Finite Developments in the λ-calculus

Part I

jean-jacques.levy@inria.fr

ISR 2021

Madrid

July 6, 2021

http://jeanjacqueslevy.net/talks/21isr

λ-calculus

function	λ-term	β-reduction
$I\,x = x$	$I = \lambda x.x$	$I\,a \longrightarrow a$
$K\,x\,y = x$	$K = \lambda x.\lambda y.x$	$K\,a\,b \longrightarrow (\lambda y.a)\,b \longrightarrow a$
$\Delta\,x = x\,x$	$\Delta = \lambda x.x\,x$	$\Delta\,a \longrightarrow a\,a$
$\Omega = \Delta\,\Delta$		$\Omega \longrightarrow \Omega$

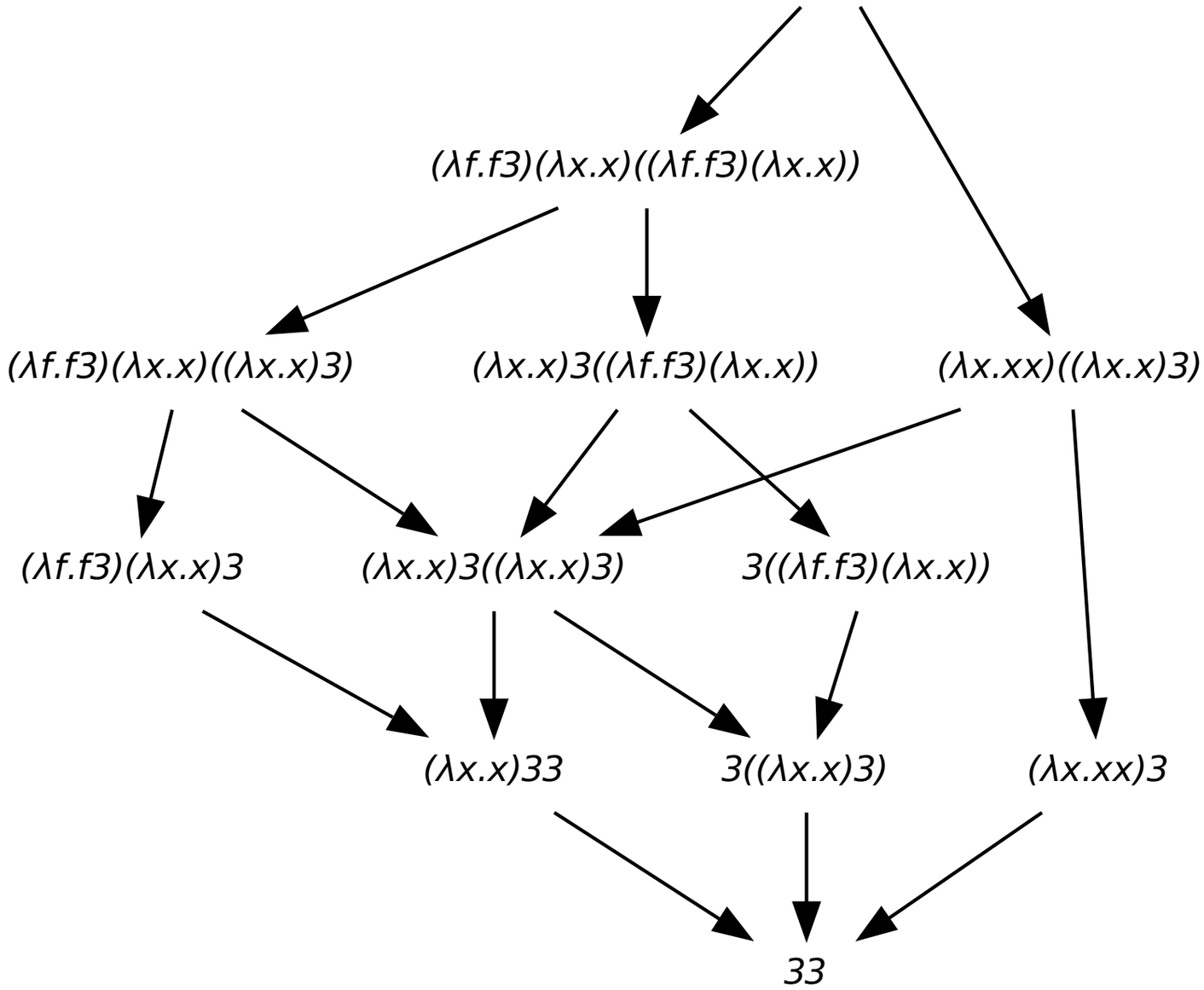
Exercise 1

$$\Delta(\lambda x.x\,x\,x) \longrightarrow \cdots$$

$$Y_f = (\lambda x.f\,(x\,x))(\lambda x.f\,(x\,x)) \longrightarrow \cdots$$

λ-calculus

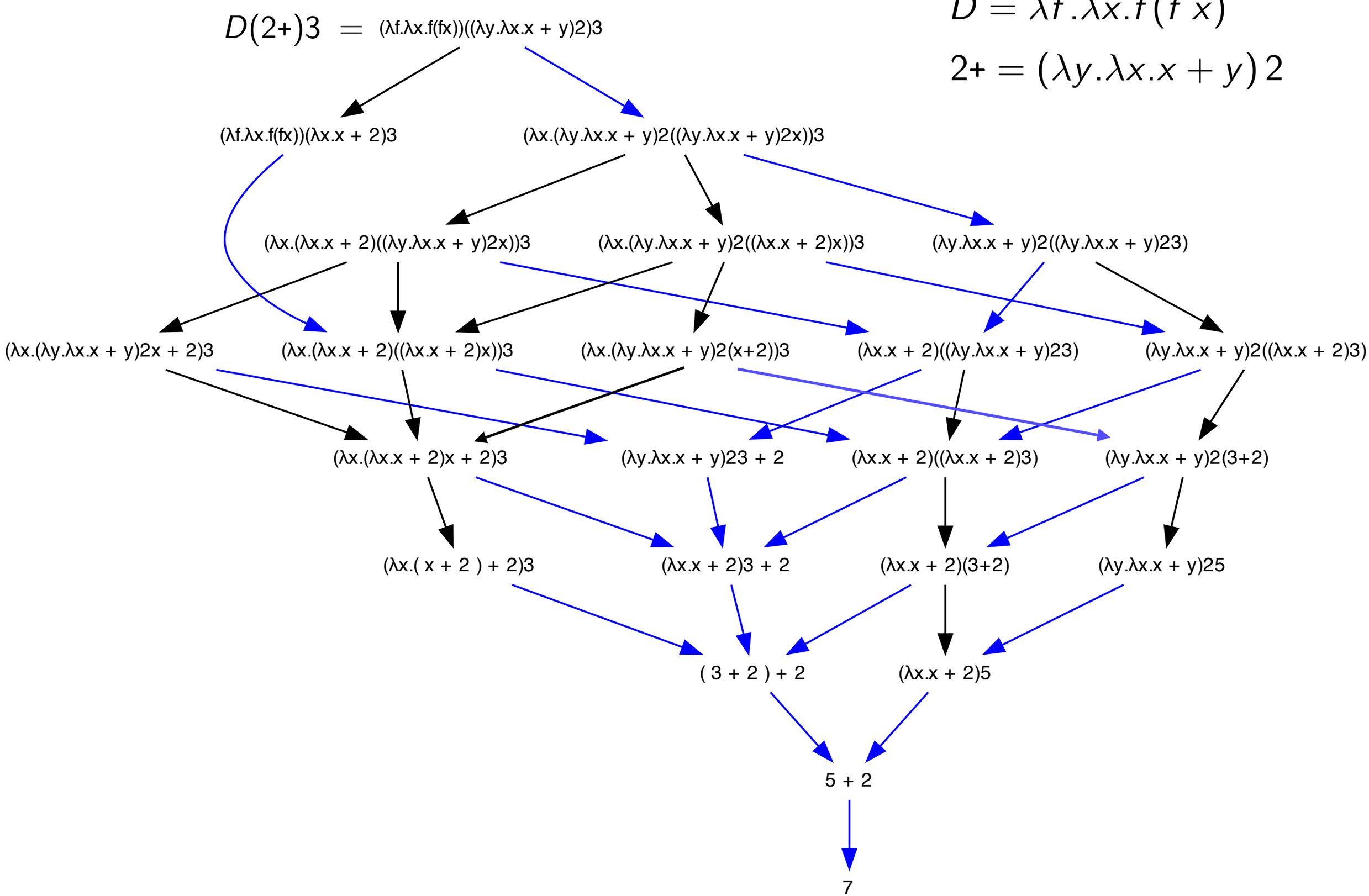
$$\Delta\left((\lambda f.3)I\right) = (\lambda x.xx)((\lambda f.f3)(\lambda x.x))$$



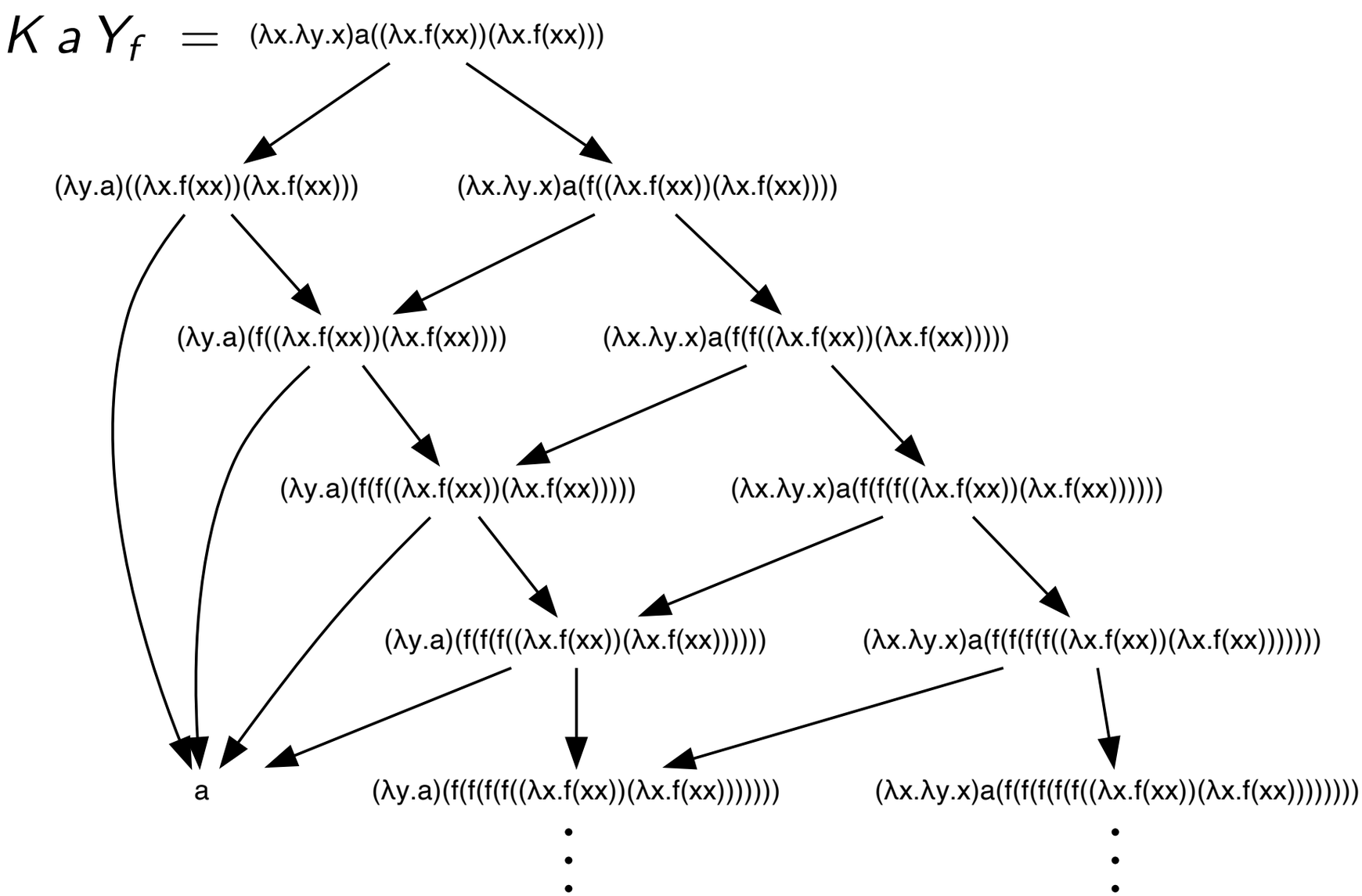
λ-calculus

$$D = \lambda f.\lambda x.f(f\,x)$$

$$2{+} = (\lambda y.\lambda x.x + y)\,2$$



λ-calculus


$K\ a\ Y_f\ =$ (λx.λy.x)a((λx.f(xx))(λx.f(xx)))
(λy.a)((λx.f(xx))(λx.f(xx)))
(λx.λy.x)a(f((λx.f(xx))(λx.f(xx))))
(λy.a)(f((λx.f(xx))(λx.f(xx))))
(λx.λy.x)a(f(f((λx.f(xx))(λx.f(xx)))))
(λy.a)(f(f((λx.f(xx))(λx.f(xx)))))
(λx.λy.x)a(f(f(f((λx.f(xx))(λx.f(xx))))))
(λy.a)(f(f(f((λx.f(xx))(λx.f(xx))))))
(λx.λy.x)a(f(f(f(f((λx.f(xx))(λx.f(xx)))))))
a
(λy.a)(f(f(f(f((λx.f(xx))(λx.f(xx)))))))
(λx.λy.x)a(f(f(f(f(f((λx.f(xx))(λx.f(xx))))))))

λ-calculus

$$K\ a\,\Omega \;=\; (\lambda x.\lambda y.x)a((\lambda x.xx)(\lambda x.xx))$$

$$\downarrow$$

$$(\lambda y.a)((\lambda x.xx)(\lambda x.xx))$$

$$\downarrow$$

$$a$$

Empirical facts

- **deterministic** result when it exists — **Church-Rosser**
- multiple reduction strategies — **CBN - CBV - ..**
- **terminating** strategy ? — **normalisation**
- **efficient** reduction strategy ? — **optimal reduction**
- **worst** reduction strategy ? — **perpetual reduction**
- when all reductions are finite ? — **strong normalisation**
- when finite, the reduction graph has a **lattice** structure ? — **NO!**

Redexes

- a redex is any reductible expression: $(\lambda x.M)N$
- the β-conversion rule is:

$$(\lambda x.M)N \longrightarrow M\{x := N\}$$

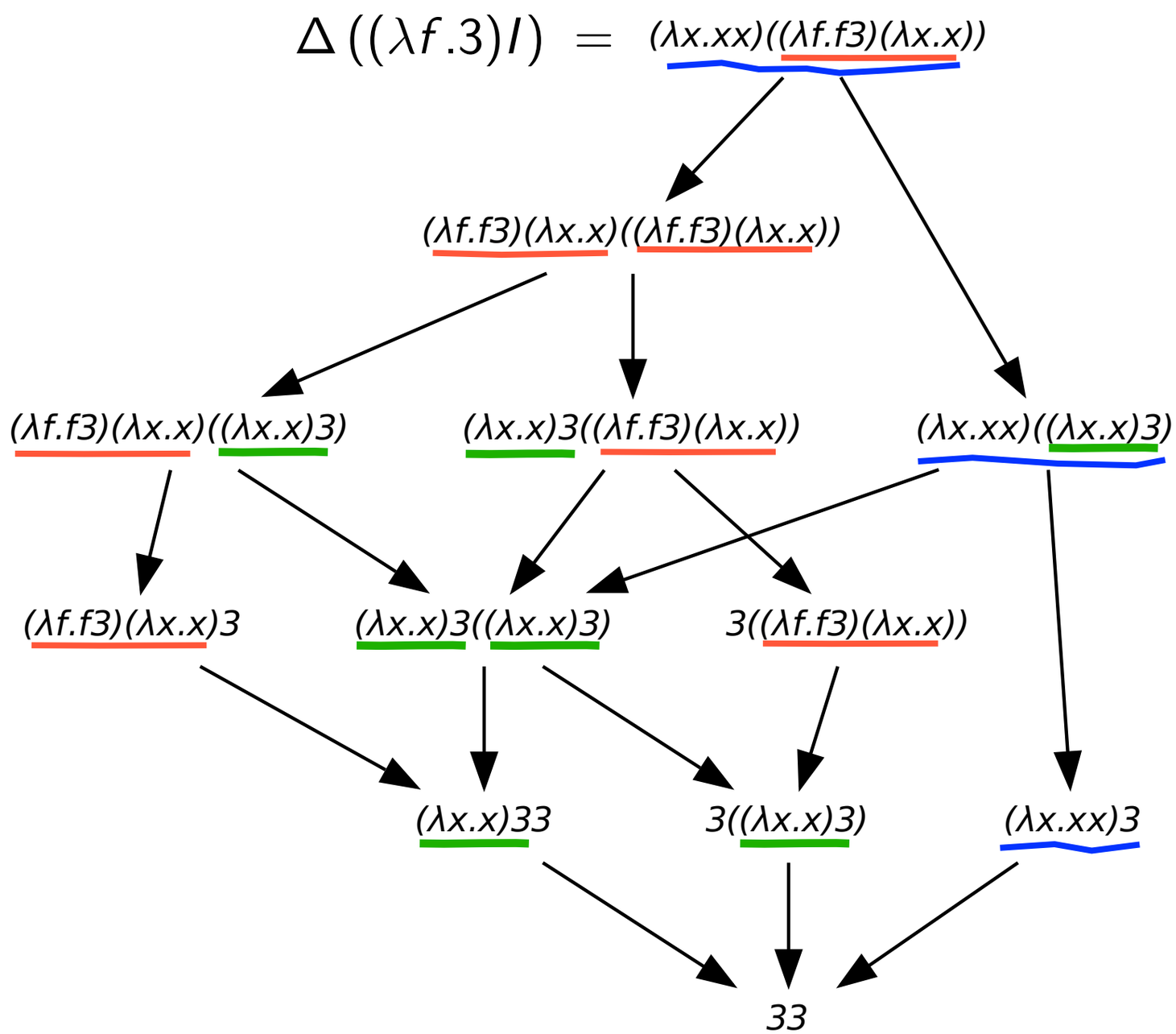
- a reduction step contracts a given redex $R = (\lambda x.A)B$

and is written: $M \xrightarrow{R} N$

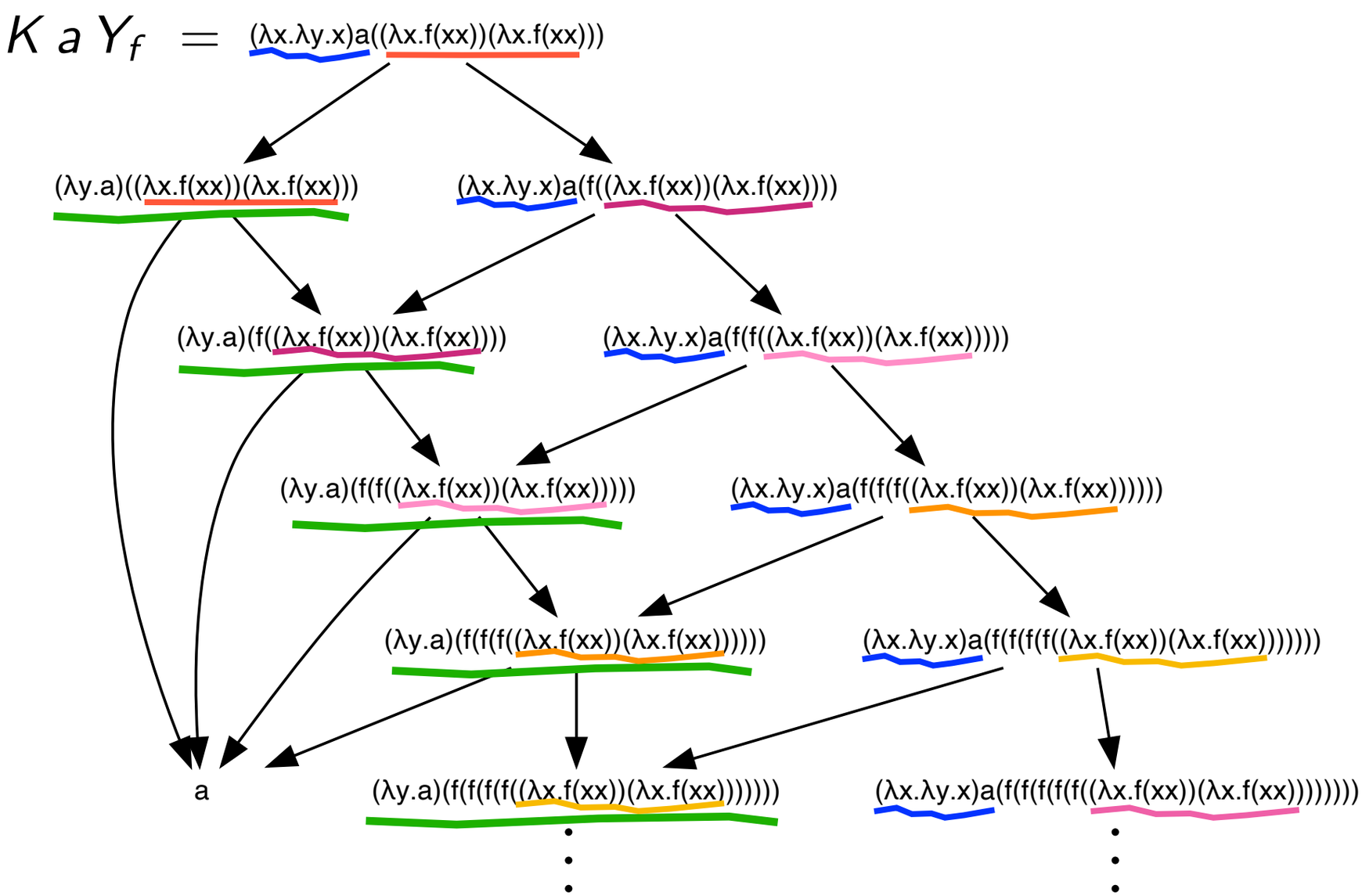
- a reduction step contracts a singleton set of redexes $M \xrightarrow{\{R\}} N$

- a more precise notation would be with occurences of subterms. We avoid it here (but it is sometimes mandatory to avoid ambiguity)
- we replaced occurences by giving names (labels) to redexes.

