

predicate logic & dependent types

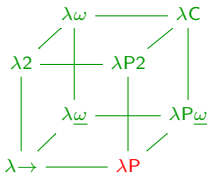
Freek Wiedijk

Type Theory & Coq

2023–2024

Radboud University Nijmegen

September 22, 2023



introduction

function types become dependent

for predicate logic one needs to generalize

$$(\lambda x : A. M) : A \rightarrow B$$

function types

to

$$(\lambda x : A. M) : \prod x : A. B$$

dependent function types

introduction

function types become dependent

for predicate logic one needs to generalize

$$(\lambda x : A. M) : A \rightarrow B$$

function types

to

can contain x

$$\downarrow$$
$$(\lambda x : A. M) : \prod x : A. B$$

dependent function types

introduction

function types become dependent

for predicate logic one needs to generalize

$$(\lambda x : A. M) : A \rightarrow B$$

function types

to

can contain x

$$\downarrow$$
$$(\lambda x : A. M) : \Pi x : A. B$$

dependent function types

$$\forall x : A. B$$

forall $x : A, B$

three different notations for the same type

we skip chapters 3 and 5 for now

inductive types



next week

program extraction



maybe last week

last week and this week

Curry-Howard correspondence:

propositional logic	$\lambda \rightarrow$	simple types
predicate logic	λP	dependent types
second order logic	$\lambda 2$	polymorphic types
beyond minimal logic	CIC	inductive types

last week and this week

Curry-Howard correspondence:

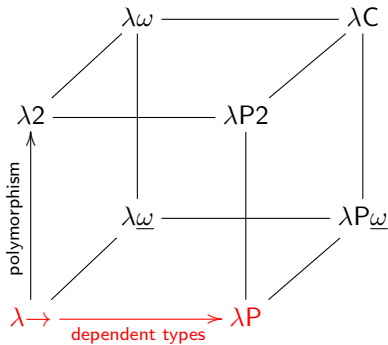
propositional logic	$\lambda \rightarrow$	simple types
predicate logic	λP	dependent types
second order logic	$\lambda 2$	polymorphic types
beyond minimal logic	CIC	inductive types

last week and this week

Curry-Howard correspondence:

propositional logic	$\lambda \rightarrow$	simple types
predicate logic	λP	dependent types
second order logic	$\lambda 2$	polymorphic types
beyond minimal logic	CIC	inductive types

dependent types in the lambda cube



recap: propositional logic

$A, B ::= a \mid A \rightarrow B \mid A \wedge B \mid A \vee B \mid \neg A \mid \top \mid \perp$

$I \rightarrow$	$E \rightarrow$
$I \wedge$	$E l \wedge \quad E r \wedge$
$I l \vee \quad I r \vee$	$E \vee$
$I \neg$	$E \neg$
$I \top$	$E \perp$

recap: simply typed lambda calculus

$$A, B ::= a \mid A \rightarrow B$$
$$M, N ::= x \mid MN \mid \lambda x : A. M$$
$$\frac{}{\Gamma \vdash x : A} \quad \text{for } (x : A) \in \Gamma$$
$$\frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash (\lambda x : A. M) : A \rightarrow B}$$
$$\frac{\Gamma \vdash F : A \rightarrow B \quad \Gamma \vdash M : A}{\Gamma \vdash FM : B}$$

predicate logic

three predicate logics

- ▶ minimal predicate logic

$$\rightarrow \forall$$

- ▶ constructive predicate logic

$$\rightarrow \wedge \vee \neg \top \perp \forall \exists$$

- ▶ classical predicate logic

$$A \vee \neg A$$
$$\neg\neg A \rightarrow A$$

propositional versus predicate logic: syntax

$$A, B ::= a \mid A \rightarrow B \mid A \wedge B \mid A \vee B \mid \neg A \mid \top \mid \perp$$

$$M, N ::= x \mid f(\vec{M})$$

$$\vec{M} ::= \cdot \mid \vec{M}, N$$

$$A, B ::= p(\vec{M}) \mid A \rightarrow B \mid A \wedge B \mid A \vee B \mid \neg A \mid \top \mid \perp \mid \\ \forall x. A \mid \exists x. A$$

\forall and \exists bind weakly

minimal propositional versus minimal predicate logic: rules

$$\begin{array}{c} [A^H] \\ \vdots \\ B \\ \hline A \rightarrow B \end{array} I[H] \rightarrow \qquad \begin{array}{c} \vdots \qquad \vdots \\ A \rightarrow B \qquad A \\ \hline B \end{array} E \rightarrow$$

$$\begin{array}{c} \vdots \\ A \\ \hline \forall x. A \end{array} I\forall \qquad \begin{array}{c} \vdots \\ \forall x. A \\ \hline A[x := M] \end{array} E\forall$$

variable condition of $I\forall$: x not free in available assumptions

same four rules with explicit assumptions

$$\frac{\Gamma, A \vdash B}{\Gamma \vdash A \rightarrow B} I_{\rightarrow}$$

$$\frac{\Gamma \vdash A \rightarrow B \quad \Gamma \vdash A}{\Gamma \vdash B} E_{\rightarrow}$$

$$\frac{\Gamma \vdash A}{\Gamma \vdash \forall x. A} I_{\forall}$$

$$\frac{\Gamma \vdash \forall x. A}{\Gamma \vdash A[x := M]} E_{\forall}$$

variable condition of I_{\forall} : x not free in Γ

rules for the existential quantifier

$$\frac{\vdots}{A[x := M]} I\exists \qquad \frac{\vdots \quad \exists x. A \quad \forall x. A \rightarrow C}{C} E\exists$$

variable condition of $E\exists$: x not free in C

example proof in minimal logic

$$\forall x. ((\forall y. p(y)) \rightarrow p(x))$$

example proof in minimal logic

$$\forall x. ((\forall y. p(y)) \rightarrow p(x))$$

$$\frac{\begin{array}{c} [A^H] \\ \vdots \\ B \end{array}}{A \rightarrow B} I[H] \rightarrow \quad \frac{\begin{array}{c} \vdots \\ A \end{array}}{\forall x. A} I\forall \quad \frac{\begin{array}{c} \vdots \\ \forall x. A \end{array}}{A[x := M]} E\forall$$

example proof in minimal logic

$$\frac{(\forall y. p(y)) \rightarrow p(x)}{\forall x. ((\forall y. p(y)) \rightarrow p(x))} I\forall$$

$$\frac{\begin{array}{c} [A^H] \\ \vdots \\ B \end{array}}{A \rightarrow B} I[H] \rightarrow \quad \frac{\begin{array}{c} \vdots \\ A \end{array}}{\forall x. A} I\forall \quad \frac{\begin{array}{c} \vdots \\ \forall x. A \end{array}}{A[x := M]} E\forall$$

example proof in minimal logic

$[\forall y. p(y)^H]$

$$\frac{\frac{p(x)}{(\forall y. p(y)) \rightarrow p(x)} I[H] \rightarrow}{\forall x. ((\forall y. p(y)) \rightarrow p(x))} I\forall$$

$$\frac{\begin{array}{c} [A^H] \\ \vdots \\ B \end{array}}{A \rightarrow B} I[H] \rightarrow \quad \frac{\begin{array}{c} \vdots \\ A \end{array}}{\forall x. A} I\forall \quad \frac{\begin{array}{c} \vdots \\ \forall x. A \end{array}}{A[x := M]} E\forall$$

example proof in minimal logic

$[\forall y. p(y)^H]$

$$\frac{\frac{\frac{\forall y. p(y)}{p(x)} E\forall}{(\forall y. p(y)) \rightarrow p(x)} I[H] \rightarrow}{\forall x. ((\forall y. p(y)) \rightarrow p(x))} I\forall$$

$$\frac{\frac{[A^H] \quad \vdots \quad B}{A \rightarrow B} I[H] \rightarrow \quad \frac{\vdots \quad A}{\forall x. A} I\forall \quad \frac{\vdots \quad \forall x. A}{A[x := M]} E\forall}{\quad}$$

