Normalization of STLC

Joos Munneke
Robert Kunst

24th of November, 2023

Overview

Grammar

Grammar of STLC

$$\tau := \text{bool} \mid \tau \to \tau$$
$$e := x \mid \text{true} \mid \text{false} \mid \text{if } e \text{ then } e \text{ else } e \mid \lambda x\colon \tau.e \mid e\, e$$
$$v := \text{true} \mid \text{false} \mid \lambda x\colon \tau.e$$
$$E := [\,] \mid \text{if } E \text{ then } e \text{ else } e \mid E\, e \mid v\, E$$

Evaluation rules

$$\frac{}{(\lambda x\colon \tau.e)v \mapsto e[x := v]}$$

$$\frac{e \mapsto e'}{E[e] \mapsto E[e']}$$

$$\frac{}{\text{if true then } e_1 \text{ else } e_2 \mapsto e_1}$$

$$\frac{}{\text{if false then } e_1 \text{ else } e_2 \mapsto e_2}$$

Typing rules 1/2

1. T-True

$$\frac{}{\Gamma \vdash \mathsf{true} : \mathsf{bool}}$$

2. T-False

$$\frac{}{\Gamma \vdash \mathsf{false} : \mathsf{bool}}$$

3. T-Var

$$\frac{\Gamma(x) = \tau}{\Gamma \vdash x : \tau}$$

Typing rules 2/2

4. T-IfThenElse

$$\frac{\Gamma \vdash e : \text{bool} \quad \Gamma \vdash e_1 : \tau \quad \Gamma \vdash e_2 : \tau}{\Gamma \vdash \text{if } e \text{ then } e_1 \text{ else } e_2 : \tau}$$

5. T-Abs

$$\frac{\Gamma, x : \tau_1 \vdash e : \tau_2}{\Gamma \vdash \lambda x : \tau_1 . e : \tau_1 \rightarrow \tau_2}$$

6. T-App

$$\frac{\Gamma \vdash e_1 : \tau_2 \rightarrow \tau_1 \quad \Gamma \vdash e_2 : \tau_2}{\Gamma \vdash e_1\, e_2 : \tau_1}$$

Normalization Theorem

Theorem

For all terms e, if $\cdot \vdash e : \tau$*, then* $e \Downarrow$.

Induction on type derivation

Proof: suppose $\cdot \vdash e : \tau$, we have to show $e \Downarrow$. Attempt: induction on typing rules.

Induction on type derivation

Proof: suppose $\cdot \vdash e : \tau$, we have to show $e \Downarrow$. Attempt: induction on typing rules.

1. T-True

$$\frac{}{\cdot \vdash \text{true} : \text{bool}}$$

Induction on type derivation

Proof: suppose $\cdot \vdash e : \tau$, we have to show $e \Downarrow$. Attempt: induction on typing rules.

1. T-True

$$\frac{}{\cdot \vdash \text{true} : \text{bool}}$$

2. T-False

$$\frac{}{\cdot \vdash \text{false} : \text{bool}}$$

Induction on type derivation

Proof: suppose $\cdot \vdash e : \tau$, we have to show $e \Downarrow$. Attempt: induction on typing rules.

1. T-True

$$\frac{}{\cdot \vdash \text{true} : \text{bool}}$$

2. T-False

$$\frac{}{\cdot \vdash \text{false} : \text{bool}}$$

3. T-Var

$$\frac{\cdot(x) = \tau}{\cdot \vdash x : \tau}$$

Induction on type derivation

Proof: suppose $\cdot \vdash e : \tau$, we have to show $e \Downarrow$. Attempt: induction on typing rules.

4. T-IfThenElse

$$\frac{\cdot \vdash e : \text{bool} \quad \cdot \vdash e_1 : \tau \quad \cdot \vdash e_2 : \tau}{\cdot \vdash \text{if } e \text{ then } e_1 \text{ else } e_2 : \tau}$$

Induction on type derivation

Proof: suppose $\cdot \vdash e : \tau$, we have to show $e \Downarrow$. Attempt: induction on typing rules.