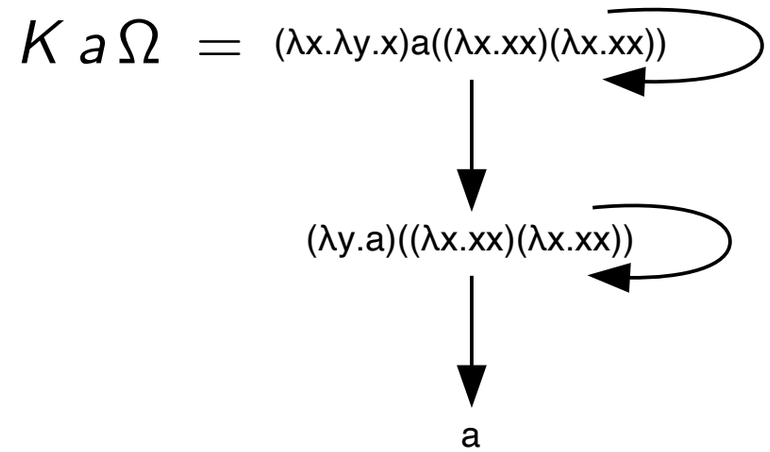
Tracing redexes

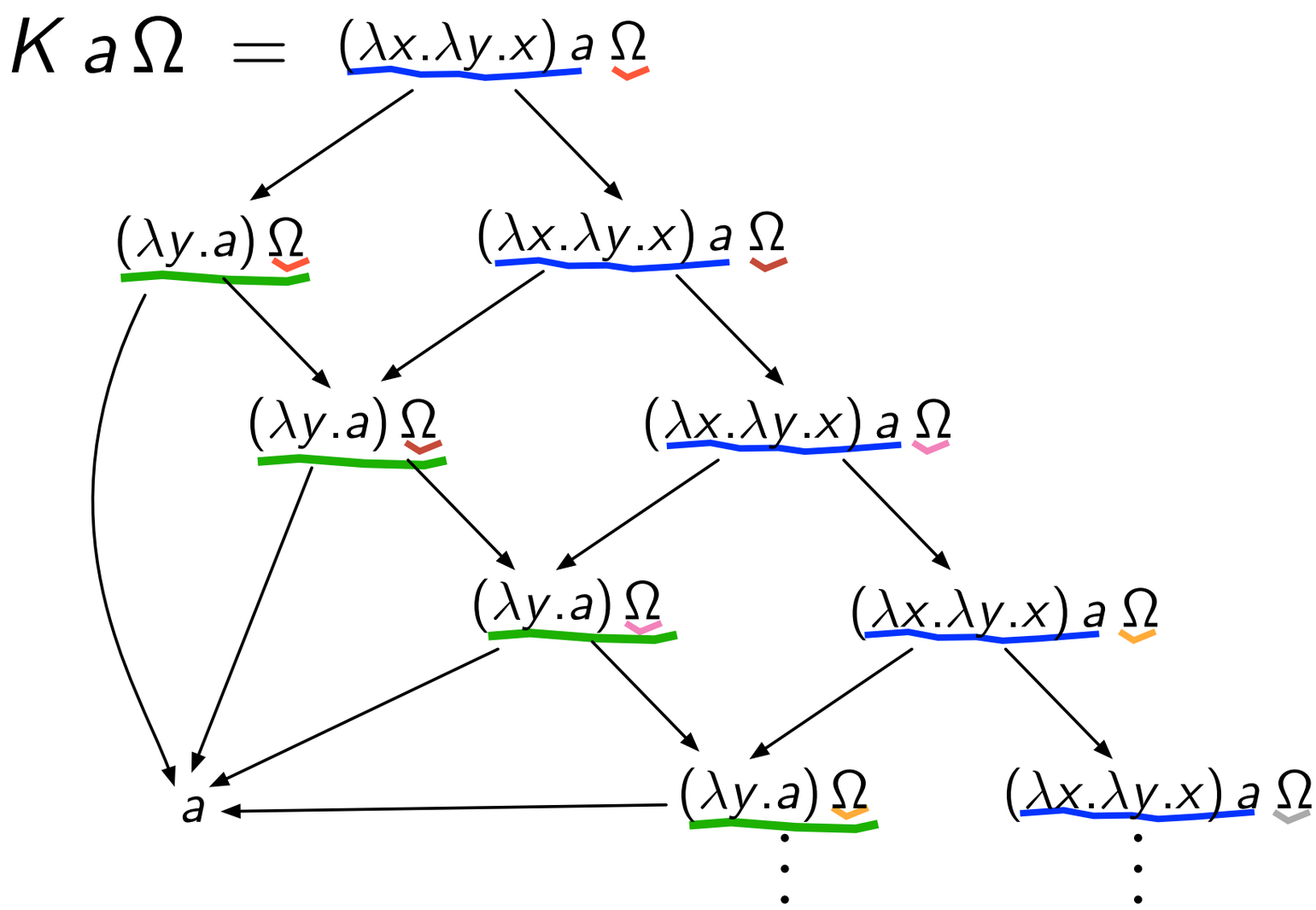


Tracing redexes

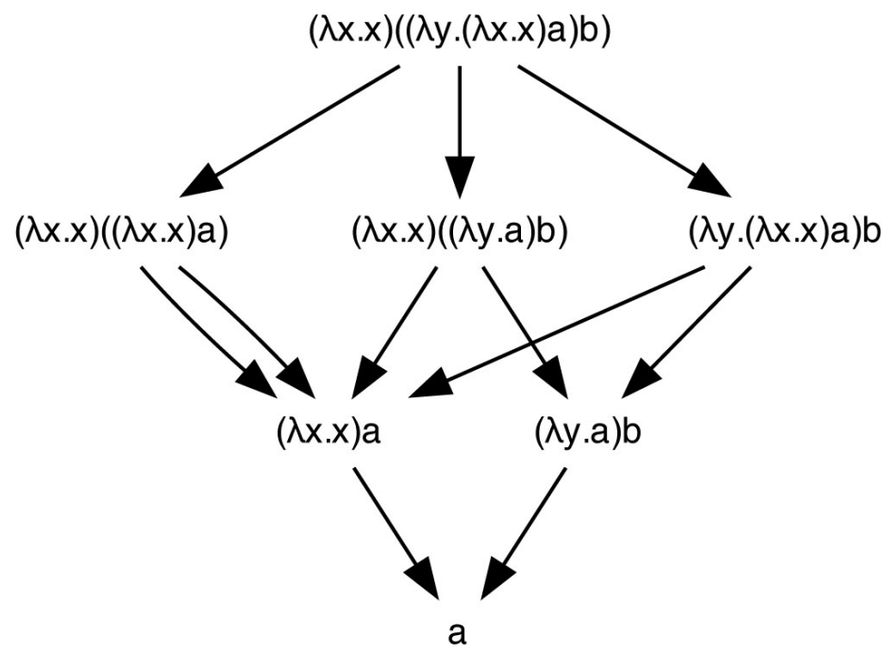


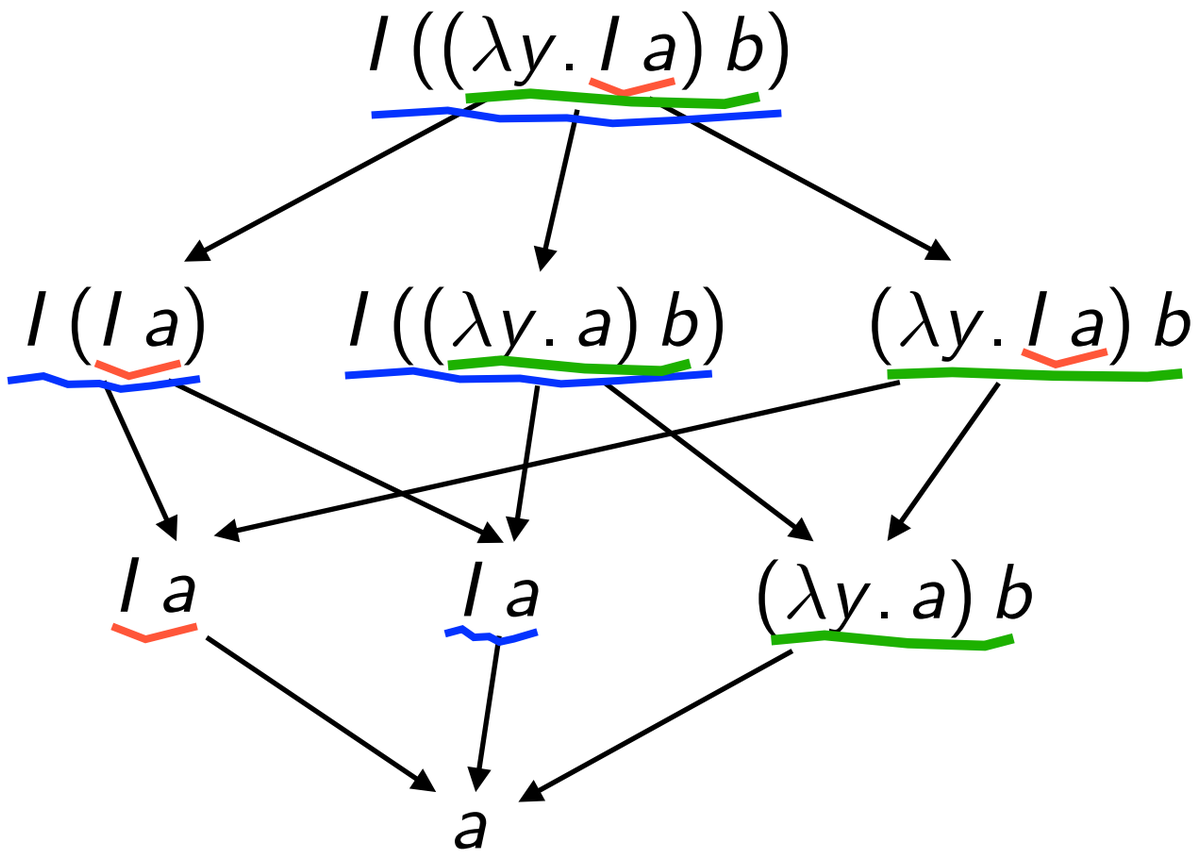
Tracing redexes





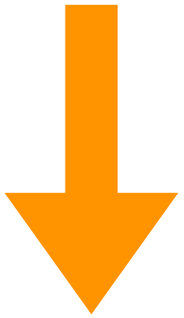
Tracing redexes





Empirical facts

- initial redexes in the initial term
- and **newly** created redexes along reductions
- **infinite** reduction iff length of creation is unbounded ?
- **deterministic** result when finite families of redexes are contracted ?



Finite Developments Theorem

Curry'50 JJL'78

Parallel reduction steps

Parallel reductions (1/3)

- permutation of reductions has to cope with copies of redexes

$$\begin{tikzcd}
(\lambda x.xx)(Ia) \arrow[r] \arrow[dr] & Ia\,(Ia) \arrow[dr, "\star"] & \\
& (\lambda x.xx)a \arrow[r] & a\,a
\end{tikzcd}$$

- in fact, a parallel reduction $Ia(Ia) \not\rightarrow aa$

- in λ-calculus, need to define parallel reductions for nested sets

Fact In the λ-calculus, disjoint redexes may become nested $(\lambda x.Ix)(\Delta y) \rightarrow I(\Delta y)$

Parallel reductions (2/3)

- the axiomatic way (à la Martin-Löf)

[Var Axiom] $x \mathrel{\not\!\!\rightarrow} x$

[Const Axiom] $c \mathrel{\not\!\!\rightarrow} c$

[App Rule] $\dfrac{M \mathrel{\not\!\!\rightarrow} M' \quad N \mathrel{\not\!\!\rightarrow} N'}{MN \mathrel{\not\!\!\rightarrow} M'N'}$

[Abs Rule] $\dfrac{M \mathrel{\not\!\!\rightarrow} M'}{\lambda x.M \mathrel{\not\!\!\rightarrow} \lambda x.M'}$

[//Beta Rule] $\dfrac{M \mathrel{\not\!\!\rightarrow} M' \quad N \mathrel{\not\!\!\rightarrow} N'}{(\lambda x.M)N \mathrel{\not\!\!\rightarrow} M'\{x := N'\}}$

inside-out (possibly void) parallel reductions

- examples:

$$(\lambda x.Ix)(Iy) \mathrel{\not\!\!\rightarrow} (\lambda x.x)y$$

$$(\lambda x.(\lambda y.yy)x)(Ia) \mathrel{\not\!\!\rightarrow} Ia(Ia)$$

$$(\lambda x.(\lambda y.yy)x)(Ia) \mathrel{\not\!\!\rightarrow} (\lambda y.yy)a$$

Parallel reductions (3/3)

- **Parallel moves lemma** [Curry 50]
  If $M \mathrel{\overset{\parallel}{\to}} N$ and $M \mathrel{\overset{\parallel}{\to}} P$, then $N \mathrel{\overset{\parallel}{\to}} Q$ and $P \mathrel{\overset{\parallel}{\to}} Q$ for some Q.

$$\begin{tikzcd}
 & M \arrow[dl, "\parallel"'] \arrow[dr, "\parallel"] & \\
N \arrow[dr, dashed, "\parallel"'] & & P \arrow[dl, dashed, "\parallel"] \\
 & Q &
\end{tikzcd}$$

lemma 1-1-1-1
(strong confluency)

Enough to prove Church Rosser theorem since $\to \subset \mathrel{\overset{\parallel}{\to}} \subset \mathrel{\overset{*}{\to}}$

[Tait--Martin Löf 60?]

Reduction of a set of redexes (1/4)

- Goal: parallel reduction of a given set of redexes

$$M, N ::= x \mid \lambda x.M \mid MN \mid (\lambda x.M)^a N$$

$$a, b, c, \ldots ::= \text{redex labels}$$

(labeled β-rule)

$$(\lambda x.M)^a N \longrightarrow M\{x := N\}$$

- Substitution as before with add-on:

$$((\lambda y.P)^a Q)\{x := N\} = (\lambda y.P\{x := N\})^a Q\{x := N\}$$

Reduction of a set of redexes (2/4)

- let $\mathcal{F}$ be a set of redex labels

[Var Axiom] $x \xrightarrow{\mathcal{F}} x$

[Const Axiom] $c \xrightarrow{\mathcal{F}} c$

[App Rule] $\dfrac{M \xrightarrow{\mathcal{F}} M' \quad N \xrightarrow{\mathcal{F}} N'}{MN \xrightarrow{\mathcal{F}} M'N'}$

[Abs Rule] $\dfrac{M \xrightarrow{\mathcal{F}} M'}{\lambda x.M \xrightarrow{\mathcal{F}} \lambda x.M'}$

[//Beta Rule] $\dfrac{M \xrightarrow{\mathcal{F}} M' \quad N \xrightarrow{\mathcal{F}} N' \quad a \in \mathcal{F}}{(\lambda x.M)^a N \xrightarrow{\mathcal{F}} M'\{x := N'\}}$

[Redex'] $\dfrac{M \xrightarrow{\mathcal{F}} M' \quad N \xrightarrow{\mathcal{F}} N' \quad a \notin \mathcal{F}}{(\lambda x.M)^a N \xrightarrow{\mathcal{F}} (\lambda x.M')^a N'}$

inside-out parallel reductions of redexes labeled in $\mathcal{F}$

- let $\mathcal{F}$, $\mathcal{G}$ be set of redexes in M and let $M \xrightarrow{\mathcal{F}} N$, then the set $\mathcal{G}/\mathcal{F}$ of residuals of $\mathcal{G}$ by $\mathcal{F}$ is the set of $\mathcal{G}$ redexes in N.

