Sequentiality in Kahn-Macqueen nets and the λ-calculus

Jean-Jacques Lévy

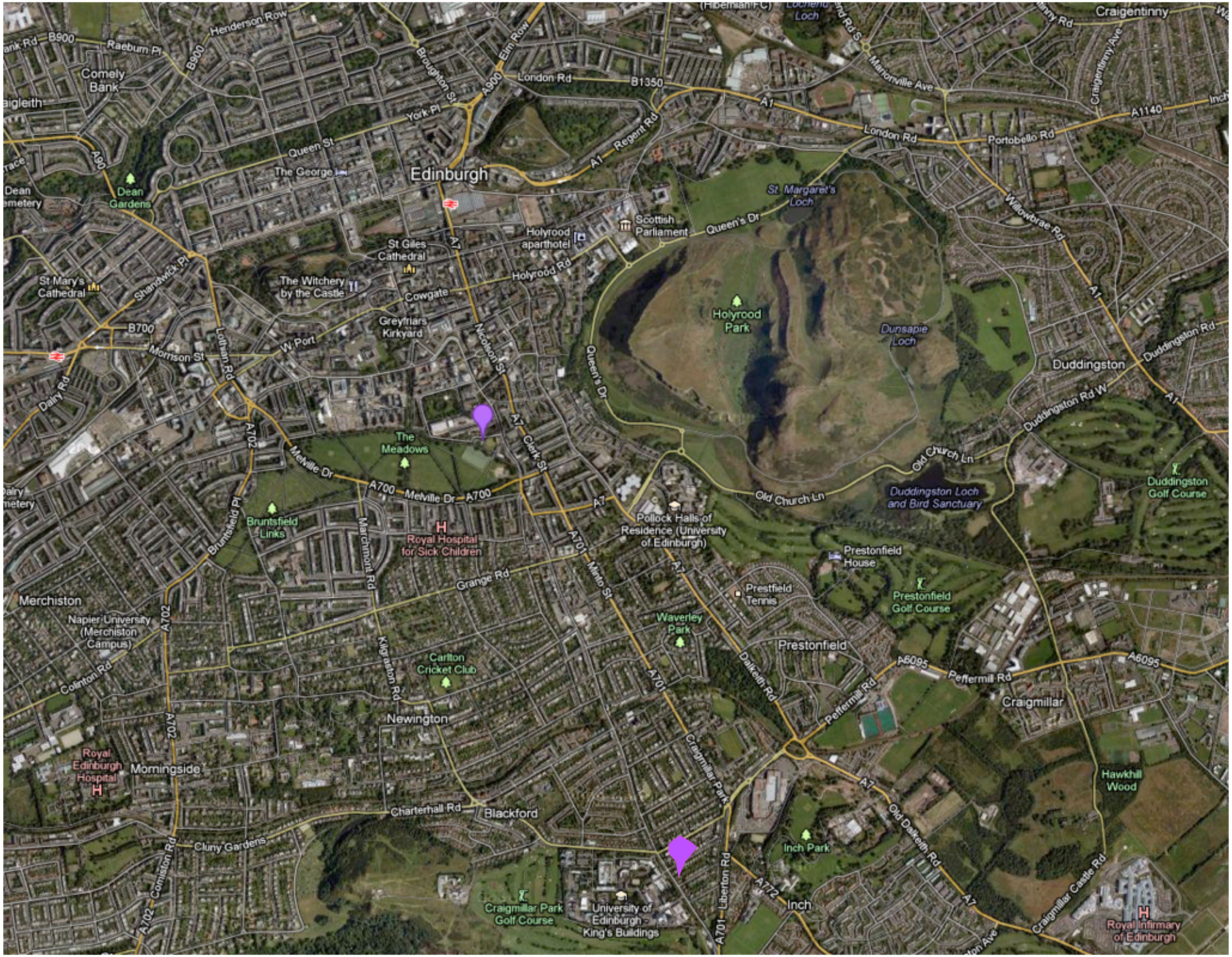
Macqueen Fest, 11-05-2012

CENTRE DE RECHERCHE COMMUN
INRIA MICROSOFT RESEARCH

Plan

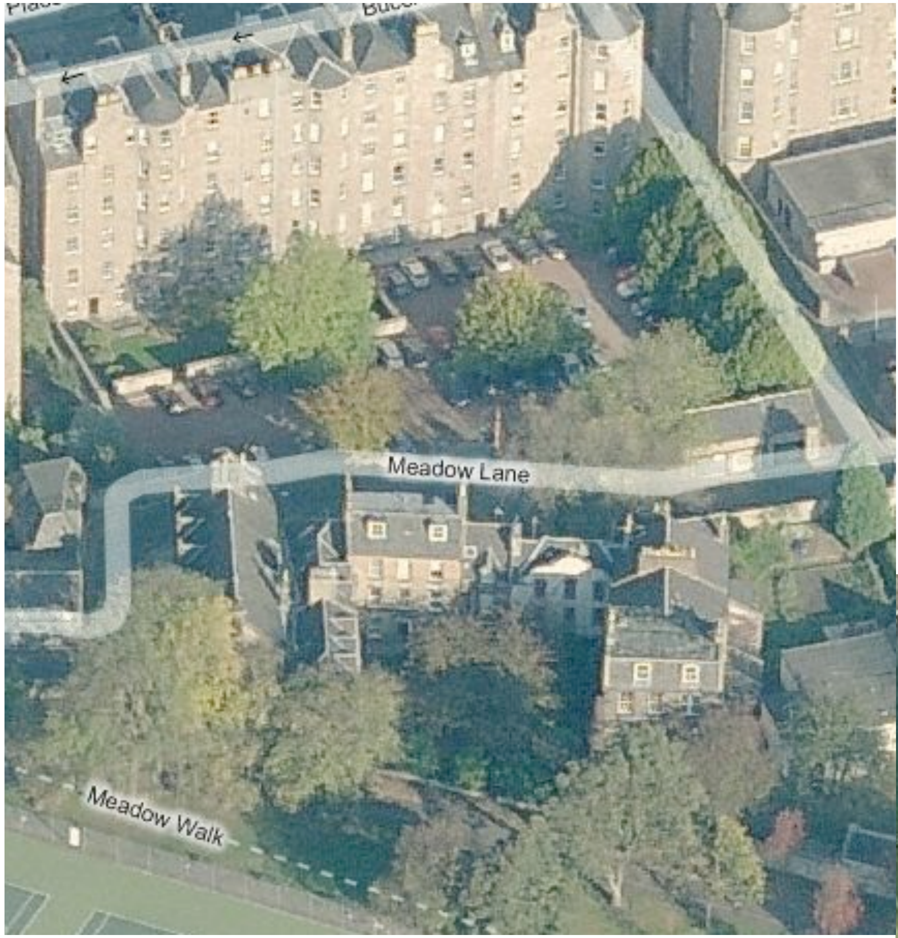
- Kahn-Macqueen networks
- Stability in the λ-calculus
- Stability in Kahn-Macqueen networks
- Revisiting stability in dynamics of the λ-calculus
- Sequentiality
- Application to Kahn-Macqueen networks

Edinburgh in 70's



Edinburgh in 70's

Hope Park Square



Scientific visits

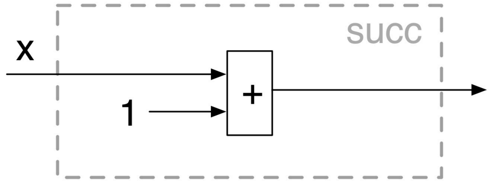
Kahn-Macqueen networks (0/4)

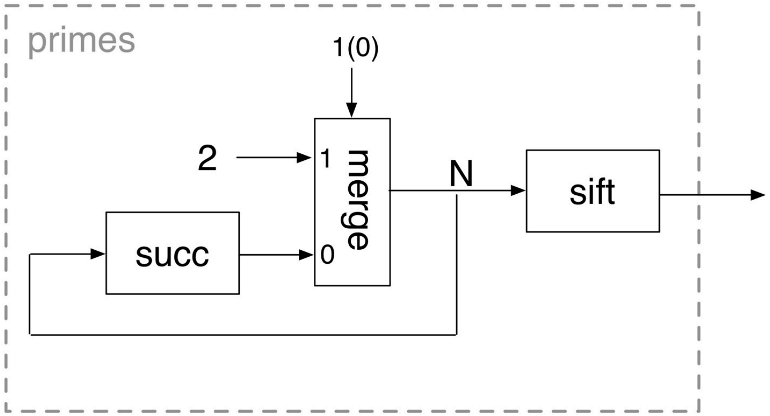
- sieve of Eratosthenes in POP-2 [GK, DBM 77]

