Lecture 4

Untyped Lambda Calculus
as a model of
computation

- variables
- (functions =) abstractions
- application (= fn call)

$$e ::= x \mid \lambda x, e_1 \mid (e_1 \sqcup e_2)$$

var (atom)

abstraction (function)

applic (fn call)

\x, e1

atoms

size — number of [illegible]

e	(size e)
x	1
$\lambda x. e_1$	$1 + (\text{size } e_1)$
$(e_1 \ e_2)$	$(\text{size } e_1) + (\text{size } e_2)$

tree height

e	(ht e)
x	0
$\lambda x. e_1$	$1 + (\text{ht } e_1)$
$(e_1 \ e_2)$	$1 + (\max e_1 \ e_2)$

term e occurs in e'

- e occurs in e
- e occurs in $\lambda x. e_1$ if
  - $e \equiv x$, or
  - e occurs in e_1

- e occurs in $e_1\, e_2$ if
  - e occurs in e_1, or
  - e occurs in e_2

$$\boxed{\text{e is a subterm of e' if e occurs in e'.}}$$

Binding occurrence

$$\lambda x.\, e_1$$

binding occurrence of x

scope of this binding

Bound occurrence of a variable

occurrence of variable x in e is bound if this occurrence is in some "subterm" $\lambda x.e_1$ of e.

Free occurrence of a variable

- an occurrence of variable x in e which is neither

binding occurrence

- a binding
- a bound occurrence.

Examples

$$\lambda x.\ \lambda y.\ x$$

free

$$\lambda x.\ \lambda y.\ (z\ x)$$

$$(\lambda x.\ x)\ x$$

binding   bound   free

fv - free variables in a term

e	$(fv\ e)$
x	$\{x\}$
$\lambda x.\ e_1$	$(fv\ e_1) - \{x\}$
$e_1\ e_2$	$(fv\ e_1) \cup (fv\ e_2)$

closed if

Closed term e is closed if

$(fv\ e) = \{\}$

Substitution

$[x := e_2]\, e_1$ or $e_1 [e_2/x]$

"substitute e_2 for all free occurrences of x in e_1"

	e_1	$[x := e_2]\, e_1$	
(a)	x	e_2	
(b)	y	y	$y \not\equiv x$
(c)	$e_{11}\ e_{12}$	$([x := e_2]e_{11})\ ([x := e_2]e_{12})$	
(d)	$\lambda x. e_{11}$	$\lambda x. e_{11}$	
(e)	$\lambda y. e_{11}$	$\lambda y. ([x := e_2]e_{11})$	$y \not\equiv x$, & $(x \notin (fv\ e_{11})$ or $y \notin (fv\ e_2))$

(f)	$\lambda y, e_{11}$	$\lambda z, ([x := e_2]([y := z]\, e_{11}))$ $y \not\equiv x$, & $y \in (fv\ e_2)$ & $x \in (fv\ e_{11})$ $z \notin (fv\ e_{11})$ $z \notin (fv\ e_2)$

Lemma

(a) $[x := x]\, e \equiv e$

(b) $x \notin (fv\ e_1) \longrightarrow$
$$[x := e_2]\, e_1 \equiv e_1$$

(c) $x \in (fv\ e_1) \longrightarrow$
$$fv\ ([x := e_2]\, e_1) = (fv\ e_2) \cup (fv\ e_1) - \{x\}$$

(d) $size\ ([x := y]\, e_1) = (size\ e_1)$

Lemma

Let x, y, z be distinct variables

No vbl bound in e_1 are free in
z, e_2, e_3

(a) $z \notin (fv\ e_1) \rightarrow$

$$[z := e_2][x := z]\, e_1 \equiv [x := e_2]\, e_1$$

(b) $z \notin (fv\ e_1) \longrightarrow$

$$[z := x][x := z]\, e_1 \equiv e_1$$

(c) $y \notin (fv\ e_2) \rightarrow$

$$[x := e_2][y := e_3]\, e_1 \equiv [y := \underbrace{[x := e_2]e_3}][x := e_2]\, e_1$$

(d) $y \notin (fv\ e_2) \rightarrow x \notin (fv\ e_3) \rightarrow$

$$[x := e_2][y := e_3]\, e_1 \equiv [y := e_3][x := e_2]\, e_1$$

(e) $[x := e_2][x := e_3]\, e_1 \equiv [x := \underbrace{[x := e_2]e_3}]\, e_1$

Change of Bound Variables

Let $\lambda x.\, e_1$ occur in e_2

Let $y \notin (fv\ e_1)$

Let $e_3 \equiv e_2$ with subterm $\lambda x.\, e_1$ replaced by $\lambda y.\, ([x := y]\, e_1)$

Bound variable x has been replaced by bound variable y (scope is e_1)

[We write $e_2 \triangleright_\alpha e_3$ iff
e_3 obtained from e_2 by a
series of 0 or more changes of
bound variables]

<u>Lemma</u>

(a) $e_1 \triangleright_\alpha e_2 \rightarrow$
$$(fv\ e_1) = (fv\ e_2)$$

(b) forall e_1,
forall variables $x_1 \ldots x_n$
exists e_2 s.t
$e_1 \triangleright_\alpha e_2$ and
forall x_i, x_i <u>not</u> bound in e_2

