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 every $x : \mathbb{N}$, 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:

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

Remark. The above treatment is in fact unfounded in our current theory. Why?

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:

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

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

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

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

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.

Computation rules

For each set, (when applicable) we specify additional *computation rules* stating that eliminating an introductory term yields a component of the latter. This is the type-theoretic formulation of *Gentzen's inversion principle*.

For example, for product types we have two computation rules:

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

More computation rules

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

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

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

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

Judgemental equality is an equivalence relation.

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

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

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

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

Once we establish that two sets are judgementally equal, we can transfer terms between the two sets.

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

Exercise. Finish deriving

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

(where $\Gamma := A : \mathcal{U}\,,\ B : A \to \mathcal{U}\,,\ C : A \to \mathcal{U}$).

More congruence rules

We can state congruence rules for dependent products and sums only after we have the conversion rule. Why?

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

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

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

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

Canonical vs non-canonical elements

Decidability of judgemental equality

Exercises

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

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

  and

  $$\Gamma \ \vdash \ \_ : (\Sigma[x:A]\ B\,x + C\,x) \leftrightarrow (\Sigma[x:A]\ B\,x) + (\Sigma[x:A]\ C\,x)$$

  What about

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

  and

  $$\Gamma \ \vdash \ \_ : (\Sigma[x:A]\ B\,x \times C\,x) \leftrightarrow (\Sigma[x:A]\ B\,x) \times (\Sigma[x:A]\ C\,x)$$

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

  $$\begin{aligned}\Gamma \ \vdash \ \_ : (\Pi[x:A]\ \Sigma[y:B]\ R\,x\,y) \to \\ \Sigma[f:A \to B]\ \Pi[x:A]\ R\,x\,(f\,x)\end{aligned}$$

More non-derivable propositions

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

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

but not

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

What about

$$\Gamma \ \vdash \ \_ : (\Pi[x:A]\ B\,x + \neg\,B\,x) \to (\neg\,\Pi[x:A]\ \neg\,B\,x) \to \Sigma[x:A]\ B\,x$$

?

Universes

In our current theory, to form types like $A \to \mathcal{U}$ where $A : \mathcal{U}$, we need to assume $\mathcal{U} : \mathcal{U}$. However, this assumption — called *impredicativity* — was shown to lead to inconsistency by Jean-Yves Girard. (This result is commonly referred to as *Girard's paradox*.)

We thus need to introduce a *predicative* hierarchy of universes $\mathcal{U}_0$, $\mathcal{U}_1$, ..., up to infinity.

- Smaller universes are elements of larger universes.

$$\frac{}{\Gamma \vdash \mathcal{U}_i : \mathcal{U}_{i+1}} \; (\mathcal{U}\mathsf{F})$$

- Elements of smaller universes are also elements of larger universes. This is called *cumulativity*.

$$\frac{\Gamma \vdash A : \mathcal{U}_i}{\Gamma \vdash A : \mathcal{U}_{i+1}} \; (\mathsf{cum})$$

Universe polymorphism

Formation rules and elimination rules have to be revised to work across the universe hierarchy.

- For example, the formation rule for coproducts becomes

$$\frac{\Gamma \vdash A : \mathcal{U}_i \qquad \Gamma \vdash B : \mathcal{U}_j}{\Gamma \vdash A + B : \mathcal{U}_{max\, i\, j}}\ (+\mathsf{F})$$

- The elimination rule for coproducts is

$$\frac{\Gamma \vdash q : A + B \quad \Gamma \vdash f : A \to C \quad \Gamma \vdash g : B \to C}{\Gamma \vdash \mathsf{case}\ q\, f\, g : C}\ (+\mathsf{E})$$

  for which we implicitly assume the following judgements:

$$\Gamma \vdash A : \mathcal{U}_i \qquad \Gamma \vdash B : \mathcal{U}_j \qquad \Gamma \vdash C : \mathcal{U}_k$$

In practice, however, we can drop the indices (as if assuming $\mathcal{U} : \mathcal{U}$) because they can be inferred most of the time.