example proof in minimal logic

$[\forall y. p(y)^H]$

$$\frac{\frac{\frac{[\forall y. p(y)^H]}{p(x)} E\forall}{(\forall y. p(y)) \rightarrow p(x)} I[H] \rightarrow}{\forall x. ((\forall y. p(y)) \rightarrow p(x))} I\forall$$

$$\frac{[\overset{\vdots}{A^H}]}{\overset{\vdots}{B}} I[H] \rightarrow \quad \frac{\overset{\vdots}{A}}{\forall x. A} I\forall \quad \frac{\overset{\vdots}{\forall x. A}}{A[x := M]} E\forall$$

example proof in minimal logic

$[\forall y. p(y)^H]$

$$\frac{\frac{\frac{[\forall y. p(y)^H]}{p(x)} E\forall}{(\forall y. p(y)) \rightarrow p(x)} I[H] \rightarrow}{\forall x. ((\forall y. p(y)) \rightarrow p(x))} I\forall$$

variable condition check:

at the $I\forall$ step the variable x does not occur in assumptions
(there are no available assumptions at that point)

example proof in minimal logic

$$\frac{\frac{\frac{[\forall y. p(y)^H]}{p(x)} E\forall}{(\forall y. p(y)) \rightarrow p(x)} I[H] \rightarrow}{\forall x. ((\forall y. p(y)) \rightarrow p(x))} I\forall$$

variable condition check:

at the $I\forall$ step the variable x does not occur in assumptions
(there are no available assumptions at that point)

example with existential quantifier

$$(\exists x. p(x)) \leftrightarrow \neg(\forall x. \neg p(x))$$

example with existential quantifier

$$A \leftrightarrow B := (A \rightarrow B) \wedge (B \rightarrow A)$$

$$(\exists x. p(x)) \leftrightarrow \neg(\forall x. \neg p(x))$$

$$(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))$$

$$\neg(\forall x. \neg p(x)) \rightarrow (\exists x. p(x))$$

example with existential quantifier

$$A \leftrightarrow B := (A \rightarrow B) \wedge (B \rightarrow A)$$

$$(\exists x. p(x)) \leftrightarrow \neg(\forall x. \neg p(x))$$

not constructively valid

$$(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))$$

$$\neg(\forall x. \neg p(x)) \rightarrow (\exists x. p(x))$$

example with existential quantifier

$$A \leftrightarrow B := (A \rightarrow B) \wedge (B \rightarrow A)$$

$$(\exists x. p(x)) \leftrightarrow \neg(\forall x. \neg p(x))$$

not constructively valid

$$\begin{aligned} & (\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x)) \\ & \neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x)) \end{aligned}$$

example from left to right

$$(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))$$

example from left to right

$$(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))$$

$$\frac{\begin{array}{c} [A^H] \\ \vdots \\ B \\ \hline A \rightarrow B \end{array} I[H] \rightarrow}{\quad} \quad \frac{\begin{array}{c} [A^H] \\ \vdots \\ \perp \\ \hline \neg A \end{array} I[H] \neg}{\quad} \quad \frac{\begin{array}{c} \vdots \\ \exists x. A \end{array} \quad \begin{array}{c} \vdots \\ \forall x. A \rightarrow C \end{array}}{C} E\exists$$

example from left to right

$[\exists x. p(x)]^{H_0}$

$$\frac{\neg(\forall x. \neg p(x))}{(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))} I[H_0] \rightarrow$$

$$\frac{\begin{array}{c} [A^H] \\ \vdots \\ B \end{array}}{A \rightarrow B} I[H] \rightarrow \quad \frac{\begin{array}{c} [A^H] \\ \vdots \\ \perp \end{array}}{\neg A} I[H] \neg \quad \frac{\begin{array}{c} \vdots \\ \exists x. A \end{array} \quad \begin{array}{c} \vdots \\ \forall x. A \rightarrow C \end{array}}{C} E\exists$$

example from left to right

$$[\exists x. p(x)^{H_0}] \quad [\forall x. \neg p(x)^{H_1}]$$

$$\frac{\frac{\perp}{\neg(\forall x. \neg p(x))} I[H_1]\neg}{(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))} I[H_0]\rightarrow$$

$$\frac{\begin{array}{c} [A^H] \\ \vdots \\ B \end{array}}{A \rightarrow B} I[H]\rightarrow \quad \frac{\begin{array}{c} [A^H] \\ \vdots \\ \perp \\ \neg A \end{array}}{I[H]\neg} \quad \frac{\begin{array}{c} \vdots \\ \exists x. A \end{array} \quad \begin{array}{c} \vdots \\ \forall x. A \rightarrow C \end{array}}{C} E\exists$$

example from left to right

$[\exists x. p(x)^{H_0}] \quad [\forall x. \neg p(x)^{H_1}]$

$$\frac{\exists x. p(x) \quad \forall x. p(x) \rightarrow \perp}{E\exists}$$

$$\frac{\perp}{\neg(\forall x. \neg p(x))} I[H_1]\neg$$

$$\frac{(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))}{I[H_0]\rightarrow}$$

$$\frac{\begin{array}{c} [A^H] \\ \vdots \\ B \end{array}}{A \rightarrow B} I[H]\rightarrow \quad \frac{\begin{array}{c} [A^H] \\ \vdots \\ \perp \end{array}}{\neg A} I[H]\neg \quad \frac{\begin{array}{c} \vdots \\ \exists x. A \\ \vdots \\ C \end{array} \quad \forall x. A \rightarrow C}{E\exists}$$

example from left to right

$[\exists x. p(x)^{H_0}] \quad [\forall x. \neg p(x)^{H_1}]$

$$\frac{[\exists x. p(x)^{H_0}] \quad \forall x. p(x) \rightarrow \perp}{\perp} E\exists$$

$$\frac{\perp}{\neg(\forall x. \neg p(x))} I[H_1]\neg$$

$$\frac{\neg(\forall x. \neg p(x))}{(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))} I[H_0]\rightarrow$$

$$\frac{[A^H] \quad \vdots \quad B}{A \rightarrow B} I[H]\rightarrow$$

$$\frac{[A^H] \quad \vdots \quad \perp}{\neg A} I[H]\neg$$

$$\frac{\vdots \quad \exists x. A \quad \forall x. A \rightarrow C \quad \vdots}{C} E\exists$$

example from left to right

$$[\exists x. p(x)^{H_0}] \quad [\forall x. \neg p(x)^{H_1}]$$

$$\frac{[\exists x. p(x)^{H_0}] \quad [\forall x. p(x) \rightarrow \perp^{H_1}]}{E\exists}$$

$$\frac{\perp}{\neg(\forall x. \neg p(x))} I[H_1]\neg$$

$$\frac{\neg(\forall x. \neg p(x))}{(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))} I[H_0]\rightarrow$$

$$\frac{[A^H] \quad \vdots \quad B}{A \rightarrow B} I[H]\rightarrow$$

$$\frac{[A^H] \quad \vdots \quad \perp}{\neg A} I[H]\neg$$

$$\frac{\vdots \quad \exists x. A \quad \vdots \quad \forall x. A \rightarrow C}{C} E\exists$$

example from left to right

$$\frac{[\exists x. p(x)^{H_0}] \quad [\forall x. p(x) \rightarrow \perp^{H_1}]}{E\exists} \frac{\perp}{\neg(\forall x. \neg p(x))} I[H_1]\neg}{(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))} I[H_0]\rightarrow$$

variable condition check:

at the $E\exists$ step the variable x does not occur in \perp

example from left to right

$$\frac{\frac{\frac{[\exists x. p(x)^{H_0}] \quad [\forall x. p(x) \rightarrow \perp^{H_1}]}{E\exists}}{\perp}}{\neg(\forall x. \neg p(x))} I[H_1]\neg}{(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))} I[H_0]\rightarrow$$

variable condition check:

at the $E\exists$ step the variable x does not occur in \perp

example from right to left

$$\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))$$

example from right to left

$$[\neg(\forall x. \neg p(x))^{H_0}]$$

$$\frac{\neg\neg(\exists x. p(x))}{\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))} I[H_0] \rightarrow$$