(c) $\triangleright_\alpha$ is Reflexive
Transitive
Symmetric
· CONGRUENCE

$\boxed{\therefore \text{ write } \equiv_\alpha \text{ instead of } \triangleright_\alpha}$

Lemma:

Assume x, y, z distinct

$(x \not\equiv y,\ y \not\equiv z,\ x \not\equiv z)$

(a) $z \notin (\mathrm{fv}\ e_1) \rightarrow$
$$[z := e_2][x := z]\, e_1 \equiv_\alpha [x := e_2]\, e_1$$

(b) $z \notin (\mathrm{fv}\ e_1) \rightarrow$
$$[z := x][x := z]\, e_1 \equiv_\alpha e_1$$

(c) $y \notin (\mathrm{fv}\ e_2) \rightarrow$
$$[x := e_2][y := e_3]\, e_1 \equiv_\alpha [y := \underbrace{[x := e_2] e_3}][x := e_2]\, e_1$$

(d) $y \notin (\mathrm{fv}\ e_2) \rightarrow x \notin (\mathrm{fv}\ e_3) \rightarrow$
$$[x := e_2][y := e_3]\, e_1 \equiv_\alpha [y := e_3][x := e_2]\, e_1$$

(e) $$[x := e_2][x := e_3]\, e_1 \equiv_\alpha [x := \underbrace{[x := e_2] e_3}]\, e_1$$

Substitution Lemma of $\equiv_\alpha$

$$e_1 \equiv_\alpha e_1' \longrightarrow$$
$$e_2 \equiv_\alpha e_2' \longrightarrow$$
$$[x := e_2]\, e_1 \equiv_\alpha [x := e_2']\, e_1'$$

$\equiv_\alpha$ is a "pain" to deal with

- Representations of λ-calc w/o bound variables
  - COMBINATORY LOGIC
  - De Bruijn Indices

Beta Reduction

β-redex $\quad (\lambda x. e_1)\, e_2$

$$(\beta) \quad (\lambda x. e_1)\, e_2 \;\triangleright_{1\beta}\; [x := e_2]\, e_1$$

Redex   contractum

Now allow this at any subterm position

$$(op)\quad \frac{e_1 \rhd_{1\beta} e_1'}{e_1\ e_2 \rhd_{1\beta} e_1'\ e_2}$$

$$(arg)\quad \frac{e_2 \rhd_{1\beta} e_2'}{e_1\ e_2 \rhd_{1\beta} e_1\ e_2'}$$

$$(\xi)\quad \frac{e \rhd_{1\beta} e'}{\lambda x.e \rhd_{1\beta} \lambda x.e'}$$

β-REDUCTION RELATION $\rhd_\beta$

$e_1 \rhd_\beta e_2$ iff e_2 obtained from e_1 by a finite series of 0 or more $\rhd_{1\beta}$ and $\equiv_\alpha$ steps.

Reflexive Transitive Closure of $\rhd_{1\beta}$ (modulo $\equiv_\alpha$)

β- NORMAL FORM

e is in β-nf if e does not contain any subterm that is a β-redex.

e has a β-nf if
$e \triangleright_\beta e'$ for some β-nf e'.

Note: $\boxed{\text{is in } \beta\text{-nf}}^{A}$ stronger property than $\boxed{\text{has a } \beta\text{-nf.}}^{B}$

$A \longrightarrow B.$

Note: $\triangleright_{1\beta}$ is a "non-deterministic" relation because a term may contain more than one redex

EXAMPLES $(\lambda x. y) z \triangleright_{1\beta} y$

Let $K \equiv (\lambda x . \lambda y . x)$

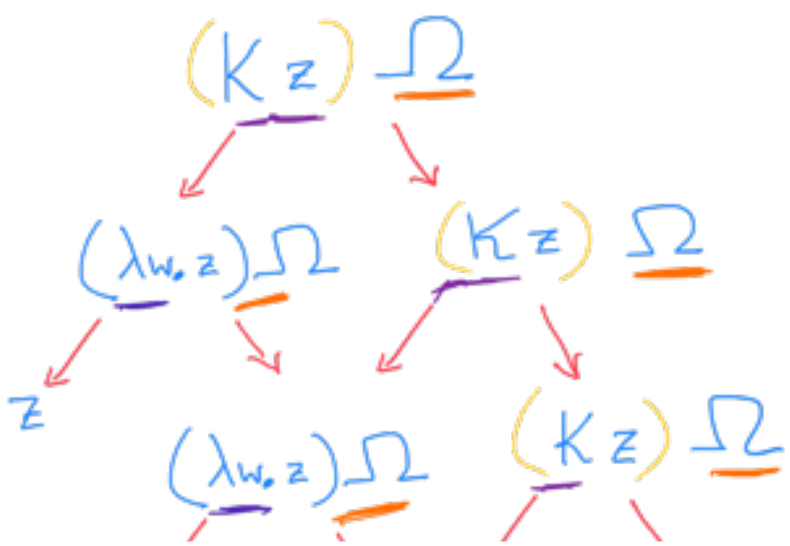
$$
\begin{aligned}
((K\ e_1)\ e_2) &\rhd_{1\beta} ([x := e_1]\ \lambda y . x)\ e_2 && ① \\
&\equiv_\alpha (\lambda w . e_1)\ e_2 && w \notin (fv\ e_1) \\
&\rhd_{1\beta} [w := e_2]\ e_1 && ② \\
&\equiv_\alpha e_1
\end{aligned}
$$