Reduction of a set of redexes (3/4)

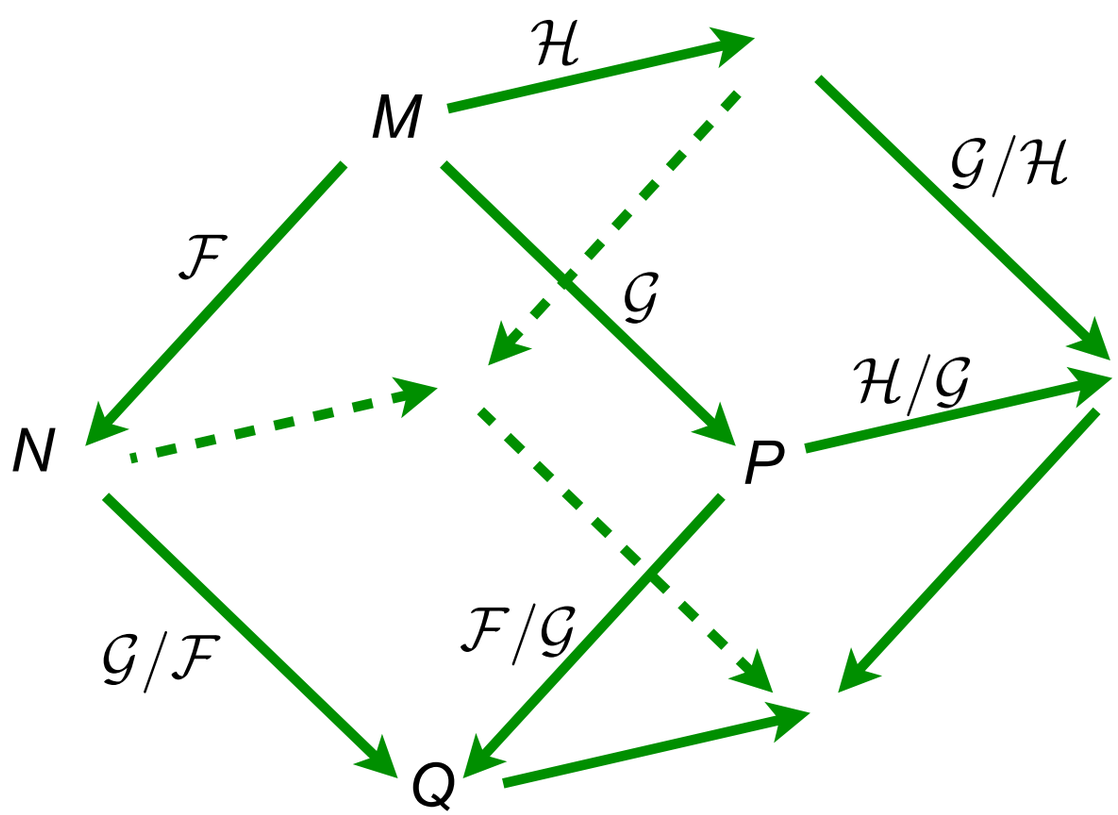
- **Parallel moves lemma+** [Curry 50]

  If $M \xrightarrow{\mathcal{F}} N$ and $M \xrightarrow{\mathcal{G}} P$, then $N \xrightarrow{\mathcal{G}/\mathcal{F}} Q$ and $P \xrightarrow{\mathcal{F}/\mathcal{G}} Q$ for some Q.

$$\begin{tikzcd}
 & M \arrow[dl, "\mathcal{F}"'] \arrow[dr, "\mathcal{G}"] & \\
N \arrow[dr, dashed, "\mathcal{G}/\mathcal{F}"'] & & P \arrow[dl, dashed, "\mathcal{F}/\mathcal{G}"] \\
 & Q &
\end{tikzcd}$$

Reduction of a set of redexes (4/4)

- **Parallel moves lemma++** `[Curry 50] The Cube Lemma`



$$(\mathcal{H}/\mathcal{F})/(\mathcal{G}/\mathcal{F}) = (\mathcal{H}/\mathcal{G})/(\mathcal{F}/\mathcal{G})$$

Residuals of redexes

Redexes

- a redex is any reductible expression: $(\lambda x.M)N$

- a reduction step contracts a given redex $R = (\lambda x.A)B$

and is written: $M \xrightarrow{R} N$

- a reduction step contracts a singleton set of redexes $M \xrightarrow{\{R\}} N$

- a more precise notation would be with occurences of subterms. We avoid it here (but it is sometimes mandatory to avoid ambiguity)

- we replaced occurences by giving names (labels) to redexes.

Residuals of redexes (1/4)

- **residuals** of redexes were defined by considering **labels**
- residuals are redexes with **same labels**
- a closer look w.r.t. their relative positions give following cases:

  let $R = (\lambda x.A)B$, let $M \xrightarrow{R} N$ and $S = (\lambda y.C)D$ be an other redex in M. Then:

Residuals of redexes (2/4)

Case 1:

$$M = \cdots\cdots R \cdots\cdots S \cdots\cdots \xrightarrow{R} \cdots\cdots R' \cdots\cdots S \cdots\cdots = N$$

or

$$M = \cdots\cdots S \cdots\cdots R \cdots\cdots \xrightarrow{R} \cdots\cdots S \cdots\cdots R' \cdots\cdots = N$$

Case 2:

$$M = \cdots\cdots R \cdots\cdots \xrightarrow{R} \cdots\cdots R' \cdots\cdots = N \quad (R \text{ and } S \text{ coincide})$$

Case 3:

$$M = \cdots (\lambda y. \cdots R \cdots) D \cdots \xrightarrow{R} \cdots (\lambda y. \cdots R' \cdots) D \cdots = N$$

Case 4:

$$M = \cdots (\lambda y. C)(\cdots R \cdots) \cdots \xrightarrow{R} \cdots (\lambda y. C)(\cdots R' \cdots) \cdots = N$$

Residuals of redexes (3/4)

Case 3:

$$M = \cdots(\lambda x.\cdots S\cdots)B\cdots \xrightarrow{R} \cdots\cdots S\{x := B\}\cdots\cdots = N$$

Case 4:

$$M = \cdots(\lambda x.\cdots x\cdots x\cdots)(\cdots S\cdots)\cdots$$

$$\xrightarrow{R} \cdots\cdots(\cdots S\cdots)\cdots(\cdots S\cdots)\cdots\cdots = N$$

Residuals of redexes (4/4)

Examples: $\Delta = \lambda x.xx, I = \lambda x.x$

$$\Delta(I\,x) \longrightarrow I\,x(I\,x)$$

$$I\,x\,(\Delta(I\,x)) \longrightarrow I\,x\,(I\,x(I\,x))$$

$$I\,(\Delta(I\,x)) \longrightarrow I\,(I\,x(I\,x))$$

$$\Delta(I\,x) \longrightarrow I\,x(I\,x)$$

$$I\,x\,(\Delta(I\,x)) \longrightarrow I\,x\,(I\,x(I\,x))$$

$$\Delta\Delta \longrightarrow \Delta\Delta$$

Residuals of reductions

Parallel reductions

- Consider reductions where each step is parallel

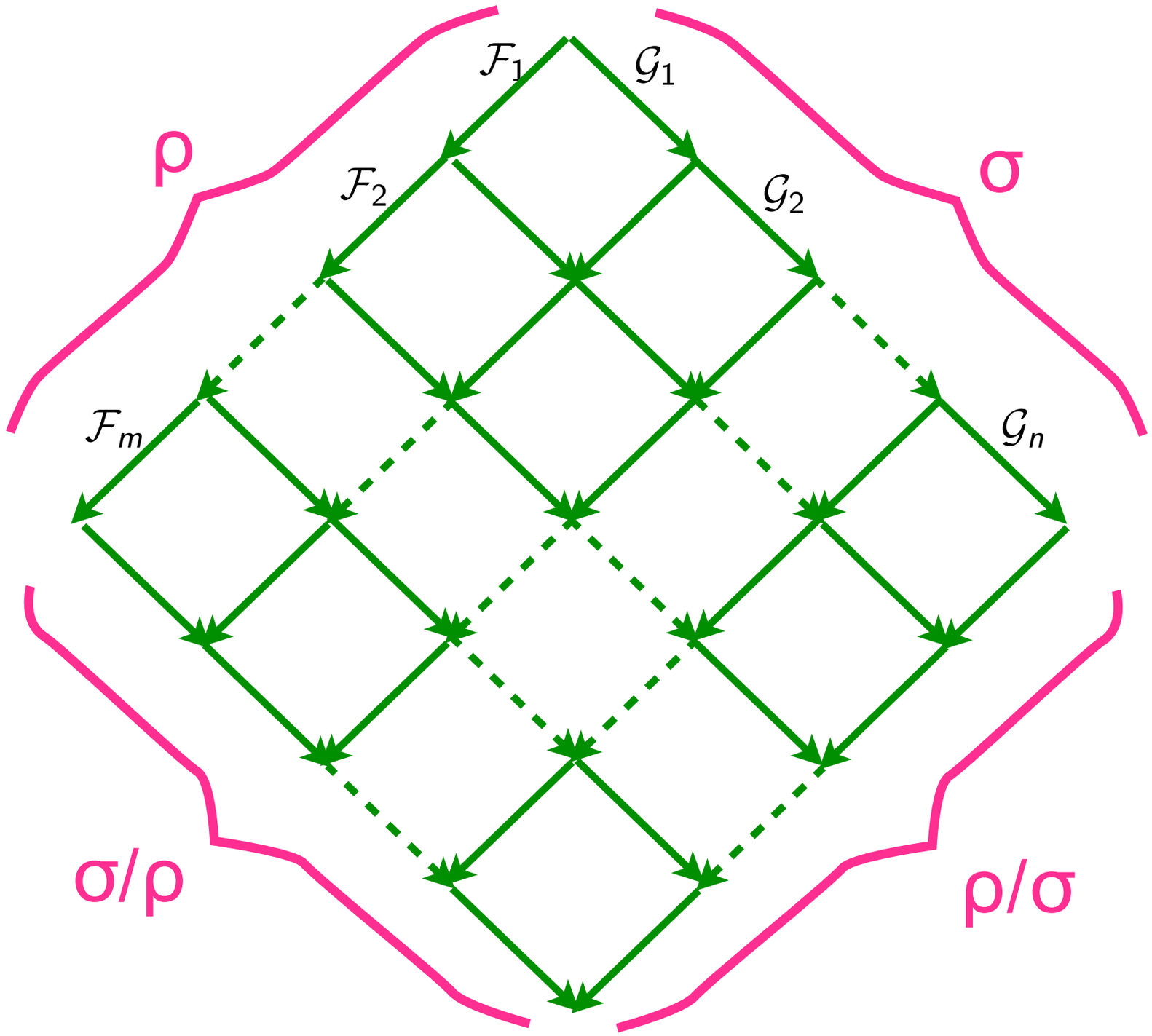
$$\rho : M = M_0 \xrightarrow{\mathcal{F}_1} M_1 \xrightarrow{\mathcal{F}_2} M_2 \cdots \xrightarrow{\mathcal{F}_n} M_n = N$$

- We also write

$\rho = 0$ when $n = 0$

$\rho = \mathcal{F}_1 \, \mathcal{F}_2 \cdots \mathcal{F}_n$ when M clear from context

Residuals of reductions (1/4)



Residuals of reductions (2/4)

- **Definition** [JJL 76]

$$\rho/0 = \rho$$

$$\rho/(\sigma\,\tau) = (\rho/\sigma)/\tau$$

$$(\rho\,\sigma)/\tau = (\rho/\tau)\,(\sigma/(\tau/\rho))$$

$\mathcal{F}/\mathcal{G}$ already defined

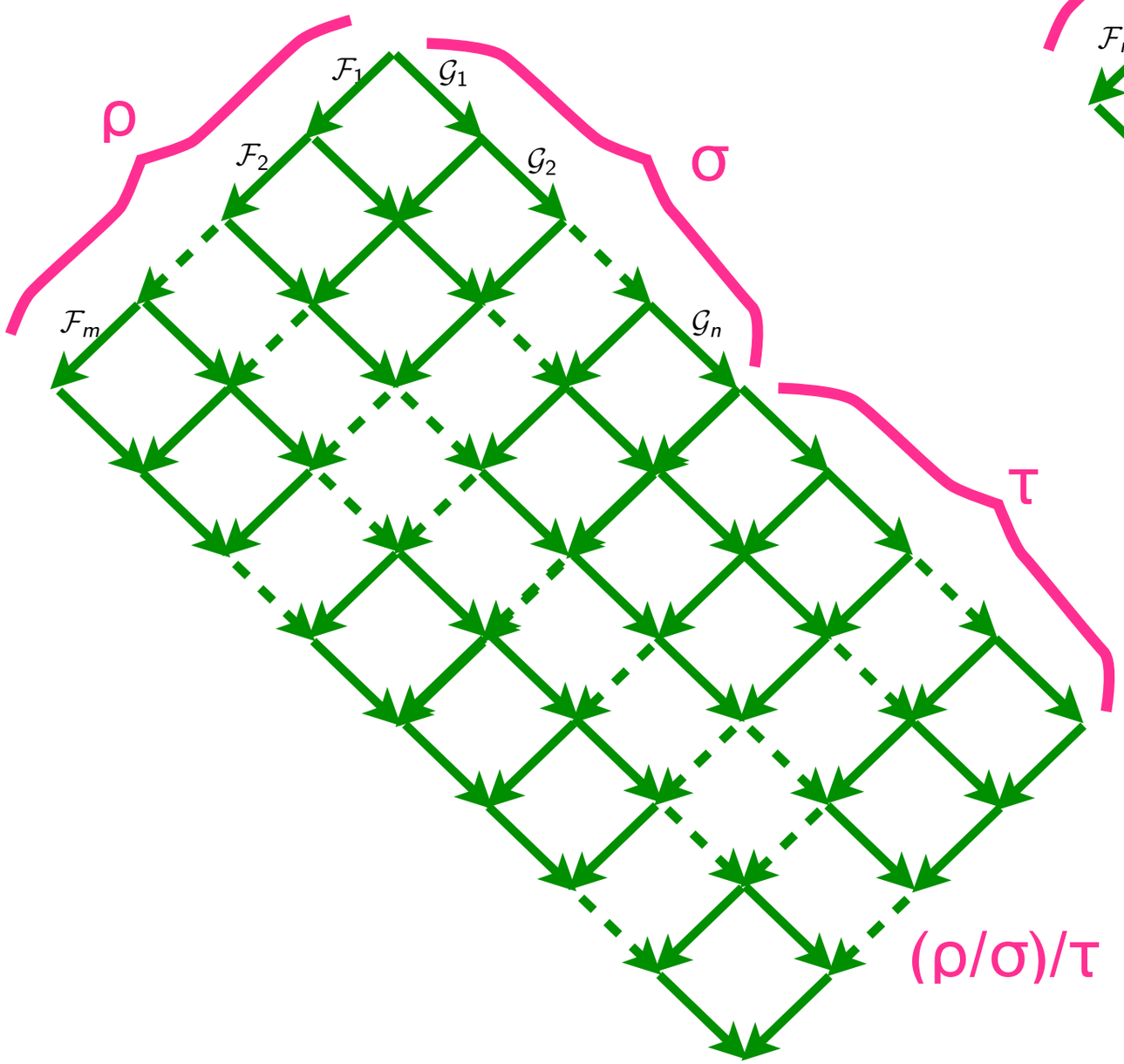
- **Notation**

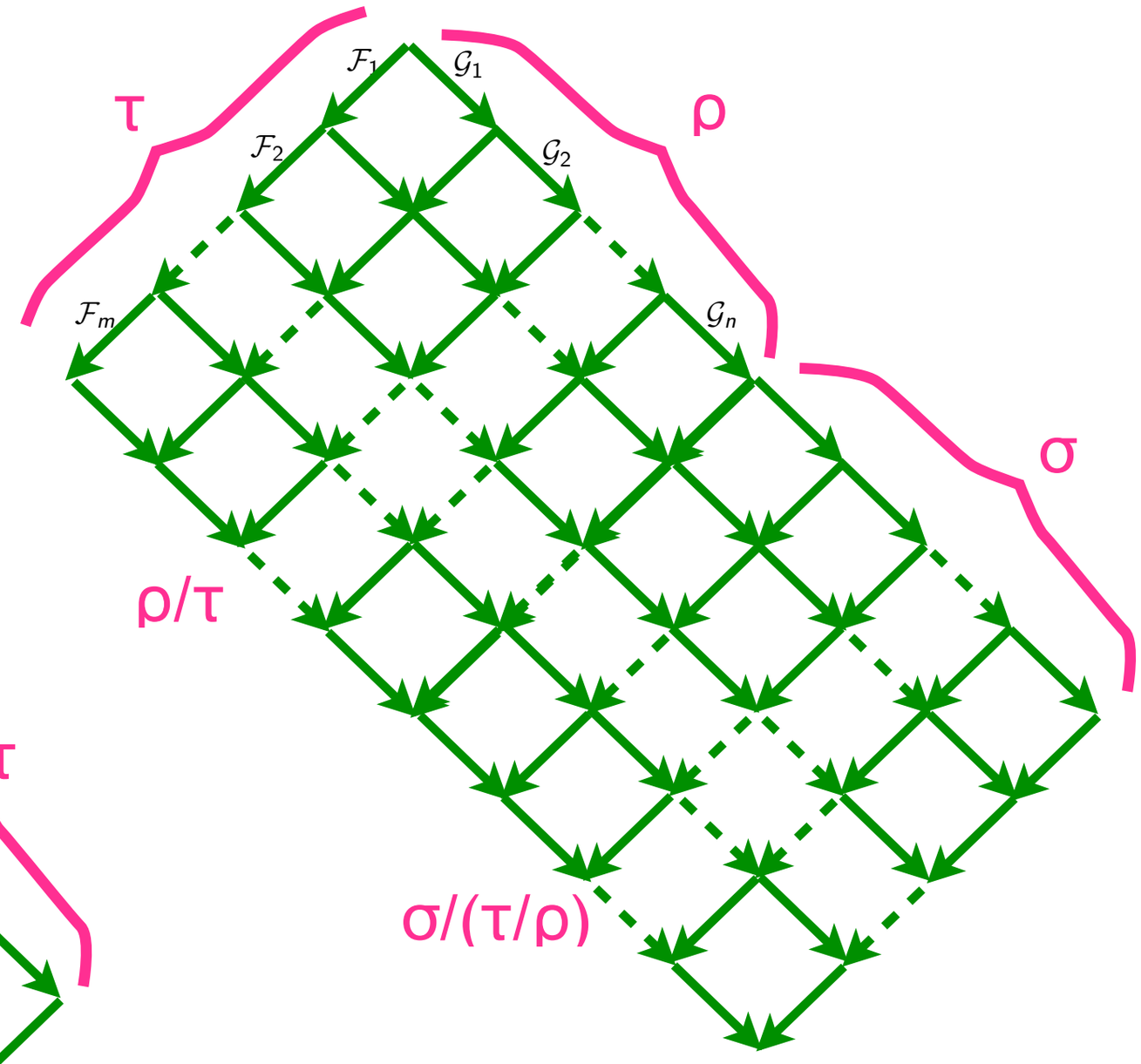
$$\rho \sqcup \sigma = \rho\,(\sigma/\rho)$$

- **Proposition** [Parallel Moves +]:

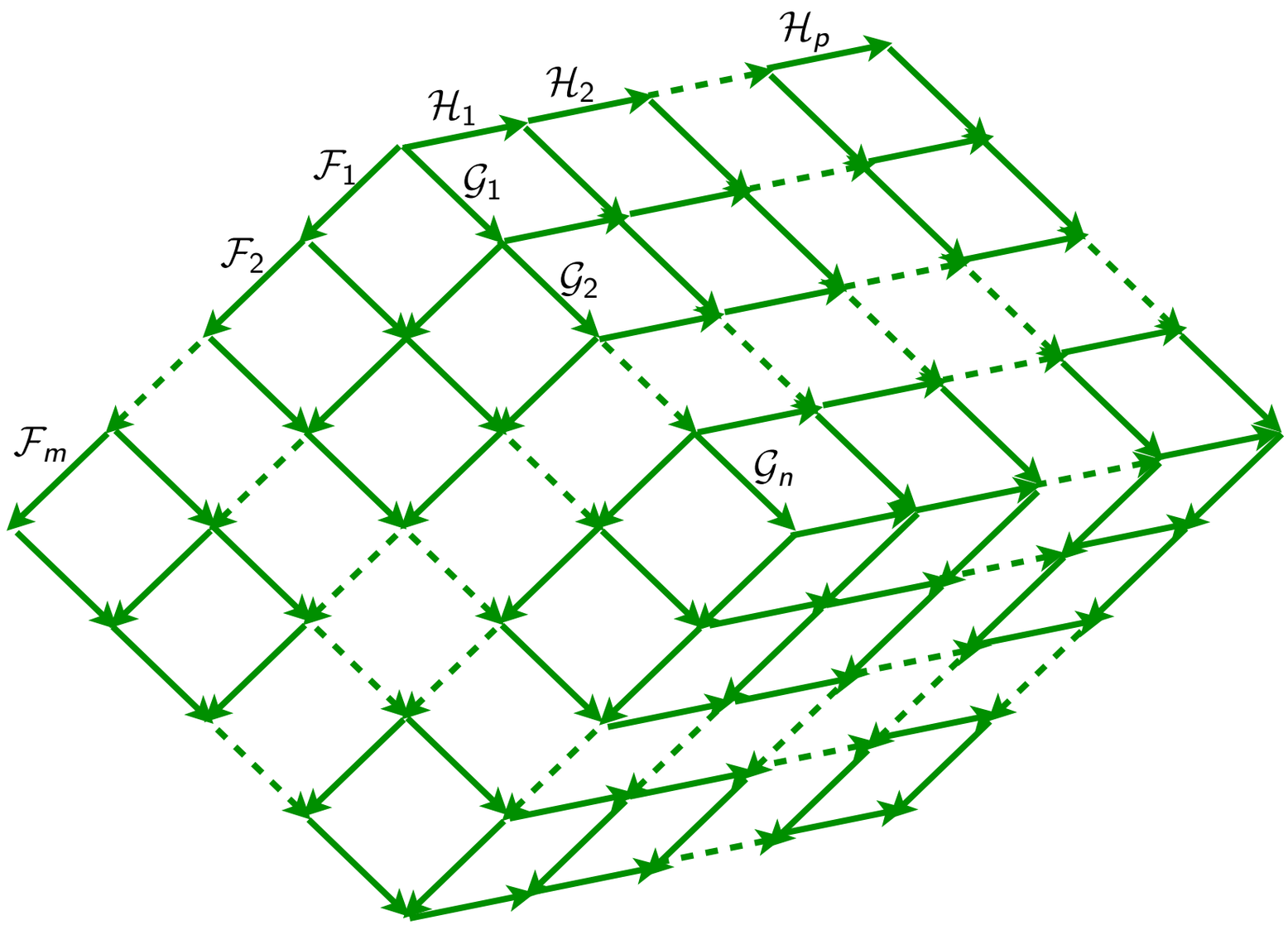
$\rho \sqcup \sigma$ and $\sigma \sqcup \rho$ are cofinal