example from right to left

$$[\neg(\forall x. \neg p(x))^{H_0}] \quad [\neg(\exists x. p(x))^{H_1}]$$

$$\frac{\frac{\perp}{\neg\neg(\exists x. p(x))} I[H_1]\neg}{\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))} I[H_0]\rightarrow$$

example from right to left

$[\neg(\forall x. \neg p(x))^{H_0}] \quad [\neg(\exists x. p(x))^{H_1}]$

$$\frac{\frac{\neg(\forall x. \neg p(x)) \quad \forall x. \neg p(x)}{E_{\neg}}}{\frac{\frac{\perp}{\neg\neg(\exists x. p(x))} I[H_1]_{\neg}}{\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))} I[H_0]_{\rightarrow}}$$

example from right to left

$[\neg(\forall x. \neg p(x))^{H_0}]$ $[\neg(\exists x. p(x))^{H_1}]$

$$\frac{\frac{\frac{[\neg(\forall x. \neg p(x))^{H_0}] \quad \forall x. \neg p(x)}{E_{\neg}}}{\perp}}{\neg\neg(\exists x. p(x))} I[H_1]_{\neg}}{I[H_0]_{\rightarrow}} \neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))$$

example from right to left

$[\neg(\forall x. \neg p(x))^{H_0}]$ $[\neg(\exists x. p(x))^{H_1}]$

$$\frac{\frac{[\neg(\forall x. \neg p(x))^{H_0}] \quad \frac{\neg p(x)}{\forall x. \neg p(x)} I\forall}{E\neg}}{\perp} \quad \frac{\perp}{\neg\neg(\exists x. p(x))} I[H_1]\neg}{\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))} I[H_0]\rightarrow$$

example from right to left

$[\neg(\forall x. \neg p(x))^{H_0}]$ $[\neg(\exists x. p(x))^{H_1}]$ $[p(x)^{H_2}]$

$$\frac{\frac{\frac{\perp}{\neg p(x)} I[H_2] \neg}{\forall x. \neg p(x)} I\forall}{[\neg(\forall x. \neg p(x))^{H_0}] E\neg} \frac{\perp}{\neg\neg(\exists x. p(x))} I[H_1] \neg}{\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))} I[H_0] \rightarrow$$

example from right to left

$[\neg(\forall x. \neg p(x))^{H_0}]$ $[\neg(\exists x. p(x))^{H_1}]$ $[p(x)^{H_2}]$

$$\begin{array}{c}
 \frac{\neg(\exists x. p(x)) \quad \exists x. p(x)}{E \rightarrow} \\
 \frac{\frac{\perp}{\neg p(x)} I[H_2] \neg}{\forall x. \neg p(x)} I \forall \\
 \frac{[\neg(\forall x. \neg p(x))^{H_0}] \quad \forall x. \neg p(x)}{E \neg} \\
 \frac{\perp}{\neg \neg(\exists x. p(x))} I[H_1] \neg \\
 \frac{\neg(\forall x. \neg p(x)) \rightarrow \neg \neg(\exists x. p(x))}{I[H_0] \rightarrow}
 \end{array}$$

example from right to left

$$[\neg(\forall x. \neg p(x))^{H_0}] \quad [\neg(\exists x. p(x))^{H_1}] \quad [p(x)^{H_2}]$$

$$\frac{\frac{\frac{[\neg(\exists x. p(x))^{H_1}] \quad \exists x. p(x)}{E \rightarrow} \quad \perp}{\neg p(x)} \quad I[H_2] \neg}{\forall x. \neg p(x)} \quad I \forall}{[\neg(\forall x. \neg p(x))^{H_0}] \quad \forall x. \neg p(x)} \quad E \neg}{\perp} \quad I[H_1] \neg}{\neg \neg (\exists x. p(x))} \quad I[H_0] \rightarrow}{\neg(\forall x. \neg p(x)) \rightarrow \neg \neg (\exists x. p(x))}$$

example from right to left

$[\neg(\forall x. \neg p(x))^{H_0}]$ $[\neg(\exists x. p(x))^{H_1}]$ $[p(x)^{H_2}]$

$$\begin{array}{c}
 \frac{[\neg(\exists x. p(x))^{H_1}] \quad \frac{p(x)}{\exists x. p(x)} I\exists}{E\rightarrow} \\
 \frac{\perp}{\neg p(x)} I[H_2]\neg \\
 \frac{[\neg(\forall x. \neg p(x))^{H_0}] \quad \frac{\forall x. \neg p(x)}{E\neg}}{I\forall} \\
 \frac{\perp}{\neg\neg(\exists x. p(x))} I[H_1]\neg \\
 \frac{\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))}{I[H_0]\rightarrow}
 \end{array}$$

example from right to left

$[\neg(\forall x. \neg p(x))^{H_0}]$ $[\neg(\exists x. p(x))^{H_1}]$ $[p(x)^{H_2}]$

$$\begin{array}{c}
 \frac{[\neg(\exists x. p(x))^{H_1}] \quad \frac{[p(x)^{H_2}]}{\exists x. p(x)} I\exists}{E\rightarrow} \\
 \frac{\perp}{\neg p(x)} I[H_2]\neg \\
 \frac{[\neg(\forall x. \neg p(x))^{H_0}] \quad \frac{\forall x. \neg p(x)}{I\forall}}{E\neg} \\
 \frac{\perp}{\neg\neg(\exists x. p(x))} I[H_1]\neg \\
 \frac{\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))}{I[H_0]\rightarrow}
 \end{array}$$

example from right to left

$$[\neg(\forall x. \neg p(x))^{H_0}] \quad [\neg(\exists x. p(x))^{H_1}] \quad [p(x)^{H_2}]$$

$$\frac{\frac{\frac{[\neg(\exists x. p(x))^{H_1}] \quad \frac{[p(x)^{H_2}]}{\exists x. p(x)} I\exists}{E\rightarrow}}{\perp} \quad \frac{\perp}{\neg p(x)} I[H_2]\neg}{\forall x. \neg p(x)} I\forall}{[\neg(\forall x. \neg p(x))^{H_0}] \quad \forall x. \neg p(x)} E\neg}{\frac{\perp}{\neg\neg(\exists x. p(x))} I[H_1]\neg} \quad \frac{\neg\neg(\exists x. p(x))}{\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))} I[H_0]\rightarrow}$$

variable condition check:

at the $I\forall$ step the variable x does not occur in assumptions

example from right to left

$$\begin{array}{c}
 \frac{[\neg(\exists x. p(x))^{H_1}] \quad \frac{[p(x)^{H_2}]}{\exists x. p(x)} I\exists}{E\rightarrow} \\
 \frac{\perp}{\neg p(x)} I[H_2]\neg \\
 \frac{[\neg(\forall x. \neg p(x))^{H_0}] \quad \frac{\forall x. \neg p(x)}{I\forall}}{E\neg} \\
 \frac{\perp}{\neg\neg(\exists x. p(x))} I[H_1]\neg \\
 \frac{\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))}{I[H_0]\rightarrow}
 \end{array}$$

variable condition check:

at the $I\forall$ step the variable x does not occur in assumptions

dependent type theory

types thus far

- ▶ atomic types
 a, b, c, \dots
- ▶ types of proof objects of propositions
 a, b, c, \dots
- ▶ function types
- ▶ datatypes
 - ▶ natural numbers
 - ▶ Booleans
 - ▶ lists
 - ▶ binary trees
 - ▶ ...

how are types introduced?

- ▶ free type variables
STT = simple type theory

how are types introduced?

- ▶ free type variables

STT = simple type theory

- ▶ in the context

PTSs = pure type systems: $\lambda \rightarrow$ λP $\lambda 2$ λC

how are types introduced?

- ▶ free type variables

STT = simple type theory

- ▶ in the context

PTSs = pure type systems: $\lambda \rightarrow$ λP $\lambda 2$ λC

$$\Gamma \vdash M : A$$

$$O : \text{nat}, S : \text{nat} \rightarrow \text{nat} \vdash S (S (S O)) : \text{nat}$$

how are types introduced?