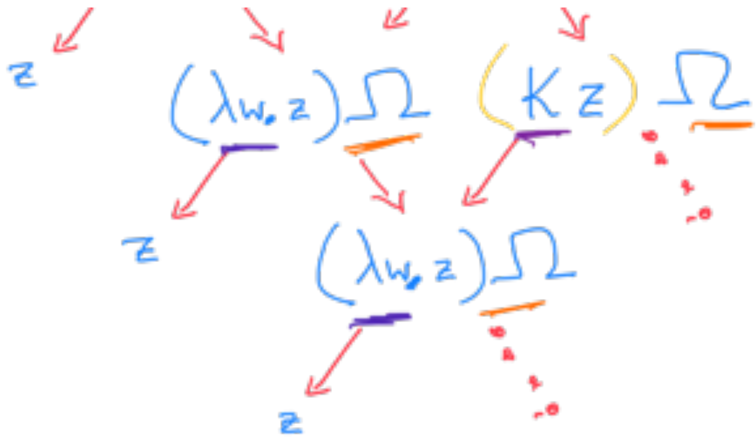
$$(\lambda x . (x\ x))\ z \ \rhd_{1\beta}\ (z\ z)$$

Let $\Delta \equiv \lambda x . (x\ x)$

$$\Delta\ \Delta \ \rhd_{1\beta}\ \Delta\ \Delta$$

Let $\Omega \equiv \Delta\ \Delta$





QUESTIONS:

- Given any term e, does it have a β-nf?
- If e has a β-nf, is that unique?
- If e has a β-nf, do we have a strategy for finding that β-nf.?

<u>Lemma</u> α-Congruence Lemma for $\rhd_\beta$

H1 $\quad e_1 \equiv_\alpha e_1' \longrightarrow$

H2 $\quad e_2 \equiv_\alpha e_2' \longrightarrow$

H3 $\quad e_1 \rhd_\beta e_2 \longrightarrow$

$$e_1' \rhd_\beta e_2'$$

Proof

$$\begin{tikzcd}
& e_1 \arrow[r, phantom, "\rhd_\beta"] \arrow[r, dash, "H3"'] & e_2 \arrow[dr, dash, "\equiv_\alpha", "H_2"'] & \\
e_1' \arrow[ur, dash, "\equiv_\alpha", "H1"'] & & & e_2'
\end{tikzcd}$$

$$\boxed{\text{Substitution Lemma for } \rhd_\beta}$$

(a) $e_1 \rhd_\beta e_2 \longrightarrow$
$$x \notin (\mathrm{fv}\ e_1) \longrightarrow x \notin (\mathrm{fv}\ e_2)$$

(b) $e_1 \rhd_\beta e_2 \longrightarrow$
$$[x := e_1]\, e_3 \rhd_\beta [x := e_2]\, e_3$$

(c) $e_1 \rhd_\beta e_2 \longrightarrow$
$$[x := e_3]\, e_1 \rhd_\beta [x := e_3]\, e_2$$

Context and redex

$$\mathcal{C}[\ ] ::= [\ ]$$
$$\mid e\,\mathcal{C}[\ ]$$
$$\mid \mathcal{C}[\ ]\, e$$
$$\mid \lambda x.\, \mathcal{C}[\ ]$$

Now all 4 rules captured by:

$$\mathcal{C}[(\lambda x.e_1)\, e_2] \rhd_{1\beta} \mathcal{C}[[x:=e_2]e]$$

—

β-reducible : Can factor into
CONTEXT $\mathcal{C}[\ ]$ and redex r

.

Recall

$\rhd_\beta$: 0 or more β-reductions
$(\rhd_{1\beta})^*$

[Allow α-conversions]

(renaming of bound variables freely whenever we wish).

CONFLUENCE OF $\triangleright_\beta$

CHURCH-ROSSER THM OF $\triangleright_\beta$

$$
\begin{tikzcd}
\text{IF} & & e \arrow[dl, "\triangleright_\beta"'] \arrow[dr, "\triangleright_\beta"] & & \\
& e_1 \arrow[dr, "\triangleright_\beta"'] & & e_2 \arrow[dl, "\triangleright_\beta"] & \\
\text{THEN} & & e_3 & & \text{FOR SOME } e_3
\end{tikzcd}
$$

β-nfs are UNIQUE (UPTO α-cnv)

· If

e_1 and e_2 in β-nf, then

$$e_1 \equiv_\alpha e_2 .$$

· If $e \triangleright_\beta e_1$ e_1 in β-nf

and $e \triangleright_\beta e_2$

then $e_2 \triangleright_\beta e_1$.

Note: THIS DOES <u>NOT</u> MEAN EVERY EXPRESSION MUST TERMINATE IN A β-nf.

