

3.2 Semantics of the Lambda Calculus

Mittwoch, 8. Juni 2016 11:00

We now define the operational semantics of the lambda calculus by introducing an interpreter for lambda terms that uses 3 reduction rules: α -reduction, β -reduction, δ -reduction

α -reduction: renaming bound variables

Def 3.2.1 (α -Reduction) (Slide 54)

The relation $\rightarrow_\alpha \subseteq \Lambda \times \Lambda$ is the smallest relation with

- $\lambda x. t \rightarrow_\alpha \lambda y. t[x/y]$ if $y \notin \text{free}(t)$
- if $t_1 \rightarrow_\alpha t_2$, then $(t_1 r) \rightarrow_\alpha (t_2 r)$,
 $(r t_1) \rightarrow_\alpha (r t_2)$, and $(\lambda y. t_1) \rightarrow_\alpha (\lambda y. t_2)$.

Ex:

$$\lambda x y. x y \rightarrow_\alpha \lambda x z. x z$$

$$\rightarrow_\alpha \lambda v z. v z$$

$$\rightarrow_\alpha \dots$$

$$\lambda x. x y \not\rightarrow_\alpha \lambda x. x z$$

No renaming
of free
variables

β -reduction is used to apply a λ -abstraction to a term: $(\lambda x. t) r$ results in t where all free occurrences of x are replaced by r

Def 3.2.2 (β -Reduction) (Slide 54)

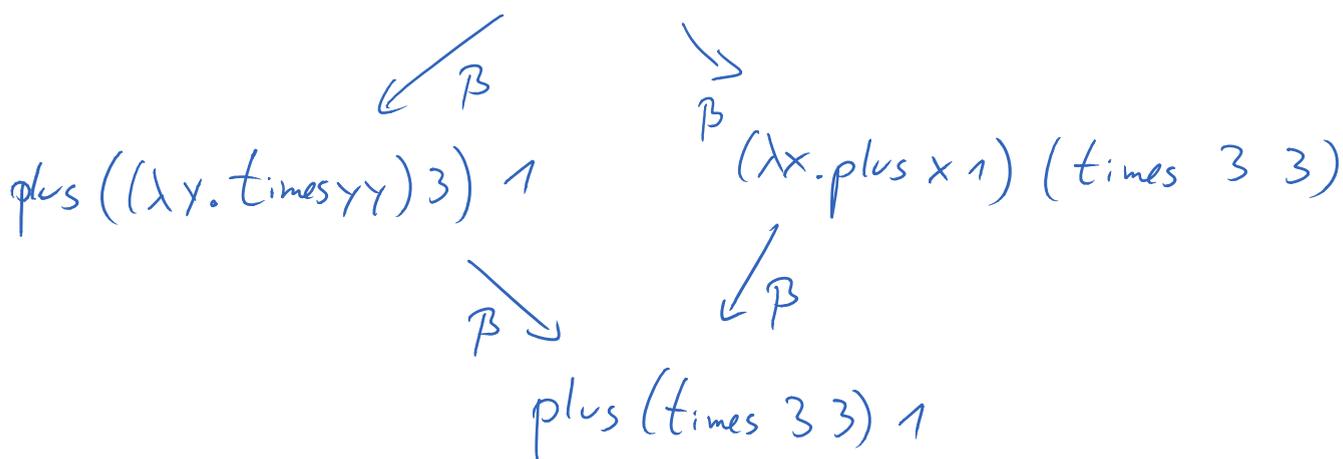
The relation $\rightarrow_{\beta} \subseteq \Lambda \times \Lambda$ is the smallest relation with

- $(\lambda x. t) r \rightarrow_{\beta} t [x/r]$
- if $t_1 \rightarrow_{\beta} t_2$, then $(t_1 r) \rightarrow_{\beta} (t_2 r)$, $(r t_1) \rightarrow_{\beta} (r t_2)$, and $(\lambda y. t_1) \rightarrow_{\beta} (\lambda y. t_2)$

Ex: $(\lambda x. x) \text{zero} \rightarrow_{\beta} x [x/\text{zero}] = \text{zero}$

$(\lambda x y. x y) y \rightarrow_{\beta} (\lambda y. x y) [x/y] = \lambda y'. y y'$

$(\lambda x. \text{plus } x \ 1) ((\lambda y. \text{times } y \ y) \ 3)$



There can be several possibilities to evaluate a

λ -term by β -reduction. Do they always yield the same result?

Def 3.2.3 (transitive-reflexive closure, normal form, confluence)

Let \rightarrow be a relation on some set N .