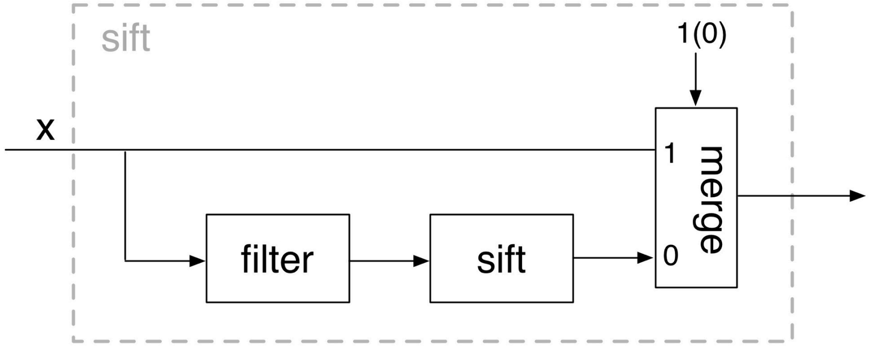
```
Process INTEGERS out QO;
    Vars N; 1 → N;
    repeat INCREMENT N; PUT(N,QO) forever
Endprocess;

Process FILTER PRIME in QI out QO;
    Vars N;
    repeat GET(QI) → N;
           if (N MOD PRIME)≠ Ø then PUT(N,QO) close
    forever
Endprocess;

Process SIFT in QI out QO;
    Vars PRIME; GET(QI) → PRIME;
    PUT (PRIME,QO); comment emit a discovered prime;
    doco channels Q;
     FILTER(PRIME,QI,Q); SIFT(Q,QO)
    closeco
Endprocess;

Process OUTPUT in QI; Comment this is a library process;
    repeat PRINT(GET(QI)) forever
Endprocess;

Start doco channels Q1 Q2;
    INTEGERS(Q1); SIFT(Q1,Q2); OUTPUT(Q2);
    closeco;
```

Fig.3. Sieve of Eratosthenes.