Recall

$$\Delta \equiv \lambda x.(x\ x)$$

Consider $\Omega \equiv (\Delta\ \Delta)$

$$\Delta\Delta \equiv_\alpha (\lambda x.(x\ x))\ \Delta$$

$$\triangleright_\beta\ [x := \Delta]\ (x\ x)$$

$$\equiv_\alpha \Delta\ \Delta$$

$$\therefore\ \Omega \triangleright_\beta \Omega \triangleright_\beta \Omega \ \cdots\cdots$$

Recall example of

$K\ z\ \Omega$

- has a β-nf
- also has an ∞ reduction (non-terminating)

SO CONFLUENCE (CHURCH-ROSSER) ONLY SAYS THAT IF AN EXPRESSION HAS A β-nf, THEN IT IS UNIQUE.

- BUT THERE MAY BE BOTH
  - TERMINATING
  - NON-TERMINATING PATHS

- WHICH TO TAKE?

(*) IF THERE IS A TERMINATING PATH, THEN LEFTMOST <u>OUTERMOST</u> REDUCTION WILL TERMINATE.

(LAZY SAFER THAN EAGER)

<u>Lemma</u>

The class of β-nfs is the smallest class s.t

- all atoms are in β-nf
- if $e_1, \dots, e_m$ are in β-nf and a is an atom, then $a\, e_1 \dots e_m$ is in β-nf
- if e is in β-nf, then $\lambda x, e$ is in β-nf.

e is in this class iff e has no β-redex.

A NOTION OF EQUALITY

$e =_\beta e'$ if for some $e_0, e_1, \dots e_n$ we have:

$e \equiv e_0 \quad , \quad e_n \equiv e'$

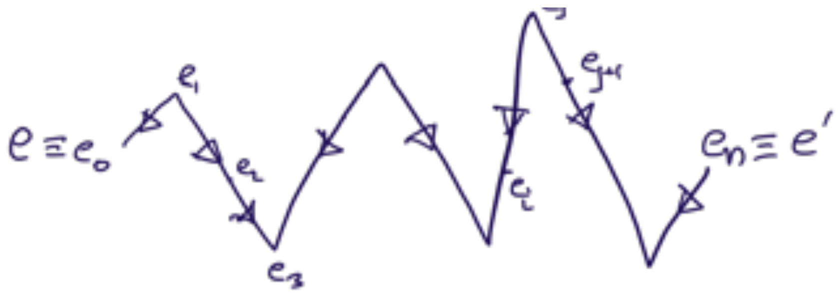
and for each $0 \le i < n$

- $e_i \rhd_\beta e_{i+1}$

or

- $e_{i+1} \rhd_\beta e_i$

e.



Substitution Lemma for $=_\beta$

(a) $e_2 =_\beta e_3 \longrightarrow$
$[x := e_2]e_1 =_\beta [x := e_3]e_1$

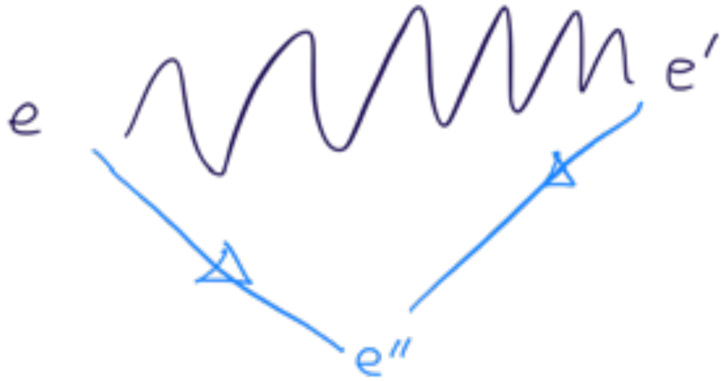
(b) $e_2 =_\beta e_3 \longrightarrow$
$[x := e_1]\, e_2 =_\beta [x := e_1]\, e_3$

CHURCH-ROSSER THEOREM OF $=_\beta$

If $e =_\beta e'$ then

there exists e'' s.t

$e \triangleright_\beta e''$ and $e' \triangleright_\beta e''$



Proof by induction on n

Base case $(n=0)$ $\quad e \equiv e'$

$e'' \equiv e$

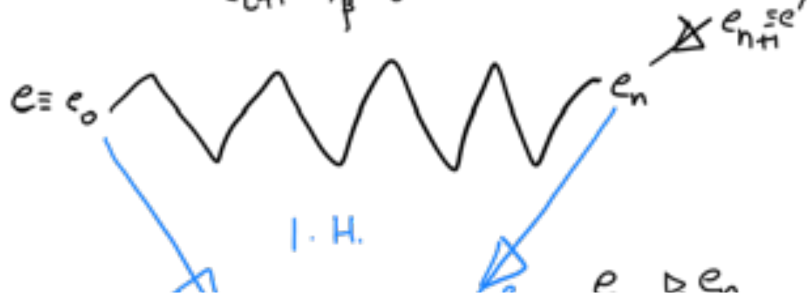
I.H: Assume if $e =_\beta e'$ by a seq of n $(e \equiv e_0, e_1 \ldots e_n \equiv e')$ then exists e'' st