Residuals of reductions (3/4)





Residuals of reductions (4/4)



- **Proposition** `[Cube Lemma ++]`:

$$\tau/(\rho \sqcup \sigma) = \tau/(\sigma \sqcup \rho)$$

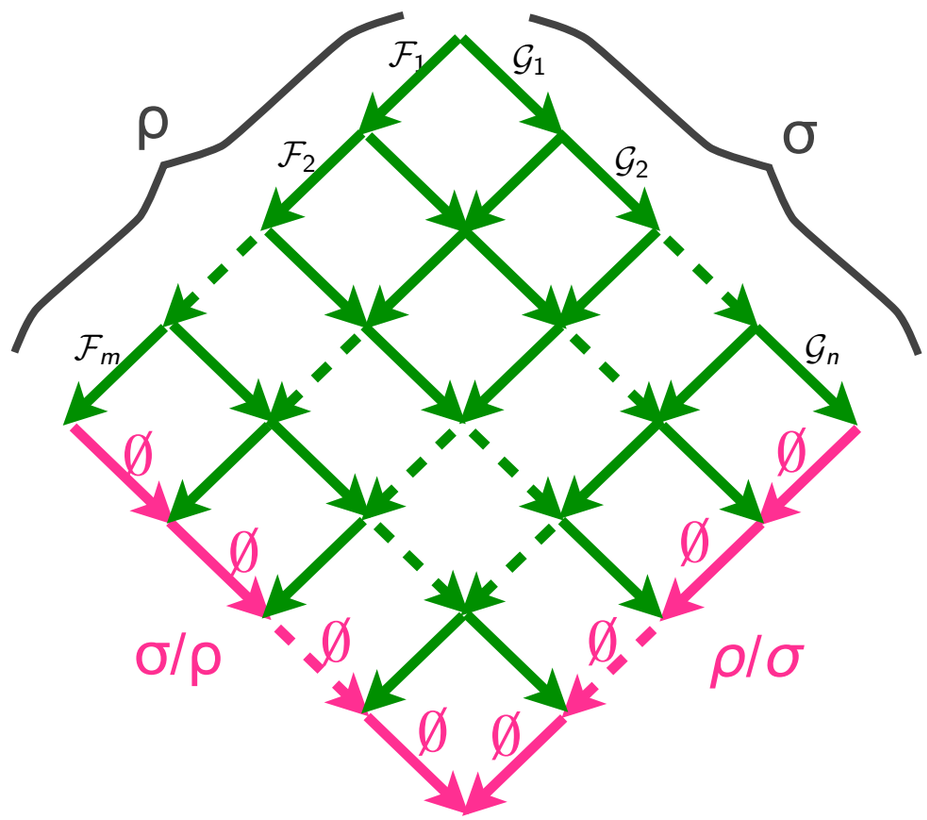
Equivalence by permutations

Equivalence by permutations (1/4)

- **Definition:**

  Let ρ and σ be 2 coinitial reductions. Then ρ is equivalent to σ by permutations, $\rho \simeq \sigma$, iff:

$$\rho/\sigma = \emptyset^m \quad \text{and} \quad \sigma/\rho = \emptyset^n$$



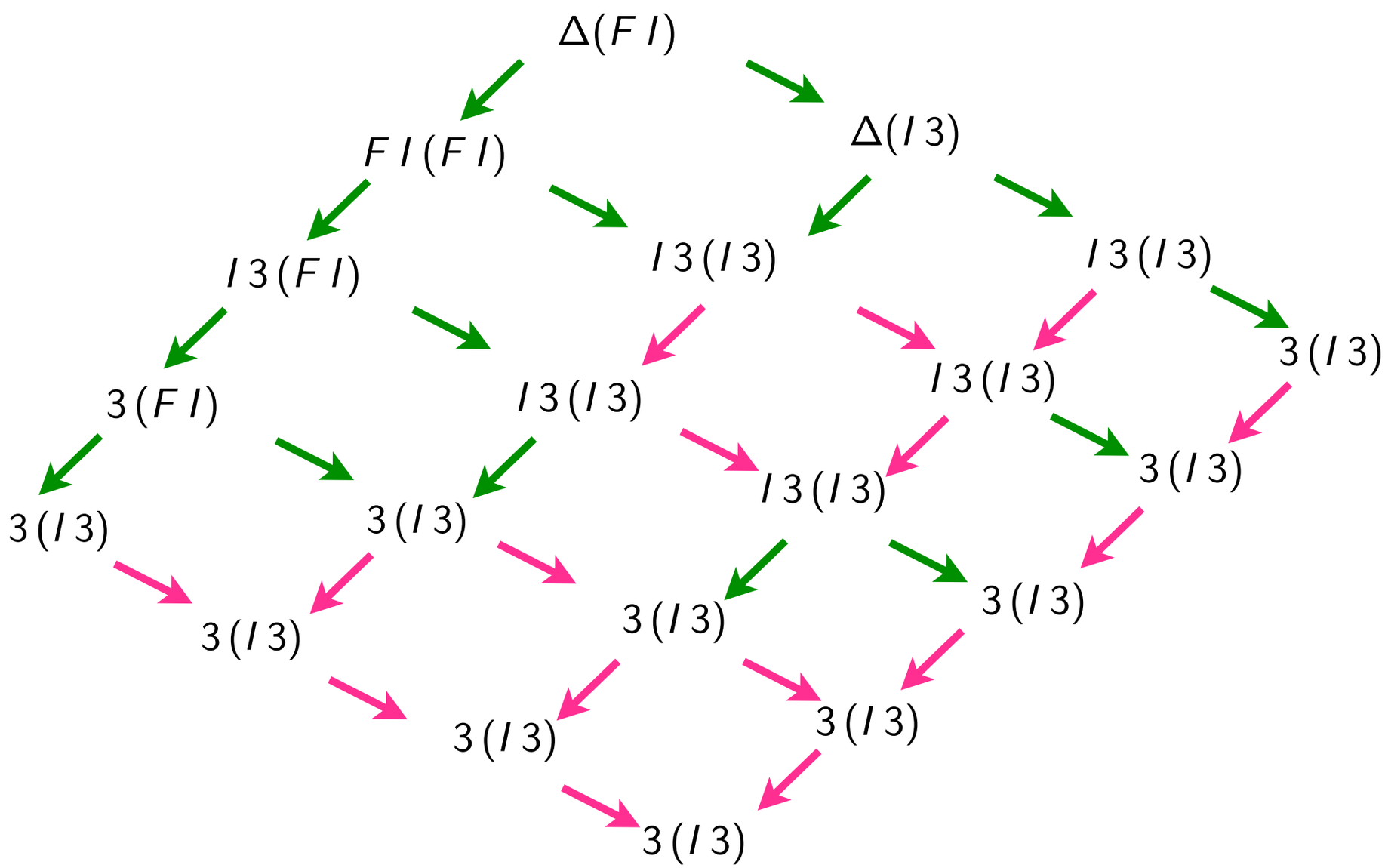
$\rho \simeq \sigma$ means that ρ and σ are coinitial and cofinal but converse is not true (see later)

Equivalence by permutations (2/4)

$\Delta = \lambda x.xx$

$F = \lambda f.f\,3$

$I = \lambda x.x$



Equivalence by permutations (3/4)

$$
\begin{tikzcd}
 & (\lambda x.x)^a((\lambda x.x)^b y) \arrow[dl, "R_a"'] \arrow[dr, "R_b"] & \\
(\lambda x.x)^b y \arrow[dr, "R_b/R_a"'] & & (\lambda x.x)^a y \arrow[dl, "R_a/R_b"] \\
 & y &
\end{tikzcd}
$$

$$\rho : M = I^a(I^b y) \xrightarrow{R_a} I^b y$$

$$\sigma : M = I^a(I^b y) \xrightarrow{R_b} I^a y$$

- Here $\rho \not\simeq \sigma$ while ρ and σ are coinitial and cofinal in the calculus with no labels

Equivalence by permutations (4/4)

- Same with $0 \not\simeq \rho$ when $\rho : \Delta\Delta \longrightarrow \Delta\Delta$

  $\Delta = \lambda x.xx$

Exercise 1: Give other examples of non-equivalent reductions between same terms.

Exercise 2: Show following equalities

$$\rho/0 = \rho \qquad \emptyset^n/\rho = \emptyset^n$$

$$0/\rho = 0 \qquad 0 \simeq \emptyset^n$$

$$\rho/\emptyset^n = \rho \qquad \rho/\rho = \emptyset^n$$

Equivalence by permutations (4/4)

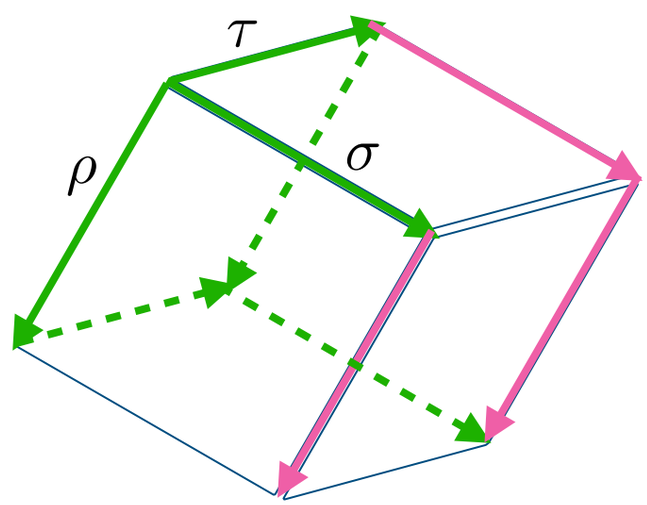
Exercise 3: Show that $\simeq$ is an equivalence relation.

Proof

(i) $\rho \simeq \rho$ obvious

(ii) same with $\rho \simeq \sigma$ implies $\sigma \simeq \rho$

(iii) $\rho \simeq \sigma \simeq \tau$ implies $\rho \simeq \tau$??



Properties of permutations (1/3)

- **Proposition**

  (i) $\rho \simeq \sigma$ iff $\forall \tau.\, \tau/\rho = \tau/\sigma$

  (ii) $\rho \simeq \sigma$ implies $\rho/\tau = \sigma/\tau$

  (iii) $\rho \simeq \sigma$ iff $\tau\rho \simeq \tau\sigma$

  (iv) $\rho \simeq \sigma$ implies $\rho\,\tau \simeq \sigma\,\tau$

  (v) $\rho \sqcup \sigma \simeq \sigma \sqcup \rho$

Proof

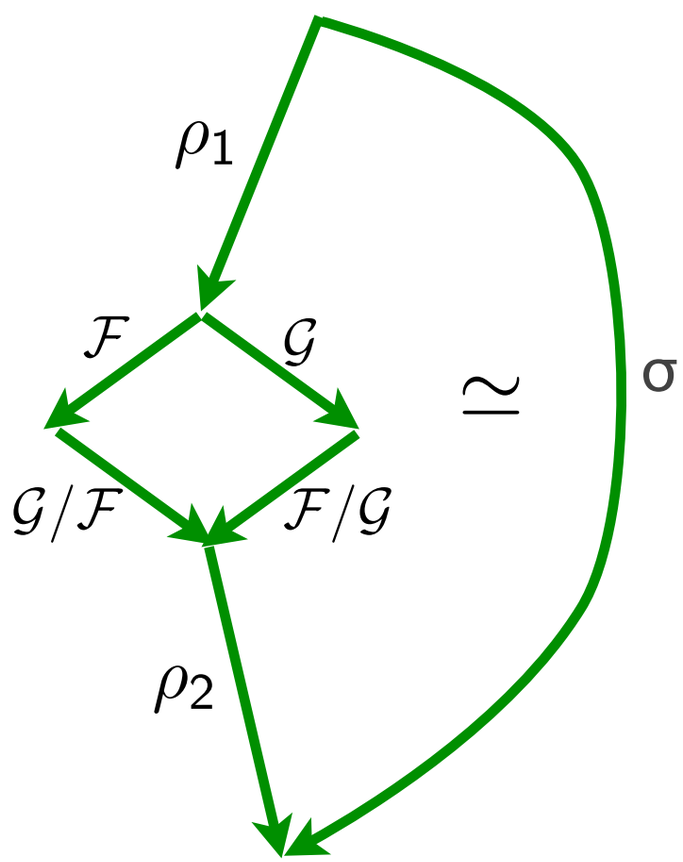
(i) $\rho \simeq \sigma$ implies $\sigma/\rho = \emptyset^n$ and $\rho/\sigma = \emptyset^m$.
Thus $\tau/(\rho \sqcup \sigma) = \tau/(\rho\,(\sigma/\rho)) = \tau/\rho/(\sigma/\rho) = \tau/\rho/\emptyset^m = \tau/\rho$
Similarly $\tau/(\sigma \sqcup \rho) = \tau/\sigma$
By cube lemma $\tau/\rho = \tau/\sigma$
Conversely, take $\tau = \rho$ and $\tau = \sigma$.

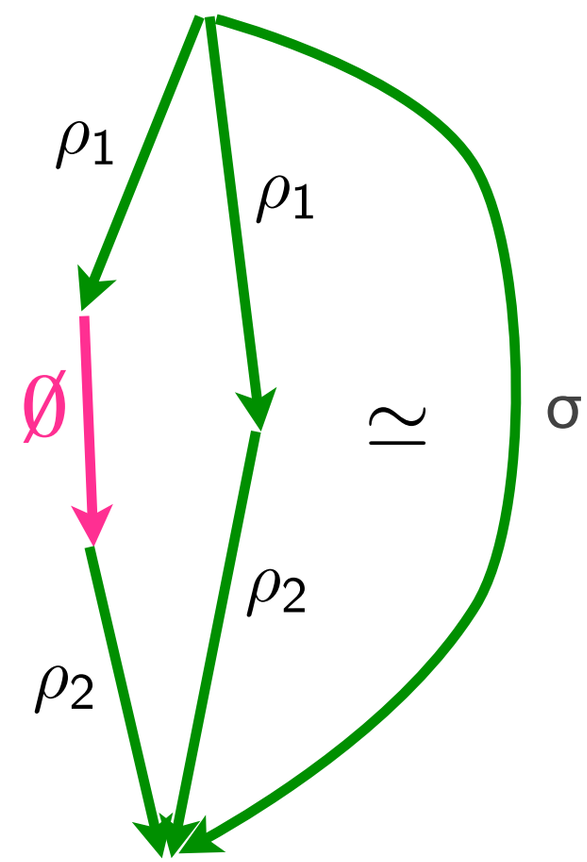
Properties of permutations (2/3)

- **Proposition** $\simeq$ is the smallest congruence containing

$$\mathcal{F}\,(\mathcal{G}/\mathcal{F}) \simeq \mathcal{G}\,(\mathcal{F}/\mathcal{G})$$

$$0 \simeq \emptyset$$



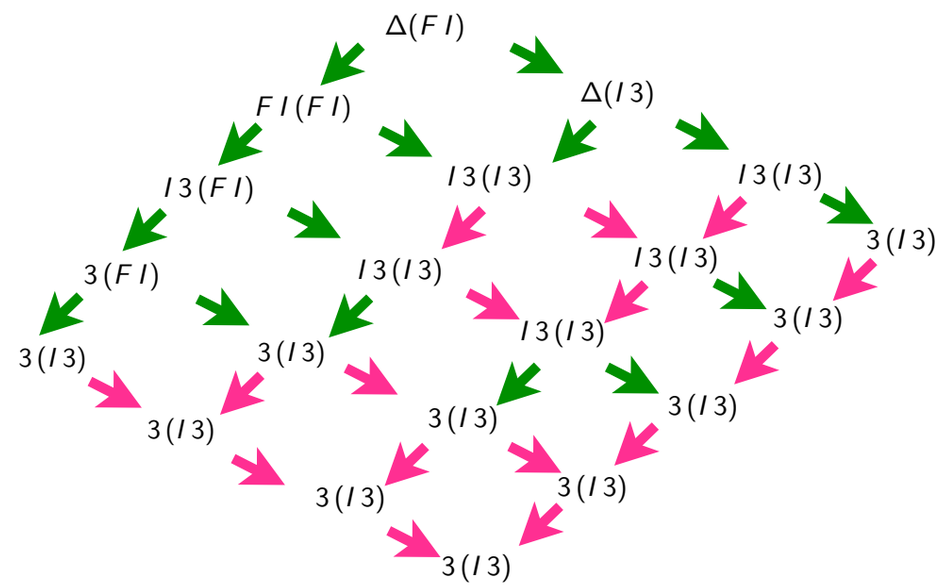


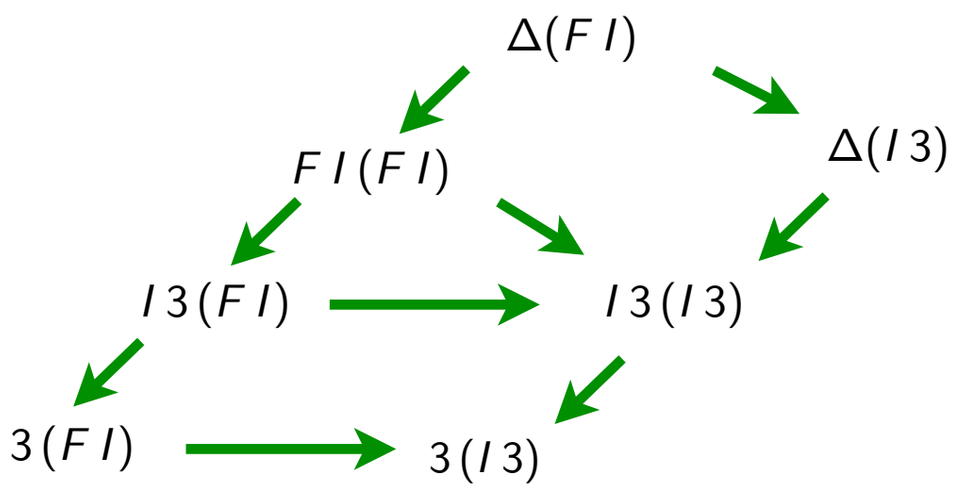
Properties of permutations (3/3)