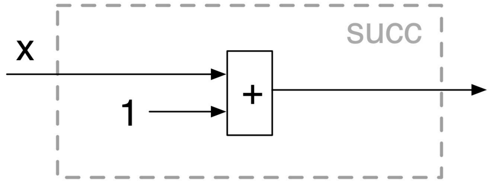
Kahn-Macqueen networks (1/4)

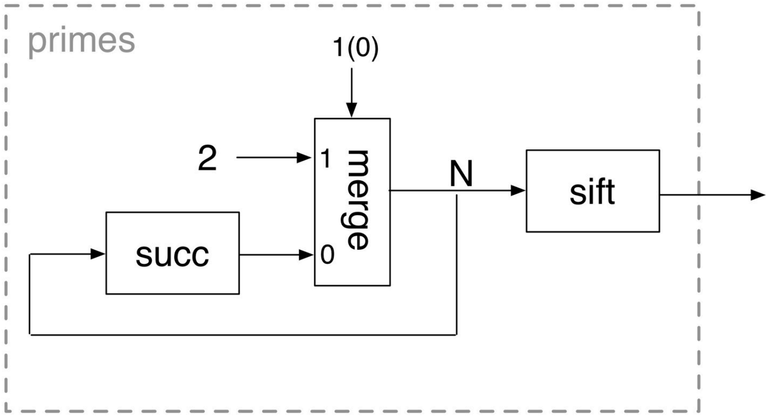


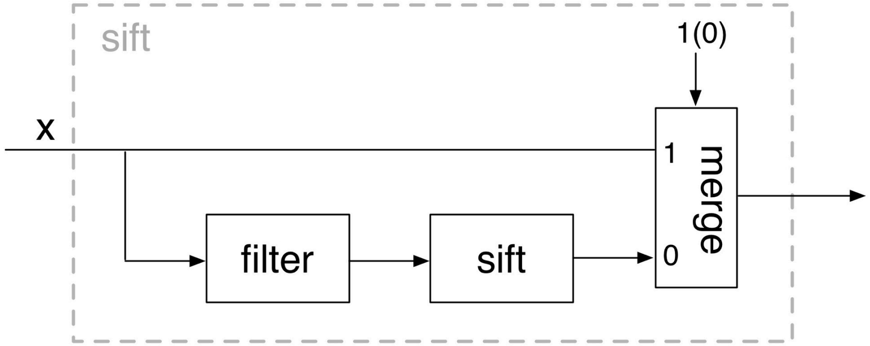




Kahn-Macqueen networks (2/4)








```
succ (x :: xs) := (x+1) :: succ xs

N := 2 :: succ N
primes := sift N


sift (x :: xs) := x :: sift (filter (x :: xs))


filter (x :: xs) := not_mult x xs
not_mult x (y :: ys) :=
  if y mod x = 0 then not_mult x ys
  else x :: (not_mult x ys)
```

Kahn-Macqueen networks (3/4)

- recursive equations on **flow histories**
- deterministic results (**determinate**)
- problem with «**fair merge**»

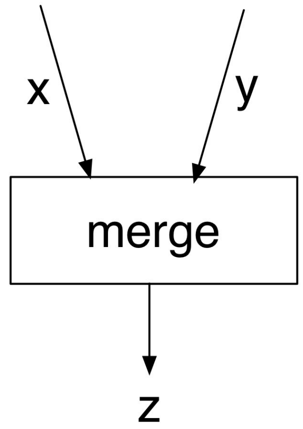
$$
\begin{aligned}
\mathrm{fmerge}(\mathrm{xs}, \epsilon) &= \{\mathrm{xs}\} \\
\mathrm{fmerge}(\epsilon, \mathrm{ys}) &= \{\mathrm{ys}\} \\
\mathrm{fmerge}(x :: \mathrm{xs}, y :: \mathrm{ys}) &= \{x :: \mathrm{zs} \mid \mathrm{zs} \in \mathrm{fmerge}(\mathrm{xs}, y :: \mathrm{ys})\} \\
&\cup \{y :: \mathrm{zs} \mid \mathrm{zs} \in \mathrm{fmerge}(x :: \mathrm{xs}, \mathrm{ys})\}
\end{aligned}
$$

- equality of traces is **not compositional** [Brock,Ackerman 81]
- powerdomain semantics, process calculi + bisimulations

[Plotkin 78] [Milner et al 78]

Kahn-Macqueen networks (4/4)

- «merge» **blocks** on its arguments x or y
- since «merge» is **sequential**
- «fair merge» is not sequential like **parallel-or**



$$\text{por}(\text{true}, x) = \text{true}$$
$$\text{por}(x, \text{true}) = \text{true}$$

meaning

$$\text{por}(\text{true}, \bot) = \text{por}(\bot, \text{true}) = \text{true}$$

$$\text{por}(\bot, \bot) = \bot$$

Sequentiality

Scott's semantics - 1st order
strict functions [Cadiou, 71]
alternative def [Vuillemin, 72]

PCF sequential [Plotkin, 75]

stable functions [Berry, 75]

concrete domains [Kahn-Plotkin, 76?]

