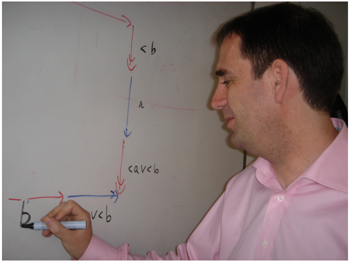
hogere orde en eerste orde

ter ere van Vincent van Oostrom (afscheid van de UU)



Clemens Grabmayer

VU Amsterdam, Informatica

Universiteit Utrecht

2 July 2014

overzicht

- wat ik van Vincent heb geleerd, en blijf leren:
  - hogere-orde herschrijven
  - optimale reductie in de λ-calculus (Lambdascope)
  - hogere-orde naar eerste-orde vertaling:
    λ-calculus naar orthogonale TRS
- toepassingen:
  - maximal sharing (met Jan Rochel)
  - geneste termgrafen (met Vincent)
- andere leermomenten:
  - paradoxaal interveniëren
  - diep
  - gewoon
  - het Vincents
  - moeten?
  - university management
  - voetbal

tijdlijn

onderzoeksprojecten:

2007–2009 NWO project *Infinity* (UU/VU/CWI)

2009–2012 NWO proj. *Realising Optimal Sharing* (UU–Phil/UU–CS)

- VvO (projectleider)
- Doaitse Swierstra
- Jan Rochel, CG

onderwijs samen met Vincent:

- datastructuren (bachelor CKI)
- models of computation (master CAI)

paradoxaal interveniëren (V)

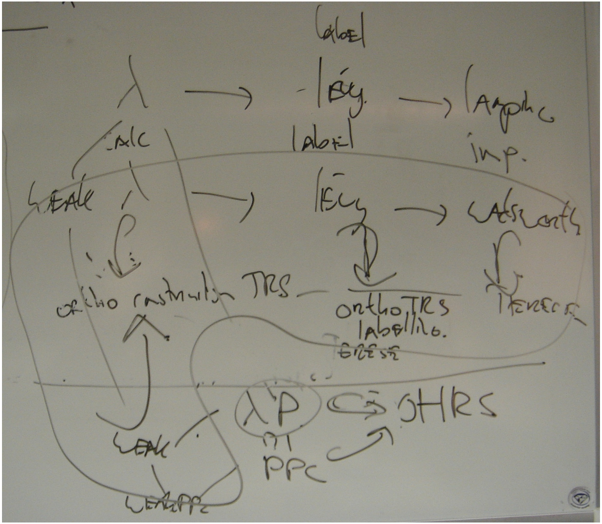
- een ‘typische Vincent opmerking’
  - volgt vaak letterlijke betekenissen van woorden, of bijbetekenissen
  - gebruikt betekenissen waar anderen helemaal niet aan denken
  - gevolgd door: ‘oooh, je bedoelde eigenlijk ...’, ‘zeg dat dan’
  - is speels, maar verruimt (in een oopopslag) het referentiekader
  - en leidt daarmee tot gezamenlijke nieuwe energie, soms ook inzichten

paradoxaal interveniëren (psychotherapie)

Alfred Adler (1870–1937)

- in gezinstherapie
- doel van de therapeut: weerstand van de client elimineren
- therapeut
  - neemt letterlijk wat de client zegt
  - bemoedigt de client om door te gaan met het symptomatisch gedrag
  - suggereert het tegenovergestelde van wat logica/rede zou ingeven
- wordt op ethisch-verantwoorde manier toegepast
- de onredelijk lijkende suggestie dwingt de client om na te denken
- wat ook de gekozen reactie is, de client beseft
  dat zij/hij meer controle heeft dan gedacht

gradaties van optimale reductie [VvO, plaatje 2008/10]



gradaties van optimale reductie

calculus (rewrite relation)	labelling	graph rewriting implementation	sharing notion
λ-calculus (β-reduction $\to_\beta$)	Lévy labelling '78	Lamping '89 Kathail '90 Abdadi/Gonthier/ /Levy '92 Asperti/Guerrini '93 VvO '03 (Terese)	context sharing
λ-calculus (weak-β red. $\to_{w\beta}$)	Blanc/Lévy/ Maranget '05/'07	Wadsworth '71 Shivers/Wand '04	extended-scope sharing
orthogonal TRS (induced rewrite relation $\to$)	VvO '03 (Terese)	Staples '80 VvO '03 (Terese)	subterm sharing

