Normalization for Typed Lambda Calculi
Lecture 2: Polymorphic Type Theory and Strong Normalization

H. Geuvers

Radboud University
Nijmegen, NL

ISR 2026
July 2026

Outline

Full Polymorphism

Strong Normalization for $\lambda\!\to$

Strong Normalization for $\lambda 2$

Why Polymorphic λ-calculus?

- Simple type theory $\lambda\!\!\rightarrow$ is not very expressive
- In simple type theory, we can not 'reuse' a function.
  E.g. $\lambda x : \alpha . x : \alpha \to \alpha$ and $\lambda x : \beta . x : \beta \to \beta$.

We want to define functions that can treat types parametrically: add types $\forall \alpha . A$:
Examples

- $\forall \alpha . \alpha \to \alpha$
  If $M : \forall \alpha . \alpha \to \alpha$, then M can map any type to itself.
- $\forall \alpha . \forall \beta . \alpha \to \beta \to \alpha$
  If $M : \forall \alpha . \forall \beta . \alpha \to \beta \to \alpha$, then M can take two inputs (of arbitrary types) and return a value of the first input type.
- $M : \forall \alpha . \alpha \to \alpha$, with $M\,\alpha = \mathrm{S}$ for $\alpha = \mathsf{Nat}$ and $M\,\alpha = \mathrm{Id}$ for $\alpha \neq \mathsf{Nat}$ ...?? No: this M is an overloaded polymorphic function, not a parametric polymorphic function.

Derivation rules of $\lambda 2$ with full (system F-style) polymorphism

$$\mathsf{Typ} := \mathsf{TVar} \mid (\mathsf{Typ} \to \mathsf{Typ}) \mid \forall \alpha . \mathsf{Typ}$$

1. Church style:

$$\frac{\Gamma \vdash M : A}{\Gamma \vdash \lambda \alpha . M : \forall \alpha . A} \, \alpha \notin \mathsf{FV}(\Gamma) \qquad \frac{\Gamma \vdash M : \forall \alpha . A}{\Gamma \vdash M\, B : A[\alpha := B]}$$

2. Curry style:

$$\frac{\Gamma \vdash M : A}{\Gamma \vdash M : \forall \alpha . A} \, \alpha \notin \mathsf{FV}(\Gamma) \qquad \frac{\Gamma \vdash M : \forall \alpha . A}{\Gamma \vdash M : A[\alpha := B]}$$

Examples

- $\lambda 2$ à la Church:
  $\lambda\alpha.\,\lambda\beta.\,\lambda x : \alpha.\,\lambda y : \beta.\,x : \forall\alpha.\,\forall\beta.\,\alpha \to \beta \to \alpha.$
- $\lambda 2$ à la Curry: $\lambda x.\,\lambda y.\,x : \forall\alpha.\,\forall\beta.\,\alpha \to \beta \to \alpha.$
- $\lambda 2$ à la Church:
  $z : \forall\alpha.\,\alpha \to \alpha \vdash \lambda\alpha.\,z\,(\alpha \to \alpha)\,(z\,\alpha) : \forall\alpha.\,\alpha \to \alpha.$
- $\lambda 2$ à la Curry: $z : \forall\alpha.\,\alpha \to \alpha \vdash z\,z : \forall\alpha.\,\alpha \to \alpha.$
- $\lambda 2$ à la Church: $\lambda x : (\forall\alpha.\,\alpha).\,\lambda y : A.\,x\,B : (\forall\alpha.\,\alpha) \to A \to B.$
- $\lambda 2$ à la Curry: $\lambda x.\,\lambda y.\,x : (\forall\alpha.\,\alpha) \to A \to B.$

More examples of typing in $\lambda 2$

Abbreviate $\bot := \forall\alpha.\,\alpha$, $\top := \forall\alpha.\,\alpha \to \alpha$.

