Call-By-Name Normalization for System F

Aaron Stump

November 10, 2014

1 Introduction

This note gives a proof that call-by-name reduction is normalizing for unannotated System F (polymorphic lambda calculus), and considers a few consequences. System F is defined with annotated terms, where λ-bound variables must be declared with their types. So we have $\lambda x : T.t$ instead of just $\lambda x.t$. For metatheoretic analysis, I prefer to work with unannotated terms. This system (with unannotated terms) is also called $\lambda 2$.

2 Syntax

term variables x		
type variables X		
terms t	$::=$	$x \mid \lambda x.t \mid t\ t'$
types T	$::=$	$X \mid T \to T' \mid \forall X.T$

3 Typing

A typing context Γ declares free term and type variables:

$$\textit{Typing context}\ \Gamma ::= \cdot \mid \Gamma, x : T \mid \Gamma, X : \star$$

We treat Γ as a function, and write $\Gamma(x) = T$ to mean that Γ contains a declaration $x : T$. We will implicitly require that Γ does not declare any variable x twice. Variables can be implicitly renamed in λ-terms to make it possible to enforce this requirement. The typing rules are in Figure 1. To ensure that types are well-formed, we use some extra rules, called *kinding* rules, in Figure 2.

$$\frac{\Gamma(x) = T}{\Gamma \vdash x : T} \qquad \frac{\Gamma, x : T \vdash t : T'}{\Gamma \vdash \lambda x.t : T \to T')} \qquad \frac{\Gamma \vdash t : T_1 \to T_2 \quad \Gamma \vdash t' : T_1}{\Gamma \vdash t\ t' : T_2}$$

$$\frac{\Gamma, X : \star \vdash t : T}{\Gamma \vdash t : \forall X.T} \qquad \frac{\Gamma \vdash t : \forall X.T \quad \Gamma \vdash T' : \star}{\Gamma \vdash t : [T'/X]T}$$

Figure 1: Typing rules for unannotated System F

$$\frac{\Gamma(X) = \star}{\Gamma \vdash X : \star} \qquad \frac{\Gamma \vdash T_1 : \star \quad \Gamma \vdash T_2 : \star}{\Gamma \vdash T_1 \to T_2 : \star} \qquad \frac{\Gamma, X : \star \vdash T : \star}{\Gamma \vdash \forall X.T : \star}$$

Figure 2: Kinding rules for unannotated System F

$$\begin{array}{lcl} [\![X]\!]_\rho & = & \rho(X) \\ [\![T_1 \to T_2]\!]_\rho & = & \{t \in \mathcal{N} \mid \forall t' \in [\![T_1]\!]_\rho.\ t\ t' \in [\![T_2]\!]_\rho\} \\ [\![\forall X.T]\!]_\rho & = & \bigcap_{R \in \mathcal{R}} [\![T]\!]_{\rho[X \mapsto R]} \end{array}$$

Figure 3: Reducibility semantics for types

4 Semantics for types

Figure 3 gives a compositional semantics $[\![T]\!]_\rho$ for types. The function ρ gives the interpretations of free type variables in T. Each free type variable is interpreted as a *reducibility candidate*, and write ρ only for functions mapping type variables X to reducibility candidates. To define what a reducibility candidate is: let us denote the set of <u>closed</u> terms which normalize using call-by-name reduction as $\mathcal{N}$. We will write $\rightsquigarrow$ for call-by-name reduction. Then a reducibility candidate R is a set of terms satisfying the following requirements:

- $R \subseteq \mathcal{N}$
- If $t \in R$ and $t' \rightsquigarrow t$, then $t' \in R$

The set of all reducibility candidates is denoted $\mathcal{R}$.

Lemma 1 ($\mathcal{R}$ is a cpo)**.** *The set $\mathcal{R}$ ordered by subset forms a complete partial order, with greatest element $\mathcal{N}$ and greatest lower bound of a nonempty set of elements of $\mathcal{R}$ given by intersection.*

Proof. $\mathcal{N}$ satisfies both requirements for a reducibility candidate, and since one of those requirements is being a subset of $\mathcal{N}$, it is clearly the largest such set to do so. Let us prove that the intersection of a nonempty set S of reducibility candidates is still a reducibility candidate. Certainly if the members of S are subsets of $\mathcal{N}$ then so is $\bigcap S$. For the second property: assume an arbitrary $t \in \bigcap S$ with $t' \rightsquigarrow t$, and show $t' \in \bigcap S$. For the latter, it suffices to show $t' \in R$ for every $R \in S$. Consider an arbitrary such R. From $t \in \bigcap S$ and $R \in S$, we have $t \in R$. Then since R is a reducibility candidate, $t \in R$ and $t' \rightsquigarrow t$ implies $t' \in R$, . □

Lemma 2 (The semantics of types computes reducibility candidates)**.** *If $\rho(X)$ is defined for every free type variable of T, then $[\![T]\!]_\rho \in \mathcal{R}$.*

Proof. The proof is by induction on the structure of the type. If T is a type variable X, then by assumption, $\rho(X)$ is a reducibility candidate, and this is the value of $[\![T]\!]_\rho$.