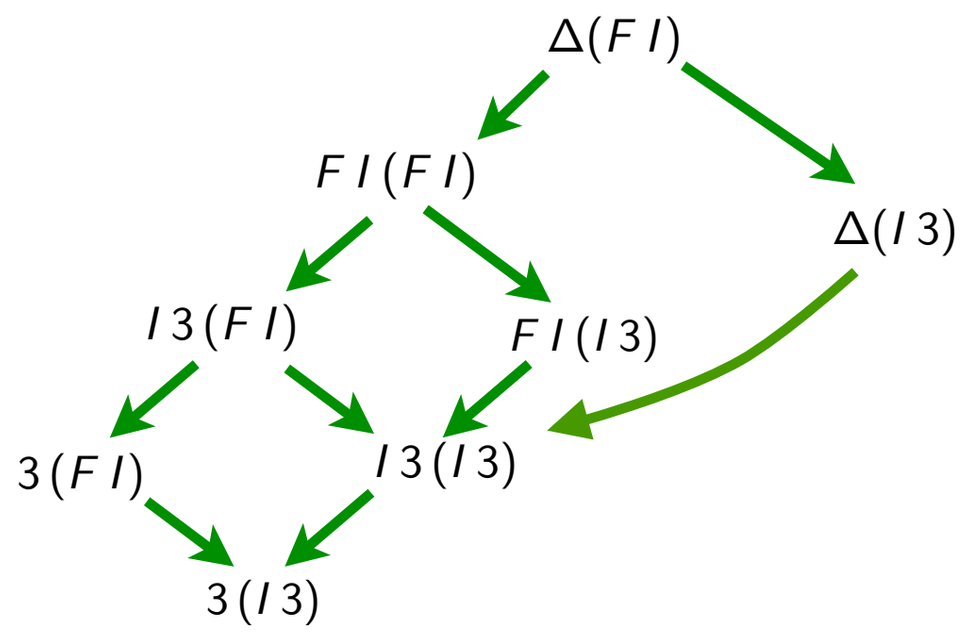
$\Delta = \lambda x.xx$

$F = \lambda f.f\,3$

$I = \lambda x.x$







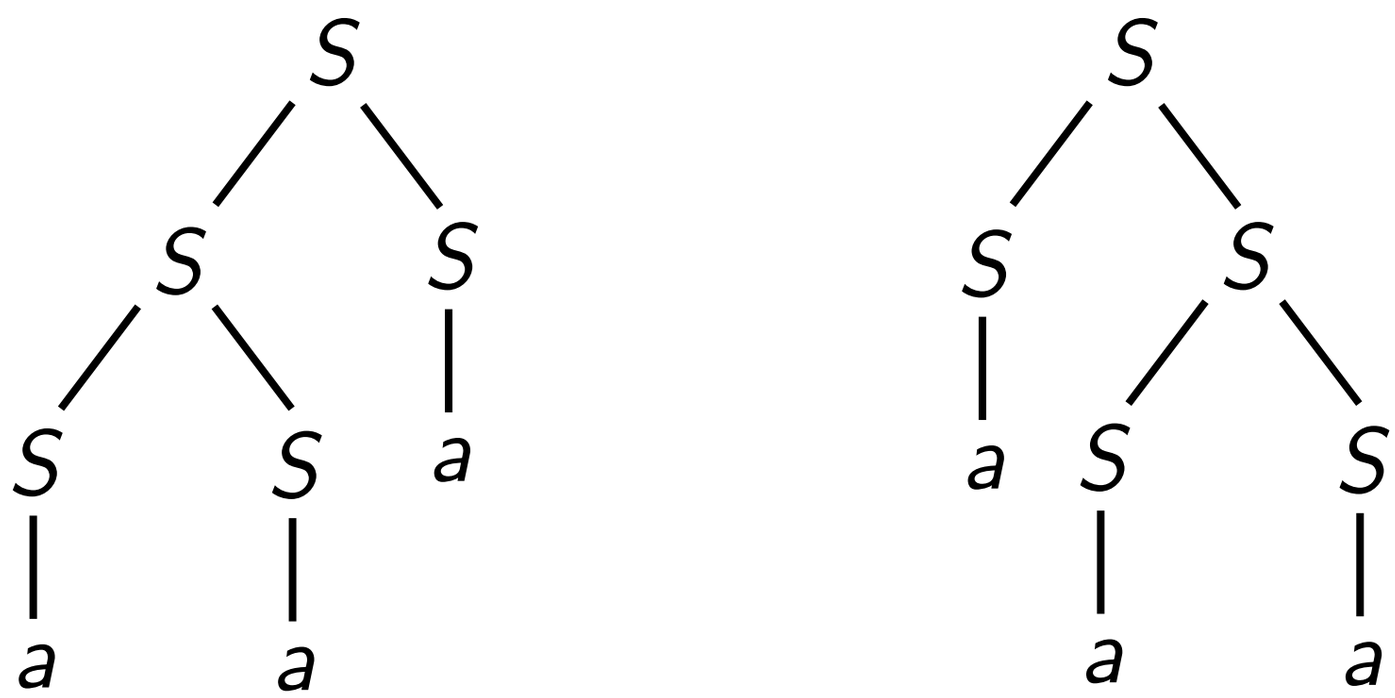
Beyond the λ-calculus

Context-free languages

- permutations of derivations in contex-free languages

$S \to SS$

$S \to a$



- each parse tree corresponds to an equivalence class

Term rewriting

- recursive program schemes [Berry-JJL'77]
- permutations of derivations in orthogonal TRS [Huet-JJL'81]
- permutations of derivations are defined with critical pairs
- critical pairs make conflicts
- only 2nd definition of equivalence works [Boudol'82]
- interaction systems [Asperti-Laneve'93]

Process algebras

- similar to TRS [Boudol-Castellani'82]
- connection to event structures [Laneve'84]

PCF

- LCF considered as a programming language [Plotkin'74]

```
M, N, P  ::= x                      variable
           | λx.M | M N             abstraction application
           | n                      integer constant
           | M ⊗ N                  ⊗ ∈ {+,−,×,÷}
           | ifz M then N then N    conditionnal
           | μx.M                   recursive definition
```

```
β      (λx.M)N → M {x := N}
op     m ⊗ n → m ⊗ n
cond1  ifz 0 then M else N → M
cond2  ifz n+1 then M else N → N
μ      μx.M → M {x := μx.M}
```

Exercises

Parallel moves (1/4)

- **Lemma** $M \xrightarrow{\mathcal{F}} N,\ M \xrightarrow{\mathcal{G}} P \ \Rightarrow\ N \xrightarrow{\mathcal{G}} Q,\ P \xrightarrow{\mathcal{F}} Q$

Proof

Case 1: $M = x = N = P = Q$. Obvious.

Case 2: $M = \lambda x.M_1,\ N = \lambda x.N_1,\ P = \lambda x.P_1$. Obvious by induction on M_1

Case 3: (App-App) $M = M_1 M_2,\ N = N_1 N_2,\ P = P_1 P_2$. Obvious by induction on $M_1,\ M_2$.

Case 4: (Red'-Red') $M = (\lambda x.M_1)^a M_2,\ N = (\lambda x.N_1)^a N_2,\ P = (\lambda x.P_1)^a P_2,\ a \notin \mathcal{F} \cup \mathcal{G}$

Then induction on $M_1,\ M_2$.

Case 4: (beta-Red') $M = (\lambda x.M_1)^a M_2,\ N = N_1\{x := N_2\},\ P = (\lambda x.P_1)^a P_2,\ a \in \mathcal{F}, a \notin \mathcal{G}$

By induction $N_1 \xrightarrow{\mathcal{G}} Q_1,\ P_1 \xrightarrow{\mathcal{F}} Q_1$. And $N_2 \xrightarrow{\mathcal{G}} Q_2,\ P_1 \xrightarrow{\mathcal{F}} Q_2$.

By lemma, $N_1\{x := N_2\} \xrightarrow{\mathcal{G}} Q_1\{x := Q_2\}$. And $(\lambda x.P_1)^a P_2 \xrightarrow{\mathcal{F}} Q_1\{x := Q_2\}$

Case 5: (beta-beta) $M = (\lambda x.M_1)^a M_2,\ N = N_1\{x := N_2\},\ P = P_1\{x := P_2\},\ a \in \mathcal{F} \cap \mathcal{G}$

As before with same lemma.

Parallel moves (1/4)

- Lemma $M \xrightarrow{\mathcal{F}} N, P \xrightarrow{\mathcal{F}} Q \;\Rightarrow\; M\{x := P\} \xrightarrow{\mathcal{F}} N\{x := Q\}$

  Proof: exercise!

- Lemma [subst] $M\{x := N\}\{y := P\} = M\{y := P\}\{x := N\{y := P\}\}$

  when x not free in P

Finite Developments in the λ-calculus

Part 2

jean-jacques.levy@inria.fr

ISR 2021

Madrid

July 6, 2021

http://jeanjacqueslevy.net/talks/21isr

A labeled lambda-calculus (1/3)

- Give names to redexes and to (some) subterms
- make names consistent with permutation equivalence.

$$M, N, \ldots ::= x \mid MN \mid \lambda x.M \mid M^{\alpha}$$

- Conversion rule is:

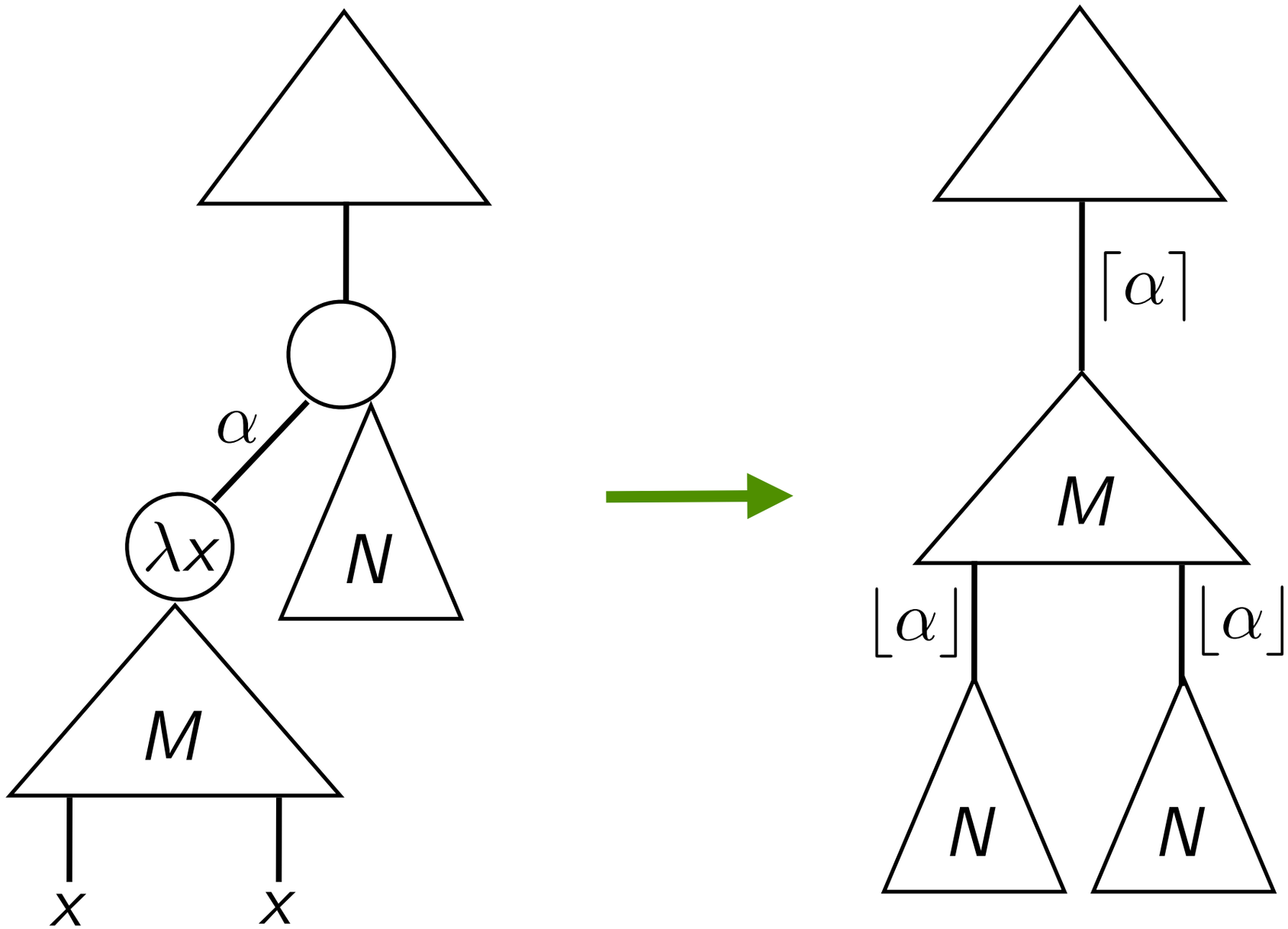
$$(\lambda x.M)^{\alpha} N \longrightarrow M^{\lceil \alpha \rceil}\{x := N^{\lfloor \alpha \rfloor}\}$$

α is **name** of redex

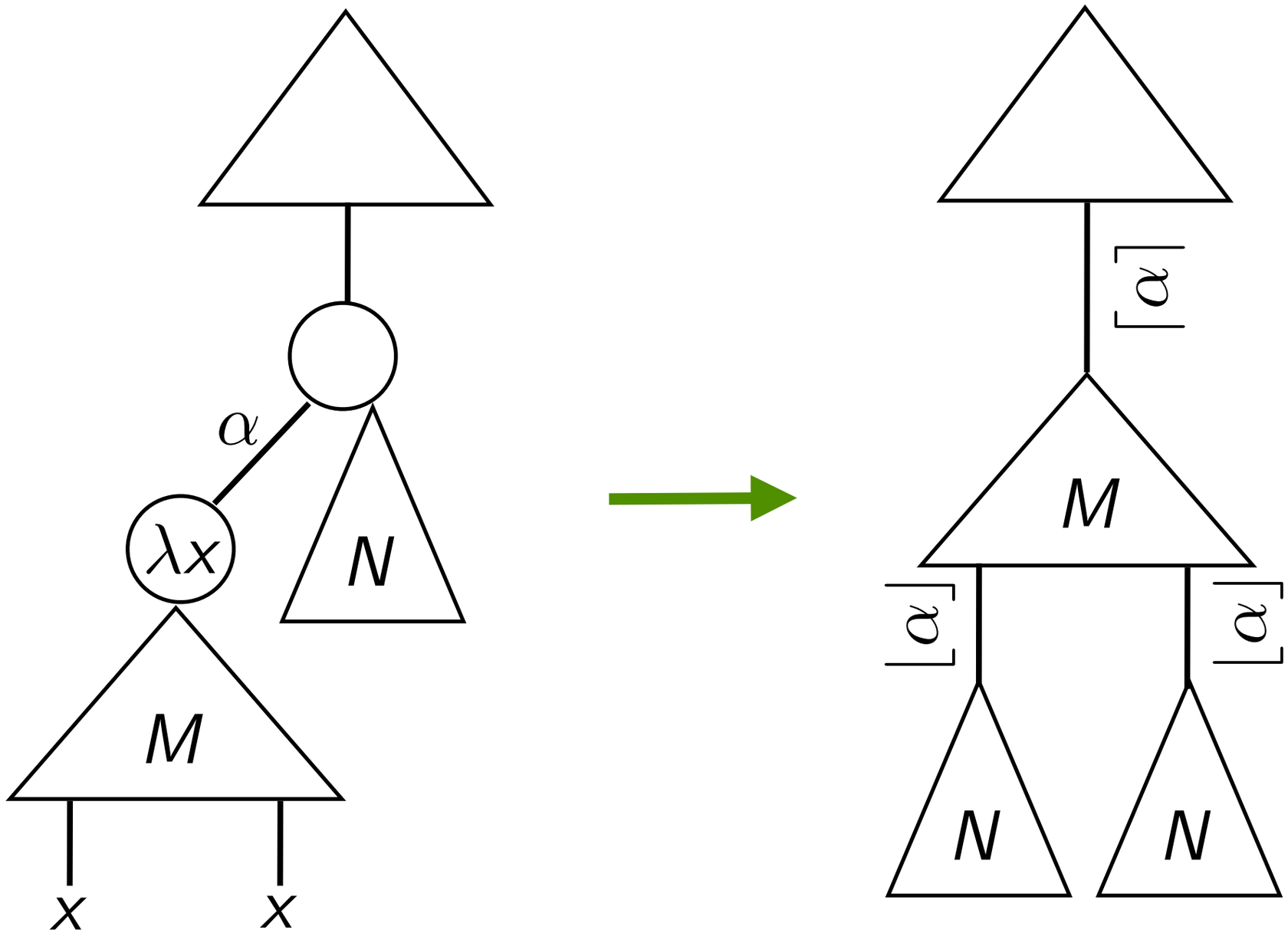
where

$$(M^{\alpha})^{\beta} = M^{\alpha\beta} \quad \text{and} \quad (M^{\alpha})\{x := N\} = (M\{x := N\})^{\alpha}$$

A labeled lambda-calculus (2/3)



A labeled lambda-calculus (2/3)



A labeled lambda-calculus (3/3)

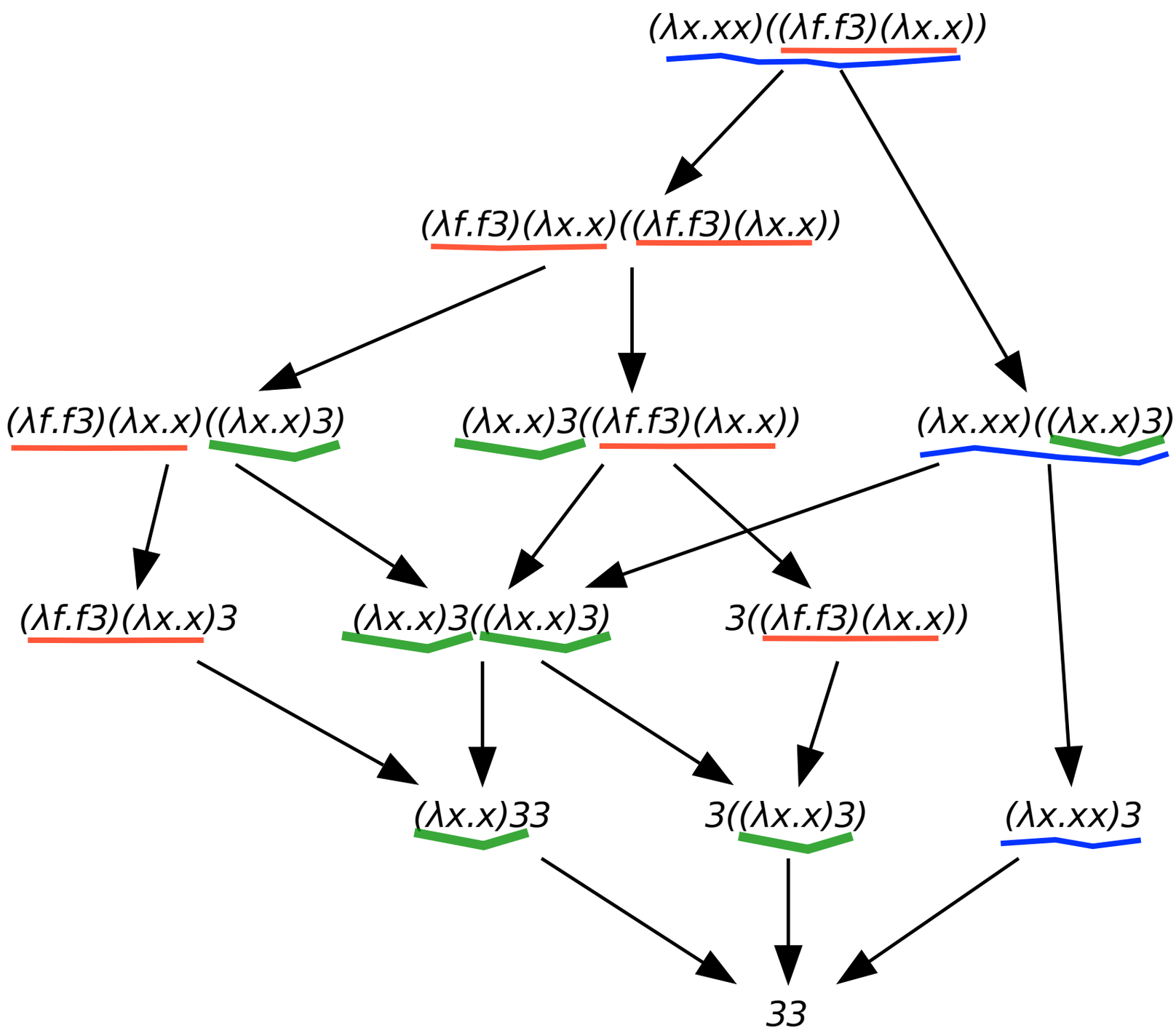
- Labels are strings of atomic labels:

$$\alpha, \beta, \ldots \quad ::= \quad \underbrace{a, b, c, \ldots \mid \lceil\alpha\rceil \mid \lfloor\alpha\rfloor}_{\texttt{atomic labels}} \mid \alpha\beta \mid \epsilon$$