4. T-IfThenElse

$$\frac{\cdot \vdash e : \text{bool} \quad \cdot \vdash e_1 : \tau \quad \cdot \vdash e_2 : \tau}{\cdot \vdash \text{if } e \text{ then } e_1 \text{ else } e_2 : \tau}$$

5. T-Abs

$$\frac{x : \tau_1 \vdash e : \tau_2}{\cdot \vdash \lambda x : \tau_1 . e : \tau_1 \to \tau_2}$$

Induction on type derivation

Proof: suppose $\cdot \vdash e : \tau$, we have to show $e \Downarrow$. Attempt: induction on typing rules.

4. T-IfThenElse

$$\frac{\cdot \vdash e : \text{bool} \quad \cdot \vdash e_1 : \tau \quad \cdot \vdash e_2 : \tau}{\cdot \vdash \text{if } e \text{ then } e_1 \text{ else } e_2 : \tau}$$

5. T-Abs

$$\frac{x : \tau_1 \vdash e : \tau_2}{\cdot \vdash \lambda x : \tau_1 . e : \tau_1 \rightarrow \tau_2}$$

6. T-App

$$\frac{\cdot \vdash e_1 : \tau_2 \rightarrow \tau_1 \quad \cdot \vdash e_2 : \tau_2}{\cdot \vdash e_1\ e_2 : \tau_1}$$

Logical Relations

A logical relation is a predicate $P_\tau(e)$ over an expression, indexed by a type.

1. The logical relation should contain well-typed terms: $\cdot \vdash e : \tau$;
2. P contains the property we want to prove;
3. P should be preserved by the elimination forms for the corresponding type τ.

Logical relation

Define the relation $N_\tau(e)$ for terms e and types τ as follows:

$$N_{\text{bool}}(e) := \cdot \vdash e : \text{bool} \ \wedge \ e \Downarrow$$
$$N_{\tau_1 \to \tau_2}(e) := \cdot \vdash e : \tau_1 \to \tau_2 \ \wedge \ e \Downarrow \ \wedge \forall e'. N_{\tau_1}(e') \implies N_{\tau_2}(e \ e')$$

Proof plan of normalization

For all terms e:

$$\cdot \vdash e : \tau \qquad \Longrightarrow \qquad e \Downarrow$$

Proof plan of normalization

For all terms e:

$$\cdot \vdash e : \tau \qquad \Longrightarrow \qquad e \Downarrow$$

$$\cdot \vdash e : \tau \quad \overset{\text{A}}{\Longrightarrow} \quad N_\tau(e) \quad \overset{\text{B}}{\Longrightarrow} \quad e \Downarrow$$

Step B

For all terms e, we have $N_\tau(e) \implies e \Downarrow$.

Step B

For all terms e, we have $N_\tau(e) \implies e \Downarrow$.

Immediate from definition of $N_\tau(e)$:

$$N_{\text{bool}}(e) := \cdot \vdash e : \text{bool} \;\wedge\; e \Downarrow$$
$$N_{\tau_1 \to \tau_2}(e) := \cdot \vdash e : \tau_1 \to \tau_2 \;\wedge\; e \Downarrow \wedge \forall e'. N_{\tau_1}(e') \implies N_{\tau_2}(e\ e')$$

Step A

For all terms e, we have $\cdot \vdash e : \tau \implies N_\tau(e)$.

Step A

For all terms e, we have $\cdot \vdash e : \tau \implies N_\tau(e)$.

Rule T-App for $e_1\ e_2 : \tau$

$$\frac{\cdot \vdash e_1 : \tau_2 \to \tau \quad \cdot \vdash e_2 : \tau_2}{\cdot \vdash e_1\ e_2 : \tau}$$

Step A

For all terms e, we have $\cdot \vdash e : \tau \implies N_\tau(e)$.

Rule T-App for $e_1\ e_2 : \tau$

$$\frac{\cdot \vdash e_1 : \tau_2 \to \tau \quad \cdot \vdash e_2 : \tau_2}{\cdot \vdash e_1\ e_2 : \tau}$$

Induction hypothesis: $N_{\tau_2 \to \tau}(e_1)$ and $N_{\tau_2}(e_2)$ hold.
To prove: $N_\tau(e_1\ e_2)$.

Step A: T-App

Induction hypothesis: $N_{\tau_2 \to \tau}(e_1)$ and $N_{\tau_2}(e_2)$ hold.
To prove: $N_\tau(e_1\ e_2)$.

