

# The untyped lambda calculus

CIS 6700, Spring 2023

February 8, 2024

## 1 Untyped lambda calculus

This document describes the untyped lambda-calculus, with the following grammar. References are to definitions in Barendregt, “The Lambda Calculus: Its Syntax and Semantics”.

$tm, t, u$	$::=$	terms
	$  x$	variables
	$  \lambda x. t$	abstractions
	$  t u$	function application

In  $\beta$ -reduction, the argument of an application is substituted for the bound variable in an abstraction. We use the  $t[x \rightsquigarrow u]$  notation for substituting the term  $u$  for the variable  $x$  in the term  $t$ .

$t$   $\beta$ -reduces to  $u$  *(Definition 3.1.3)*

$$\begin{array}{c} \text{BETA-REDUCT} \\ \hline (\lambda x. t) u \beta\text{-reduces to } t[x \rightsquigarrow u] \end{array}$$

We can define a deterministic, small-step evaluation relation by reducing in the heads of applications. This is a *call-by-name* semantics and performs head-reduction.

$t \rightsquigarrow t'$  *(Small-step evaluation)*

$$\begin{array}{cc} \text{S-APP} & \text{S-BETA} \\ \frac{t \rightsquigarrow t'}{t u \rightsquigarrow t' t} & \frac{}{(\lambda x. t) u \rightsquigarrow t[x \rightsquigarrow u]} \end{array}$$

We can also define a nondeterministic single-step full-reduction by performing  $\beta$ -reduction in any subterm. Iterating this reduction will convert a term into its  $\beta$ -normal form.

$$\boxed{t \longrightarrow_{\beta} u} \quad (\text{Full } \beta\text{-reduction})$$

$$\begin{array}{c}
\text{F-BETA} \\
\frac{}{(\lambda x.t) u \longrightarrow_{\beta} t[x \rightsquigarrow u]}
\end{array}
\quad
\begin{array}{c}
\text{F-ABS} \\
\frac{t \longrightarrow_{\beta} t'}{\lambda x.t \longrightarrow_{\beta} \lambda x.t'}
\end{array}
\quad
\begin{array}{c}
\text{F-APP1} \\
\frac{t \longrightarrow_{\beta} t'}{t u \longrightarrow_{\beta} t' u}
\end{array}$$

$$\begin{array}{c}
\text{F-APP2} \\
\frac{u \longrightarrow_{\beta} u}{t u \longrightarrow_{\beta} t u'}
\end{array}$$

We can define when two terms are equivalent up to  $\beta$ .

$$\boxed{t \equiv_{\beta} u} \quad (\beta\text{-conversion or } \beta\text{-equality (Definition 2.1.4)})$$

$$\begin{array}{c}
\text{EQ-BETA} \\
\frac{}{(\lambda x.t) u \equiv_{\beta} t[x \rightsquigarrow u]}
\end{array}
\quad
\begin{array}{c}
\text{EQ-REFL} \\
\frac{}{t \equiv_{\beta} t}
\end{array}
\quad
\begin{array}{c}
\text{EQ-SYM} \\
\frac{u \equiv_{\beta} t}{t \equiv_{\beta} u}
\end{array}
\quad
\begin{array}{c}
\text{EQ-TRANS} \\
\frac{t_1 \equiv_{\beta} t_2 \quad t_2 \equiv_{\beta} t_3}{t_1 \equiv_{\beta} t_3}
\end{array}$$

$$\begin{array}{c}
\text{EQ-APP1} \\
\frac{t \equiv_{\beta} t'}{t u \equiv_{\beta} t' u}
\end{array}
\quad
\begin{array}{c}
\text{EQ-APP2} \\
\frac{u \equiv_{\beta} u'}{t u \equiv_{\beta} t u'}
\end{array}
\quad
\begin{array}{c}
\text{EQ-ABS} \\
\frac{t \equiv_{\beta} t'}{\lambda x.t \equiv_{\beta} \lambda x.t'}
\end{array}$$

Finally, the Church-Rosser Theorem relies on the definition of parallel reduction. This is a version of reduction that is confluent.

$$\boxed{t \Longrightarrow_{\beta} u} \quad (\text{Parallel reduction (3.2.3)})$$

$$\begin{array}{c}
\text{P-BETA} \\
\frac{t \Longrightarrow_{\beta} \lambda x.t' \quad u \Longrightarrow_{\beta} u'}{t u \Longrightarrow_{\beta} t'[x \rightsquigarrow u']}
\end{array}
\quad
\begin{array}{c}
\text{P-VAR} \\
\frac{}{x \Longrightarrow_{\beta} x}
\end{array}
\quad
\begin{array}{c}
\text{P-ABS} \\
\frac{t \Longrightarrow_{\beta} t'}{\lambda x.t \Longrightarrow_{\beta} \lambda x.t'}
\end{array}
\quad
\begin{array}{c}
\text{P-APP} \\
\frac{t \Longrightarrow_{\beta} t' \quad u \Longrightarrow_{\beta} u'}{t u \Longrightarrow_{\beta} t' u'}
\end{array}$$

## 2 Relation operations

Many of the operations above can be generated by applying the following closure operations to the  $\beta$  reduction relation. These operations are parameterized by an arbitrary relation  $R$ .