- ▶ free type variables

STT = simple type theory

- ▶ in the context

PTSs = pure type systems: $\lambda \rightarrow$ λP $\lambda 2$ λC

$$\Gamma \vdash M : A$$

$\text{nat} : *$, $O : \text{nat}$, $S : \text{nat} \rightarrow \text{nat} \vdash S (S (S O)) : \text{nat}$

how are types introduced?

- ▶ free type variables

STT = simple type theory

- ▶ in the context = 'axiomatic'

PTSs = pure type systems: $\lambda \rightarrow$ λP $\lambda 2$ λC

$$\Gamma \vdash M : A$$

$\text{nat} : *$, $O : \text{nat}$, $S : \text{nat} \rightarrow \text{nat} \vdash S (S (S O)) : \text{nat}$

Coq: Parameter

how are types introduced?

- ▶ free type variables

STT = simple type theory

- ▶ in the context = 'axiomatic'

PTSs = pure type systems: $\lambda \rightarrow$ λP $\lambda 2$ λC

$$\Gamma \vdash M : A$$

$\text{nat} : *$, $O : \text{nat}$, $S : \text{nat} \rightarrow \text{nat} \vdash S (S (S O)) : \text{nat}$

Coq: Parameter

- ▶ definitions \rightarrow next week

CIC = Calculus of Inductive Constructions

$$E[\Gamma] \vdash M : A$$

Coq: Definition Inductive

star and box

the **sorts** of the lambda cube:

* = the type of types

□ = the type of *

star and box

the **sorts** of the lambda cube:

* = the type of types

□ = the type of *

the sort * is a **kind**
= third of four levels

object : type

type : kind

kind : □

object : type : **kind** : □

S : nat → nat : * : □

example contexts

natural numbers:

nat : *

O : nat

S : nat \rightarrow nat

add : nat \rightarrow nat \rightarrow nat

Booleans:

bool : *

true : bool

false : bool

lists

lists of Booleans:

`list : *`

`nil : list`

`cons : bool → list → list`

`hd : list → bool`

`tl : list → list`

`append : list → list → list`

`reverse : list → list`

lists

lists of Booleans:

```
list : *
nil : list
cons : bool → list → list
hd : list → bool
tl : list → list
append : list → list → list
reverse : list → list
```

cons has two arguments
(the head and tail to be put together)

trueslist

the function 'trueslist' maps a number n to a list of length n consisting of n copies of 'true'

trueslist 0 = nil

trueslist (S 0) = cons true nil

trueslist (S (S 0)) = cons true (cons true nil)

trueslist (S (S (S 0))) = cons true (cons true (cons true nil))

...

trueslist : nat \rightarrow list

vectors and truesvec

vectors of Booleans:

`vec : nat → *`

`vec 0 : *` vectors of length 0

`vec (S 0) : *` vectors of length 1

`vec (S (S 0)) : *` vectors of length 2

...

vectors and truesvec

vectors of Booleans:

`vec : nat → *`

`vec O : *` vectors of length 0

`vec (S O) : *` vectors of length 1

`vec (S (S O)) : *` vectors of length 2

...

`truestlist : nat → list`

`truestvec : nat → vec`

vectors and truesvec

vectors of Booleans:

$\text{vec} : \text{nat} \rightarrow *$

$\text{vec } 0 : *$ vectors of length 0

$\text{vec } (\text{S } 0) : *$ vectors of length 1

$\text{vec } (\text{S } (\text{S } 0)) : *$ vectors of length 2

...

$\text{trueslist} : \text{nat} \rightarrow \text{list}$

$\text{truesvec} : \text{nat} \rightarrow \text{vec } n$

vectors and truesvec

vectors of Booleans:

$\text{vec} : \text{nat} \rightarrow *$

$\text{vec } 0 : *$ vectors of length 0

$\text{vec } (\text{S } 0) : *$ vectors of length 1

$\text{vec } (\text{S } (\text{S } 0)) : *$ vectors of length 2

...

$\text{trueslist} : \text{nat} \rightarrow \text{list}$

$\text{truesvec} : \text{nat} \rightarrow \text{vec } n$

$\text{truesvec} : \prod n : \text{nat}. \text{vec } n$

vnil and vcons

$\text{vnil} : \text{vec } 0$

$\text{vcons} : \text{bool} \rightarrow \text{vec } n \rightarrow \text{vec } (\text{S } n)$

vnil and vcons

$\text{vnil} : \text{vec } 0$

$\text{vcons} : \text{nat} \rightarrow \text{bool} \rightarrow \text{vec } n \rightarrow \text{vec } (\text{S } n)$

vnil and vcons

$\text{vnil} : \text{vec } 0$

$\text{vcons} : \text{nat} \rightarrow \text{bool} \rightarrow \text{vec } n \rightarrow \text{vec } (\text{S } n)$

$\text{vcons} : \prod n : \text{nat}. \text{bool} \rightarrow \text{vec } n \rightarrow \text{vec } (\text{S } n)$

vnil and vcons

$\text{vnil} : \text{vec } 0$

$\text{vcons} : \text{nat} \rightarrow \text{bool} \rightarrow \text{vec } n \rightarrow \text{vec } (\text{S } n)$

$\text{vcons} : \Pi n : \text{nat}. \text{bool} \rightarrow \text{vec } n \rightarrow \text{vec } (\text{S } n)$

vcons has three arguments!

vnil and vcons

vnil : vec 0

vcons : nat → bool → vec n → vec (S n)

vcons : $\Pi n : \text{nat}. \text{bool} \rightarrow \text{vec } n \rightarrow \text{vec } (\text{S } n)$

vcons has three arguments!

truesvec (S (S 0)) : vec (S (S 0))

truesvec (S (S 0)) = vcons (S 0) true (vcons 0 true vnil)

types when applying vcons

$\text{truesvec } (S\ O) = \text{vcons } O\ \text{true}\ \text{vnil}$

$\text{truesvec} : \prod n : \text{nat}. \text{vec } n$

$\text{truesvec } (S\ O) : \text{vec } (S\ O)$

$\text{vcons} : \prod n : \text{nat}. \text{bool} \rightarrow \text{vec } n \rightarrow \text{vec } (S\ n)$

$O : \text{nat}$

$\text{vcons } O : \text{bool} \rightarrow \text{vec } O \rightarrow \text{vec } (S\ O)$

$\text{true} : \text{bool}$

$\text{vcons } O\ \text{true} : \text{vec } O \rightarrow \text{vec } (S\ O)$

$\text{vnil} : \text{vec } O$

$\text{vcons } O\ \text{true}\ \text{vnil} : \text{vec } (S\ O)$

BHK-interpretation

proof of $A \rightarrow B$

proof of $\neg A$

proof of $A \wedge B$

proof of $A \vee B$

function from proofs of A to proofs of B

function from proofs of A to proofs of \perp

pair of a proof of A and a proof of B

either a proof of A and a proof of B

Coq

BHK-interpretation

proof of $A \rightarrow B$	function from proofs of A to proofs of B
proof of $\neg A$	function from proofs of A to proofs of \perp
proof of $A \wedge B$	pair of a proof of A and a proof of B
proof of $A \vee B$	either a proof of A and a proof of B
proof of $\forall x. A$	dependent function from objects to proofs of A
proof of $\exists x. A$	dependent pair of an object and a proof of A

dependent = A is not fixed but *depends* on the object x

Coq

BHK-interpretation

proof of $A \rightarrow B$	function from proofs of A to proofs of B
proof of $\neg A$	function from proofs of A to proofs of \perp
proof of $A \wedge B$	pair of a proof of A and a proof of B
proof of $A \vee B$	either a proof of A and a proof of B
proof of $\forall x. A$	dependent function from objects to proofs of A
proof of $\exists x. A$	dependent pair of an object and a proof of A

dependent = A is not fixed but *depends* on the object x

$$A \vee B \leftrightarrow \exists x \in \{\text{left}, \text{right}\}. (x = \text{left} \rightarrow A) \wedge (x = \text{right} \rightarrow B)$$