(a) The transitive-reflexive closure \rightarrow^* is the smallest relation such that:

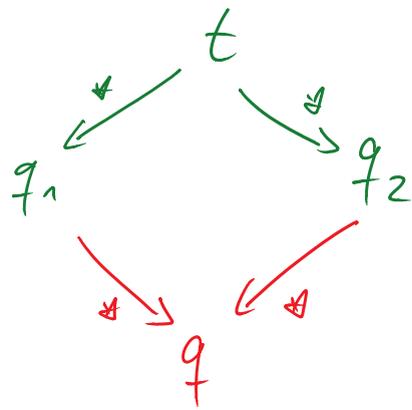
- $t_1 \rightarrow t_2$ implies $t_1 \rightarrow^* t_2$
- $t_1 \rightarrow t_2 \rightarrow^* t_3$ implies $t_1 \rightarrow^* t_3$
- $t_1 \rightarrow^* t_1$

In other words: $t_0 \rightarrow^* t_n$ iff $t_0 \rightarrow t_1 \rightarrow t_2 \rightarrow \dots \rightarrow t_n$
for $n \geq 0$

(b) An object $q \in N$ is a normal form iff there is no $q' \in N$ with $q \rightarrow q'$.

We say that q is a normal form of t iff $t \rightarrow^* q$ and q is a normal form.

(c) The relation \rightarrow is confluent iff for all $t, q_1, q_2 \in N$ we have: if $t \rightarrow^* q_1$ and $t \rightarrow^* q_2$, then there exists a $q \in N$ with $q_1 \rightarrow^* q$ and $q_2 \rightarrow^* q$.

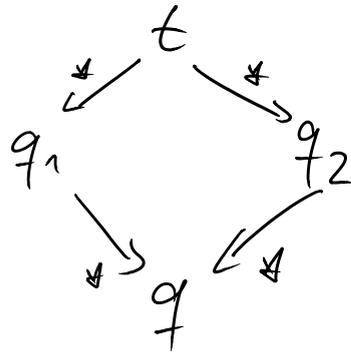


if the green part holds,
then the red part holds as well.

Lemma 3.2.4 (Confluence implies unique normal forms)

Let \rightarrow be a confluent relation on a set N . Then every $t \in N$ has at most one normal form.

Proof: Let $t \in N$ have 2 normal forms t_1, t_2 .



By confluence, there exists a $q \in N$ with $t_1 \rightarrow^* q, t_2 \rightarrow^* q$.

Since t_1, t_2 are normal forms, we have

$$t_1 = q = t_2.$$

□

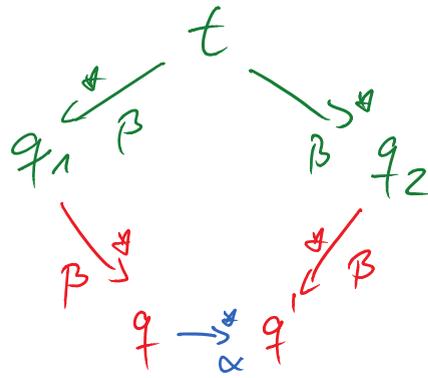
Thm 3.2.5 (Confluence of the λ -calculus with β -reduction, Church & Rosser)

\rightarrow_β is confluent, i.e.,

if $t \rightarrow_\beta^* t_1$ and $t \rightarrow_\beta^* t_2$,

then there exist $q, q' \in \Lambda$ with

$$t_1 \rightarrow_\beta^* q, t_2 \rightarrow_\beta^* q', \text{ and } q \rightarrow_\beta^* q'$$



The last form of reduction rules in the λ -calculus is needed to evaluate terms built with constants from \mathcal{C} . In particular, these constants could correspond to pre-defined functions of Haskell.

But: we want to make sure that δ -reduction does not destroy the confluence of the λ -calculus.

Solution: Define δ -reduction by a set of δ -rules.

These rules must have a certain restricted form which ensures that evaluation with β - and δ -reduction remains confluent.

Def 326 (δ -Reduction) (Slide 55)