If T is an arrow type $T_1 \to T_2$, we must prove the two properties listed above for being a reducibility candidate. Certainly $[\![T]\!]_\rho \subseteq \mathcal{N}$, because the semantics of arrow types requires this explicitly. Now suppose that $t \in [\![T_1 \to T_2]\!]_\rho$ and $t' \rightsquigarrow t$. We must show $t' \in [\![T_1 \to T_2]\!]_\rho$. Since t is normalizing and $t' \rightsquigarrow t$, we know that t' is also normalizing (there is a reduction sequence from t' to t and from t to a normal form). So let us assume an arbitrary $t'' \in [\![T_1]\!]_\rho$, and show that $t'\ t'' \in [\![T_2]\!]_\rho$. Since $t' \rightsquigarrow t$, by the definition of call-by-name reduction, we have

$$t'\ t'' \rightsquigarrow t\ t''$$

Since $t \in [\![T_1 \to T_2]\!]_\rho$, we know by the semantics of types that $t\ t'' \in [\![T_2]\!]_\rho$, since $t'' \in [\![T_1]\!]_\rho$. By the IH, $[\![T_2]\!]_\rho$ is a reducibility candidate. So since $t'\ t'' \rightsquigarrow t\ t''$ and $t\ t'' \in [\![T_2]\!]_\rho$, we also have $t'\ t'' \in [\![T_2]\!]_\rho$. This was all we had to prove in this case.

Finally, if T is a universal type $\forall X.T'$, then by IH, the set $[\![T']\!]_{\rho[X \mapsto R]}$ is a reducibility candidate for all $R \in \mathcal{R}$. Since $\mathcal{R}$ is a complete partial order, $\bigcap_{R \in \mathcal{R}} [\![T']\!]_{\rho[X \mapsto R]}$ is then also a reducibility candidate.

□

5 Soundness of Typing Rules

The goal of this section is to prove that terms which can be assigned a type using the rules of Figure 1 are normalizing. We will actually prove a stronger statement, based on an interpretation of typing judgments. First, we must define an interpretation $[\![\Gamma]\!]$ for typing contexts Γ. This interpretation will be a set of pairs (σ, ρ), where ρ is, as above, a function mapping type variables to reducibility candidates; and σ maps term variables to terms. The definition is by recursion on the structure of Γ:

$$\begin{array}{lcl} (\sigma,\rho) \in [\![x : T, \Gamma]\!] & \Leftrightarrow & \sigma(x) \in [\![T]\!]_\rho \ \wedge\ (\sigma,\rho) \in [\![\Gamma]\!] \\ (\sigma,\rho) \in [\![X : *, \Gamma]\!] & \Leftrightarrow & \rho(x) \in \mathcal{R} \ \wedge\ (\sigma,\rho) \in [\![\Gamma]\!] \\ (\sigma,\rho) \in [\![\cdot]\!] & & \end{array}$$

In the statement of the theorem below, we write σt to mean the result of simultaneously substituting $\sigma(x)$ for x in t, for all x in the domain of σ.

Lemma 3. *Suppose $(\sigma,\rho) \in [\![\Gamma]\!]$. If $t \in [\![T]\!]_\rho$, then $(\sigma[x \mapsto t], \rho) \in [\![\Gamma, x : T]\!]$. Also, if $R \in \mathcal{R}$, then $(\sigma, \rho[x \mapsto R]) \in [\![\Gamma, X : *]\!]$.*

Proof. The proof of the first part is by induction on Γ. If $\Gamma = \cdot$, then to show $(\sigma[x \mapsto t], \rho) \in [\![\cdot, x : T]\!]$, it suffices to show $t \in [\![T]\!]_\rho$, which holds by assumption. If $\Gamma = y : T', \Gamma'$, then we have $(\sigma, \rho) \in [\![\Gamma']\!]$ by the definition of $[\![\Gamma]\!]$, and we may apply the IH to conclude $(\sigma[x \mapsto t], \rho) \in [\![\Gamma', x : T]\!]$, from which we can conclude the desired $(\sigma[x \mapsto t], \rho) \in [\![\Gamma, x : T]\!]$, again by the definition of $[\![\Gamma]\!]$. Similar reasoning applies if $\Gamma = X : \star, \Gamma'$. The proof of the second part of the lemma is exactly analogous. □