Curry-Howard correspondence

type theory

$A \rightarrow B$

$\Pi x : D. A$

Coq

$A \rightarrow B$

`forall x : D, A`

minimal logic

$A \rightarrow B$

$\forall x. A$

Curry-Howard correspondence

type theory

$A \rightarrow B$

$\Pi x : D. A$

Coq

$A \rightarrow B$

`forall x : D, A`

minimal logic

$A \rightarrow B$

$\forall x. A$

Curry-Howard correspondence

type theory

$A \rightarrow B$

$\Pi x : D. A$

Coq

$A \rightarrow B$

`forall x : D, A`

minimal logic

$A \rightarrow B$

$\forall x. A$

$D : *$ is called **Terms** in the notes

Curry-Howard correspondence

type theory	Coq	minimal logic
$A \rightarrow B$	$A \rightarrow B$	$A \rightarrow B$
$\prod x : D. A$	<code>forall x : D, A</code>	$\forall x. A$

$D : *$ is called `Terms` in the notes

proof term	type	statement proved
H	$: A \rightarrow B$	$A \rightarrow B$
H'	$: A$	A
HH'	$: B$	B

Curry-Howard correspondence

type theory	Coq	minimal logic
$A \rightarrow B$	$A \rightarrow B$	$A \rightarrow B$
$\prod x : D. A$	<code>forall x : D, A</code>	$\forall x. A$

$D : *$ is called Terms in the notes

proof term	type	statement proved
H	$A \rightarrow B$	$A \rightarrow B$
H'	A	A
HH'	B	B
H	$\prod x : D. Px$	$\forall x. P(x)$
HM	PM	$P(M)$

Coq version of the first example

Parameter D : Set.

D is declared

Coq version of the first example

```
Parameter D : Set.                p is declared
Parameter p : D -> Prop.
```

Coq version of the first example

Parameter D : Set.

1 subgoal (ID 1)

Parameter p : D -> Prop.

Lemma four :

forall x : D,
 (forall y : D, p y) ->
 p x.

=====
forall x : D,
 (forall y : D, p y) -> p x

$$\forall x. (\forall y. p(y)) \rightarrow p(x)$$

Coq version of the first example

```
Parameter D : Set.  
Parameter p : D -> Prop.
```

```
Lemma four :  
  forall x : D,  
    (forall y : D, p y) ->  
    p x.  
intros x H.
```

```
1 subgoal (ID 3)
```

```
x : D
```

```
H : forall y : D, p y
```

```
=====
```

```
p x
```

$$\frac{\frac{p(x)}{(\forall y. p(y)) \rightarrow p(x)} I[H] \rightarrow}{\forall x. (\forall y. p(y)) \rightarrow p(x)} I\forall$$

Coq version of the first example

Parameter D : Set.

No more subgoals.

Parameter p : D -> Prop.

Lemma four :

forall x : D,

(forall y : D, p y) ->

p x.

intros x H.

apply H.

$$\frac{\frac{\frac{[\forall y. p(y)^H]}{p(x)} E\forall}{(\forall y. p(y)) \rightarrow p(x)} I[H] \rightarrow}{\forall x. (\forall y. p(y)) \rightarrow p(x)} I\forall$$

Coq version of the first example

Parameter D : Set.

Parameter p : D -> Prop.

Lemma four :

```
forall x : D,  
  (forall y : D, p y) ->  
  p x.
```

```
intros x H.
```

```
apply H.
```

Qed.

$$\frac{\frac{\frac{[\forall y. p(y)^H]}{p(x)} E\forall}{(\forall y. p(y)) \rightarrow p(x)} I[H] \rightarrow}{\forall x. (\forall y. p(y)) \rightarrow p(x)} I\forall$$

Coq version of the first example

```
Parameter D : Set.
Parameter p : D -> Prop.

Lemma four :
  forall x : D,
    (forall y : D, p y) ->
      p x.
intros x H.
apply H.
Qed.
```

four =

```
fun (x : D) (H : forall y : D, p y)
=> H x
      : forall x : D,
        (forall y : D, p y) -> p x
```

Arguments four _ _%function_scope

Print four.

$$\frac{\frac{\frac{[\forall y. p(y)^H]}{p(x)} E\forall}{(\forall y. p(y)) \rightarrow p(x)} I[H] \rightarrow}{\forall x. (\forall y. p(y)) \rightarrow p(x)} I\forall$$

$(\lambda x : D. \lambda H : (\Pi y : D. p y). H x) : \Pi x : D. (\Pi y : D. p y) \rightarrow p x$

Coq version of the first example

```
Parameter D : Set.  
Parameter p : D -> Prop.
```

```
Lemma four :  
  forall x : D,  
    (forall y : D, p y) ->  
      p x.  
intros x H.  
apply H.  
Qed.
```

```
Print four.
```

```
four =  
fun (x : D) (H : forall y : D, p y)  
=> H x
```

```
: forall x : D,  
  (forall y : D, p y) -> p x
```

```
Arguments four _ _%function_scope
```

$$\frac{\frac{\frac{[\forall y. p(y)^H]}{p(x)} E\forall}{(\forall y. p(y)) \rightarrow p(x)} I[H] \rightarrow}{\forall x. (\forall y. p(y)) \rightarrow p(x)} I\forall$$

$(\lambda x : D. \lambda H : (\Pi y : D. p y). H x) : \Pi x : D. (\Pi y : D. p y) \rightarrow p x$

understanding the proof term

$$\forall x. (\forall y. p(y)) \rightarrow p(x)$$

understanding the proof term

$$\forall x. (\forall y. p(y)) \rightarrow p(x)$$

$$\lambda x : D.$$

understanding the proof term

$$\forall x. (\forall y. p(y)) \rightarrow p(x)$$

$$\lambda x : D. \lambda H : \underbrace{(\Pi y : D. py)}_{\forall y. p(y)}.$$

understanding the proof term

$$\forall x. (\forall y. p(y)) \rightarrow p(x)$$

$$\lambda x : D. \lambda H : (\Pi y : D. py). \underbrace{Hx}_{: px}$$

understanding the proof term

$$\forall x. (\forall y. p(y)) \rightarrow p(x)$$

$$\lambda x : D. \lambda H : (\Pi y : D. py). Hx$$

the second example in Coq, from left to right

Parameter D : Set.

1 subgoal (ID 3)

Parameter p : D -> Prop.

Lemma five :

(exists x : D, p x) ->
~ (forall x : D, ~ p x).

=====
(exists x : D, p x) ->
~ (forall x : D, ~ p x)

$$(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))$$

the second example in Coq, from left to right

```
Parameter D : Set.                1 subgoal (ID 6)
Parameter p : D -> Prop.
                                   H0 : exists x : D, p x
                                   H1 : forall x : D, ~ p x
                                   =====
                                   False

Lemma five :
  (exists x : D, p x) ->
  ~ (forall x : D, ~ p x).
intros H0 H1.
```

$$\frac{\frac{\perp}{\neg(\forall x. \neg p(x))} I[H_1]\neg}{(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))} I[H_0]\rightarrow$$

the second example in Coq, from left to right

```
Parameter D : Set.  
Parameter p : D -> Prop.  
  
Lemma five :  
  (exists x : D, p x) ->  
  ~ (forall x : D, ~ p x).  
intros H0 H1.  
elim H0.
```

1 subgoal (ID 7)

H0 : exists x : D, p x

H1 : forall x : D, ~ p x

=====
forall x : D, p x -> False

$$\frac{[\exists x. p(x)^{H_0}] \quad \forall x. p(x) \rightarrow \perp}{\perp} E\exists$$
$$\frac{\perp}{\neg(\forall x. \neg p(x))} I[H_1]\neg$$
$$\frac{\neg(\forall x. \neg p(x))}{(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))} I[H_0]\rightarrow$$

the second example in Coq, from left to right

Parameter D : Set.

No more subgoals.

Parameter p : D -> Prop.

