FLOLAC'14

Type theory and logic

Lecture II: dependent type theory

2 July 2014

柯向上

Department of Computer Science
University of Oxford

Hsiang-Shang.Ko@cs.ox.ac.uk

Indexed families of sets (predicates)

Common mathematical statements involve predicates and universal/existential quantification.

For example: "For all natural number $x : \mathbb{N}$, if x is not zero, then there exists $y : \mathbb{N}$ such that x is equal to $1 + y$."

In type theory, a predicate on A has type $A \to \mathcal{U}$ — a *family of sets* indexed by the domain A. For example:

$$\vdash \lambda x.\ \text{"if } x \text{ is zero then } \mathbf{0} \text{ else } \mathbf{1}\text{"} : \mathbb{N} \to \mathcal{U}$$

(Note that the above treatment is in fact unfounded in our current theory. Why? (We will fix it on Thursday.))

Dependent product types (universal quantification)

- Formation:

$$\frac{\Gamma \vdash A : \mathcal{U} \qquad \Gamma \vdash B : A \to \mathcal{U}}{\Gamma \vdash \Pi\, A\, B : \mathcal{U}}\ (\Pi\mathsf{F})$$

- Introduction:

$$\frac{\Gamma, x : A \vdash t : B\, x}{\Gamma \vdash \lambda x.\, t : \Pi\, A\, B}\ (\Pi\mathsf{I})$$

- Elimination:

$$\frac{\Gamma \vdash f : \Pi\, A\, B \qquad \Gamma \vdash a : A}{\Gamma \vdash f\, a : B\, a}\ (\Pi\mathsf{E})$$

Notation. We usually write $\Pi[x : A]\ B\, x$ for $\Pi\ A\ B$, regarding '$\Pi[x : A]$' as a quantifier.

Exercise. Let $\Gamma := A : \mathcal{U}, B : \mathcal{U}, C : A \to B \to \mathcal{U}$. Derive

$$\Gamma \vdash \_ : (\Pi[x : A]\ \Pi[y : B]\ C\, x\, y) \to \Pi[y : B]\ \Pi[x : A]\ C\, x\, y$$

Dependent sum types (existential quantification)

- Formation:

$$\frac{\Gamma \ \vdash \ A : \mathcal{U} \qquad \Gamma \ \vdash \ B : A \to \mathcal{U}}{\Gamma \ \vdash \ \Sigma\ A\ B : \mathcal{U}}\ (\Sigma\mathsf{F})$$

- Introduction:

$$\frac{\Gamma \ \vdash \ a : A \qquad \Gamma \ \vdash \ b : B\ a}{\Gamma \ \vdash \ (a\,,\ b) : \Sigma\ A\ B}\ (\Sigma\mathsf{I})$$

- Elimination:

$$\frac{\Gamma \ \vdash \ p : \Sigma\ A\ B}{\Gamma \ \vdash \ \mathtt{fst}\ p : A}\ (\Sigma\mathsf{EL}) \qquad \frac{\Gamma \ \vdash \ p : \Sigma\ A\ B}{\Gamma \ \vdash \ \mathtt{snd}\ p : B\,(\mathtt{fst}\ p)}\ (\Sigma\mathsf{ER})$$

Notation. We usually write $\Sigma\,[x : A]\ B\,x$ for $\Sigma\ A\ B$.

Exercise. Let $\Gamma := A : \mathcal{U}\,,\ B : \mathcal{U}\,,\ C : A \to B \to \mathcal{U}$. Derive the *axiom of choice*:

$$\Gamma \ \vdash \ \_ : (\Pi\,[x : A]\ \Sigma\,[y : B]\ C\,x\,y) \to \Sigma\,[f : A \to B]\ \Pi\,[x : A]\ C\,x\,(f\,x)$$

Computation

Let $\Gamma := A : \mathcal{U}\,,\ B : A \to \mathcal{U}\,,\ C : A \to \mathcal{U}$. Try to derive

$$\Gamma \ \vdash \ \_ : (\Pi[\,p : \Sigma\ A\ B]\ \ C\,(\texttt{fst}\ p)) \to (\Pi[\,x : A]\ \ B\,x \to\ C\,x)$$

... and you should notice some problems.

So far we have been concentrating on the *statics* of type theory; here we need to formally invoke the *dynamics* of the theory.