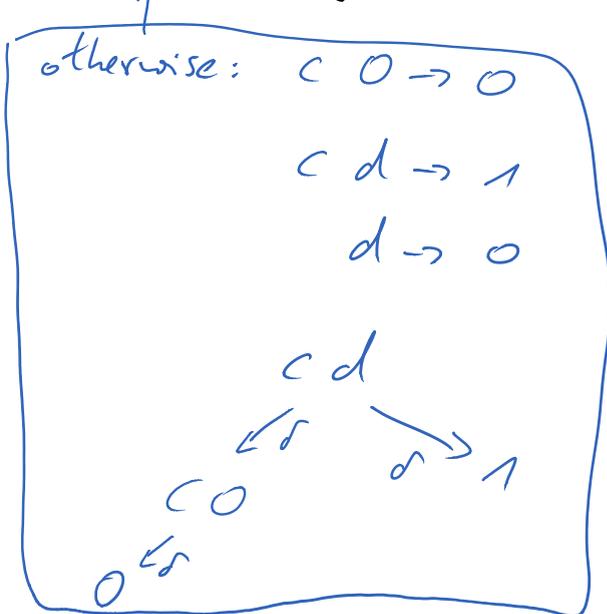
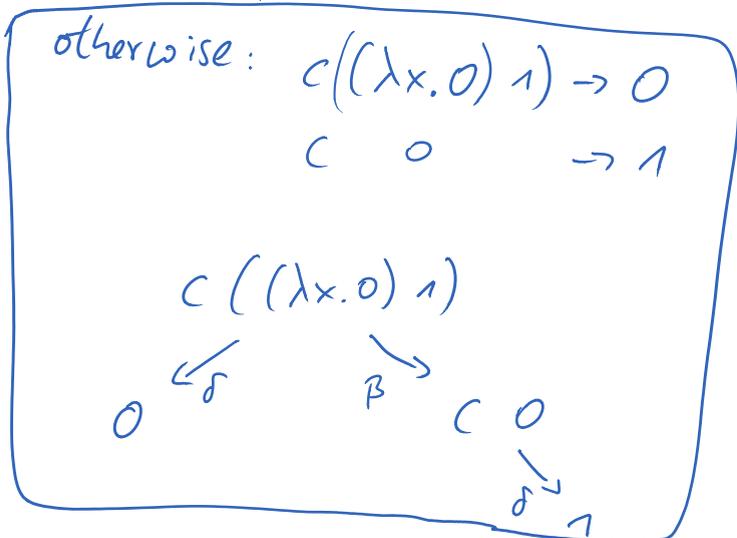
A set of rules δ of the form $c t_1 \dots t_n \rightarrow r$ with $c \in \mathcal{C}$, $t_1, \dots, t_n, r \in \Lambda$ is a Delta-Rule-Set iff t_1, \dots, t_n, r are closed λ -terms

otherwise: $\left. \begin{array}{l} c 0 \rightarrow 0 \\ c x \rightarrow 1 \end{array} \right\} \delta$

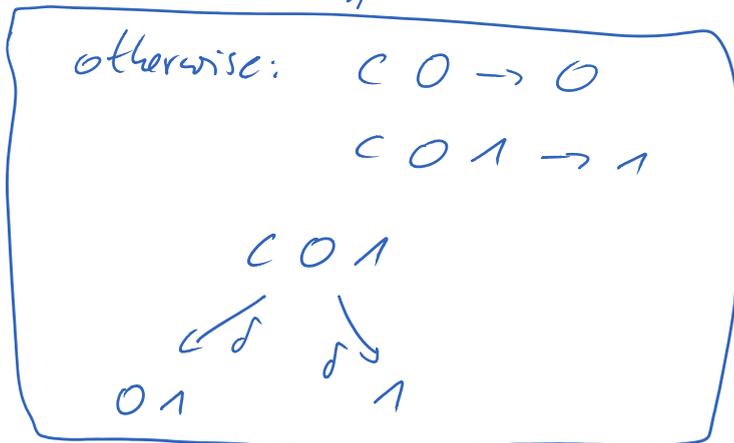
Then $\begin{array}{c} c 0 \\ \swarrow \delta \quad \searrow \end{array}$



• t_1, \dots, t_n are in \rightarrow_{β} -normal form and they do not contain any left-hand side of a rule from δ



• In δ , there are no two rules $ct_1 \dots t_n \rightarrow r$, $ct_1 \dots t_n \dots t_m \rightarrow r'$ with $m \geq n$.



For such a set δ , \rightarrow_{δ} is the smallest relation with

- $l \rightarrow_{\delta} r$ for all $l \rightarrow r \in \delta$
- if $t_1 \rightarrow_{\delta} t_2$, then $(t_1 r) \rightarrow_{\delta} (t_2 r)$, $(r t_1) \rightarrow_{\delta} (r t_2)$, and $\lambda y. t_1 \rightarrow_{\delta} \lambda y. t_2$

We define $\rightarrow_{\beta\delta} = \rightarrow_{\beta} \cup \rightarrow_{\delta}$.

Ex. for a Delta-Rule-Set:

$\delta = \{ \text{isa}_{\text{Succ}} (\text{Succ } t) \rightarrow \text{True} \mid t \in \underline{A}, t \text{ is closed,}$
and in $\rightarrow_{\beta\delta}$ -normal form $\} \cup$

$\{ \text{isa}_{\text{Succ}} \text{ zero} \rightarrow \text{False} \}$

Thm 3.2.7 (Confluence for the λ -calculus with
 \rightarrow_{β} and \rightarrow_{δ})

$\rightarrow_{\beta\delta}$ is confluent, i.e.,

if $t \xrightarrow{\delta}_{\beta\delta} q_1$ and $t \xrightarrow{\delta}_{\beta\delta} q_2$,

then there exist $q, q' \in \underline{A}$ with

$q_1 \xrightarrow{\delta}_{\beta\delta} q$, $q_2 \xrightarrow{\delta}_{\beta\delta} q'$, and $q \xrightarrow{\delta}_{\alpha} q'$.

So $\rightarrow_{\beta\delta}$ defines an operational semantics for the
 λ -calculus (and \rightarrow_{α} defines that we regard certain
terms as being "equal").

(A denotational semantics for the λ -calculus
was only invented 30 years later by D. Scott.
Crucial idea: continuous functions).