Lemma five :

(exists x : D, p x) ->
~ (forall x : D, ~ p x).

intros H0 H1.

elim H0.

apply H1.

$$\frac{\frac{\frac{[\exists x. p(x)]^{H_0} \quad [\forall x. p(x) \rightarrow \perp]^{H_1}}{E\exists}}{\perp}}{\neg(\forall x. \neg p(x))} I[H_1]\neg}{(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))} I[H_0]\rightarrow$$

the second example in Coq, from left to right

```
Parameter D : Set.  
Parameter p : D -> Prop.
```

```
Lemma five :  
  (exists x : D, p x) ->  
  ~ (forall x : D, ~ p x).  
intros H0 H1.  
elim H0.  
apply H1.  
Qed.
```

$$\frac{\frac{[\exists x. p(x)^{H_0}] \quad [\forall x. p(x) \rightarrow \perp^{H_1}]}{E\exists} \quad \frac{\perp}{\neg(\forall x. \neg p(x))} I[H_1]\neg}{(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))} I[H_0]\rightarrow$$

the second example in Coq, from left to right

```
Parameter D : Set.  
Parameter p : D -> Prop.
```

```
Lemma five :
```

```
  (exists x : D, p x) ->  
  ~ (forall x : D, ~ p x).
```

```
intros H0 H1.
```

```
elim H0.
```

```
apply H1.
```

```
Qed.
```

$$\frac{\frac{\frac{[\exists x. p(x)]^{H_0} \quad [\forall x. p(x) \rightarrow \perp]^{H_1}}{E\exists}}{\perp}}{\neg(\forall x. \neg p(x))} I[H_1]\neg}{(\exists x. p(x)) \rightarrow \neg(\forall x. \neg p(x))} I[H_0]\rightarrow$$

the second example in Coq, from right to left

Parameter D : Set.

1 subgoal (ID 5)

Parameter p : D -> Prop.

Lemma six :

~ (forall x : D, ~ p x) ->

~ ~ (exists x : D, p x).

=====

~ (forall x : D, ~ p x) ->

~ ~ (exists x : D, p x)

$$\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))$$

the second example in Coq, from right to left

```
Parameter D : Set.                1 subgoal (ID 8)
Parameter p : D -> Prop.
Lemma six :
  ~ (forall x : D, ~ p x) ->      H0 : ~ (forall x : D, ~ p x)
  ~ ~ (exists x : D, p x).       H1 : ~ (exists x : D, p x)
                                  =====
                                  False
intros H0 H1.
```

$$\frac{\frac{\perp}{\neg\neg(\exists x. p(x))} I[H_1]\neg}{\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))} I[H_0]\rightarrow$$

the second example in Coq, from right to left

```
Parameter D : Set.                1 subgoal (ID 9)
Parameter p : D -> Prop.

Lemma six :                       H0 : ~ (forall x : D, ~ p x)
  ~ (forall x : D, ~ p x) ->     H1 : ~ (exists x : D, p x)
  ~ ~ (exists x : D, p x).       =====
                                  forall x : D, ~ p x

intros H0 H1.
apply H0.
```

$$\frac{\frac{\frac{[\neg(\forall x. \neg p(x))^{H_0}] \quad \forall x. \neg p(x)}{E_{\neg}}}{\perp}}{\neg\neg(\exists x. p(x))} I[H_1]_{\neg}}{\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))} I[H_0]_{\rightarrow}$$

the second example in Coq, from right to left

```
Parameter D : Set.                1 subgoal (ID 12)
Parameter p : D -> Prop.

Lemma six :
  ~ (forall x : D, ~ p x) ->      H0 : ~ (forall x : D, ~ p x)
  ~ ~ (exists x : D, p x).       H1 : ~ (exists x : D, p x)
                                  x : D
                                  H2 : p x
intros H0 H1.                    =====
apply H0.                        False
intros x H2.
```

$$\frac{\frac{\frac{\perp}{\neg p(x)} I[H_2] \neg}{\forall x. \neg p(x)} I\forall}{[\neg(\forall x. \neg p(x))]^{H_0}} E\neg}{\frac{\frac{\perp}{\neg\neg(\exists x. p(x))} I[H_1] \neg}{\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))} I[H_0] \rightarrow}$$

the second example in Coq, from right to left

Parameter D : Set.

1 subgoal (ID 15)

Parameter p : D -> Prop.

Lemma six :

~ (forall x : D, ~ p x) ->

~ ~ (exists x : D, p x).

intros H0 H1.

apply H0.

intros x H2.

apply H1.

exists x.

H0 : ~ (forall x : D, ~ p x)

H1 : ~ (exists x : D, p x)

x : D

H2 : p x

=====

p x

$$\begin{array}{c}
 \frac{\frac{p(x)}{\exists x. p(x)} I\exists}{[\neg(\exists x. p(x))]^{H_1}} E\rightarrow \\
 \frac{\perp}{\neg p(x)} I[H_2]\neg \\
 \frac{[\neg(\forall x. \neg p(x))]^{H_0}}{\forall x. \neg p(x)} I\forall \\
 \frac{\perp}{\neg\neg(\exists x. p(x))} I[H_1]\neg \\
 \frac{\frac{[\neg(\forall x. \neg p(x))]^{H_0}}{\forall x. \neg p(x)} I\forall}{\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))} I[H_0]\rightarrow
 \end{array}$$

the second example in Coq, from right to left

Parameter D : Set.

No more subgoals.

Parameter p : D -> Prop.

Lemma six :

~ (forall x : D, ~ p x) ->

~ ~ (exists x : D, p x).

intros H0 H1.

apply H0.

intros x H2.

apply H1.

exists x.

apply H2.

$$\frac{\frac{[\neg(\exists x. p(x))^{H_1}] \quad \frac{[p(x)^{H_2}]}{\exists x. p(x)} I\exists}{E\rightarrow}}{\perp} \quad \frac{\frac{\frac{[\neg(\forall x. \neg p(x))^{H_0}] \quad \frac{\frac{\perp}{\neg p(x)} I[H_2]\neg}{\forall x. \neg p(x)} I\forall}{E\neg}}{\perp}}{\neg\neg(\exists x. p(x))} I[H_1]\neg}{\neg(\forall x. \neg p(x)) \rightarrow \neg\neg(\exists x. p(x))} I[H_0]\rightarrow$$

the second example in Coq, from right to left

Parameter D : Set.

Parameter p : D -> Prop.

Lemma six :

~ (forall x : D, ~ p x) ->

~ ~ (exists x : D, p x).

intros H0 H1.

apply H0.

intros x H2.

apply H1.

exists x.

apply H2.

Qed.

$$\frac{\frac{[\neg(\exists x. p(x))^{H_1}]}{\frac{\frac{\perp}{\neg p(x)} I[H_2] \neg}{\forall x. \neg p(x)} I\forall}{\exists x. p(x)} I\exists}{E\rightarrow} \frac{\frac{[\neg(\forall x. \neg p(x))^{H_0}]}{\frac{\frac{\perp}{\neg\neg(\exists x. p(x))} I[H_1] \neg}{\neg\neg(\exists x. p(x))} I[H_0] \rightarrow}}{E\neg} \frac{\frac{\frac{\perp}{\neg p(x)} I[H_2] \neg}{\forall x. \neg p(x)} I\forall}{\exists x. p(x)} I\exists}{E\rightarrow}$$

the second example in Coq, from right to left

Parameter D : Set.

Parameter p : D -> Prop.

Lemma six :

~ (forall x : D, ~ p x) ->

~ ~ (exists x : D, p x).

intros H0 H1.

apply H0.

intros x H2.

apply H1.

exists x.

apply H2.