CDS [Berry-Curien, 79]

⋮

fully abstract models [Abramsky et al, 93]

Stability

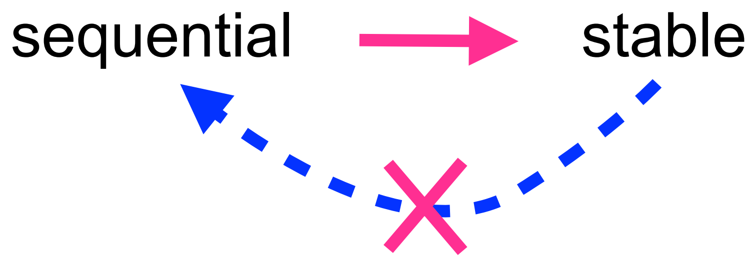
- f **stable** function iff $x \uparrow y \Rightarrow f(x \sqcap y) = f(x) \sqcap f(y)$

- **por** is not stable :

$$\bot = \mathsf{por}(\bot, \bot) \neq \mathsf{por}(\bot, \mathsf{true}) \sqcap \mathsf{por}(\mathsf{true}, \bot) = \mathsf{true}$$

- semantics of (strongly) stable functions

- with strange Berry's function



Stability inside calculi

- PCF [Plotkin,75]

$$M, N, P ::= x \mid \lambda x.M \mid MN \mid \underline{n} \mid M \oplus N \mid \texttt{ifz}\, P \,\texttt{then}\, M \,\texttt{else}\, N$$

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

$$\underline{m} \oplus \underline{n} \longrightarrow \underline{m+n}$$

$$\texttt{ifz}\ \underline{0}\ \texttt{then}\ M\ \texttt{else}\ N \longrightarrow M$$

$$\texttt{ifz}\ \underline{n+1}\ \texttt{then}\ M\ \texttt{else}\ N \longrightarrow N$$

- PCF cannot express por.

Stability inside the λ-calculus (1/3)

$$M, N ::= x \mid \lambda x.M \mid MN$$

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

- Impossible to get:

$$C[\Omega, \Omega] \not\longrightarrow^{*} \text{nf}$$

$$C[\Omega, \lambda x.x] \longrightarrow^{*} \text{nf}$$

$$C[\lambda x.x, \Omega] \longrightarrow^{*} \text{nf}$$

Lemma «has a nf» is a stable function.

Stability inside the λ-calculus (2/3)

$$M, N ::= x \mid \lambda x.M \mid MN$$

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

- Impossible to get:

$$C[\Omega, \Omega] \not\xrightarrow{\star} \text{hnf}$$

$$C[\Omega, H'] \xrightarrow{\star} \text{hnf}$$

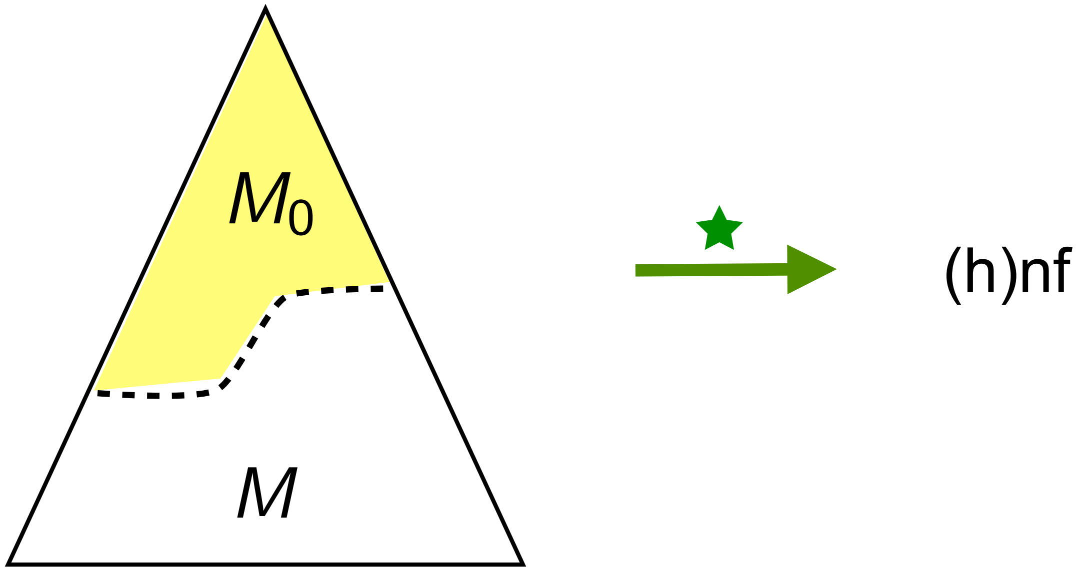
$$C[H, \Omega] \xrightarrow{\star} \text{hnf} \qquad (H,\ H' \text{ with hnf})$$

Lemma «has a hnf» is a stable function.