Definition of $N_\tau(e)$:

$$N_{\text{bool}}(e) := \cdot \vdash e : \text{bool} \ \wedge \ e \Downarrow$$
$$N_{\tau_1 \to \tau_2}(e) := \cdot \vdash e : \tau_1 \to \tau_2 \ \wedge \ e \Downarrow \ \wedge \forall e'. N_{\tau_1}(e') \implies N_{\tau_2}(e\ e')$$

Step A: T-App

Induction hypothesis: $N_{\tau_2 \to \tau}(e_1)$ and $N_{\tau_2}(e_2)$ hold.
To prove: $N_\tau(e_1\ e_2)$.

Definition of $N_{\tau_2 \to \tau}(e_1)$:

$$N_{\tau_2 \to \tau}(e_1) := \cdot \vdash e_1 : \tau_2 \to \tau \ \wedge \ e_1 \Downarrow \ \wedge \forall e'. N_{\tau_2}(e') \implies N_\tau(e_1\ e')$$

Step A: T-App

Induction hypothesis: $N_{\tau_2 \to \tau}(e_1)$ and $N_{\tau_2}(e_2)$ hold.
To prove: $N_\tau(e_1\ e_2)$.

Definition of $N_{\tau_2 \to \tau}(e_1)$:

$$N_{\tau_2 \to \tau}(e_1) := \cdot \vdash e_1 : \tau_2 \to \tau \ \wedge \ e_1 \Downarrow \ \wedge \forall e'. N_{\tau_2}(e') \implies N_\tau(e_1\ e')$$

Case T-App is fixed.

Step A

For all terms e, we have $\cdot \vdash e : \tau \implies N_\tau(e)$.

Step A

For all terms e, we have $\cdot \vdash e : \tau \implies N_\tau(e)$.

Now look at the case T-Abs:

$$\frac{x : \tau_1 \vdash e : \tau_2}{\cdot \vdash \lambda x : \tau_1 . e : \tau_1 \rightarrow \tau_2}$$

Step A

For all terms e, we have $\cdot \vdash e : \tau \implies N_\tau(e)$.

Now look at the case T-Abs:

$$\frac{x : \tau_1 \vdash e : \tau_2}{\cdot \vdash \lambda x : \tau_1 . e : \tau_1 \to \tau_2}$$

To prove: $N_{\tau_1 \to \tau_2}(\lambda x : \tau_1 . e)$.

Step A: T-Abs

To prove: $N_{\tau_1 \to \tau_2}(\lambda x : \tau_1 . e)$.

Step A: T-Abs

To prove: $N_{\tau_1 \to \tau_2}(\lambda x : \tau_1.e)$.
Definition of $N_{\tau_1 \to \tau_2}(\lambda x : \tau_1.e)$

$$N_{\tau_1 \to \tau_2}(\lambda x : \tau_1.e) := \cdot \vdash \lambda x : \tau_1.e : \tau_1 \to \tau_2 \ \wedge \ \lambda x : \tau_1.e \Downarrow \\ \wedge \ \forall e'.N_{\tau_1}(e') \implies N_{\tau_2}((\lambda x : \tau_1.e)\ e')$$

Step A: T-Abs

To prove: $N_{\tau_1 \to \tau_2}(\lambda x : \tau_1 . e)$.
Definition of $N_{\tau_1 \to \tau_2}(\lambda x : \tau_1 . e)$

$$N_{\tau_1 \to \tau_2}(\lambda x : \tau_1 . e) := \cdot \vdash \lambda x : \tau_1 . e : \tau_1 \to \tau_2 \ \wedge \ \lambda x : \tau_1 . e \Downarrow \\ \wedge \ \forall e' . N_{\tau_1}(e') \implies N_{\tau_2}((\lambda x : \tau_1 . e) \ e')$$

Case T-Abs broke!

Substitutions

We allow for open terms by modifying step A to step A':

A. For all terms e, if $\cdot \vdash e : \tau$, then $N_\tau(e)$.

Substitutions

We allow for open terms by modifying step A to step A':

A. For all terms e, if $\cdot \vdash e : \tau$, then $N_\tau(e)$.

A'. For all contexts Γ, "suitable substitutions" γ and terms e, if $\Gamma \vdash e : \tau$, then $N_\tau(\gamma(e))$.