Qed.

$$\frac{\frac{\frac{[\neg(\exists x. p(x))^{H_1}]}{\perp} I[H_2] \neg}{\neg p(x)} I \forall}{\forall x. \neg p(x)} E \neg}{\frac{[\neg(\forall x. \neg p(x))^{H_0}]}{\perp} I[H_1] \neg}{\neg \neg(\exists x. p(x))} I[H_0] \rightarrow} E \rightarrow$$

proof rules versus Coq tactics

$I \rightarrow$	$I\forall$	intro intros	
$E \rightarrow$	$E\forall$	apply	
$E\wedge$	$E\vee$	$E\exists$	elim destruct intro-patterns
	$I\wedge$	split	
	$I\vee$	left right	
	$I\exists$	exists	

detours

proof normalization

detour = introduction rule directly followed by a elimination rule for the same connective

detours

proof normalization

detour = **introduction** rule directly followed by a **elimination** rule for the same connective

cut = corresponding notion in sequent calculus

can be **eliminated** \longrightarrow reduction of the proof term

detours

proof normalization

detour = **introduction** rule directly followed by a **elimination** rule for the same connective

cut = corresponding notion in sequent calculus

can be **eliminated** \rightarrow reduction of the proof term

detour for implication:

'prove a **lemma** A and then prove B using this lemma'

detour elimination is inlining the proof₁ of the lemma everywhere

$$\frac{\frac{\frac{[A^H]}{\vdots_2} B}{A \rightarrow B} I[H] \rightarrow \quad \frac{\vdots_1}{A} E \rightarrow}{B} \rightarrow_{\beta} \quad \frac{\vdots_1}{A} \vdots_2 B$$

detours for predicate logic

detour for the universal quantifier

generalize the statement $A[x := M]$ to A with arbitrary x
elimination is specializing the proof₁ to M

$$\frac{\frac{\vdots_1}{A} \text{IV}}{\forall x. A} \text{EV} \quad \longrightarrow_{\beta} \quad \frac{\vdots_1[x := M]}{A[x := M]}$$

$$(\lambda x : D. H_1)M \rightarrow_{\beta} H_1[x := M]$$

λP

STT versus λP syntax

STT:

$$\begin{aligned} A, B &::= a \mid A \rightarrow B \\ M, N &::= x \mid MN \mid \lambda x : A. M \\ \Gamma &::= \cdot \mid \Gamma, x : A \\ \mathcal{J} &::= \Gamma \vdash M : A \end{aligned}$$

λP :

$$\begin{aligned} s &::= * \mid \square \\ M, N, A, B &::= x \mid MN \mid \lambda x : A. M \mid \overline{\Pi x : A. B} \mid s \\ \Gamma &::= \cdot \mid \Gamma, x : A \\ \mathcal{J} &::= \Gamma \vdash M : A \end{aligned}$$

λP

STT versus λP syntax

STT:

$$\begin{aligned} A, B &::= a \mid A \rightarrow B \\ M, N &::= x \mid MN \mid \lambda x : A. M \\ \Gamma &::= \cdot \mid \Gamma, x : A \\ \mathcal{J} &::= \Gamma \vdash M : A \end{aligned}$$

λP :

$$\begin{aligned} s &::= * \mid \square \\ M, N, A, B &::= x \mid MN \mid \lambda x : A. M \mid \overline{\Pi x : A. B} \mid s \\ \Gamma &::= \cdot \mid \Gamma, x : A \\ \mathcal{J} &::= \Gamma \vdash M : A \end{aligned}$$

$$a : *, b : *, f : a \rightarrow b, x : a \vdash fx : b$$

λP

STT versus λP syntax

STT:

$$\begin{aligned} A, B &::= a \mid A \rightarrow B \\ M, N &::= x \mid MN \mid \lambda x : A. M \\ \Gamma &::= \cdot \mid \Gamma, x : A \\ \mathcal{J} &::= \Gamma \vdash M : A \end{aligned}$$

λP :

$$\begin{aligned} s &::= * \mid \square \\ M, N, A, B &::= x \mid MN \mid \lambda x : A. M \mid \overbrace{\Pi x : A. B}^{A \rightarrow B} \mid s \\ \Gamma &::= \cdot \mid \Gamma, x : A \\ \mathcal{J} &::= \Gamma \vdash M : A \end{aligned}$$

$$\overbrace{a : *, b : *, f : a \rightarrow b, x : a}^{\Gamma} \vdash \overbrace{fx}^M : \overbrace{b}^A$$

STT versus λP

STT

3 rules

$$x \mid MN \mid \lambda x : A. M$$

λP

7 rules

$$x \mid MN \mid \lambda x : A. M \mid \Pi x : A. M \mid * \mid \square$$

STT versus λP

STT

3 rules

$$x \mid MN \mid \lambda x : A. M$$

λP

7 rules

$$x \mid MN \mid \lambda x : A. M \mid \Pi x : A. M \mid * \mid \square$$

box does not have a type
→ no rule for box

STT versus λP

STT

3 rules

$$x \mid MN \mid \lambda x : A. M$$

λP

7 rules

$$x \mid MN \mid \lambda x : A. M \mid \Pi x : A. M \mid *$$

box does not have a type
→ no rule for box

variables have two rules

STT versus λP

STT

3 rules

$$x \mid MN \mid \lambda x : A. M$$

λP

7 rules

$$x \mid MN \mid \lambda x : A. M \mid \Pi x : A. M \mid *$$

box does not have a type
→ no rule for box

variables have two rules

+ conversion rule

typing rules

λP

$\overline{\vdash * : \square}$

typing rules

λP

$\overline{\vdash * : \square}$

axiom rule

start rule

λP

$\overline{\vdash * : \square}$

axiom rule

start rule

$$\frac{\Gamma \vdash A : * \quad \Gamma, x : A \vdash B : s}{\Gamma \vdash \Pi x : A. B : s}$$

product rule

λP

STT

$$\overline{\vdash * : \square}$$

axiom rule
start rule

$$\frac{\Gamma \vdash A : * \quad \Gamma, x : A \vdash B : s}{\Gamma \vdash \Pi x : A. B : s}$$

product rule

$$\frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x : A. M : A \rightarrow B}$$

abstraction rule

typing rules

λP

STT

$$\overline{\vdash * : \square}$$

axiom rule
start rule

$$\frac{\Gamma \vdash A : * \quad \Gamma, x : A \vdash B : s}{\Gamma \vdash \Pi x : A. B : s}$$

product rule

$$\frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x : A. M : \Pi x : A. B}$$

abstraction rule

$$\frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x : A. M : A \rightarrow B}$$

typing rules

λP

STT

$$\overline{\vdash * : \square}$$

axiom rule

start rule

$$\frac{\Gamma \vdash A : * \quad \Gamma, x : A \vdash B : s}{\Gamma \vdash \Pi x : A. B : s}$$

product rule

$$\frac{\Gamma, x : A \vdash M : B \quad \Gamma \vdash \Pi x : A. B : s}{\Gamma \vdash \lambda x : A. M : \Pi x : A. B}$$

abstraction rule

$$\frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x : A. M : A \rightarrow B}$$

typing rules (continued)

λP

STT

$$\frac{\Gamma \vdash F : A \rightarrow B \quad \Gamma \vdash M : A}{\Gamma \vdash FM : B}$$

application rule

typing rules (continued)

λP

$$\frac{\Gamma \vdash F : \Pi x : A. B \quad \Gamma \vdash M : A}{\Gamma \vdash FM : B}$$

application rule

STT

$$\frac{\Gamma \vdash F : A \rightarrow B \quad \Gamma \vdash M : A}{\Gamma \vdash FM : B}$$

typing rules (continued)

λP

$$\frac{\Gamma \vdash F : \prod x : A. B \quad \Gamma \vdash M : A}{\Gamma \vdash FM : B[x := M]}$$

application rule

STT

$$\frac{\Gamma \vdash F : A \rightarrow B \quad \Gamma \vdash M : A}{\Gamma \vdash FM : B}$$

truesvec : $\prod n : \text{nat}. \text{vec } n$

truesvec (S (S O)) : $\text{vec } (\text{S } (\text{S } \text{O}))$

typing rules (continued)

λP

STT

variable rule

$$\frac{}{\Gamma \vdash x : A} (x : A) \in \Gamma$$

typing rules (continued)

λP

(context Γ is a list)

correct: $a : *, x : a \vdash x : a$

incorrect: $x : a, a : * \vdash x : a$

STT

(context Γ is a set)

$$\frac{}{\Gamma \vdash x : A} (x : A) \in \Gamma$$

variable rule

typing rules (continued)

λP

(context Γ is a list)

correct: $a : *, x : a \vdash x : a$

incorrect: $x : a, a : * \vdash x : a$

$$\frac{\Gamma \vdash A : s}{\Gamma, x : A \vdash x : A}$$

variable rule

$$\frac{\Gamma \vdash M : A \quad \Gamma \vdash B : s}{\Gamma, y : B \vdash M : A}$$

weakening rule

STT

(context Γ is a set)

$$\frac{}{\Gamma \vdash x : A} (x : A) \in \Gamma$$

the final typing rule

$\text{vecappend} : \prod n_1 : \text{nat}. \prod n_2 : \text{nat}.$

$\text{vec } n_1 \rightarrow \text{vec } n_2 \rightarrow \text{vec } (\text{add } n_1 \ n_2)$

