftrightarrow H \mid \mathsf{Grpd} \simeq_3 \mathsf{Grpd} \mid a \sim_e b \\
& \mid \mathtt{tt} \mid x \mid (\Pi x{:}A)B \mid (\Sigma x{:}A)B \mid \lambda x{:}A.t \mid st \mid (s,t) \mid \pi_1 t \mid \pi_2 t \\
& \mid *^* \mid \Pi^*[x, x', x^*]{:}A^*.B^* \mid \Sigma^*[x, x', x^*]{:}A^*.B^* \mid \simeq^* A^* B^* \\
& \mid \mathsf{r}(t) \mid e^+(s) \mid e^-(t) \mid e^=(s,t) \mid \overrightarrow{e}(s) \mid \overleftarrow{e}(t)
\end{aligned}
$$

$\lambda{\simeq}_2$

$$
\begin{aligned}
t, A, e ::= \;& \mathbf{1} \mid \mathrm{Prop} \mid \mathrm{Set} \mid \mathrm{Grpd} \mid *_3 \\
& \mid P \leftrightarrow Q \mid A \Leftrightarrow B \mid G \Leftrightarrow H \mid \mathrm{Grpd} \simeq_3 \mathrm{Grpd} \mid a \sim_e b \\
& \mid \mathtt{tt} \mid x \mid (\Pi x{:}A)B \mid (\Sigma x{:}A)B \mid \lambda x{:}A.t \mid st \mid (s, t) \mid \pi_1 t \mid \pi_2 t \\
& \mid *^* \mid \Pi^*[x, x', x^*]{:}A^*.B^* \mid \Sigma^*[x, x', x^*]{:}A^*.B^* \mid \simeq^* A^* B^* \\
& \mid \mathsf{r}(t) \mid e^+(s) \mid e^-(t) \mid e^=(s, t) \mid \overrightarrow{e}(s) \mid \overleftarrow{e}(t)
\end{aligned}
$$

$$
\boxed{\dfrac{A : *_k \qquad B : *_k}{A \simeq_k B : *_k}}
\qquad
\boxed{\dfrac{a : A \quad b : B \quad e : A \simeq_k B}{a \sim_e b : *_{k-1}}}
$$

$\lambda{\simeq_2}$

$$
\begin{aligned}
t, A, e ::= \;& \mathbf{1} \mid \mathsf{Prop} \mid \mathsf{Set} \mid \mathsf{Grpd} \mid *_3 \\
& \mid P \leftrightarrow Q \mid A \Leftrightarrow B \mid G \Leftrightarrow H \mid \mathsf{Grpd} \simeq_3 \mathsf{Grpd} \mid a \sim_e b \\
& \mid \mathtt{tt} \mid x \mid (\Pi x{:}A)B \mid (\Sigma x{:}A)B \mid \lambda x{:}A.t \mid st \mid (s,t) \mid \pi_1 t \mid \pi_2 t \\
& \mid *^* \mid \Pi^*[x,x',x^*]{:}A^*.B^* \mid \Sigma^*[x,x',x^*]{:}A^*.B^* \mid \simeq^* A^* B^* \\
& \mid \mathsf{r}(t) \mid e^+(s) \mid e^-(t) \mid e^=(s,t) \mid \overrightarrow{e}(s) \mid \overleftarrow{e}(t)
\end{aligned}
$$

$$
\boxed{\dfrac{A : *_k \qquad B : *_k}{A \simeq_k B : *_k}} \qquad \boxed{\dfrac{a : A \qquad b : B \qquad e : A \simeq_k B}{a \sim_e b : *_{k-1}}}
$$

$$
\begin{aligned}
*_2 &:= \mathsf{Grpd} & \qquad A \simeq_2 B &:= A \Leftrightarrow B \\
*_1 &:= \mathsf{Set} & A \simeq_1 B &:= A \Leftrightarrow B \\
*_0 &:= \mathsf{Prop} & A \simeq_0 B &:= A \leftrightarrow B \\
*_{-1} &:= \mathbf{1} & a \simeq_A a' &:= a \sim_{\mathsf{r}(A)} a'
\end{aligned}
$$

Typing rules

$$\frac{k \in \{0,1,2,3\}}{\vdash *_{k-1} : *_k}$$

$$\frac{\Gamma \vdash A : *_j \qquad \Gamma, x : A \vdash B : *_k \qquad j,k \in \{0,1,2\}}{\Gamma \vdash \Sigma x{:}A.B : *_{\max(j,k)}}$$

$$\frac{\Gamma \vdash A : *_j \qquad \Gamma, x : A \vdash B : *_k \qquad j,k \in \{0,1,2\}}{\Gamma \vdash \Pi x{:}A.B : \begin{cases} *_2 & j = 2 = k+1 \\ *_k & \text{otherwise} \end{cases}}$$