Equality judgements and computation rules

We introduce a new kind of judgements stating that two terms should be regarded as the same during typechecking:

$$\Gamma \ \vdash \ t = u \ \in \ A$$

for which we also have a well-formedness requirement that A and everything appearing on the right of the colons in Γ are judged to be sets, and t and u are judged to be elements of A.

For each set, (when applicable) we specify additional *computational rules* stating that eliminating an introductory term yields a component of the latter. For example, for product types we have two computation rules:

$$\frac{\Gamma \ \vdash \ a : A \qquad \Gamma \ \vdash \ b : B}{\Gamma \ \vdash \ \texttt{fst}\,(a\,,\, b) = a \ \in \ A}\,(\times\mathrm{CL}) \qquad \frac{\Gamma \ \vdash \ a : A \qquad \Gamma \ \vdash \ b : B}{\Gamma \ \vdash \ \texttt{snd}\,(a\,,\, b) = b \ \in \ B}\,(\times\mathrm{CR})$$

More computation rules

$$\frac{\Gamma, x : A \ \vdash \ t : B \qquad \Gamma \ \vdash \ a : A}{\Gamma \ \vdash \ (\lambda x.\ t)\ a = t[a/x] \ \in A \to B}\ (\to\mathrm{C})$$

$$\frac{\Gamma \ \vdash \ a : A \qquad \Gamma \ \vdash \ f : A \to C \qquad \Gamma \ \vdash \ g : B \to C}{\Gamma \ \vdash \ \texttt{case}\ (\texttt{left}\ a)\ f\, g = f\, a \ \in \ C}\ (+\mathrm{CL})$$

$$\frac{\Gamma \ \vdash \ b : B \qquad \Gamma \ \vdash \ f : A \to C \qquad \Gamma \ \vdash \ g : B \to C}{\Gamma \ \vdash \ \texttt{case}\ (\texttt{right}\ b)\ f\, g = g\, b \ \in \ C}\ (+\mathrm{CR})$$

More computation rules

$$\frac{\Gamma, x : A \vdash t : B\,x \qquad \Gamma \vdash a : A}{\Gamma \vdash (\lambda x.\, t)\, a = t[a/x] \in B\, a}\ (\Pi\mathrm{C})$$

$$\frac{\Gamma \vdash a : A \qquad \Gamma \vdash b : B\, a}{\Gamma \vdash \mathtt{fst}\,(a, b) = a \in A}\ (\Sigma\mathrm{CL})$$

$$\frac{\Gamma \vdash a : A \qquad \Gamma \vdash b : B\, a}{\Gamma \vdash \mathtt{snd}\,(a, b) = b \in B\, a}\ (\Sigma\mathrm{CR})$$

Equivalence rules

$$\frac{\Gamma \;\vdash\; t : A}{\Gamma \;\vdash\; t = t \;\in A} \text{ (refl)}$$

$$\frac{\Gamma \;\vdash\; t = u \;\in A}{\Gamma \;\vdash\; u = t \;\in A} \text{ (sym)}$$

$$\frac{\Gamma \;\vdash\; t = u \;\in A \qquad \Gamma \;\vdash\; u = v \;\in A}{\Gamma \;\vdash\; t = v \;\in A} \text{ (trans)}$$

Congruence rules

We need a congruence rule for each constant we introduce:

$$\frac{\Gamma \;\vdash\; a = a' \;\in\; A \qquad \Gamma \;\vdash\; b = b' \;\in\; B}{\Gamma \;\vdash\; (a\,,\, b) = (a'\,,\, b') \;\in\; A \times B}$$

$$\frac{\Gamma \;\vdash\; p = p' \;\in\; A \times B}{\Gamma \;\vdash\; \mathtt{fst}\; p = \mathtt{fst}\; p' \;\in\; A} \qquad \frac{\Gamma \;\vdash\; p = p' \;\in\; A \times B}{\Gamma \;\vdash\; \mathtt{snd}\; p = \mathtt{snd}\; p' \;\in\; B}$$

$$\frac{\Gamma\,,\, x : A \;\vdash\; t = t' \;\in\; B}{\Gamma \;\vdash \lambda x.\, t = \lambda x.\, t' \;\in\; A \to B}$$

$$\frac{\Gamma \;\vdash\; f = f' \;\in\; A \qquad \Gamma \;\vdash\; a = a' \;\in\; A}{\Gamma \;\vdash\; f\, a = f'\, a' \;\in\; B}$$

… and similar rules for `left`, `right`, `case`, and `absurd`.