$\text{vecappend } 3 \ 4 \ v_1 \ v_2 : \text{vec } (\text{add } 3 \ 4)$

$\text{vecappend } 3 \ 4 \ v_1 \ v_2 : \text{vec } 7$

the final typing rule

$\text{vecappend} : \Pi n_1 : \text{nat}. \Pi n_2 : \text{nat}.$

$\text{vec } n_1 \rightarrow \text{vec } n_2 \rightarrow \text{vec } (\text{add } n_1 \ n_2)$

$\text{vecappend } 3 \ 4 \ v_1 \ v_2 : \text{vec } (\text{add } 3 \ 4)$

$\text{vecappend } 3 \ 4 \ v_1 \ v_2 : \text{vec } 7$

λP

$$\frac{\Gamma \vdash M : A \quad \Gamma \vdash A' : s}{\Gamma \vdash M : A'} A =_{\beta} A'$$

conversion rule

A and A' are convertible

$=_{\beta}$ is defined on preterms

$M, F, A, B ::= x \mid FM \mid \lambda x : A. M \mid \Pi x : A. B \mid s$
 $s ::= * \mid \square$

$$\frac{}{\vdash * : \square} \quad \frac{\Gamma \vdash A : s}{\Gamma, x : A \vdash x : A} \quad \frac{\Gamma \vdash M : A \quad \Gamma \vdash B : s}{\Gamma, y : B \vdash M : A}$$

$$\frac{\Gamma, x : A \vdash M : B \quad \Gamma \vdash \Pi x : A. B : s}{\Gamma \vdash \lambda x : A. M : \Pi x : A. B} \quad \frac{\Gamma \vdash A : * \quad \Gamma, x : A \vdash B : s}{\Gamma \vdash \Pi x : A. B : s}$$

$$\frac{\Gamma \vdash F : \Pi x : A. B \quad \Gamma \vdash M : A}{\Gamma \vdash FM : B[x := M]} \quad \frac{\Gamma \vdash M : A \quad \Gamma \vdash A' : s}{\Gamma \vdash M : A'} A =_{\beta} A'$$

$$\frac{}{\Gamma \vdash s_1 : s_2} (s_1, s_2) \in \mathcal{A}$$

$$\frac{\Gamma \vdash A : s_1 \quad \Gamma, x : A \vdash B : s_2}{\Gamma \vdash \Pi x : A. B : s_3} (s_1, s_2, s_3) \in \mathcal{R}$$

pure type systems

$$\frac{}{\Gamma \vdash s_1 : s_2} (s_1, s_2) \in \mathcal{A}$$

$$\frac{\Gamma \vdash A : s_1 \quad \Gamma, x : A \vdash B : s_2}{\Gamma \vdash \Pi x : A. B : s_3} (s_1, s_2, s_3) \in \mathcal{R}$$

\mathcal{S}
sorts

\mathcal{A}
axioms

\mathcal{R}
rules

$\lambda \rightarrow$

$\{*, \square\}$

$\{(*, \square)\}$

$\{(*, *, *)\}$

λP

$\{*, \square\}$

$\{(*, \square)\}$

$\{(*, *, *), (*, \square, \square)\}$

pure type systems

$$\frac{}{\Gamma \vdash s_1 : s_2} (s_1, s_2) \in \mathcal{A}$$

$$\frac{\Gamma \vdash A : s_1 \quad \Gamma, x : A \vdash B : s_2}{\Gamma \vdash \Pi x : A. B : s_3} (s_1, s_2, s_3) \in \mathcal{R}$$

\mathcal{S}
sorts

\mathcal{A}
axioms

\mathcal{R}
rules

$\lambda*$	$\{*\}$	$\{(*, *)\}$	$\{(*, *, *)\}$
$\lambda \rightarrow$	$\{*, \square\}$	$\{(*, \square)\}$	$\{(*, *, *)\}$
λP	$\{*, \square\}$	$\{(*, \square)\}$	$\{(*, *, *), (*, \square, \square)\}$

example PTS type derivation

$$a : *, x : a \vdash (\lambda y : a. y)x : a$$

example PTS type derivation

$a : *, x : a \vdash (\lambda y : a. y)x : a$

$$\frac{}{\vdash * : \square} \quad \frac{\Gamma \vdash A : s}{\Gamma, x : A \vdash x : A} \quad \frac{\Gamma \vdash M : A \quad \Gamma \vdash B : s}{\Gamma, y : B \vdash M : A}$$

$$\frac{\Gamma, x : A \vdash M : B \quad \Gamma \vdash \Pi x : A. B : s}{\Gamma \vdash \lambda x : A. M : \Pi x : A. B} \quad \frac{\Gamma \vdash F : \Pi x : A. B \quad \Gamma \vdash M : A}{\Gamma \vdash FM : B[x := M]}$$

example PTS type derivation

$$\frac{a : *, x : a \vdash (\lambda y : a. y) : a \rightarrow a \quad a : *, x : a \vdash x : a}{a : *, x : a \vdash (\lambda y : a. y)x : a}$$

$$\frac{}{\vdash * : \square} \quad \frac{\Gamma \vdash A : s}{\Gamma, x : A \vdash x : A} \quad \frac{\Gamma \vdash M : A \quad \Gamma \vdash B : s}{\Gamma, y : B \vdash M : A}$$

$$\frac{\Gamma, x : A \vdash M : B \quad \Gamma \vdash \Pi x : A. B : s}{\Gamma \vdash \lambda x : A. M : \Pi x : A. B} \quad \frac{\Gamma \vdash F : \Pi x : A. B \quad \Gamma \vdash M : A}{\Gamma \vdash FM : B[x := M]}$$

example PTS type derivation

$$\frac{a : *, x : a, y : a \vdash y : a \quad a : *, x : a \vdash a \rightarrow a : *}{\frac{a : *, x : a \vdash (\lambda y : a. y) : a \rightarrow a \quad a : *, x : a \vdash x : a}{a : *, x : a \vdash (\lambda y : a. y)x : a}}$$

$$\frac{}{\vdash * : \square} \quad \frac{\Gamma \vdash A : s}{\Gamma, x : A \vdash x : A} \quad \frac{\Gamma \vdash M : A \quad \Gamma \vdash B : s}{\Gamma, y : B \vdash M : A}$$

$$\frac{\Gamma, x : A \vdash M : B \quad \Gamma \vdash \Pi x : A. B : s}{\Gamma \vdash \lambda x : A. M : \Pi x : A. B} \quad \frac{\Gamma \vdash F : \Pi x : A. B \quad \Gamma \vdash M : A}{\Gamma \vdash FM : B[x := M]}$$

example PTS type derivation

$$\frac{
 \frac{
 \frac{
 a : *, x : a \vdash a : *
 }{
 a : *, x : a, y : a \vdash y : a
 }
 \quad
 a : *, x : a \vdash a \rightarrow a : *
 }{
 a : *, x : a \vdash (\lambda y : a. y) : a \rightarrow a
 }
 \quad
 a : *, x : a \vdash x : a
 }{
 a : *, x : a \vdash (\lambda y : a. y)x : a
 }$$

$$\frac{}{\vdash * : \square} \quad
 \frac{\Gamma \vdash A : s}{\Gamma, x : A \vdash x : A} \quad
 \frac{\Gamma \vdash M : A \quad \Gamma \vdash B : s}{\Gamma, y : B \vdash M : A}$$

$$\frac{\Gamma, x : A \vdash M : B \quad \Gamma \vdash \Pi x : A. B : s}{\Gamma \vdash \lambda x : A. M : \Pi x : A. B} \quad
 \frac{\Gamma \vdash F : \Pi x : A. B \quad \Gamma \vdash M : A}{\Gamma \vdash FM : B[x := M]}$$