- Curry $\lambda 2$: $\lambda x.\, x\, x : \bot \to \bot$
- Church $\lambda 2$: $\lambda x : \bot.\, x\,(\bot \to \bot)\, x : \bot \to \bot$.
- Church $\lambda 2$: $\lambda x : \bot.\, \lambda\alpha.\, x\,(\alpha \to \alpha)\,(x\,\alpha) : \bot \to \bot$.

LEMMA. In $\lambda 2$ à la Curry, every λ-term in normal form is typable by giving all variables type $\bot$.

Exercises:

- Verify that in Church $\lambda 2$: $\lambda x : \top.\, x\, \top\, x : \top \to \top$.
- Verify that in Curry $\lambda 2$: $\lambda x.\, x\, x : \top \to \top$
- Find a type in Curry $\lambda 2$ for $\lambda x.\, x\, x\, x$
- Find a type in Curry $\lambda 2$ for $\lambda x.\, x\, x\,(\lambda y.\, y\, y)\, x\, x$

Erasure from $\lambda 2$ à la Church to $\lambda 2$ à la Curry

$$
\begin{array}{lcllcl}
|x| & := & x & & & \\
|\lambda x{:}\, A.\, M| & := & |\lambda x.\, M| & |\lambda \alpha.\, M| & := & |M| \\
|M\, N| & := & |M|\,|N| & |M\, A| & := & |M|
\end{array}
$$

THEOREM If $\Gamma \vdash M : A$ in $\lambda 2$ à la Church, then $\Gamma \vdash |M| : A$ in $\lambda 2$ à la Curry.

THEOREM If $\Gamma \vdash P : A$ in $\lambda 2$ à la Curry, then there is an M such that $|M| \equiv P$ and $\Gamma \vdash M : A$ in $\lambda 2$ à la Church.

PROOF. By induction on the derivation of $\Gamma \vdash M : A$.

THEOREM[Wells 1993] In $\lambda 2$ à la Curry, type checking is undecidable

Recall: Important Properties

$\Gamma \vdash M : A?$	TCP
$\Gamma \vdash M : ?$	TSP
$\vdash ? : A$	TIP

Properties of polymorphic λ-calculus

- TIP is undecidable, TCP and TSP are equivalent.
TCP	à la Church	à la Curry
ML-style	decidable	decidable
System F-style	decidable	undecidable

  With full polymorphism (system F), untyped terms contain too little information to compute the type.

Meta properties of $\lambda 2$

- For $\lambda 2$ à la Church: Uniqueness of types
  If $\Gamma \vdash M : A$ and $\Gamma \vdash M : B$, then $A = B$.
- Subject Reduction
  If $\Gamma \vdash M : A$ and $M \to_{\beta\eta} N$, then $\Gamma \vdash N : A$.
- Strong Normalization
  If $\Gamma \vdash M : A$, then all $\beta\eta$-reductions from M terminate.

Strong Normalization for $\lambda\!\to$ à la Curry

This is proved by constructing a model of $\lambda\!\to$.
Method originally due to Tait (1967)[1]

Definition

- $[\![\alpha]\!] :=$ SN (the set of strongly normalizing λ-terms).
- $[\![A \to B]\!] := \{M \mid \forall N \in [\![A]\!](M\,N \in [\![B]\!])\}$.

Lemma

1. $x\,N_1 \ldots N_k \in [\![A]\!]$ for all x, A and $N_1, \ldots, N_k \in$ SN.
2. $[\![A]\!] \subseteq$ SN
3. If $M[x := N]\vec{P} \in [\![A]\!]$, $N \in$ SN, then $(\lambda x.\, M)\,N\,\vec{P} \in [\![A]\!]$.

Proof. By induction on A, (1) and (2) simultaneously.

[1] Also direct "arithmetical" methods exist, that use a decreasing ordering (David 2001, David & Nour)

Fundamental Property (soundness)

PROPOSITION.

$$\text{If } \Gamma \vdash M : A, \text{ then } \Gamma \models M : A,$$

where $\Gamma \models M : A$ denotes

$$\forall \sigma (\sigma \models \Gamma \Rightarrow M\,\sigma \in [\![A]\!])$$

and $\sigma \models \Gamma$ says that σ is a substitution such that for all $x : B \in \Gamma$, we have $\sigma(x) \in [\![B]\!]$.