Lemma «Bohm tree» is a stable function.

Stability inside the λ-calculus (3/3)

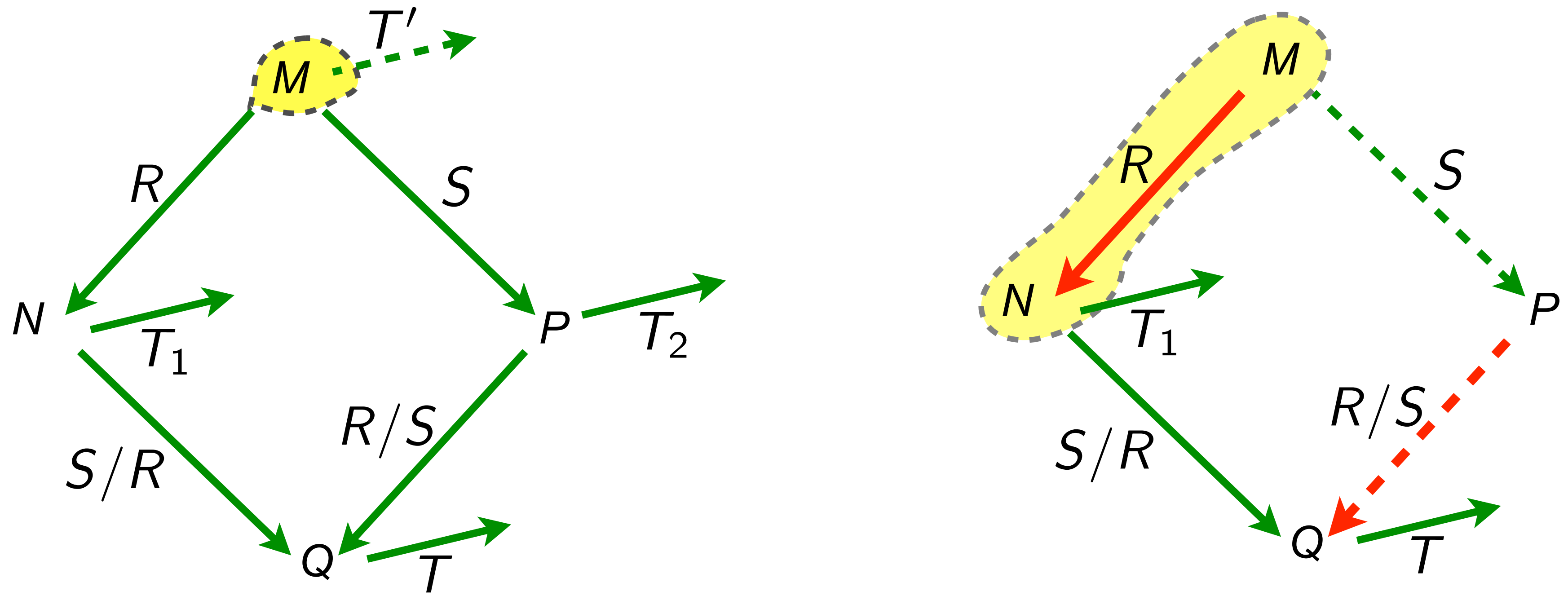
Lemma Let $M \xrightarrow{\star} $ (h)nf, then there is a unique minimum prefix M_0 of M such that $M_0 \xrightarrow{\star}$ (h)nf.



Stability inside redexes (1/2)

Lemma [stability of redex creation] When $R \neq S$,

$$T \in T_1/(S/R) \text{ and } T \in T_2/(R/S) \text{ implies } T \in T'/(R \sqcup S)$$