Note that  $t \rightarrow_R u$  with  $R$  equal to  $\beta$  is the same relation as  $t \longrightarrow_{\beta} u$ . And, the compatible, reflexive, symmetric and transitive closure of  $\beta$  is the same relation as  $t \equiv_{\beta} u$ .

$$\boxed{t \rightarrow_R u} \quad (\text{Compatible closure of } R \text{ (aka one-step reduction (3.1.5))})$$

$$\begin{array}{c}
\text{CC-REL} \\
\frac{R t u}{t \rightarrow_R u}
\end{array}
\quad
\begin{array}{c}
\text{CC-ABS} \\
\frac{t \rightarrow_R u}{\lambda x.t \rightarrow_R \lambda x.u}
\end{array}
\quad
\begin{array}{c}
\text{CC-APP1} \\
\frac{t \rightarrow_R t'}{t u \rightarrow_R t' u}
\end{array}
\quad
\begin{array}{c}
\text{CC-APP2} \\
\frac{u \rightarrow_R u'}{t u \rightarrow_R t u'}
\end{array}$$

$$\boxed{t \rightarrow_R^- u} \quad (\text{reflexive closure of } R \text{ (3.1.4)})$$

$$\frac{\text{R-REL} \quad R t u}{t \rightarrow_R^- u}$$

$$\frac{\text{R-REFL}}{t \rightarrow_R^- t}$$

$$\boxed{t \rightarrow_R^+ u} \quad (\text{transitive closure of } R \text{ (3.1.4)})$$

$$\frac{\text{T-REL} \quad R t u}{t \rightarrow_R^+ u}$$

$$\frac{\text{T-TRANS} \quad t_1 \rightarrow_R^+ t_2 \quad t_2 \rightarrow_R^+ t_3}{t_1 \rightarrow_R^+ t_3}$$

$$\boxed{t \rightarrow_R^* u} \quad (\text{reflexive-transitive closure of } R)$$

$$\frac{\text{RT-REL} \quad R t u}{t \rightarrow_R^* u}$$

$$\frac{\text{RT-REFL}}{t \rightarrow_R^* t}$$

$$\frac{\text{RT-TRANS} \quad t_1 \rightarrow_R^* t_2 \quad t_2 \rightarrow_R^* t_3}{t_1 \rightarrow_R^* t_3}$$

$$\boxed{t \leftrightarrow_R u} \quad (\text{symmetric-transitive closure of } R)$$

$$\frac{\text{ST-REL} \quad R t u}{t \leftrightarrow_R u}$$

$$\frac{\text{ST-SYM} \quad u \leftrightarrow_R t}{t \leftrightarrow_R u}$$

$$\frac{\text{ST-TRANS} \quad t_1 \leftrightarrow_R t_2 \quad t_2 \leftrightarrow_R t_3}{t_1 \leftrightarrow_R t_3}$$

### 3 Eta-reduction

By changing the definition of the underlying primitive reduction, we can also reason about  $\eta$ -reduction and  $\beta\eta$ -equivalence. Note that the rule for  $\eta$ -reduction has been stated in a way that generates the appropriate output for the locally-nameless representation in Coq. In this output, the fact that  $x \notin \text{fv}t$  is implicit and does not need to be added as a precondition to the rule.

$$\boxed{t \text{ } \eta\text{-reduces to } u} \quad (\text{Definition 3.1.1 (i)})$$

$$\frac{\text{ETA-REDUCT} \quad t' = t x}{\lambda x. t' \text{ } \eta\text{-reduces to } t}$$

$$\boxed{t \text{ } \beta\eta\text{-reduces to } u} \quad (\text{Either } \beta \text{ or } \eta \text{ reduction (Definition 3.1.1 (ii))})$$

$$\frac{\text{ETA-BETA} \quad t \text{ } \beta\text{-reduces to } u}{t \text{ } \beta\eta\text{-reduces to } u}$$

$$\frac{\text{ETA-ETA} \quad t \text{ } \eta\text{-reduces to } u}{t \text{ } \beta\eta\text{-reduces to } u}$$