- Labels are strings of atomic labels:

$a, b, c, \ldots$	atomic letters
$\lceil\alpha\rceil, \lfloor\alpha\rfloor, \ldots$	overlined, underlined labels
$\alpha\beta$	compound labels
$\epsilon = \lfloor\epsilon\rfloor = \lceil\epsilon\rceil$	empty label

Example



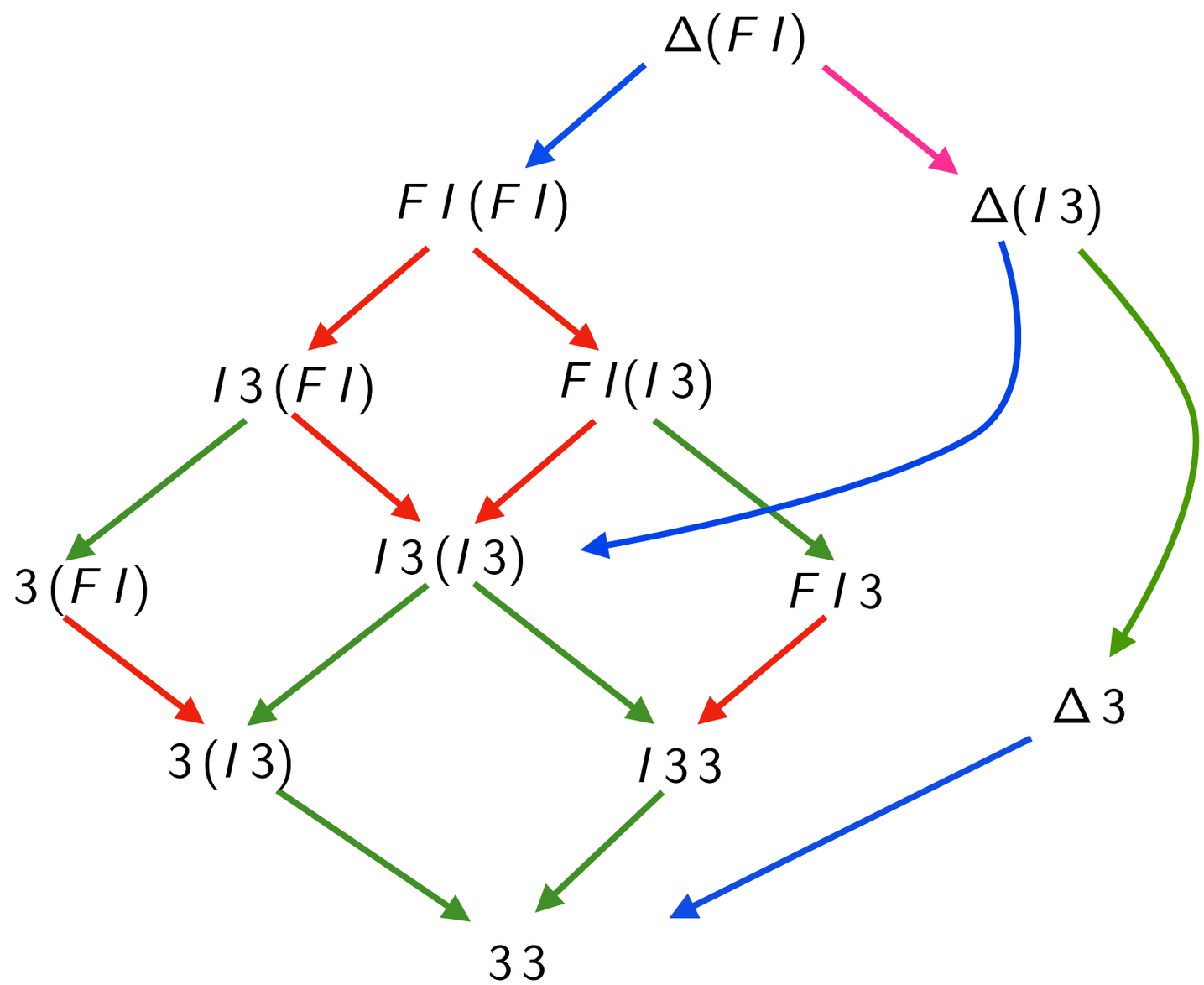
- 3 redex families: **red**, **blue**, **green**.

Example

$\Delta = \lambda x.x\,x$

$F = \lambda f.f\,3$

$I = \lambda x.x$



Example

$\Delta = \lambda x.(x^c\, x^d)^b$

$F = \lambda f.(f^k\, 3^\ell)^j$

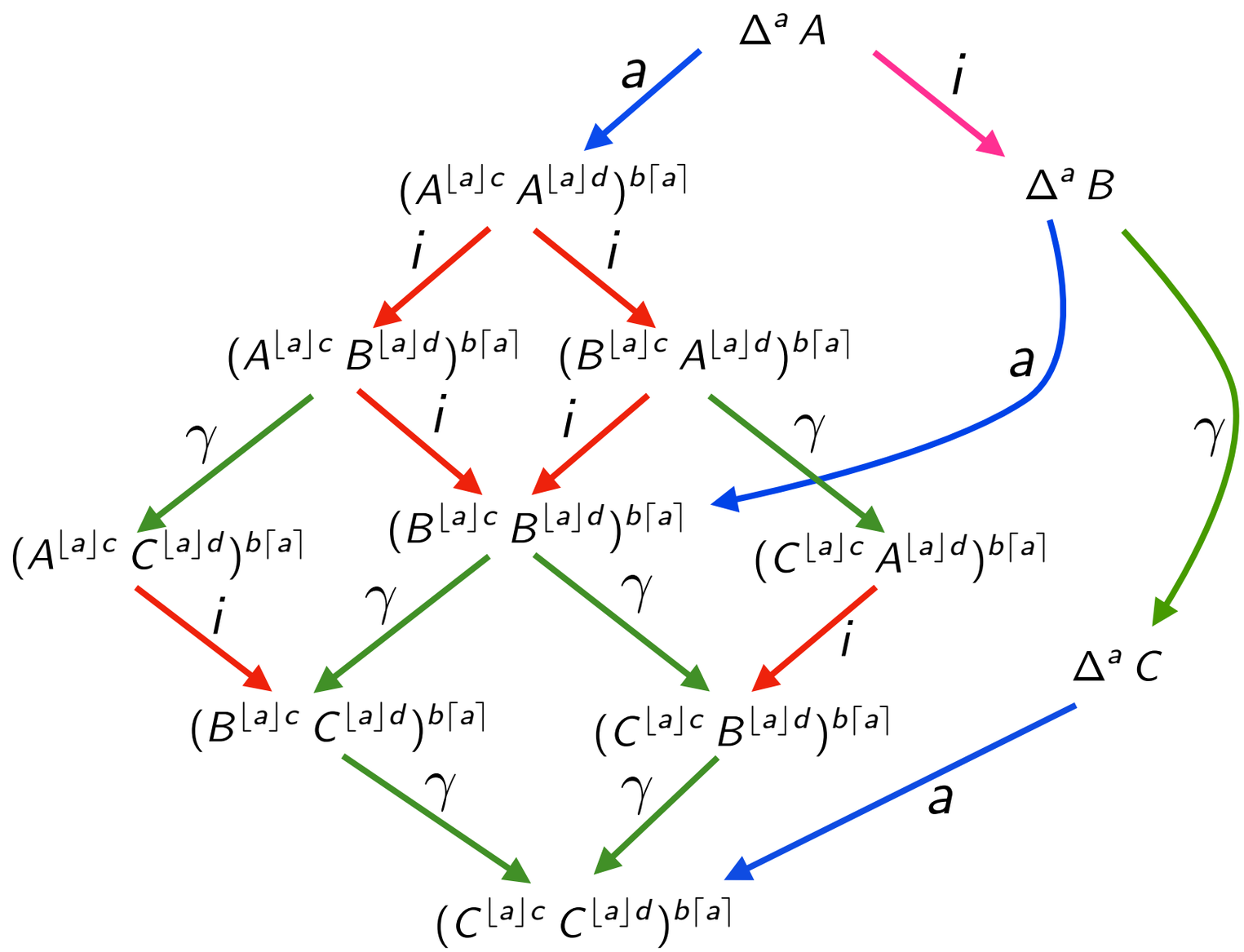
$I = \lambda x.x^v$

$A = (F^i\, I^u)^q$

$B = (I^\gamma\, 3^\ell)^q$

$C = 3^{\ell\lfloor\gamma\rfloor v\lceil\gamma\rceil q}$

$\gamma = u\lfloor i\rfloor k$



3 redexes names: $a, i, \gamma = u\lfloor i\rfloor k$

Example

$$\Omega = D^a\,\Delta^e$$

$$\Big\downarrow a$$

$$\Omega_1 = (\Delta^{\gamma_1}\,\Delta^{\delta_1})^{b\lceil a\rceil}$$

$$\Big\downarrow \gamma_1$$

$$\Omega_2 = (\Delta^{\gamma_2}\,\Delta^{\delta_2})^{f\lceil\gamma_1\rceil b\lceil a\rceil}$$

$$\Big\downarrow \gamma_2$$

$$\Omega_3 = (\Delta^{\gamma_3}\,\Delta^{\delta_3})^{f\lceil\gamma_2\rceil f\lceil\gamma_1\rceil b\lceil a\rceil}$$

$$\Big\downarrow \gamma_3$$

$$\Omega_4 = (\Delta^{\gamma_4}\,\Delta^{\delta_4})^{f\lceil\gamma_3\rceil f\lceil\gamma_2\rceil f\lceil\gamma_1\rceil b\lceil a\rceil}$$

$$\Big\downarrow \gamma_4$$

redexes names: $a, \gamma_1, \gamma_2, \gamma_3, \ldots$

$$D = \lambda x.(x^c\,x^d)^b$$

$$\Delta = \lambda x.(x^g\,x^h)^f$$

$$\gamma_1 = e\lfloor a\rfloor c$$

$$\gamma_2 = \delta_1\lfloor\gamma_1\rfloor g$$

$$\gamma_3 = \delta_2\lfloor\gamma_2\rfloor g$$

$$\gamma_4 = \delta_3\lfloor\gamma_3\rfloor g$$

$$\delta_1 = e\lfloor a\rfloor d$$

$$\delta_2 = \delta_1\lfloor\gamma_1\rfloor h$$

$$\delta_3 = \delta_2\lfloor\gamma_2\rfloor h$$

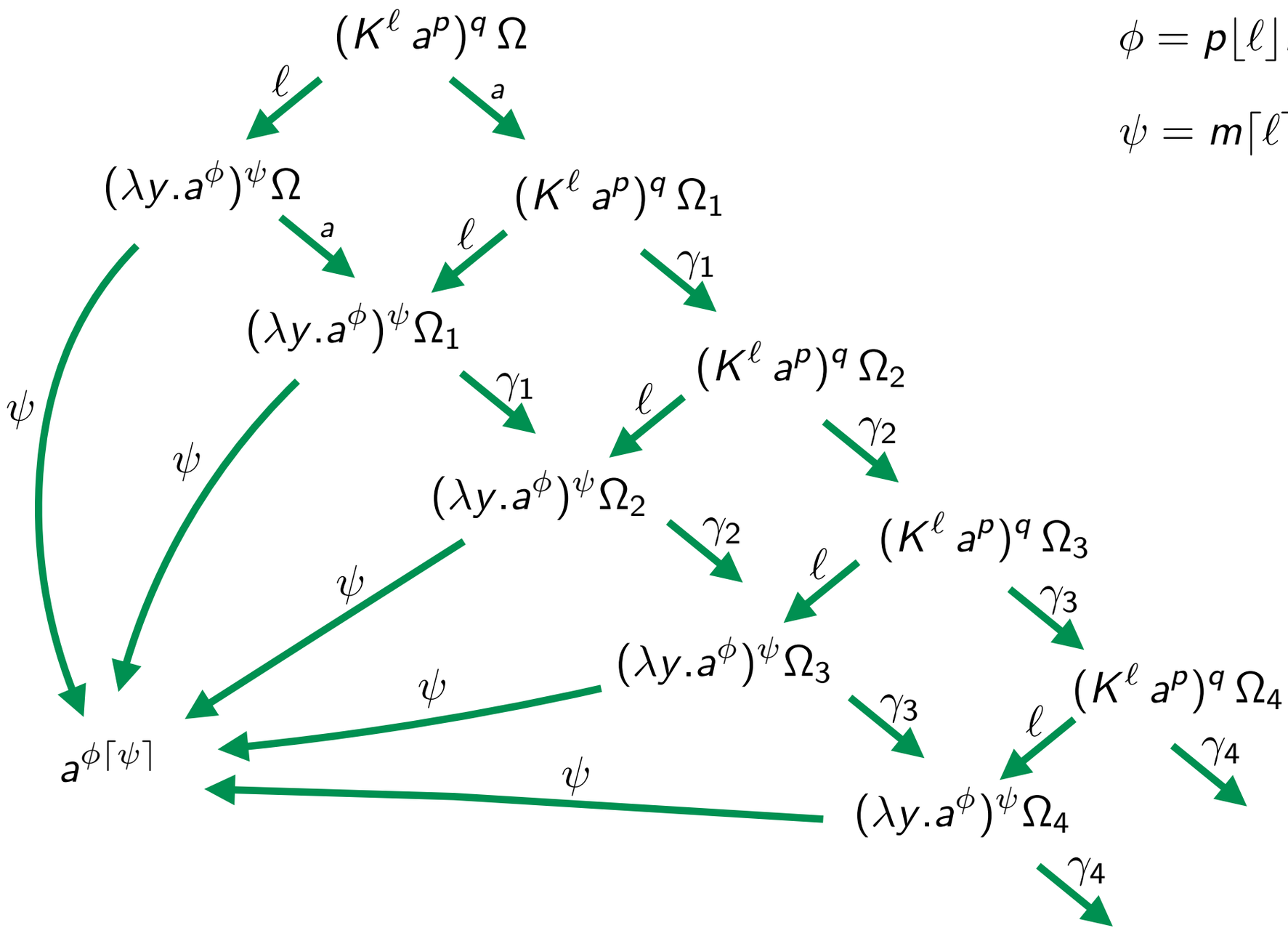
$$\delta_4 = \delta_2\lfloor\gamma_2\rfloor h$$

Example

$K = \lambda x.(\lambda y.x^n)^m$

$\phi = p\lfloor\ell\rfloor n$

$\psi = m\lceil\ell\rceil q$



redexes names: $\ell, \psi, a, \gamma_1, \gamma_2, \gamma_3, \ldots$

Example

$$F = \lambda f.(f^c\, 3^d)^b$$
$$I = \lambda x.x^v$$
$$\Delta = \lambda x.(x^k\, x^\ell)^j$$

$$\begin{tikzcd}
 & F^a(I^u\,\Delta^i)^q \arrow[dl, "a"'] \arrow[dr, "u"] & \\
((I^u\,\Delta^i)^{q\lfloor a\rfloor c}\, 3^d)^{b\lceil a\rceil} \arrow[dr, "u"'] & & F^a\,\Delta^{i\lfloor u\rfloor v\lceil u\rceil q} \arrow[dl, "a"] \\
 & (\Delta^{i\lfloor u\rfloor v\lceil u\rceil q\lfloor a\rfloor c}\, 3^d)^{b\lceil a\rceil} &
\end{tikzcd}$$

2 independent redexes a and u creates the new one $i\lfloor u\rfloor v\lceil u\rceil q\lfloor a\rfloor c$

Empirical facts (bis)

- **deterministic** result when it exists — **Church-Rosser**
- multiple reduction strategies
- **terminating** strategy ?
- **efficient** reduction strategy ? — **optimal reduction**
- **worst** reduction strategy ?
- when all reductions are finite ? — **strong normalisation**
- when finite, the reduction graph has a **lattice** structure ? — **YES!**

Permutation equivalence (1/7)

- **Proposition** `[residuals of labeled redexes]`

  $S \in R/\rho$ implies $\mathsf{name}(R) = \mathsf{name}(S)$

- **Definition** `[created redexes]` Let $\rho : M \xrightarrow{\star} N$

  we say that ρ **creates** R in M when $\nexists R'$, $R \in R'/\rho$.

- **Proposition** `[created labeled redexes]`

  If S creates R, then $\mathsf{name}(S)$ is strictly contained in $\mathsf{name}(R)$.

Permutation equivalence (2/7)

Proof (cont'd) Created redexes contains names of creator

$$\underbrace{(\lambda x.\cdots(x^{\beta}\,N)\cdots)^{\alpha}}_{\alpha}\,(\lambda y.M)^{\gamma} \longrightarrow \cdots\underbrace{((\lambda y.M)^{\gamma\lfloor\alpha\rfloor\beta}\,N')}_{\gamma\lfloor\alpha\rfloor\beta}\cdots$$

creates

$$\underbrace{((\lambda x.(\lambda y.M)^{\gamma})^{\alpha}\,N)^{\beta}}_{\alpha}\,P \longrightarrow \underbrace{(\lambda y.M')^{\gamma\lceil\alpha\rceil\beta}}_{\gamma\lceil\alpha\rceil\beta}\,P$$

creates

$$\underbrace{((\lambda x.\,x^{\gamma})^{\alpha}\,(\lambda y.M)^{\delta})^{\beta}}_{\alpha}\,N \longrightarrow \underbrace{(\lambda y.M)^{\delta\lfloor\alpha\rfloor\gamma\lceil\alpha\rceil\beta}}_{\delta\lfloor\alpha\rfloor\gamma\lceil\alpha\rceil\beta}\,N$$

creates

Permutation equivalence (3/7)

- **Labeled laws** $M^\alpha\{x := N\} = (M\{x := N\})^\alpha \qquad (M^\alpha)^\beta = M^{\alpha\beta}$

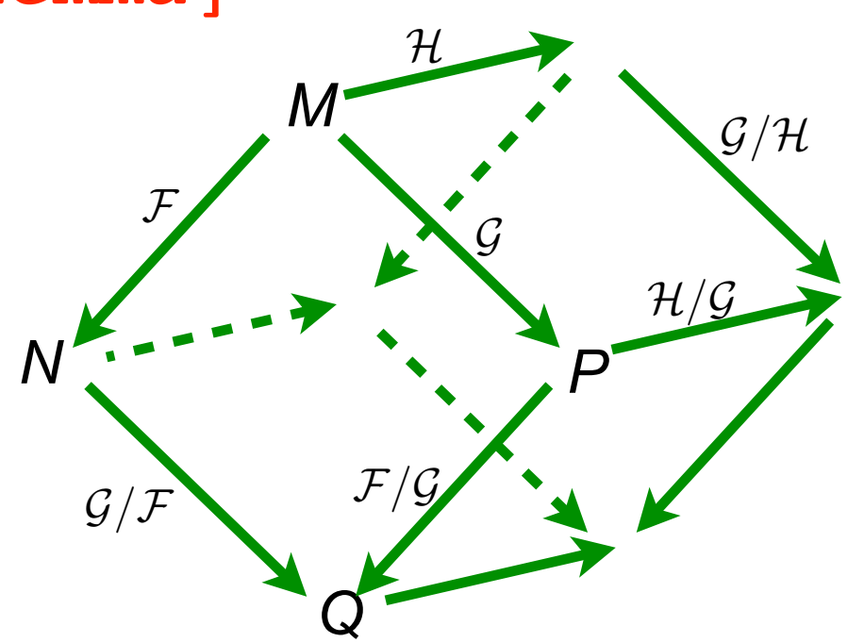
  If $M \longrightarrow N$, then $M^\alpha \longrightarrow N^\alpha$

- **Labeled parallel moves lemma+** [74]

  If $M \xrightarrow{\mathcal{F}} N$ and $M \xrightarrow{\mathcal{G}} P$, then $N \xrightarrow{\mathcal{G}/\mathcal{F}} Q$ and $P \xrightarrow{\mathcal{F}/\mathcal{G}} Q$ for some Q.