$e \triangleright_\beta e''$ and $e' \triangleright_\beta e''$

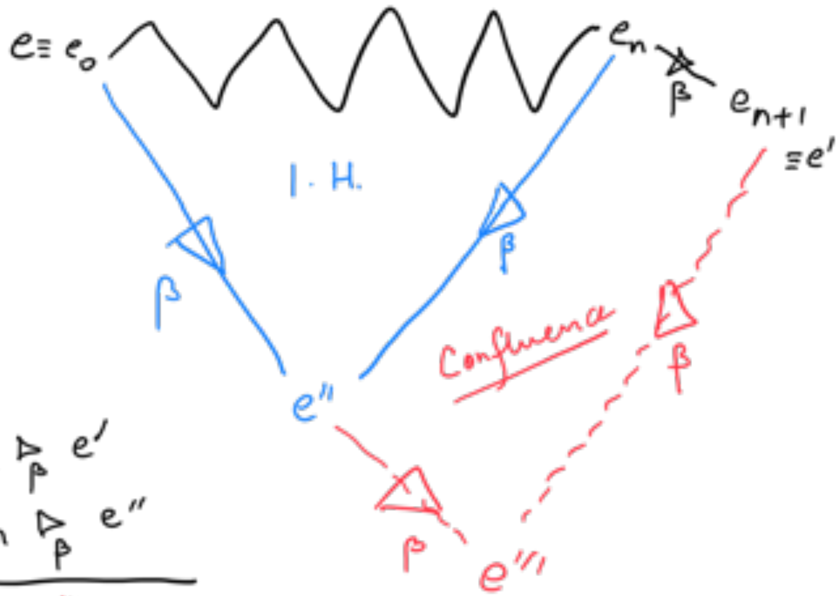
Induction Step. Suppose $e =_\beta e'$ by a seq. of $n+1$ $\quad e \equiv e_0, e_1, \ldots e_n, e_{n+1} \equiv e'$

where $\cdot\ e_i \triangleright_\beta e_{i+1}$

or $\cdot\ e_{i+1} \triangleright_\beta e_i$



$$\frac{\begin{array}{c} e_{n+1} \triangleright_\beta e' \\ e_n \triangleright_\beta e'' \end{array}}{\therefore\ e_{n+1} \triangleright_\beta e''}$$



$$\frac{\begin{array}{c} e_n \triangleright_\beta e' \\ e_n \triangleright_\beta e'' \end{array}}{\begin{array}{c} e'' \triangleright_\beta e''' \\ e' \triangleright_\beta e''' \end{array}}$$

$$\frac{\begin{array}{c} e \triangleright_\beta e'' \\ e'' \triangleright_\beta e''' \end{array}}{e \triangleright_\beta e'''}$$

note: If $e_1 \triangleright_\beta e_2$
then $e_1 =_\beta e_2$

STRATEGY FOR PROVING EQUALITY OF expression that have a β-nf.

- FIND THEIR β-nf
- CHECK IF THESE ARE EQUAL (mod $\equiv_\alpha$)

COROLLARIES

1. If $e_1 =_\beta e_2$ and e_2 is in β-nf then $e_1 \twoheadrightarrow_\beta e_2$

2. If $e_1 =_\beta e_2$, then either
   - e_1 and e_2 do not have any β-nf
   - e_1 and e_2 have the same β-nfs.

3. If $e_1 =_\beta e_2$ and e_1, e_2 in β-nf then $e_1 \equiv_\alpha e_2$

4. A term can be $=_\beta$ to at most one β-nf modulo $\equiv_\alpha$.

5. If $x\, e_1 \cdots e_m =_\beta y\, e'_1 \cdots e'_n$ then
   - $x \equiv y$
   - $m = n$
   - $e_i =_\beta e'_i$ for all $i \in \{1, \ldots, m\}$

Modelling the <u>booleans</u>

2 "values"

$$T \triangleq \lambda x.\lambda y.\, x$$

$$F \triangleq \lambda x.\lambda y.\, y$$

How do we show $T \neq F$

- Show that if $T =_\beta F$ then all expressions are equal.

Suppose $T =_\beta F$

then for all e_1, e_2

$$\begin{tikzcd}
T\, e_1\, e_2 \arrow[d, "\beta"] & =_\beta & F\, e_1\, e_2 \arrow[d, "\beta"] \\
e_1 \arrow[dr, dashed] & & e_2 \arrow[dl, dashed] \\
& \circ &
\end{tikzcd}$$

All terms are equal!
(Useless theory).

Using T, F — need an
"if _ then $\underline{e_1}$ else $\underline{e_2}$"

Define $D \triangleq \lambda t. \lambda a. \lambda b. (t\, a\, b)$

$$D\, T\, e_1\, e_2 =_\beta T\, e_1\, e_2$$
$$=_\beta e_1$$

$$D\, F\, e_1\, e_2 =_\beta F\, e_1\, e_2$$
$$=_\beta e_2$$

Can this be generalised to
- Sets of cardinality n
  (n - a finite integer)
- n-ary case analysis. ?