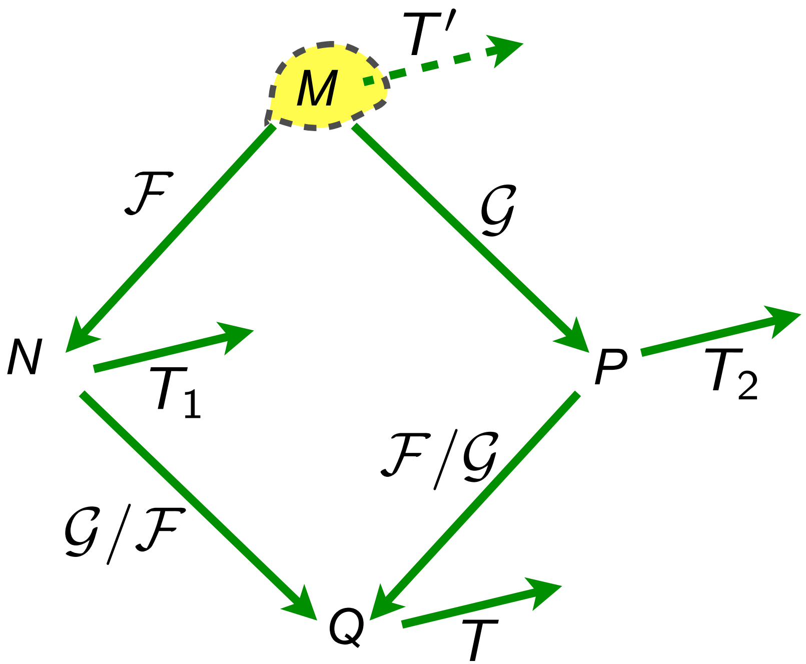
Corollary When $R \neq S$,

If $T \in T_1/(S/R)$ and R creates T_1, then $\exists R' \in R/S,\ \ R'$ creates T.

Stability inside redexes (2/2)

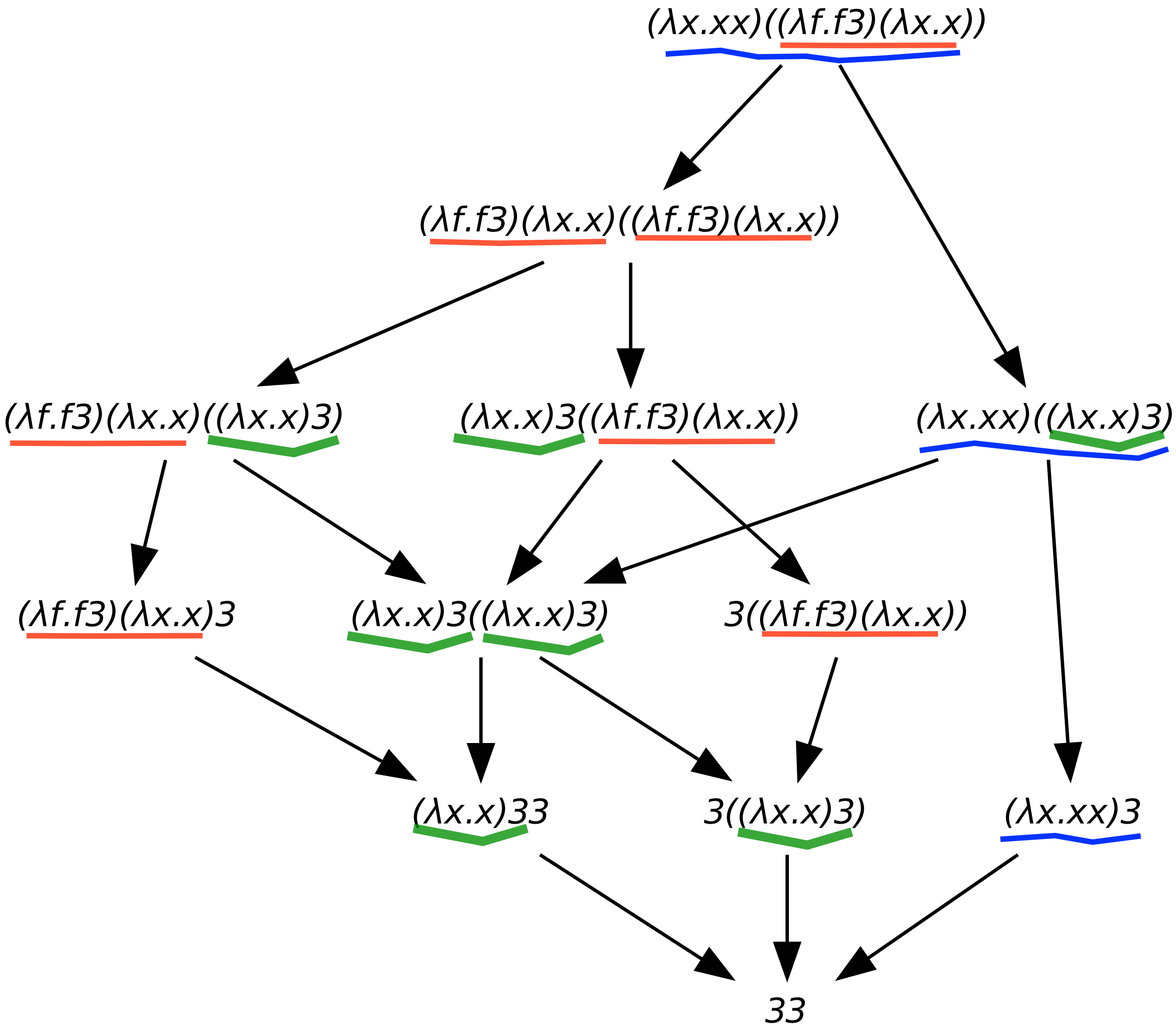
Lemma [stability of redex creation] When $\mathcal{F} \cap \mathcal{G} = \emptyset$,

$$T \in T_1/(\mathcal{G}/\mathcal{F}) \text{ and } T \in T_2/(\mathcal{F}/\mathcal{G}) \text{ implies } T \in T'/(\mathcal{F} \sqcup \mathcal{G})$$