Alternatively, we can write the statement of this Proposition as

$$\left.\begin{array}{c} x_1 : B_1, \ldots, x_n : B_n \vdash M : A \\ N_1 \in [\![B_1]\!], \ldots, N_n \in [\![B_n]\!] \end{array}\right\} \Rightarrow M[x_1 := N_1, \ldots x_n := N_n] \in [\![A]\!]$$

PROOF. By induction on the derivation of $\Gamma \vdash M : A$.
(Using (3) of the previous Lemma:
If $M[x := N]\vec{P} \in [\![A]\!]$, $N \in \mathsf{SN}$, then $(\lambda x.\, M)\, N\, \vec{P} \in [\![A]\!]$.) $\square$

Strong Normalization

COROLLARY. $\lambda\!\!\to$ is SN

Remember the Fundamental Property:

$$\left.\begin{array}{c} x_1 : B_1, \ldots, x_n : B_n \vdash M : A \\ N_1 \in [\![B_1]\!], \ldots, N_n \in [\![B_n]\!] \end{array}\right\} \Rightarrow M[x_1 := N_1, \ldots x_n := N_n] \in [\![A]\!]$$

We take $N_i := x_i$. (That can be done, because $x_i \in [\![B_i]\!]$ by (1) of the Lemma.)
Then $M \in [\![A]\!] \subseteq \mathrm{SN}$, using (2) of the Lemma. $\square$

Exercise Verify the details of the Strong Normalization proof. (That is, prove the missing details of the Lemma and the Proposition.)

Consistency

Normalization (weak or strong) implies logical consistency of the type theory: there is a type A that has no closed inhabitant:

$$\neg\exists M(\vdash M : A)$$

Proof.

A little bit on semantics

$\lambda\!\!\to$ has a simple set-theoretic model. Given sets $[\![\alpha]\!]$ for type variables α, define

$$[\![A \to B]\!] := [\![B]\!]^{[\![A]\!]} \text{ (set theoretic function space } [\![A]\!] \to [\![B]\!])$$

If any of the base sets $[\![\alpha]\!]$ is infinite, then there are higher and higher (uncountable) cardinalities among the $[\![A]\!]$
There are smaller models, e.g.

$$[\![A \to B]\!] := \{f \in [\![A]\!] \to [\![B]\!] | f \text{ is definable}\}$$

where definability means that it can be constructed in some formal system. This restricts the collection to a countable set.
For example

$$[\![A \to B]\!] := \{f \in [\![A]\!] \to [\![B]\!] | f \text{ is } \lambda\text{-definable}\}$$

Recall $\lambda 2$

Church style:

$$\frac{\Gamma \vdash M : A}{\Gamma \vdash \lambda\alpha.\, M : \forall\alpha.\, A}\ \alpha \notin \mathrm{FV}(\Gamma) \qquad \frac{\Gamma \vdash M : \forall\alpha.\, A}{\Gamma \vdash M\, B : A[\alpha := B]}\ \text{for } B \text{ a } \lambda 2\text{-type}$$

Curry style:

$$\frac{\Gamma \vdash M : A}{\Gamma \vdash M : \forall\alpha.\, A}\ \alpha \notin \mathrm{FV}(\Gamma) \qquad \frac{\Gamma \vdash M : \forall\alpha.\, A}{\Gamma \vdash M : A[\alpha := B]}\ \text{for } B \text{ a } \lambda 2\text{-type}$$

Strong Normalization of β for $\lambda 2$

- For $\lambda 2$ a la Church, there are two kinds of β-reductions:
  - $(\lambda x : A.\, M)P \to_\beta M[x := P]$ term reduction
  - $(\lambda \alpha.\, M)B \to_\beta M[\alpha := B]$ type reduction
- The second doesn't do any harm, so we can just look at $\lambda 2$ à la Curry

  More precisely:

  LEMMA
  - type reduction is terminating
  - if there is an infinite combined term reduction / type reduction path in $\lambda 2$ a la Church, then there is an infinite term reduction path in $\lambda 2$ a la Curry.