Conversion rule

$$\frac{\Gamma \;\vdash\; t : A \qquad \Gamma \;\vdash\; A = B \;\in \mathcal{U}}{\Gamma \;\vdash\; t : B}\ \text{(conv)}$$

More congruence rules

$$\frac{\Gamma \ \vdash \ a = a' \ \in \ A \qquad \Gamma \ \vdash \ b = b' \ \in \ B\,a}{\Gamma \ \vdash \ (a\,,\,b) = (a'\,,\,b') \ \in \ \Sigma\, A\, B}$$

$$\frac{\Gamma \ \vdash \ p = p' \ \in \ \Sigma\, A\, B}{\Gamma \ \vdash \ \mathtt{fst}\ p = \mathtt{fst}\ p' \ \in \ A} \qquad \frac{\Gamma \ \vdash \ p = p' \ \in \ \Sigma\, A\, B}{\Gamma \ \vdash \ \mathtt{snd}\ p = \mathtt{snd}\ p' \ \in \ B\,(\mathtt{fst}\ p)}$$

$$\frac{\Gamma\,,\, x : A \ \vdash \ t = t' \ \in \ B\,x}{\Gamma \ \vdash \lambda x.\, t = \lambda x.\, t' \ \in \ \Pi\, A\, B}$$

$$\frac{\Gamma \ \vdash \ f = f' \ \in \ A \qquad \Gamma \ \vdash \ a = a' \ \in \ A}{\Gamma \ \vdash \ f\,a = f'\,a' \ \in \ B\,a}$$

Predicates respect computation.

Natural numbers

- Formation:

$$\frac{}{\Gamma \vdash \mathbb{N} : \mathcal{U}}\ (\mathbb{N}\mathsf{F})$$

- Introduction:

$$\frac{}{\Gamma \vdash \mathtt{zero} : \mathbb{N}}\ (\mathbb{N}\mathsf{IZ}) \qquad \frac{\Gamma \vdash n : \mathbb{N}}{\Gamma \vdash \mathtt{suc}\ n : \mathbb{N}}\ (\mathbb{N}\mathsf{IS})$$

Natural numbers — elimination rule

- Elimination:

$$\frac{\begin{array}{l}\Gamma \;\vdash\; P : \mathbb{N} \to \mathcal{U} \\ \Gamma \;\vdash\; z : P\ \texttt{zero} \\ \Gamma \;\vdash\; s : \Pi[x:\mathbb{N}]\ P\,x \to P\,(\texttt{suc}\ x) \\ \Gamma \;\vdash\; n : \mathbb{N}\end{array}}{\Gamma \;\vdash\; \texttt{ind}\ P\,z\,s\,n : P\,n}\ (\mathbb{N}\mathrm{E})$$

Natural numbers — computation rule

- Computation:

$$\frac{\begin{array}{l}\Gamma \ \vdash \ P : \mathbb{N} \to \mathcal{U} \\ \Gamma \ \vdash \ z : P \ \texttt{zero} \\ \Gamma \ \vdash \ s : \Pi[x : \mathbb{N}] \ P\,x \to P\,(\texttt{suc}\ x)\end{array}}{\Gamma \ \vdash \ \texttt{ind}\ P\,z\,s\ \texttt{zero} = z \ \in \ P\ \texttt{zero}}\ (\mathbb{N}\mathrm{CZ})$$

$$\frac{\begin{array}{l}\Gamma \ \vdash \ P : \mathbb{N} \to \mathcal{U} \\ \Gamma \ \vdash \ z : P \ \texttt{zero} \\ \Gamma \ \vdash \ s : \Pi[x : \mathbb{N}] \ P\,x \to P\,(\texttt{suc}\ x) \\ \Gamma \ \vdash \ n : \mathbb{N}\end{array}}{\Gamma \ \vdash \ \texttt{ind}\ P\,z\,s\,(\texttt{suc}\ n) = s\,n\,(\texttt{ind}\ P\,z\,s\,n) \ \in \ P\,(\texttt{suc}\ n)}\ (\mathbb{N}\mathrm{CS})$$