- **Parallel moves lemma++** [The Cube Lemma]

  still holds.



Permutation equivalence (4/7)

- Labels do not break Church-Rosser, nor residuals

- Labels refine λ-calculus:
  - any unlabeled reduction can be performed in the labeled calculus
  - but two cofinal unlabeled reductions may no longer be cofinal

Take $I(I3)$ with $I = \lambda x.x$.

$$I^b(J^e 3^g)^d$$

$$I = \lambda x.x^c$$
$$J = \lambda x.x^f$$

$$(J^e\ 3^g)^{d\lfloor b\rfloor c\lceil b\rceil} \qquad I^b 3^{g\lfloor e\rfloor f\lceil e\rceil d}$$

Permutation equivalence (5/7)

- **Definition** `[pure labeled calculus]`

  Pure labeled terms are labeled terms where all subterms have non empty labels.

- **Theorem** `[labeled permutation equivalence, 76]`

  Let ρ and σ be coinitial pure labeled reductions.
  Then $\rho \simeq \sigma$ iff ρ and σ are labeled cofinal.

Proof Let $\rho \simeq \sigma$. Then obvious because of labeled parallel moves lemma.
Conversely, we apply standardization thm and following lemma.

Permutation equivalence (6/7)

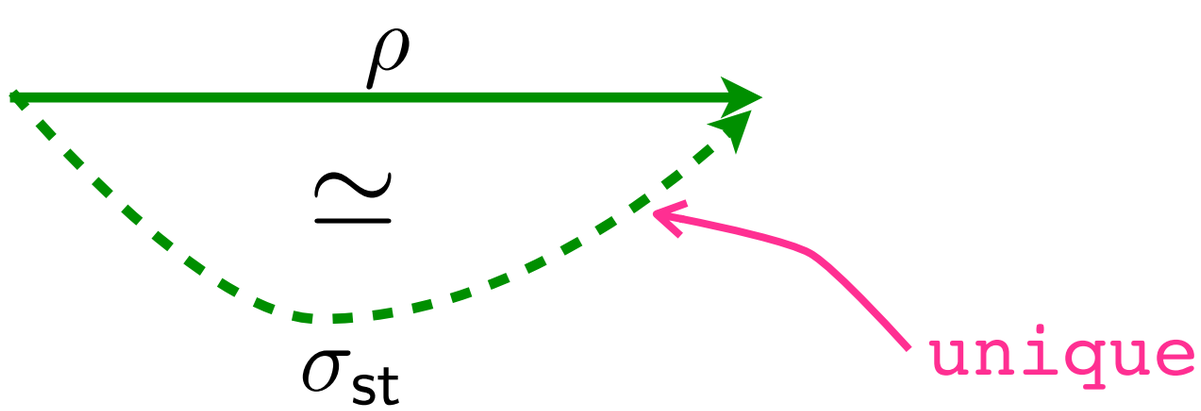
- **Definition:** The following reduction is **standard**

$$\rho : M = M_0 \xrightarrow{R_1} M_1 \xrightarrow{R_2} M_2 \cdots \xrightarrow{R_n} M_n = N$$

iff for all i and j, $i < j$, then R_j is not residual along ρ of some R'_j to the left of R_i in M_{i-1}.

- **Standardization** [Curry 50] Let $M \xrightarrow{\star} N$. Then $M \xrightarrow[\text{st}]{\star} N$.

- **Labeled standardization** $\forall \rho,\ \exists!\, \sigma_{\text{st}},\ \rho \simeq \sigma_{\text{st}}$



Permutation equivalence (7/7)

- **Notation** `[prefix ordering]` $\rho \sqsubseteq \sigma$ for $\exists \tau . \rho \tau \simeq \sigma$

- **Corollary** `[labeled prefix ordering]`
  Let $\rho : M \xrightarrow{\star} N$ and $\sigma : M \xrightarrow{\star} P$ be coinitial pure labeled reductions.
  Then $\rho \sqsubseteq \sigma$ iff $N \xrightarrow{\star} P$.

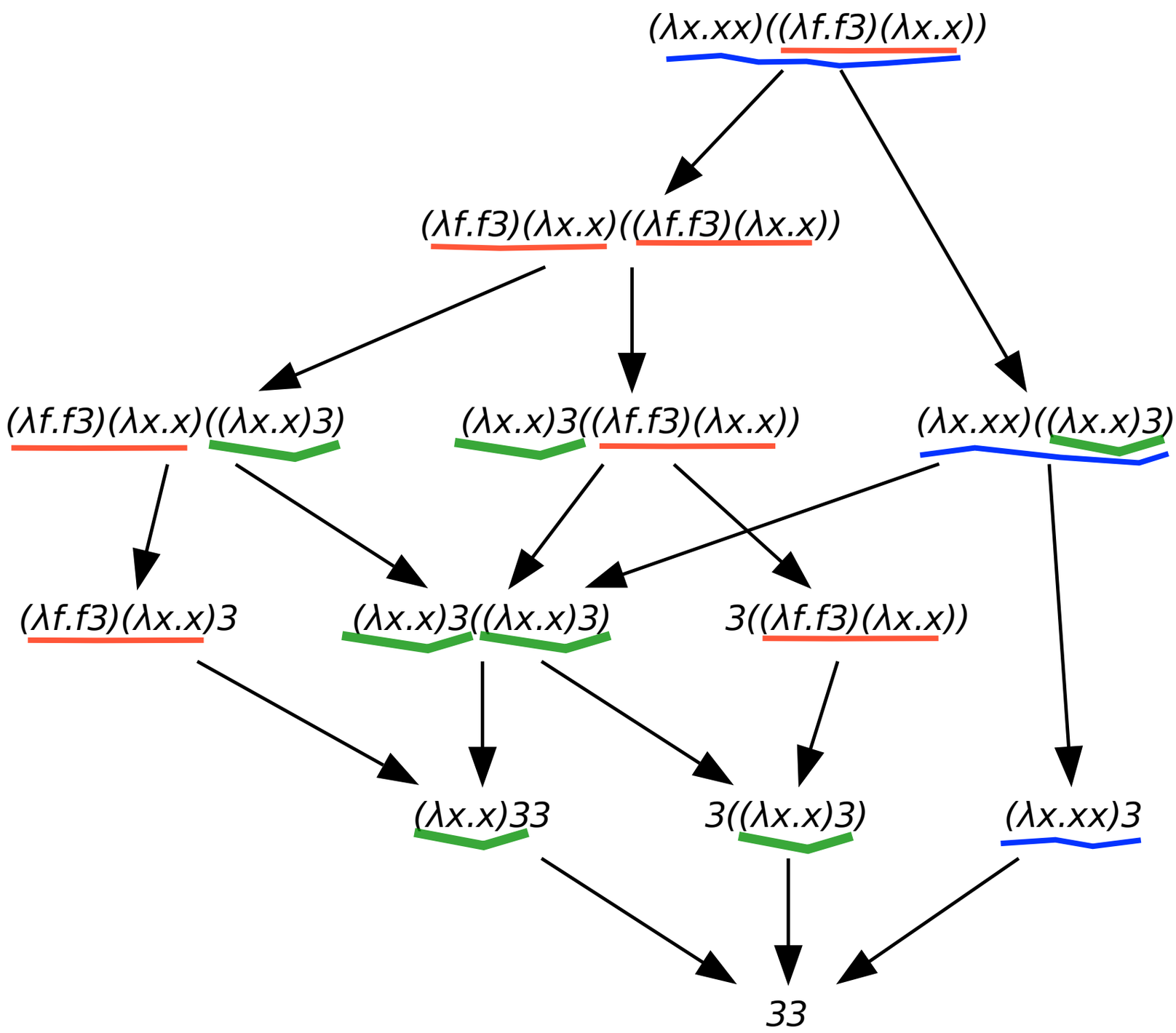
- **Corollary** `[lattice of labeled reductions]`
  Labeled reduction graphs are upwards semi lattices for any pure labeling.

In other terms, reductions up-to permutation equivalence is a push-out category.

Exercise Try on $(\lambda x.x)((\lambda y.(\lambda x.x)a)b)$ or $(\lambda x.xx)(\lambda x.xx)$

Redex families

Example



- 3 redex families: **red**, **blue**, **green**.

hRedexes

- **Definition** `[hRedex]`

  hRedex is a pair $\langle \rho, R \rangle$ where R is a redex in final term of ρ

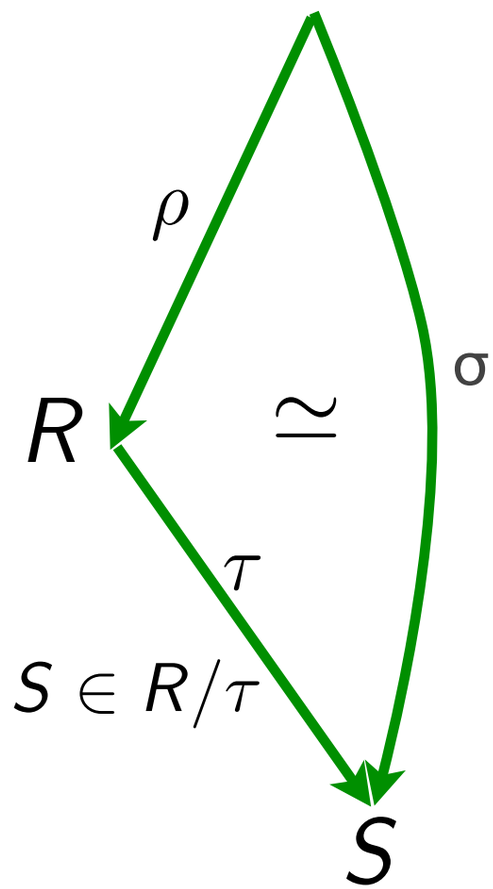
- **Definition** `[copies of hRedex]`

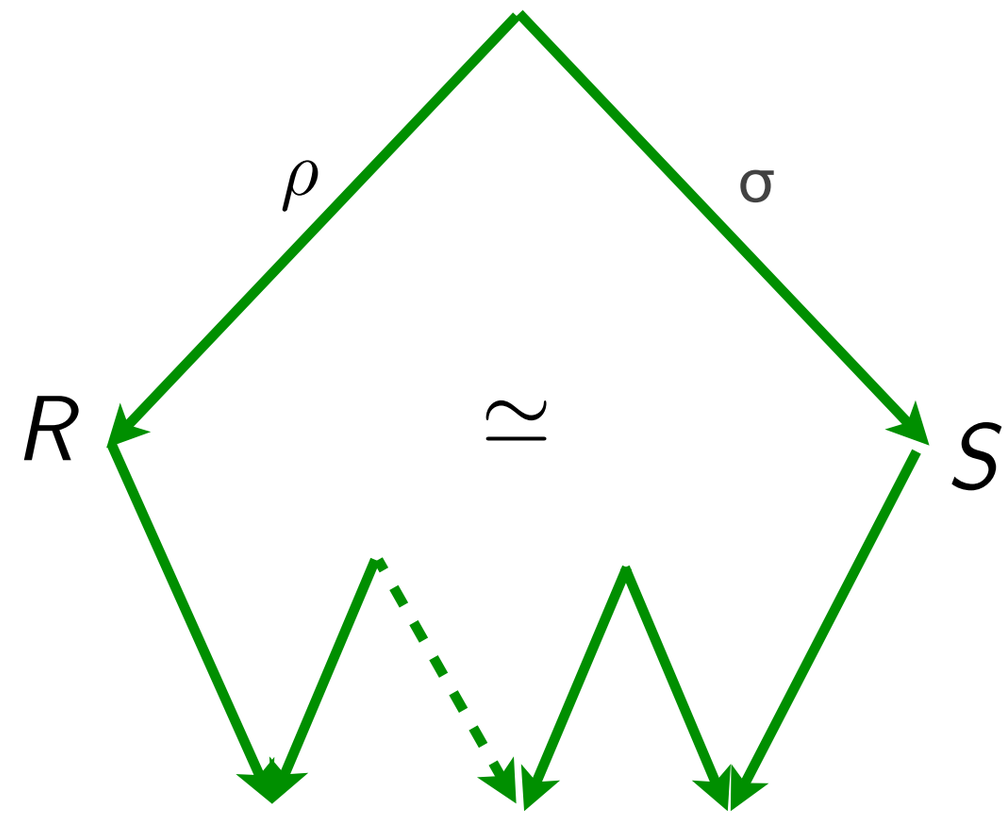
  $\langle \rho, R \rangle \leq \langle \sigma, S \rangle$ when $\exists \tau.\, \rho\tau \simeq \sigma$ and $S \in R/\tau$

- **Definition** `[families of hRedexes]`

  $\langle \rho, R \rangle \sim \langle \sigma, S \rangle$ for reflexive, symmetric, transitive closure of the copy relation.

Labels and history (1/4)





$\langle \rho, R \rangle \leq \langle \sigma, S \rangle$

$\langle \rho, R \rangle \sim \langle \sigma, S \rangle$

$\mathrm{name}(R) = \mathrm{name}(S)$

Labels and history (2/4)

- **Proposition** `[same history → same name]`

  In the labeled λ-calculus, for any labeling, we have:

  $\langle \rho, R \rangle \sim \langle \sigma, S \rangle$ implies $\mathsf{name}(R) = \mathsf{name}(S)$

- The opposite direction is clearly not true for any labeling

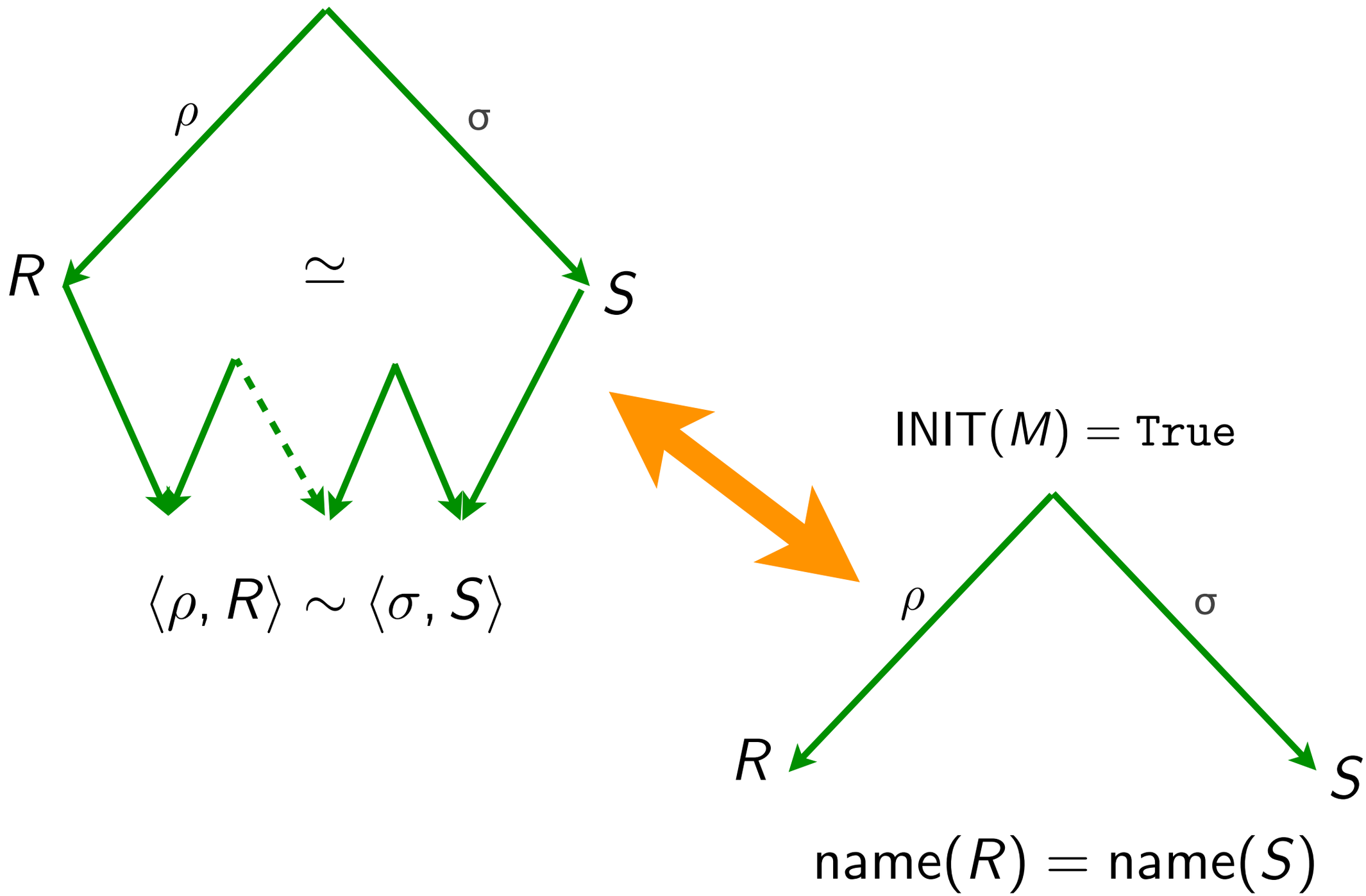
  (For instance, take all labels equal)

- But it is true when all labels are distinct atomic letters in the initial term.

- **Definition** `[all labels distinct letters]`

  $\mathsf{INIT}(M) = \mathtt{True}$ when all labels in M are distinct letters.

Labels and history (3/4)



Labels and history (4/4)

- **Theorem** `[same history = same name, 76]`

  When INIT(M) and reductions ρ and σ start from M :

  $\langle \rho, R \rangle \sim \langle \sigma, S \rangle$ iff $\mathrm{name}(R) = \mathrm{name}(S)$

- **Corollary** `[decidability of family relation]`

  The family relation is decidable (although complexity is proportional to length of standard reduction).

Finite developments

Parallel steps revisited (1/3)

- parallel steps were defined with inside-out strategy
  [à la Martin-Löf]

  Can we take any order as a reduction strategy ?

- **Definition** A **reduction relative** to a set $\mathcal{F}$ of redexes in M is any reduction contracting only residuals of $\mathcal{F}$.
  A **development** of $\mathcal{F}$ is any maximal relative reduction of $\mathcal{F}$.