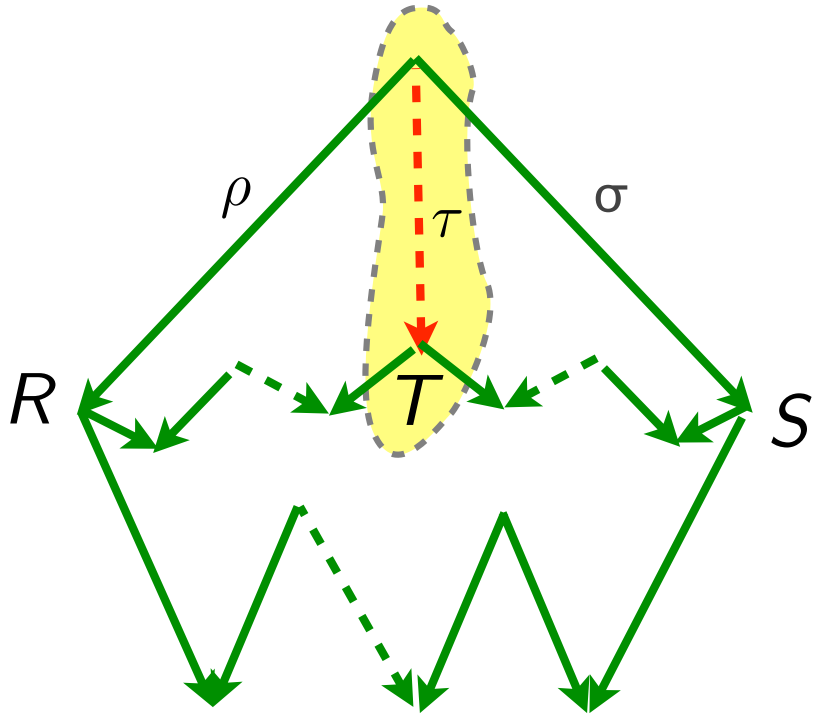
Corollary When $\mathcal{F} \cap \mathcal{G} = \emptyset$, if $\mathcal{F}$ creates T, then $\mathcal{G}/\mathcal{F}$ creates $T/(\mathcal{G}/\mathcal{F})$.