Modelling Pairs

Modelling Pairs

$\langle e_1 , e_2 \rangle$ for any e_1, e_2

Define $P \equiv \lambda a. \lambda b. \lambda t. (t\ a\ b)$

$$P\ e_1\ e_2 =_\beta \lambda t. (t\ e_1\ e_2)$$

Using pairs — need to have proj1

proj2

such that

$$\text{proj1}\ \langle e_1, e_2 \rangle =_\beta e_1 \qquad 1$$

$$\text{proj2}\ \langle e_1, e_2 \rangle =_\beta e_2 \qquad 2$$

Define

$$\text{proj}_1 \equiv \lambda p. (p\ \underline{(\lambda x. \lambda y. x)}) \qquad T$$

$$\text{proj}_2 \equiv \lambda p. (p\ \underline{(\lambda x. \lambda y. y)}) \qquad F$$

EXERCISE

Check (1) & (2) HOLD.

EXERCISE

How CAN ONE GENERALISE to k-tuples for any finite k?

CHURCH NUMERALS

$$\underline{0} \equiv \lambda f. \lambda x. x$$

$$\underline{1} \equiv \lambda f. \lambda x. (f\ x)$$

$$\underline{n} \equiv \lambda f. \lambda x. \underbrace{f\ (f \ldots (f}_{n\ f\text{'s.}}\ x) \ldots)$$

x — "constructor" for Zero

f - "constructor" for Succ

choose your atom for Succ

choose your atom for zero.

Note: $\underline{n}\ g\ y \triangleright_\beta$

$$\underbrace{g(\ldots\ldots(g}_{n \text{ times}}\; y))\ldots)$$

$$\underline{succ} \equiv \lambda n. \lambda g. \lambda y.\; n\, g\, (g\, y)$$

$$\begin{aligned}
\underline{succ}\ \underline{m} &\equiv (\lambda n. \lambda g. \lambda y.\; n\, g\, (g\, y))\ \underline{m} \\
&\rhd_\beta \lambda g. \lambda y. (\underline{m}\ g)\, (g\, y) \\
&\rhd_\beta \lambda g. \lambda y ((\lambda f. \lambda x. (f^m\, x))\, g\ (g\, y)) \\
&\rhd_\beta \lambda g. \lambda y.\ (\lambda x. (g^m\, x))\ (g\, y) \\
&\rhd_\beta \lambda g. \lambda y.\ g^m\, (g\, y) \\
&\equiv \lambda g. \lambda y. (g^{m+1}\, y)
\end{aligned}$$

How about

$$\underline{succ'} \equiv \lambda n. \lambda g. \lambda y. (g\, (n\, g\, y)) ?$$

$$\underline{add} \equiv \lambda m. \lambda n. \lambda h. \lambda z.\ (m\, h)(n\, h\, z)$$

Check that $\underline{add}\ \underline{m}\ \underline{n} \rhd_\beta \underline{m+n}$

Intuition $(\underline{n}\ h\ z) \rhd_\beta \underbrace{h^n\, z}_{w}$

$(\underline{m}\ h\ w) \rhd_\beta h^m\, w$

$$\therefore \quad \underline{m}\, h\, (\underline{n}\, h\, z) \rhd_\beta h^m (h^n z)$$
$$\equiv h^{m+n} z$$

$$\therefore \quad \underline{add}\ \underline{m}\ \underline{n} \rhd_\beta \lambda h. \lambda z.\ \underline{m}\ h\ (\underline{n}\ h\ z)$$
$$\rhd_\beta \lambda h. \lambda z. (h^{m+n} z)$$
$$\equiv \underline{m+n}$$

$$mult := \lambda m. \lambda n. \lambda h. \lambda z. (m\ (n\ h)\ z)$$
$$=_\eta \lambda m. \lambda n. \lambda h.\ m\ (n\ h)$$

Check $\underline{mult}\ \underline{m}\ \underline{n} \rhd_\beta \underline{m * n}$

$$\underline{mult}\ \underline{m}\ \underline{n} \rhd_\beta \underline{m}\ (\underline{n}\ h)$$
$$\underline{n}\ h \rhd_\beta \lambda x. (h^n\ x)$$
$$\underline{m}\ w \rhd_\beta \lambda x.\ (w^m\ x)$$

$$(\lambda x. (h^n x))\ x$$
$$\rhd_\beta h^n x$$

$$(\lambda x. h^n x)\ (h^n x)$$
$$\rhd_\beta h^{n+n} x$$

Repeat m times

$\therefore \ \underline{\text{mult}}\ \underline{m}\ \underline{n} \ \triangleright\ \underbrace{\lambda h.\lambda x.(h^{m*n}\ x)}_{\underline{m*n}}$

$$\text{exp} \equiv \lambda m.\lambda n.\ n\ m$$

Recall

$$\Delta \equiv \lambda x.(x\ x)$$

$$\Omega \equiv \Delta\Delta$$

$$\Omega \triangleright_{1\beta} \Omega \triangleright_{1\beta} \Omega \cdots\cdots$$

Consumes "energy" but reproduces itself.