Strong Normalization of β for $\lambda 2$ a la Curry

Recall the proof for $\lambda{\to}$:

- $[\![\alpha]\!] := \mathsf{SN}$.
- $[\![A \to B]\!] := \{M \mid \forall N \in [\![A]\!] (M\, N \in [\![B]\!])\}$.

Question:
How to define $[\![\forall \alpha.\, A]\!]$??

$$[\![\forall \alpha.\, A]\!] := \Pi_{X \in U} [\![A]\!]_{\alpha := X} ??$$

Interpretation of types

Question: How to define $[\![\forall\alpha.\, A]\!]$??

$$[\![\forall\alpha.\, A]\!] := \Pi_{X\in U}[\![A]\!]_{\alpha:=X} ??$$

- What should U be?
  The collection of "all possible interpretations" of types
- $\Pi_{X\in U}[\![A]\!]_{\alpha:=X}$ gets very big: $\mathrm{card}(\Pi_{X\in U}[\![A]\!]_{\alpha:=X}) > \mathrm{card}(U)$

Girard:

- $[\![\forall\alpha.\, A]\!]$ should be small

$$\bigcap_{X\in U}[\![A]\!]_{\alpha:=X}$$

- Characterization of U.

Saturated sets

U := SAT, the collection of saturated sets of (untyped) λ-terms.

DEFINITION. $X \subseteq \Lambda$ is saturated if

- $x\, P_1 \ldots P_n \in X$ (for all $x \in \mathrm{Var}$, $P_1, \ldots, P_n \in \mathrm{SN}$)
- $X \subseteq \mathrm{SN}$
- If $M[x := N]\vec{P} \in X$ and $N \in \mathrm{SN}$, then $(\lambda x.\, M)\, N\vec{P} \in X$.

DEFINITION. Let $\rho : \mathrm{TVar} \to \mathrm{SAT}$ be a valuation of type variables. Define the interpretation of types $[\![A]\!]_\rho$ as follows.

- $[\![\alpha]\!]_\rho := \rho(\alpha)$
- $[\![A \to B]\!]_\rho := \{M | \forall N \in [\![A]\!]_\rho (M\, N \in [\![B]\!]_\rho)\}$
- $[\![\forall\alpha.\, A]\!]_\rho := \cap_{X \in \mathrm{SAT}} [\![A]\!]_{\rho, \alpha := X}$

Fundamental Property (soundness)

PROPOSITION.

$$\text{If } \Gamma \vdash M : A, \text{ then } \Gamma \models M : A,$$

where $\Gamma \models M : A$ denotes

$$\forall \rho, \sigma (\rho, \sigma \models \Gamma \Rightarrow M\,\sigma \in [\![A]\!]_\rho)$$

and $\rho, \sigma \models \Gamma$ says that σ is a substitution such that for all $x : B \in \Gamma$, we have $\sigma(x) \in [\![B]\!]_\rho$.

Again, this can be read as follows

$$x_1 : B_1, \ldots, x_n : B_n \vdash M : A \Rightarrow M[x_1 := P_1, \ldots, x_n := P_n] \in [\![A]\!]_\rho$$

for all valuations ρ and $P_1 \in [\![B_1]\!]_\rho, \ldots, P_n \in [\![B_n]\!]_\rho$

PROOF. By induction on the derivation of $\Gamma \vdash M : A$. $\square$.

COROLLARY. $\lambda 2$ is SN.
PROOF. Again we take P_1 to be x_1, ..., P_n to be x_n. $\square$

A little bit on semantics

$\lambda 2$ does not have a set-theoretic model! [Reynolds]

THEOREM. If

$$[\![A \to B]\!] := [\![B]\!]^{[\![A]\!]} \text{ (set theoretic function space)}$$

then $[\![A]\!]$ is a singleton set for every A.

So: in a $\lambda 2$-model, $[\![A \to B]\!]$ must be 'small'.

Questions?