Theorem 4 (Soundness of typing rules with respect to the semantics)**.** *If $\Gamma \vdash t : T$, then for all $(\sigma, \rho) \in [\![\Gamma]\!]$, we have $\sigma t \in [\![T]\!]_\rho$.*

Proof. The proof is by induction on the structure of the assumed typing derivation. In each case, we will implicitly assume an arbitrary $(\sigma, \rho) \in [\![\Gamma]\!]$.

<u>Case:</u>

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

We proceed by inner induction on Γ. If Γ is empty, then $\Gamma(x) = T$ is false, and this case cannot arise. Suppose Γ is of the form $x : T, \Gamma'$. Then $\sigma(x) \in [\![T]\!]_\rho$ by definition of $[\![\Gamma]\!]$, which suffices to prove the conclusion. Suppose Γ is of the form $y : T, \Gamma'$, where $y \neq x$, or of the form $X : *, \Gamma'$. Then $\Gamma'(x) = T$ and $(\sigma, \rho) \in [\![\Gamma']\!]$, and we use the induction hypothesis to conclude $\sigma x \in [\![T]\!]_\rho$.

<u>Case:</u>

$$\frac{\Gamma, x : T \vdash t : T'}{\Gamma \vdash \lambda x.t : T \to T')}$$

To prove $(\lambda x.\sigma t) \in [\![T \to T']\!]_\rho$, it suffices to assume an abitrary $t' \in [\![T]\!]_\rho$ and prove $(\lambda x.\sigma t)\ t' \in [\![T']\!]_\rho$. Since $[\![T']\!]_\rho$ is a reducibility candidate, it suffices to prove $[t'/x]\sigma t \in [\![T']\!]_\rho$, since $(\lambda x.\sigma t)\ t' \rightsquigarrow [t'/x](\sigma t)$. But if we let $\sigma' = \sigma[x \mapsto t']$, then we have $(\sigma', \rho) \in [\![\Gamma, x : T]\!]$ by Lemma 3, so we may apply the IH to conclude $\sigma' t \in [\![T']\!]_\rho$, as required.

Case:

$$\frac{\Gamma \vdash t : T_1 \to T_2 \quad \Gamma \vdash t' : T_1}{\Gamma \vdash t\ t' : T_2}$$

By the IH, $\sigma t \in [\![T_1 \to T_2]\!]_\rho$ and $\sigma t' \in [\![T_1]\!]_\rho$. By the semantics of arrow types, this immediately implies $(\sigma t)\ (\sigma t') \in [\![T_2]\!]_\rho$, as required.

Case:

$$\frac{\Gamma, X : \star \vdash t : T}{\Gamma \vdash t : \forall X.T}$$

We must prove $\sigma t \in [\![\forall X.T]\!]_\rho$. By the semantics of universal types, it suffices to assume an arbitrary $R \in \mathcal{R}$, and prove $\sigma t \in [\![T]\!]_{\rho[X \mapsto R]}$. But this follows by the IH, which we can apply because $(\sigma, \rho[X \mapsto R]) \in [\![\Gamma, X : \star]\!]$, by Lemma 3.

Case:

$$\frac{\Gamma \vdash t : \forall X.T \quad \Gamma \vdash T' : \star}{\Gamma \vdash t : [T'/X]T}$$

By the IH, we know $\sigma t \in [\![\forall X.T]\!]_\rho$, which by the semantics of universal types is equivalent to

$$\sigma t \in \bigcap_{R \in \mathcal{R}} T_{\rho[X \mapsto R]} \tag{1}$$

Since $(\sigma, \rho) \in [\![\Gamma]\!]$, we may easily observe that ρ is defined for all the free type variables of T'. So by Lemma 2, $[\![T']\!]_\rho \in \mathcal{R}$. From the displayed formula above (1), we can conclude $\sigma t \in [\![T]\!]_{\rho[X \mapsto [\![T']\!]_\rho]}$. Now we must apply the following lemma, whose easy proof by induction on T we omit, to conclude $\sigma t \in [\![[T'/X]T]\!]_\rho$.

Lemma 5. $[\![[T'/X]T]\!]_\rho = [\![T]\!]_{\rho[X \mapsto T']}$

□