Parallel steps revisited (2/3)

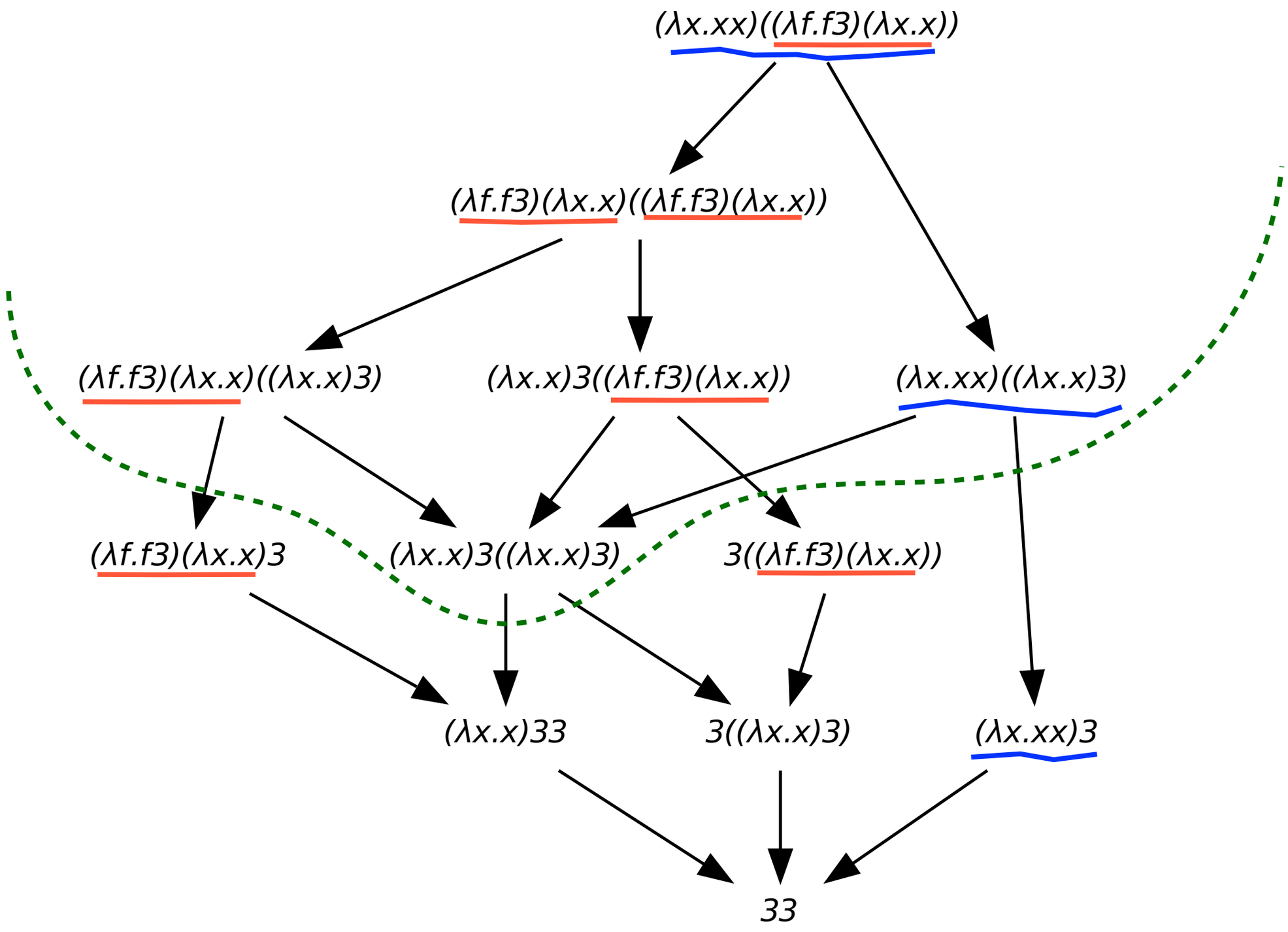
- **Theorem** `[Finite Developments, Curry, 50]`

  Let $\mathcal{F}$ be set of redexes in M.

  (1) there are no infinite relative reductions of $\mathcal{F}$,
  (2) they all finish on same term N
  (3) Let R be redex in M. Residuals of R by all finite developments of $\mathcal{F}$ are the same.

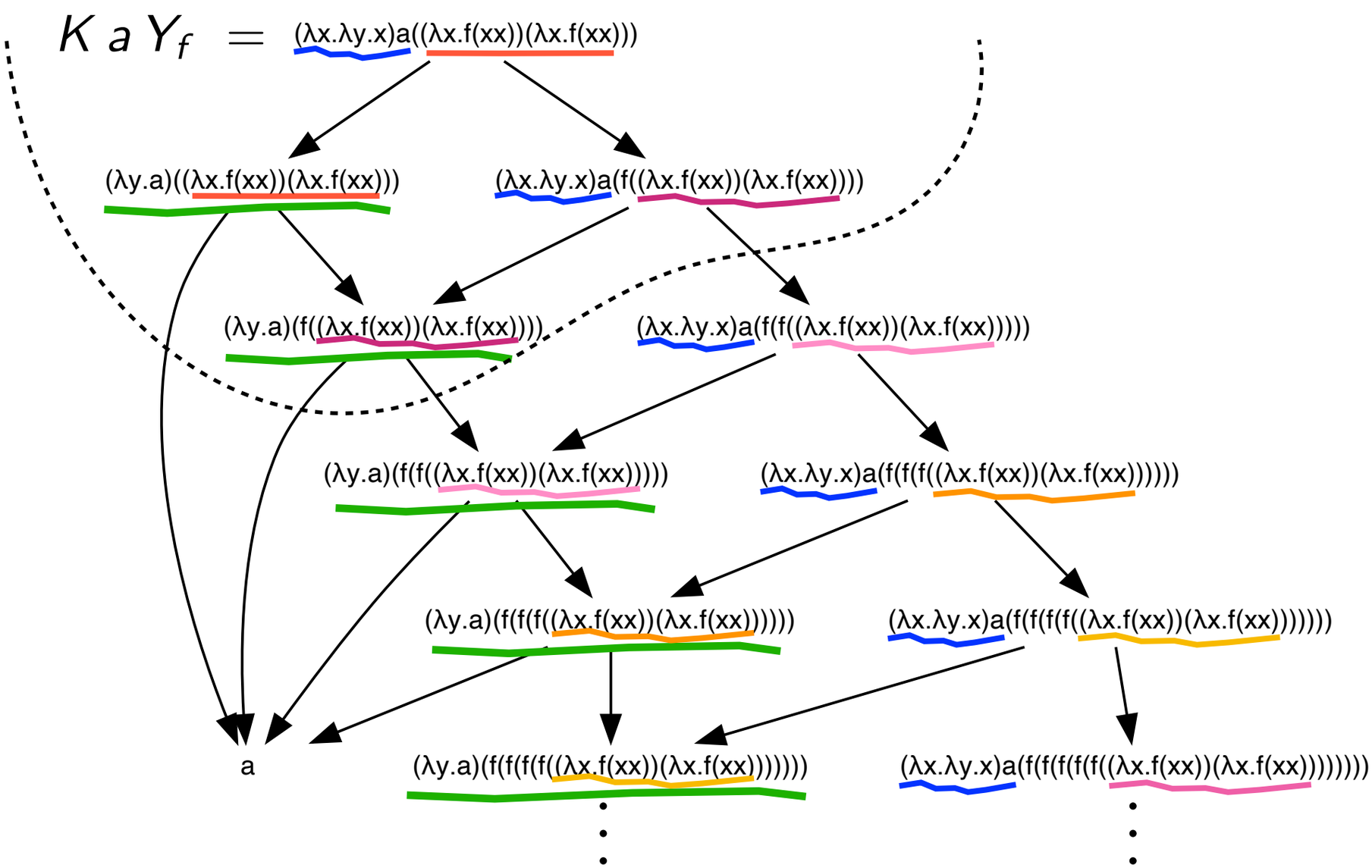
- Similar to the parallel moves lemma, but we considered a particular inside-out reduction strategy.

Example



developments of **red**, **blue**.

Example



developments of **red**, **blue**.

Parallel steps revisited (3/3)

- **Notation** `[parallel reduction steps]`
  Let $\mathcal{F}$ be set of redexes in M. We write $M \xrightarrow{\mathcal{F}} N$
  if a development of $\mathcal{F}$ connects M to N.

- This notation is consistent with previous definition
  (since inside-out parallel step is a particular development)

- Corollaries of FD thm are also parallel moves + cube lemmas

Finite and infinite reductions (1/3)

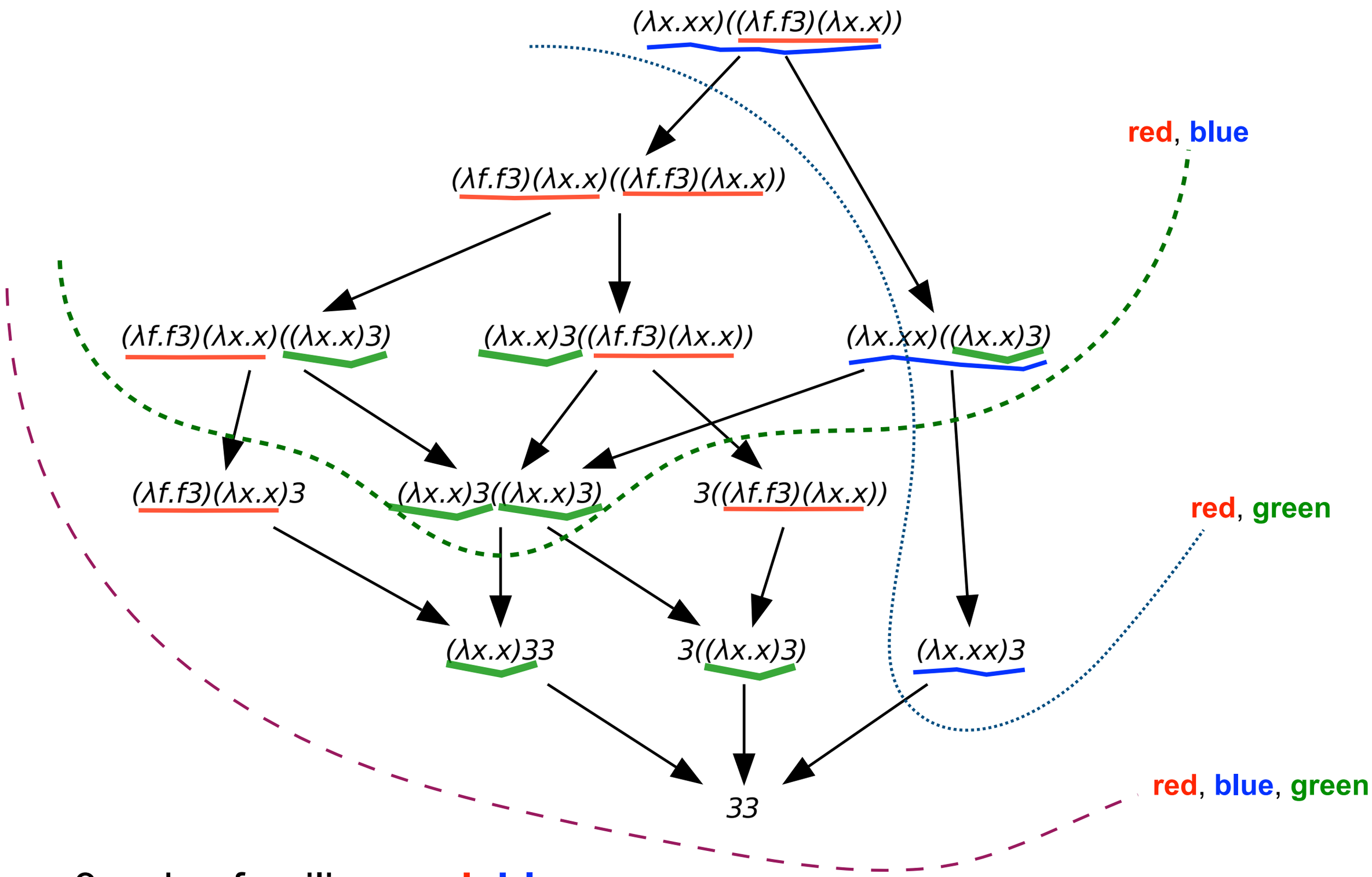
- **Definition** A **reduction relative** to a set $\mathcal{F}$ of redex families is any reduction contracting redexes in families of $\mathcal{F}$.

  A **development** of $\mathcal{F}$ is any maximal relative reduction.

- **Theorem** [Generalized Finite Developments+, 76]
  Let $\mathcal{F}$ be a finite set of redex families.

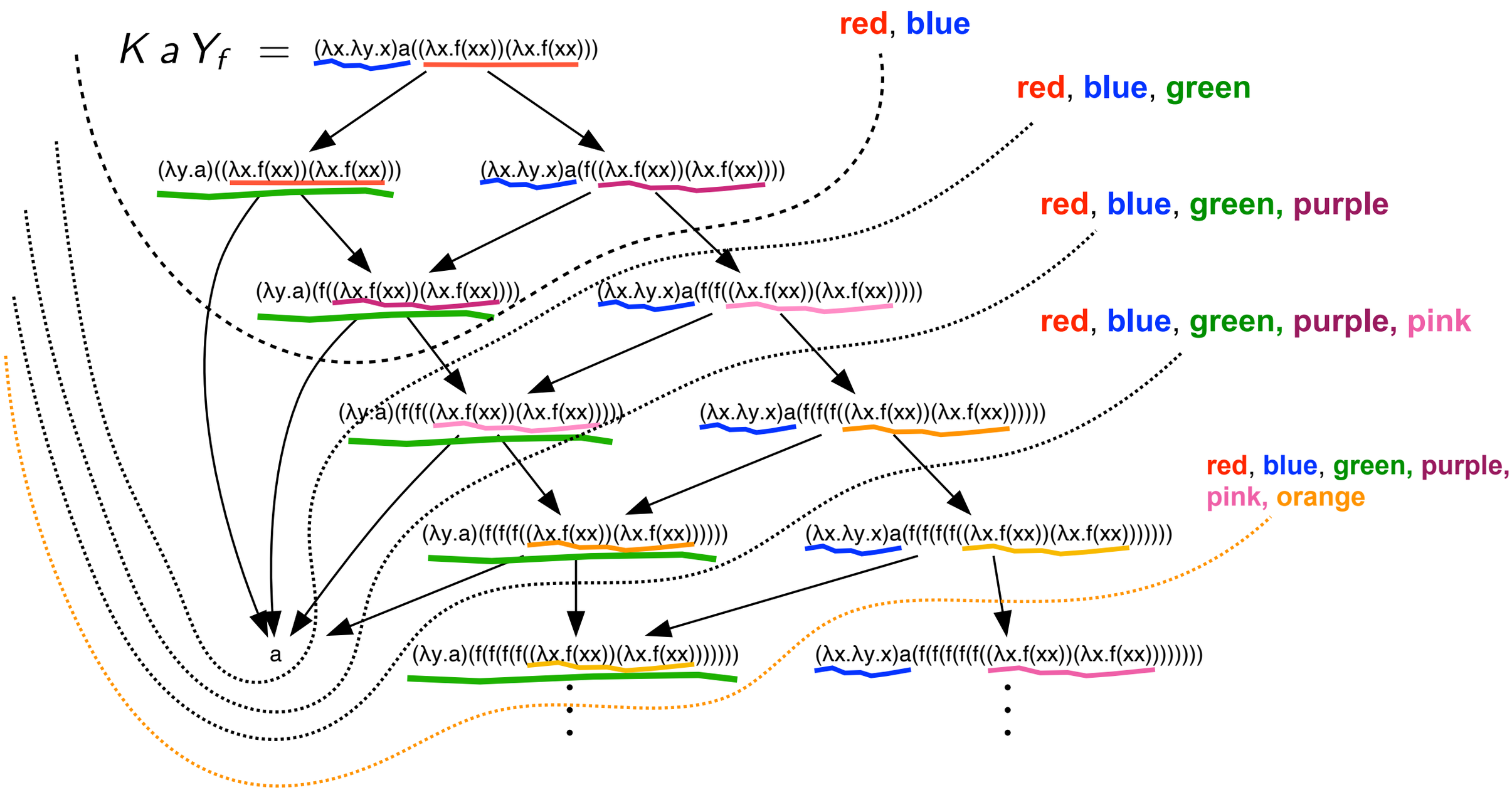
  (1) there are no infinite reductions relative to $\mathcal{F}$,
  (2) they all finish on same term N
  (3) All developments are equivalent by permutations.

Example



- 3 redex families: **red**, **blue**, **green**.

Example



developments of families.

Finite and infinite reductions (2/3)

- **Corollary** An **infinite reduction** contracts an **infinite set of redex families.**

- **Corollary** Any term generating a finite number of redex families strongly normalizes

finite number of redex families

strong normalization

Strong normalization

1st-order typed λ-calculus (1/2)

Residuals of redexes keep their types (of names)

Created redexes have lower types

$$\underbrace{(\lambda x.\cdots xN\cdots)}_{s\to t}\underbrace{(\lambda y.M)}_{s} \to \cdots\underbrace{(\lambda y.M)}_{s}N'\cdots$$

creates

Finite number of redexes families

$$\underbrace{(\lambda x.\underbrace{\lambda y.M}_{t})}_{s\to t}NP \to \underbrace{(\lambda y.M')}_{t}P$$

creates

$$\underbrace{(\lambda x.x)}_{s\to s}\underbrace{(\lambda y.M)}_{s}N \to \underbrace{(\lambda y.M)}_{s}N$$

creates

Strong normalization

1st-order typed λ-calculus (2/2)

- **Typed λ-calculus** as a specific labeled calculus

$$s,\ t\ ::=\ \mathbb{N}, \mathbb{B}\ \mid\ s \to t$$

Decorate subterms with their types

$$(\lambda f.\,(f^{\mathbb{N}\to\mathbb{N}} 3^{\mathbb{N}})^{\mathbb{N}})^{(\mathbb{N}\to\mathbb{N})\to\mathbb{N}}\, I^{\mathbb{N}\to\mathbb{N}}$$

$$(I^{\mathbb{N}\to\mathbb{N}} 3^{\mathbb{N}})^{\mathbb{N}} \longrightarrow 3^{\mathbb{N}}$$

Apply following rules to labeled λ-calculus

$$\lceil s \to t \rceil = t$$

$$\lfloor s \to t \rfloor = s$$

$$s\,t = s$$

Strong normalization (1/2)

- Another labeled λ-calculus was considered to study Scott D-infinity model [Hyland-Wadsworth, 74]

- D-infinity projection functions on each subterm (n is any integer):

$$M, N, \ldots ::= x^n \mid (MN)^n \mid (\lambda x.M)^n$$

- Conversion rule is:

$$((\lambda x.M)^{n+1} N)^p \longrightarrow M\{x := N_{[n]}\}_{[n][p]}$$

$n+1$ is **degree** of redex

$$U_{[m][n]} = U_{[p]} \quad \text{where} \quad p = \min\{m, n\}$$

$$x^n \{x := M\} = M_{[n]}$$

Strong normalization (2/2)

- **Proposition** Hyland-Wadsworth calculus is derivable from labeled calculus by simple homomorphism on labels.

  Proof Assign an integer to any atomic letter and take:

$$\mathsf{h}(\alpha\beta) = \min\{\mathsf{h}(\alpha),\ \mathsf{h}(\beta)\}$$

$$\mathsf{h}(\lceil\alpha\rceil) = \mathsf{h}(\lfloor\alpha\rfloor) = \mathsf{h}(\alpha) - 1$$

- Redex degrees are bounded by maximum of labels in initial term. therefore a finite number of redex families

- **Proposition** Hyland-Wadsworth calculus strongly normalizes.