Now consider

$$V_f \equiv \lambda x.f(x\ x)$$

$$Y_{\text{Curry}} \equiv \lambda f.(V_f\ V_f)$$

$$Y_{\text{Curry}}\ e \equiv (\lambda f.(V_f\ V_f))\ e$$

$$\triangleright_{1\beta} \quad \underbrace{(\lambda x.\, e\,(x\;x))}_{V_e}\underbrace{(\lambda x.\, e\,(x\,x))}_{V_e}$$

$$\triangleright_{1\beta} \quad e\,(V_e\;V_e)$$

$$\begin{array}{ccc}
Y_{curry}\; e & & e\,(Y_{curry}\; e) \\
\searrow_{1\beta} & & \swarrow_{1\beta} \\
\searrow_{1\beta} & e\,(V_e\;V_e) &
\end{array}$$

$$\therefore \quad Y_{curry}\; e \;=_\beta\; e\,(Y_{curry}\; e)$$

e — arbitrary!!!

$$Y_{Turing} \equiv Z\,Z$$

$$Z \equiv \lambda z.\lambda x.\; x\,(z\,z\,x)$$

$$\begin{aligned}
Y_{Turing}\; e &\equiv Z\,Z\,e \\
&\equiv (\lambda z.\lambda x.\; x\,(z\,z\,x))\,Z\,e \\
&\to_\beta (\lambda x.\; x\,(Z\,Z\,x))\,e \\
&\to_\beta e\,(\underbrace{Z\,Z\,e}) \\
&\equiv e\,(Y_{Turing}\; e)
\end{aligned}$$

$$Y_{Turing}\, e \rhd_\beta e\,(Y_{Turing}\, e)$$

for any e !!!

Theorem (Fixed point)

- There is a combinator Y such that

(a) $\cdot$ $Yx =_\beta x(Yx)$

(b) $Yx \rhd_\beta x(Yx)$

Note: Y is not unique

$Y_{curry}, Y_{Turing}, \ldots$

Thm: For any and $n \geq 0$

the equation

$$x\, y_1 \cdots y_n =_\beta e$$

can be solved for x

i.e., there is a term t s.t

$$t\, y_1 \cdots y_n =_\beta \boxed{[x := t]}$$

Proof: Let $t = Y(\lambda x\, \lambda y_1 \ldots \lambda y_n.\, e)$

— D Let $L = Y(\lambda x. x y_1 \cdots y_n \;\;)$

Corollary: Every finite set of simultaneous equations

$$\left.\begin{aligned} x_1\, y_1 \cdots y_n &= t_1 \\ &\vdots \\ x_k\, y_1 \cdots y_n &= t_k \end{aligned}\right\} \quad \begin{matrix} n \geq 0 \\ k \geq 1 \end{matrix}$$

is solvable for $x_1 \ldots x_k$.

Double fixed-point theorem

for any X, Y, there exist P, Q s.t

$$X P Q =_\beta P$$
$$Y P Q =_\beta Q$$

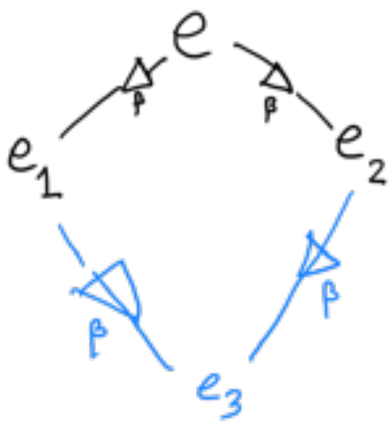
Proof: There exist X_1, X_2 s.t

$$X_i\, y_1\, y_2 =_\beta y_i\, (X_1 y_1 y_2)(X_2 y_1 y_2)$$

$$P \equiv X_1 X Y$$
$$Q \equiv X_2 X Y.$$

CHURCH-ROSSER THM OF $\triangleright_\beta$

IF



THEN

FOR SOME e_3

STRONG DIAMOND

WEAK DIAMOND

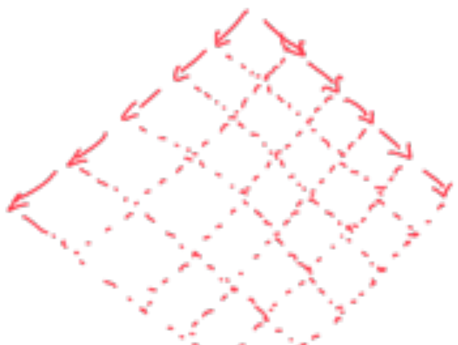
CONFLUENCE

STRONG DIAMOND

$\Rightarrow$ CONFLUENCE (1)

$\Rightarrow$ WEAK DIAMOND (2)

(1) TILING ARGUMENT.



(2) SPECIAL CASE

$\swarrow$ is an instance of $\overset{*}{\swarrow}$

$\searrow$ " " " " $\overset{*}{\searrow}$

CAN'T USE "TILING" WITH WEAK DIAMOND

NO INDUCTION SCHEME...