Redex families



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

Redexes and their unique origin

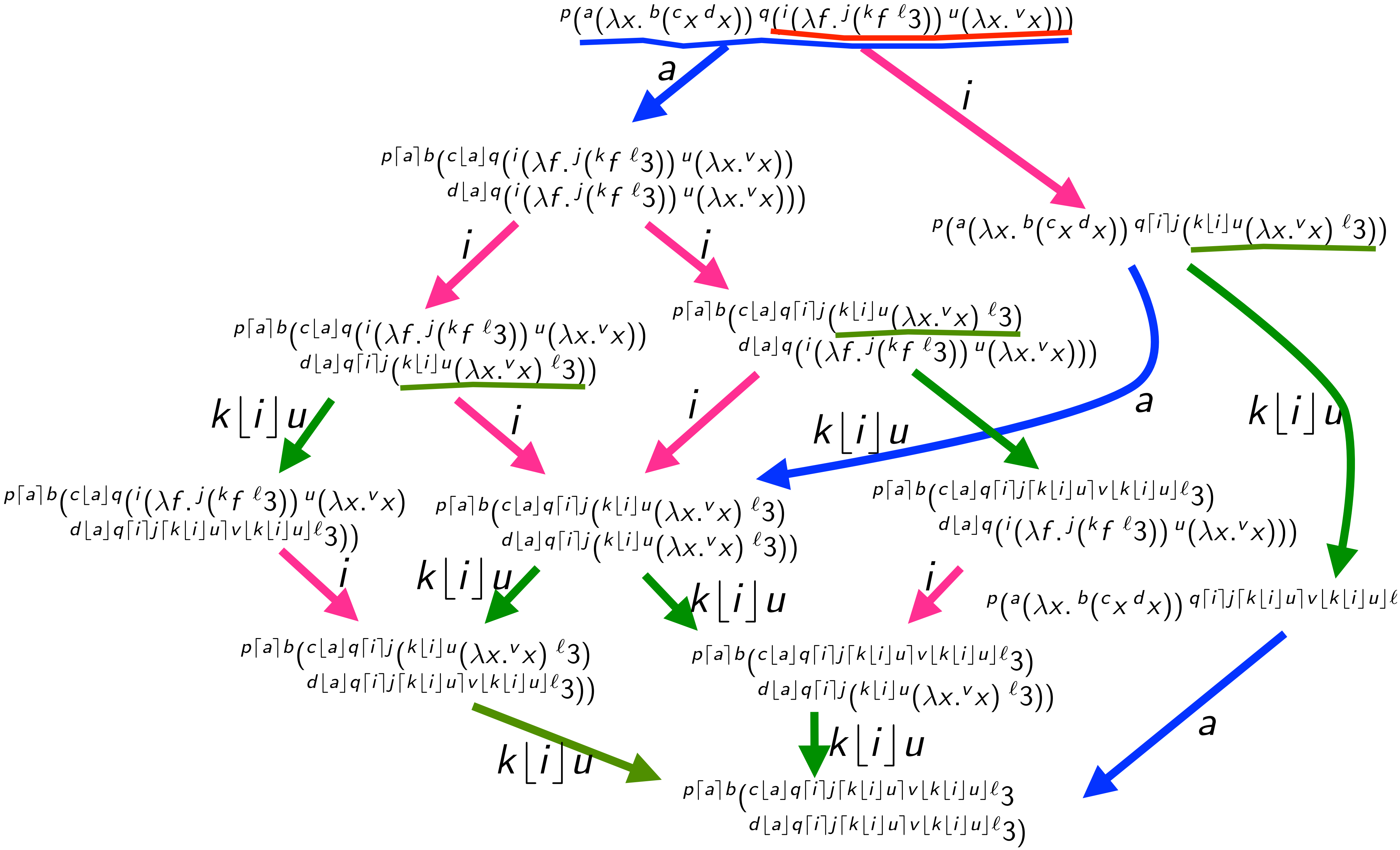


Proposition

There is a unique $\langle \tau, T \rangle$ with τ standard reduction of minimum length in each redex family.

Redex families

3 families and their names: a i $k\lfloor i \rfloor u$

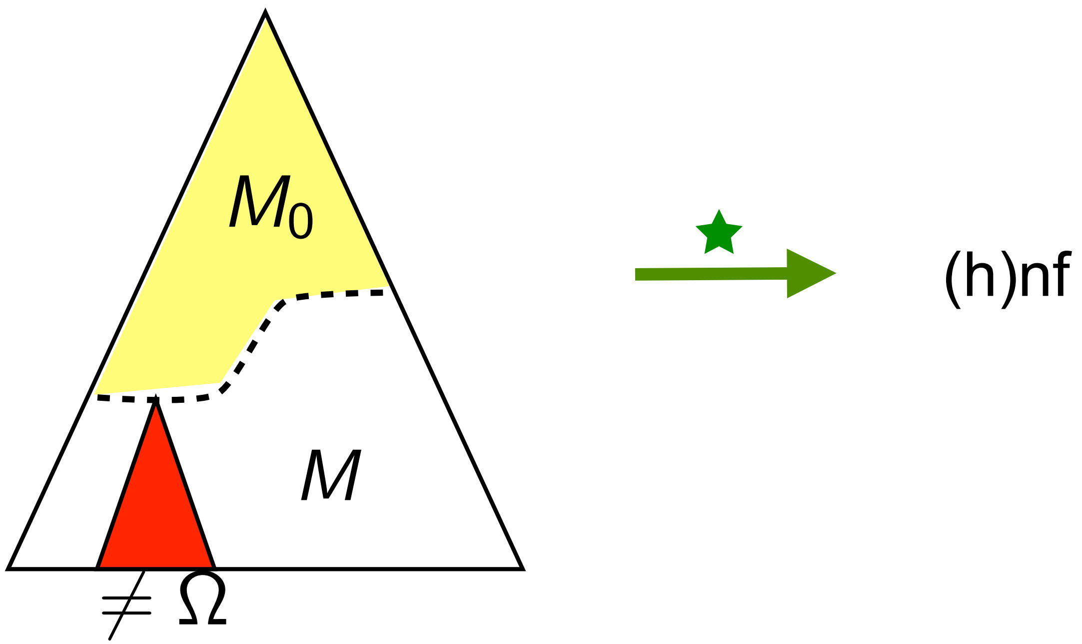


Stability in Kahn-Macqueen nets

- Equations on history flows are left-linear orthogonal TRS
- Stability for prefixes `[Huet, JJL, 81; Klop 90]`
- Stability inside their redexes `[Maranget 91]`

Sequentiality (1/2)

Lemma Let $M_0 \not\xrightarrow{*}$ (h)nf, then there is an Ω occurrence such that you cannot get a (h)nf without strictly increasing it.



Sequentiality (2/2)

- «Bohm-tree» is a sequential function [Berry, JJL, 78]

  $C[\Omega, \Omega] \not\xrightarrow{*} \text{nf}$

  $C[M, N] \xrightarrow{*} \text{nf}$ for some M and N

  one of the Ω's is such that $C[\Omega, N] \not\xrightarrow{*} \text{nf}$ for all N

- Theory of strongly sequential TRS
  [Huet, JJL, 81, Klop 90]

- Call by need calculations for Kahn-Macqueen nets

Todo-list

- From strongly sequential TRS to Kahn-Macqueen networks
- Theory of sequentiality for redexes
- Need to work with subcontexts ?

Enjoy retirement Dave!

CENTRE DE RECHERCHE
COMMUN

INRIA
MICROSOFT RESEARCH