Substitutions

A substitution $\gamma = \{x_1 \mapsto v_1, \ldots, x_n \mapsto v_n\}$ is suitable for a context Γ if:

Substitutions

A substitution $\gamma = \{x_1 \mapsto v_1, \ldots, x_n \mapsto v_n\}$ is suitable for a context Γ if:

- Both γ and Γ contain the same variables, that is, they have equal domain.

Substitutions

A substitution $\gamma = \{x_1 \mapsto v_1, \ldots, x_n \mapsto v_n\}$ is suitable for a context Γ if:

- Both γ and Γ contain the same variables, that is, they have equal domain.
- The substitutes of γ must be correctly typed according to Γ, and even comply with our normalization predicate N.

Substitutions

A substitution $\gamma = \{x_1 \mapsto v_1, \ldots, x_n \mapsto v_n\}$ is suitable for a context Γ if:

- Both γ and Γ contain the same variables, that is, they have equal domain.
- The substitutes of γ must be correctly typed according to Γ, and even comply with our normalization predicate N.

Definition

A substitution γ is suitable for a context Γ, write $\gamma \vDash \Gamma$, is defined by:

$$\gamma \vDash \Gamma := \text{dom}(\gamma) = \text{dom}(\Gamma) \wedge (\forall x \in \text{dom}(\Gamma).N_{\Gamma(x)}(\gamma(x))).$$

Step A'

A'. For all contexts Γ, "suitable substitutions" γ and terms e, if $\Gamma \vdash e : \tau$, then $N_\tau(\gamma(e))$.

Step A'

A'. For all contexts Γ, substitutions γ and terms e, if $\Gamma \vdash e : \tau$ and $\gamma \vDash \Gamma$, then $N_\tau(\gamma(e))$.

Lemmas

Lemma (Substitution Lemma)

Substitution preserves type:
if $\Gamma \vdash e : \tau$ *and* $\gamma \vDash \Gamma$, *then* $\cdot \vdash \gamma(e) : \tau$.

Lemmas

Lemma (Substitution Lemma)

Substitution preserves type:
if $\Gamma \vdash e : \tau$ *and* $\gamma \vDash \Gamma$, *then* $\cdot \vdash \gamma(e) : \tau$.

Lemma (Reduction Lemma)

$N_\tau(e)$ *respects reduction:*
if $\cdot \vdash e : \tau$ *and* $e \mapsto e'$, *then* $N_\tau(e) \iff N_\tau(e')$.

Step A'

For all contexts Γ, substitutions γ and terms e, if $\Gamma \vdash e : \tau$ and $\gamma \vDash \Gamma$, then $N_\tau(\gamma(e))$.

Step A'

For all contexts Γ, substitutions γ and terms e, if $\Gamma \vdash e : \tau$ and $\gamma \vDash \Gamma$, then $N_\tau(\gamma(e))$.
We prove step A' by induction on the typing derivation.

Step A'

For all contexts Γ, substitutions γ and terms e, if $\Gamma \vdash e : \tau$ and $\gamma \vDash \Gamma$, then $N_\tau(\gamma(e))$.

Step A'

For all contexts Γ, substitutions γ and terms e, if $\Gamma \vdash e : \tau$ and $\gamma \vDash \Gamma$, then $N_\tau(\gamma(e))$.

Look at the case T-Abs:

$$\frac{\Gamma, x : \tau_1 \vdash e : \tau_2}{\Gamma \vdash \lambda x : \tau_1 . e : \tau_1 \to \tau_2}$$

Step A'

For all contexts Γ, substitutions γ and terms e, if $\Gamma \vdash e : \tau$ and $\gamma \vDash \Gamma$, then $N_\tau(\gamma(e))$.

Look at the case T-Abs:

$$\frac{\Gamma, x : \tau_1 \vdash e : \tau_2}{\Gamma \vdash \lambda x : \tau_1 . e : \tau_1 \to \tau_2}$$

To prove: for any substitution $\gamma \vDash \Gamma$, we have $N_{\tau_1 \to \tau_2}(\gamma(\lambda x : \tau_1 . e))$.