RESIDUALS

Let r, s be β-redexes in e

Suppose $e \overset{[r]}{\triangleright}_{1\beta} e'$

$$e \equiv \mathcal{C}[r] \;\triangleright_{1\beta}\; \mathcal{C}[t] \equiv e'$$

The residuals of s w.r.t contracting (reducing) r are those β-redexes in e' s.t

Case 1 $\quad r, s$ are non-overlapping in e

- neither r subterm of s
- nor s subterm of r

s remains unchanged in

$$e \rhd_{1\beta} e'$$

So this occurrence of s in e' the residual

Case 2 $\quad r \equiv s$ (same occurrence)

$\therefore$ contracting r is contracting s

So no residual of s in e'

Case 3 $\quad r$ subterm of s but $r \not\equiv s$.

$$s \equiv (\lambda x.e_1)\, e_2$$

$\therefore$ r subterm of e_1 — 3a

or r subterm of e_2 — 3b

(3a) $\quad e_1 \overset{[r]}{\rhd}_{1\beta} e_1'$

$$\therefore (\lambda x.e_1)\, e_2 \overset{[r]}{\rhd}_{1\beta} (\lambda x.e_1')\, e_2$$

residual of s

(3b) $e_2 \triangleright^{[r]}_{1\beta} e_2'$

$$\therefore (\lambda x.e_1)\, e_2 \triangleright^{[r]}_{1\beta} \underbrace{(\lambda x.e_1)\, e_2'}_{\text{residual of } s}$$

<u>Case 4</u> s subterm of r $s \not\equiv r$

$$r \equiv (\lambda x.e_1)\, e_2 \triangleright_{1\beta} [x := e_2]\, e_1$$

- s subterm of e_1 (4a)
- s subterm of e_2 (4b)

(4a) s becomes one of

$$[x := e_2]\, s$$

or

$$[x := e_2]\,[y_1 := z_1] \ldots [y_n := z_n]\, s$$

or

$$s$$

possibilities for residuals

depending on # of times case (f) employed in $[x := e_2]\, e_1$

(4b) <u>Each</u> copy of s in e_2 in

$[x := e_2]\, e_1$

is a residual of s

Minimal Complete Development (mcd)

Let $r_1 \ldots r_n$ $(n \geq 0)$ be some redexes in e

r_i is called <u>minimal</u> if

no <u>other</u> r_j subterm of r_i

(forall r_j, r_j subterm of r_i $\rightarrow$ $r_j \equiv r_i$)

<u>Define</u>

$$e \rhd_{mcd} e'$$

as:

Pick a minimal r_i in $r_1 \ldots r_n$

(w.l.o.g r_1 in some ordering)

- get residuals $r_2' \ldots r_n'$ of $r_2 \ldots r_n$

...ill have

Note: Each of $r_2 \ldots r_n$ will have

only 1 residual each

$\because r_1$ is minimal.

Now repeat with any minimal r_j'

⋮

Repeat until NO residuals left.

Make as many α-conversions

(Process is not unique)

- In any non-empty set of redexes, always exists at least 1 minimal.
- No redex, $\triangleright_{mcd}$ is just $\equiv_\alpha$ moves
- $\triangleright_{1\beta}$ is a special case of $\triangleright_{mcd}$ on a singleton set of redexes.
- Non mcd's exist

$$(\lambda x.(x\ y))(\lambda z.z) \ \underset{①}{\triangleright_{1\beta}} \ (\lambda z.z)\ y$$

$$\underset{②}{\triangleright_{1\beta}} \ y$$

- $\triangleright_{mcd}$ is not transitive

no mcd to directly do ①-②

$$\ast \quad \frac{e_1 \rhd_{mcd} e_1' \qquad e_2 \rhd_{mcd} e_2'}{e_1 e_2 \rhd_{mcd} e_1' e_2'}$$ However

Lemma

If $e_1 \rhd_{mcd} e_2$

and $e_1 \equiv_\alpha e_1'$

then $e_1' \rhd_{mcd} e_2$

Lemma

If $e_1 \rhd_{mcd} e_1'$

and $e_2 \rhd_{mcd} e_2'$

then $[x := e_2]e_1 \rhd_{mcd} [x := e_2']e_1'$

Assume that bound variables in e_1 do not appear in x or in e_2

Induction on e_1

Cases

1. $e_1 \equiv x$

2. $x \notin (fv\ e_1)$

3. $e_1 \equiv \lambda y. e_{11}$

4. $e_1 \equiv e_{11}\ e_{12}$

5. $e_1 \equiv (\lambda y. e_{11})\ e_{12}$

- $\rhd_{1\beta} \subseteq \rhd_{mcd}$
- $\rhd_{1\beta}^{\ *} = \rhd_{mcd}^{\ *}$
- $\rhd_{mcd}$ forms strong diamonds.

$$
\begin{tikzcd}
 & e \arrow[dl, "\rhd_{mcd}"'] \arrow[dr, "\rhd_{mcd}"] & \\
e' \arrow[dr, "\rhd_{mcd}"'] & & e'' \arrow[dl, "\rhd_{mcd}"] \\
 & e''' &
\end{tikzcd}
$$

Proof

Cases

Cases

- $e \equiv x$
- $e \equiv \lambda x. e_1$
- $e \equiv e_1\ e_2$
- $e \equiv (\lambda x. e_1)\ e_2$

Main Theorem's proof