Typing rules

$$\frac{\Gamma \vdash A : *_k \qquad k \in \{0,1,2,3\}}{\Gamma, x : A \vdash x : A}$$

$$\frac{\Gamma \vdash M : A \qquad \Gamma \vdash B : *_k \qquad k \in \{0,1,2,3\}}{\Gamma, y : B \vdash M : A}$$

$$\frac{\Gamma \vdash A : *_k \quad \Gamma \vdash B : *_k \qquad \Gamma \vdash M : A \qquad A = B \quad k \in \{0,1,2\}}{\Gamma \vdash M : B}$$

Typing rules

$$\frac{e : A \simeq_k B \qquad k \in \{0,1,2\}}{\begin{array}{c} e^+ : A \to B \\ e^- : B \to A \end{array}}$$

$$\frac{a : A \qquad b : B \qquad e : A \simeq_k B \qquad k \in \{1,2\}}{\begin{array}{c} e^=(a,b) : (a \simeq_A e^-(b)) \simeq_{k-1} (e^+(a) \simeq_B b) \\ \overrightarrow{e}(a) : a \sim_e e^+(a) \\ \overleftarrow{e}(b) : e^-(b) \sim_e a \end{array}}$$

$$\frac{A : *_k \qquad a : A \qquad k \in \{0,1,2\}}{\mathrm{r}(a) : a \simeq_A a}$$

Substitutions

$$\frac{\Gamma, x{:}A \vdash t : B \qquad \Gamma \vdash a : A}{\Gamma \vdash t[a/x] : B[a/x]}$$

$$\boxed{\frac{\Gamma, \Delta \vdash t : B \qquad [\vec{a}/\vec{x}] \vdash \Gamma \Rightarrow \Delta}{\Gamma \vdash t[\vec{a}/\vec{x}] : B[\vec{a}/\vec{x}]}}$$

Substitutions

$$\frac{\Gamma, x{:}A \vdash t : B \qquad \Gamma \vdash a^* : a \simeq_A a'}{\Gamma \vdash t[a^*//x] : t[a/x] \sim_{B[a^*//x]} t[a'/x]}$$

$$\boxed{\frac{\Gamma, \Delta \vdash t : B \qquad [\vec{a}^*//\vec{x}] : [\vec{a}/\vec{x}] \simeq [\vec{a}'/\vec{x}] \vdash \Gamma \Rightarrow \Delta}{\Gamma \vdash t[\vec{a}^*//\vec{x}] : t[\vec{a}/\vec{x}] \sim_{B[\vec{a}^*//\vec{x}]} t[\vec{a}'/\vec{x}]}}$$

Substitutions

$$\frac{}{\between \vdash \Gamma \Rightarrow \varnothing}$$

$$\frac{}{\between^{*} : \between \simeq \between \vdash \Gamma \Rightarrow \varnothing}$$

Substitutions

$$\frac{[\vec{a}/\vec{x}] \vdash \Gamma \Rightarrow \Delta \qquad \Gamma, \Delta[\vec{a}/\vec{x}] \vdash b : B[\vec{a}/\vec{x}]}{[\vec{a}, b/\vec{x}, y] \vdash \Gamma \Rightarrow \Delta, y : B}$$

$$\frac{\begin{array}{l}[\vec{a}, b/\vec{x}, y], [\vec{a}', b'/\vec{x}, y] \vdash \Gamma \Rightarrow \Delta, y : B \\ [\vec{a}^*//\vec{x}] : [\vec{a}/\vec{x}] \simeq [\vec{a}'/\vec{x}] \vdash \Gamma \Rightarrow \Delta \\ \qquad\qquad\qquad \Gamma, \Delta[\vec{a}^*//\vec{x}] \vdash b^* : b \sim_{B[\vec{a}^*//\vec{x}]} b'\end{array}}{[\vec{a}^*, b^*//\vec{x}, y] : [\vec{a}, b/\vec{x}, y] \simeq [\vec{a}', b'/\vec{x}, y] \vdash \Gamma \Rightarrow \Delta, y : B}$$

The intended model

Our system has a natural set-theoretic semantics:

- $[\![\mathrm{Prop}]\!] = \{0, 1\}$;
- $[\![\mathrm{Set}]\!] = \boldsymbol{V}_\kappa$, for strongly inaccessible κ;
- $[\![\mathrm{Grpd}]\!]$ = the collection of (locally) κ-small groupoids;
- $p [\![\leftrightarrow]\!] q = \mathrm{Iff}(p, q)$;
- $A [\![\Leftrightarrow]\!] B = \mathrm{Iso}(A, B)$;
- $G [\![\Leftrightarrow]\!] H = \mathrm{Eq}(G, H)$;
- $[\![(\Pi x{:}A)B]\!] = \prod_{a \in [\![A]\!]} [\![B]\!]_{x:=a}$
- $[\![(\Sigma x{:}A)B]\!] = \bigsqcup_{a \in [\![A]\!]} [\![B]\!]_{x:=a}$

A strict model

Similar to the previous one, except

- $[\![\mathrm{Grpd}]\!] = \boldsymbol{V}_{\kappa'}$, where $\kappa' > \kappa$.
- $[\![\Leftrightarrow]\!] = [\![\Leftrightarrow]\!] = [\![\sim_e]\!] := \ (=_{\mathrm{ZF}})$
- This model actually validates the rule

$$\frac{a \sim_{\mathrm{r}(A)} b}{a = b}$$

A meta-theoretic fact

PROPOSITION. Let $\Gamma \vdash A : \mathsf{Prop}$. Let B be such that

$$\begin{array}{ll} & \Gamma, x{:}A, \Delta \vdash B : \mathsf{Set} \\ \text{or} & \Gamma, x{:}A, \Delta \vdash B : \mathsf{Grpd} \end{array}$$

Then B is convertible to a term where x does not occur.

Plan

- Background
- The system
- Formalization

The problem of definitional equalities

- A central issue arising in formalization of type theories is the interpretation of definitional equalities.
- One approach consists of interpreting *all* equalities propositionally — including beta conversion, and the substitution lemma

$$[\![M[N/x]]\!]_\rho = [\![M]\!]_{\rho, x \mapsto [\![N]\!]_\rho}$$

- The conversion rule is thereby *not* validated "on the nose": if A, B are convertible types, the interpretation of $M : A$ is *coerced* from $[\![A]\!]$ to $[\![B]\!]$ by identity elimination.
- Due to this mismatch, it becomes necessary to prove coherence of interpretation with respect to all equalities — major pain in the brain!

"Kipling"-style formalization

- In "Outrageous but meaningful coincidences", McBride (2008) shows how dependent type theory can be interpreted in itself preserving all definitional equalities.
- The heart of the idea: all binders in the language are indexed by the *interpretation* of their domains. In particular, types A in context Γ are families indexed by $[\![\Gamma]\!]$.
- In particular, if $x : A \vdash B(x) : *$, and $\vdash (a, a') : A \times A'$, then $[\![B(\pi_1(a, a'))]\!] = [\![B(a)]\!]$, since $[\![\pi_1(a, a')]\!] = [\![a]\!]$.
- Using McBride's technique, we formalized a strict interpretation of $\lambda{\simeq}$ into a semantic universe defined by induction-recursion.

Recognizing degeneracies

- Definitional equalities are instances of *degenerate paths*.
- Once these are added into the syntax (in the form of the $\mathrm{r}(t)$-constructor), a strict interpretation must preserve them as well.
- This requirement already raises problems when one wants to lift the model to the level of *setoids*.

Strict fibrations

- A *setoid* is a type A with an equivalence relation $\simeq_A : A \to A \to *$.
- A *fibration* of setoids consists of:
  - A family of setoids $B(x)$ indexed by $x : A$;
  - For each $e : a \simeq_A a'$, a setoid isomorphism

    $$B(e) : B(a) \simeq B(a')$$

  - The map $e \mapsto B(e)$ must be functorial.
  - In particular, $B(\mathrm{r}(a))$ should be *exactly* the identity isomorphism.
- This last condition is related to "decidability of degeneracies".

Our approach

- The contexts Γ are interpreted by *freely generated setoids*: these are given as the setoid of paths in a graph.
- The fibration of a setoid over a freely generated one need only specify isomorphisms over the generating edges.
- This data generates a strict fibration over the generated setoid.
- Context extension preserves the property of being freely generated.

Conclusion

- A new type system for reasoning about equalities up to the groupoid level.
- A strict formalization of $\lambda\simeq$.
- A strict formalization of the $\lambda\simeq_2$ up to setoids (in progress).
- A strict formalization of the $\lambda\simeq_2$ (prospective).

The Shoutout

- R. Gandy, *On the Axiom of Extensionality*, 1956
- T. Coquand, *Equality and dependent type theory.* `http://www.cse.chalmers.se/~coquand/equality.pdf`
- Tait, *Extensional equality in the classical theory of types*, 1995
- Altenkirch et al, *Observational type theory*, 2006
- McBride: *Outrageous but meaningful coincidences*, 2010
- Martin-Löf: Lectures given at CMU, March 2013