Step A'

For all contexts Γ, substitutions γ and terms e, if $\Gamma \vdash e : \tau$ and $\gamma \vDash \Gamma$, then $N_\tau(\gamma(e))$.

Look at the case T-Abs:

$$\frac{\Gamma, x : \tau_1 \vdash e : \tau_2}{\Gamma \vdash \lambda x : \tau_1 . e : \tau_1 \to \tau_2}$$

To prove: for any substitution $\gamma \vDash \Gamma$, we have $N_{\tau_1 \to \tau_2}(\gamma(\lambda x : \tau_1 . e))$.
Induction hypothesis: for any substitution $\gamma' \vDash \Gamma, x : \tau_1$, we have $N_{\tau_2}(\gamma'(e))$.

Step A': case T-Abs

Induction hypothesis: for any substitution $\gamma' \vDash \Gamma, x : \tau_1$, we have $N_{\tau_2}(\gamma'(e))$.
To prove: for any substitution $\gamma \vDash \Gamma$, we have $N_{\tau_1 \to \tau_2}(\gamma(\lambda x : \tau_1 . e))$.

Step A': case T-Abs

Induction hypothesis: for any substitution $\gamma' \vDash \Gamma, x : \tau_1$, we have $N_{\tau_2}(\gamma'(e))$.
To prove: for any substitution $\gamma \vDash \Gamma$, we have $N_{\tau_1 \to \tau_2}(\gamma(\lambda x : \tau_1 . e))$.

Take any substitution $\gamma \vDash \Gamma$. Proving $N_{\tau_1 \to \tau_2}(\gamma(\lambda x : \tau_1 . e))$ means proving three things:

1. $\cdot \vdash \gamma(\lambda x : \tau_1 . e) : \tau_1 \to \tau_2$
2. $\gamma(\lambda x : \tau_1 . e) \Downarrow$
3. $\forall e' . N_{\tau_1}(e') \implies N_{\tau_2}(\gamma(\lambda x : \tau_1 . e)\ e')$

Step A': case T-Abs (1)

Induction hypothesis: for all $\gamma' \vDash \Gamma, x : \tau_1$, we have $N_{\tau_2}(\gamma'(e))$.
Assumptions: $\Gamma \vdash \lambda x : \tau_1.e : \tau_1 \to \tau_2$ and $\gamma \vDash \Gamma$

1. $\cdot \vdash \gamma(\lambda x : \tau_1.e) : \tau_1 \to \tau_2$

Step A': case T-Abs (2)

Induction hypothesis: for all $\gamma' \vDash \Gamma, x : \tau_1$, we have $N_{\tau_2}(\gamma'(e))$.
Assumptions: $\Gamma \vdash \lambda x : \tau_1.e : \tau_1 \to \tau_2$ and $\gamma \vDash \Gamma$

2. $\gamma(\lambda x : \tau_1.e) \Downarrow$

Step A': case T-Abs (3)

Induction hypothesis: for all $\gamma' \vDash \Gamma, x : \tau_1$, we have $N_{\tau_2}(\gamma'(e))$.
Assumptions: $\Gamma \vdash \lambda x : \tau_1.e : \tau_1 \to \tau_2$ and $\gamma \vDash \Gamma$

3. $\forall e'.N_{\tau_1}(e') \implies N_{\tau_2}(\gamma(\lambda x : \tau_1.e)\ e')$

Normalization

Theorem

For all terms e, if $\cdot \vdash e : \tau$, *then* $e \Downarrow$.

A'. For all contexts Γ, substitutions γ and terms e, if $\Gamma \vdash e : \tau$ and $\gamma \vDash \Gamma$, then $N_\tau(\gamma(e))$.

B. For all terms e, we have $N_\tau(e) \implies e \Downarrow$.

Normalization

Theorem

For all terms e, if $\cdot \vdash e : \tau$*, then* $e \Downarrow$.

A'. For all contexts Γ, substitutions γ and terms e, if $\Gamma \vdash e : \tau$ and $\gamma \vDash \Gamma$, then $N_\tau(\gamma(e))$.

B. For all terms e, we have $N_\tau(e) \implies e \Downarrow$.

We have shown A', and we had already shown B, so the normalization theorem follows.