SUBJECT REDUCTION FOR PURE TYPE SYSTEMS

ZHAOSHEN ZHAI

Throughout, fix a countably infinite set V, whose element we call *variables*. For each of the following type systems, there will be a notion of 'types' and 'terms'. Once they are defined, we can speak of the following:

Definition. A *context* is a finite set $\Gamma \coloneqq \{x_1 : \tau_1, \dots, x_n : \tau_n\}$ of pairs $(x_i : \tau_i)$, where each $x_i \in V$ and each τ_i is a 'type'. If $(x : \tau) \in \Gamma$, we write $\Gamma(x) = \tau$, and we let

$$\operatorname{dom}\Gamma \coloneqq \{x \in V : (x : \tau) \text{ for some } \tau\} \quad \text{and} \quad \operatorname{im}\Gamma \coloneqq \{\tau \text{ type} : (x : \tau) \in \Gamma \text{ for some } x\}.$$

A *judgement* is a triple $\Gamma \vdash M : \tau$ consisting of a context Γ, a term M, and a type τ.

1. The Simply-typed λ-calculus

Definition 1.1. A *simple type* is a propositional formula in the language $\to$.

Definition 1.2. A *λ-term* is a string defined by the grammar $M \coloneqq x \mid M\,M \mid (\lambda x\, M)$. We denote by Λ the set of λ-terms. The set of *free variables* of a λ-term M is defined inductively by

$$FV(x) \coloneqq \{x\}, \quad FV(\lambda x\, M) \coloneqq FV(M) \setminus \{x\}, \quad FV(MN) \coloneqq FV(M) \cup FV(N).$$

Definition 1.3. We say that a judgement $\Gamma \vdash M : \tau$ is *derivable in $\lambda_\to$* if there is a finite tree of judgements rooted at $\Gamma \vdash M : \tau$ whose leaves are instances of Var and such that each parent is obtained from its children using either Abs or App.

$$\frac{}{\Gamma, x : \tau \vdash x : \tau}\ \textsc{Var} \qquad \frac{\Gamma, x : \sigma \vdash M : \tau}{\Gamma \vdash (\lambda x\, M) : \sigma \to \tau}\ \textsc{Abs} \qquad \frac{\Gamma \vdash M : \sigma \to \tau \quad \Gamma \vdash N : \sigma}{\Gamma \vdash (M\,N) : \tau}\ \textsc{App}$$

If $\Gamma \vdash M : \tau$ is derivable, we say that M *has type τ in Γ*.

Lemma 1.4 (Generation Lemma for $\lambda_\to$)**.** *Suppose that $\Gamma \vdash M : \tau$.4*

(1) If $M = x$, then $\Gamma(x) = \tau$.
(2) If $M = PQ$, then $\Gamma \vdash P : \sigma \to \tau$ and $\Gamma \vdash Q : \sigma$ for some type σ.
(3) If $M = \lambda x\, N$ and $x \notin \operatorname{dom}\Gamma$, then $\tau = \tau_1 \to \tau_2$ and $\Gamma, x : \tau_1 \vdash N : \tau_2$.

Proof. ∎

Lemma 1.5 (Change of Context)**.** *If $\Gamma \vdash M : \sigma$ and $\Gamma(x) = \Gamma'(x)$ for all $x \in FV(M)$, then $\Gamma' \vdash M : \sigma$.*

Proof. ∎

Lemma 1.6 (Substitution Lemma for $\lambda_\to$)**.** *If $\Gamma, x : \tau \vdash M : \sigma$ and $\Gamma \vdash N : \tau$, then $\Gamma \vdash M[N/x] : \sigma$.*

Proof. ∎

Definition 1.7. A relation $\succ$ on Λ is *compatible* if for any $M, N \in \Lambda$ with $M \succ N$, we have $MP \succ NP$ and $PM \succ PN$ for each $P \in \Lambda$, and $\lambda x\, M \succ \lambda x\, N$ for each $x \in V$.

The least compatible relation $\to_\beta$ on Λ such that $(\lambda x\, M)N \to_\beta M[N/x]$ is called *β-reduction*.

Notation 1.8.

Theorem 1.9 (Subject Reduction for $\lambda_\to$)**.** *If $\Gamma \vdash M : \sigma$ and $M \twoheadrightarrow_\beta N$, then $\Gamma \vdash N : \sigma$.*

Proof. ∎

Date: April 2, 2025.

2. The polymorphic λ-calculus: System $\mathbf{F}$

Definition 2.1.

Lemma 2.2.

Theorem 2.3 (Subject Reduction for $\mathbf{F}$)**.**

3. Dependent Types: $\lambda\mathbf{P}$

Definition 3.1.

Lemma 3.2.

Theorem 3.3 (Subject Reduction for $\lambda\mathbf{P}$)**.**

4. The λ-cube and beyond: Pure Type Systems

Definition 4.1.

Lemma 4.2.

Theorem 4.3 (Subject Reduction for Pure Type Systems)**.**