hogere orde termherschrijven

Vincent en Femke van Raamsdonk

hogere orde termherschrijven

Yuri Gurevich

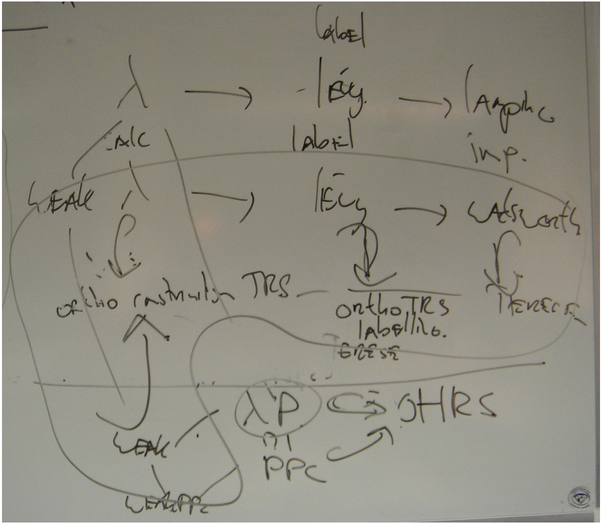
hogere orde termherschrijven

herschrijven: 'equationeel redeneren, één kant op, naar het antwoord toe'

hogere-orde herschrijven: herschr. van syntactische objecten met binding

- prototypische voorbeeld: λ-calculus ($\boldsymbol{\lambda}$)
- verschillende formaten zoals:
  - Combinatory Reduction Systems (CRSs) [Klop]
  - Higher-Order Rewrite Systems (HRSs) [Nipkow]
- HRS-introductie voor filosofen geschreven door Vincent in:

  G, Joop Leo, Vincent van Oostrom, Albert Visser: *On the termination of Russell's description elimination algorithm*, RSL'2011.


label
λ
label
λ
TRS
ortho TRS
labelling.
λP
PPC

gradaties van optimale reductie

calculus (rewrite relation)	labelling	graph rewriting implementation	sharing notion
λ-calculus (β-reduction $\rightarrow_\beta$)	Lévy labelling '78	Lamping '89 Kathail '90 Abdadi/Gonthier/ /Levy '92 Asperti/Guerrini '93 VvO '03	context sharing
λ-calculus (weak-β red. $\rightarrow_{w\beta}$)	Blanc/Lévy/ Maranget '05/'07	Wadsworth '71 Shivers/Wand '04	extended-scope sharing
orthogonal TRS (induced rewrite relation $\rightarrow$)	VvO '03 (Terese)	Staples '80 VvO '03 (Terese)	subterm sharing

gradaties van optimale reductie

optimale evaluatie in λ-calculus vermijdt:

- onnodig werk
- dubbel werk

calculus (rewrite relation)	labelling	graph rewriting implementation	sharing notion
λ-calculus (β-reduction $\to_\beta$)	Lévy labelling '78	Lamping '89 Kathail '90 Abdadi/Gonthier/ /Levy '92 Asperti/Guerrini '93 VvO '03	context sharing
λ-calculus (weak-β red. $\to_{w\beta}$)	Blanc/Lévy/ Maranget '05/'07	Wadsworth '71 Shivers/Wand '04	extended-scope sharing
orthogonal TRS (induced rewrite relation $\to$)	VvO '03 (Terese)	Staples '80 VvO '03 (Terese)	subterm sharing

gradaties van optimale reductie

optimale evaluatie in λ-calculus vermijdt:

- onnodig werk
- dubbel werk

calculus (rewrite relation)	labelling	graph rewriting implementation	sharing notion
λ-calculus (β-reduction $\rightarrow_\beta$)	Lévy labelling '78	Lamping '89 Kathail '90 Abdadi/Gonthier/ /Levy '92 Asperti/Guerrini '93 VvO '03	context sharing

gradaties van optimale reductie

optimale evaluatie in λ-calculus vermijdt:

- onnodig werk
- dubbel werk

calculus (rewrite relation)	labelling	graph rewriting implementation	sharing notion
λ-calculus (β-reduction $\to_\beta$)	Lévy labelling '78	Lamping '89 Kathail '90 Abdadi/Gonthier/ /Levy '92 Asperti/Guerrini '93 VvO '03	context sharing

gradaties van optimale reductie

optimale evaluatie in λ-calculus vermijdt:

- onnodig werk
- dubbel werk

calculus (rewrite relation)	labelling	graph rewriting implementation	sharing notion
λ-calculus (β-reduction $\rightarrow_\beta$)	Lévy labelling '78	Lamping '89 Kathail '90 Abdadi/Gonthier/ /Levy '92 Asperti/Guerrini '93 VvO '03	context sharing

gradaties van optimale reductie

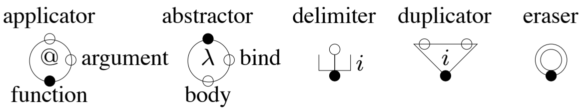
optimale evaluatie in λ-calculus vermijdt:

- onnodig werk
- dubbel werk

calculus (rewrite relation)	labelling	graph rewriting implementation	sharing notion
λ-calculus (β-reduction $\to_\beta$)	Lévy labelling '78	Lamping '89 Kathail '90 Abdadi/Gonthier/ /Levy '92 Asperti/Guerrini '93 VvO '03	context sharing

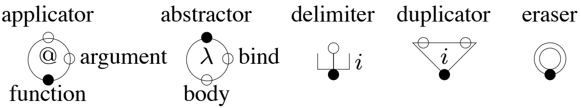
Lambdascope [VvO, van de Looij, Zwitserlood, 2003]

agents:

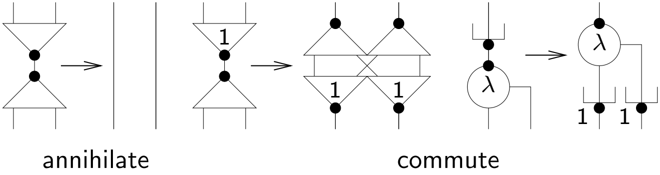


Lambdascope [VvO, van de Looij, Zwitserlood, 2003]

agents:

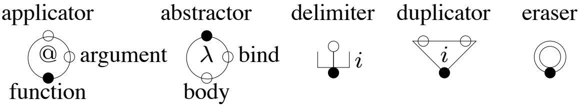


basic rules:

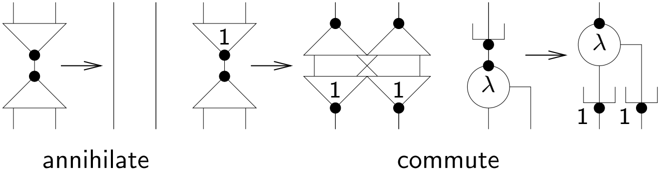


Lambdascope [VvO, van de Looij, Zwitserlood, 2003]

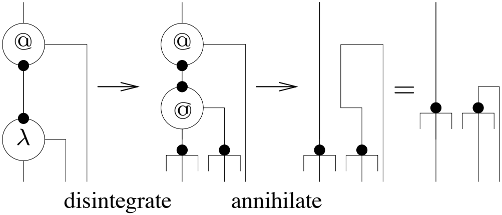
agents:



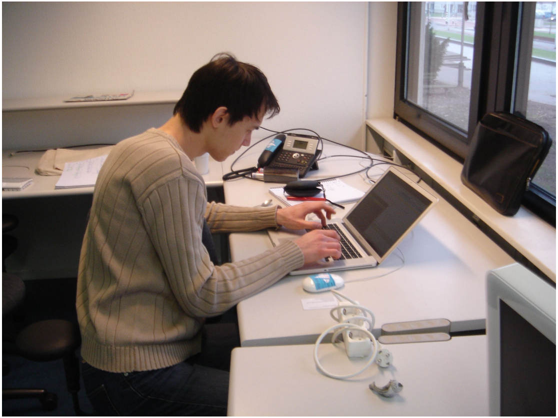
basic rules:



β-rule:



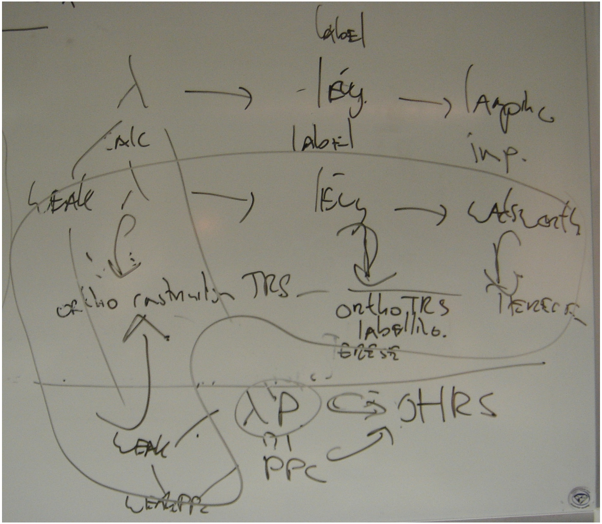
graafherschrijf-tool Jan Rochel



graph rewrite tool on Hackage:

`http://hackage.haskell.org/package/graph-rewriting-0.7.5`

gradaties van optimale reductie [VvO, plaatje 2008/10]



gradaties van optimale reductie

calculus (rewrite relation)	labelling	graph rewriting implementation	sharing notion
λ-calculus (β-reduction $\rightarrow_\beta$)	Lévy labelling '78	Lamping '89 Kathail '90 Abdadi/Gonthier/ /Levy '92 Asperti/Guerrini '93 VvO '03 (Terese)	context sharing
λ-calculus (weak-β red. $\rightarrow_{w\beta}$)	Blanc/Lévy/ Maranget '05/'07	Wadsworth '71 Shivers/Wand '04	extended-scope sharing
orthogonal TRS (induced rewrite relation $\rightarrow$)	VvO '03 (Terese)	Staples '80 VvO '03 (Terese)	subterm sharing

gradaties van optimale reductie

calculus (rewrite relation)	labelling	graph rewriting implementation	sharing notion
λ-calculus (weak-β red. $\to_{w\beta}$)	Blanc/Lévy/ Maranget '05/'07	Wadsworth '71 Shivers/Wand '04	extended-scope sharing
orthogonal TRS (induced rewrite relation $\to$)	VvO '03 (Terese)	Staples '80 VvO '03 (Terese)	subterm sharing

gradaties van optimale reductie

calculus (rewrite relation)	labelling	graph rewriting implementation	sharing notion
λ-calculus (weak-β red. $\rightarrow_{w\beta}$)	Blanc/Lévy/ Maranget '05/'07	Wadsworth '71 Shivers/Wand '04	extended-scope sharing
orthogonal TRS (induced rewrite relation $\rightarrow$)	VvO '03 (Terese)	Staples '80 VvO '03 (Terese)	subterm sharing

gradaties van optimale reductie

calculus (rewrite relation)	labelling	graph rewriting implementation	sharing notion
λ-calculus (weak-β red. $\to_{w\beta}$)	Blanc/Lévy/ Maranget '05/'07	Wadsworth '71 Shivers/Wand '04	extended-scope sharing
orthogonal TRS (induced rewrite relation $\to$)	VvO '03 (Terese)	Staples '80 VvO '03 (Terese)	subterm sharing

gradaties van optimale reductie

calculus (rewrite relation)	labelling	graph rewriting implementation	sharing notion
λ-calculus (weak-β red. $\to_{w\beta}$)	Blanc/Lévy/ Maranget '05/'07	Wadsworth '71 Shivers/Wand '04	extended-scope sharing
orthogonal TRS (induced rewrite relation $\to$)	VvO '03 (Terese)	Staples '80 VvO '03 (Terese)	subterm sharing

review [VvO, 2005]

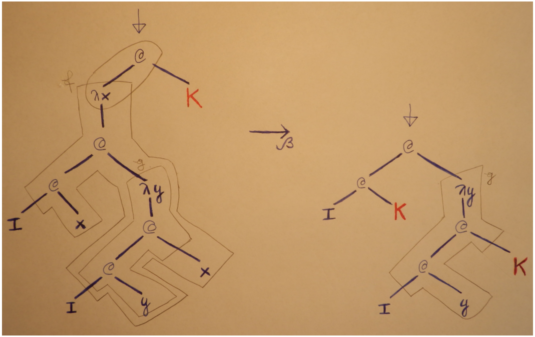
[redacted] That being said, by the λ-calculus considered being weak, its theory becomes closer to first-order TRSs than to λ-calculus; a rather useful connection (corroborated by *ad hoc* first-order implementations like supercombinators), which the authors do not mention. Therefore, since for TRSs it is known that dags suffice, it is not that surprising that they suffice for the weak λ-calculus as well. Even stronger, I think that this connection could and should have been exploited also technically (see the Interpretation in TRSs below). It would not only show that the results are not as difficult as the authors suggest now, but it could also help to improve the presentation[redacted]

weak β-step

$$(\lambda x.I\,x\,(\lambda y.I\,y\,x))\,K \quad \rightarrow_{w\beta} \quad I\,K\,(\lambda y.I\,K)$$

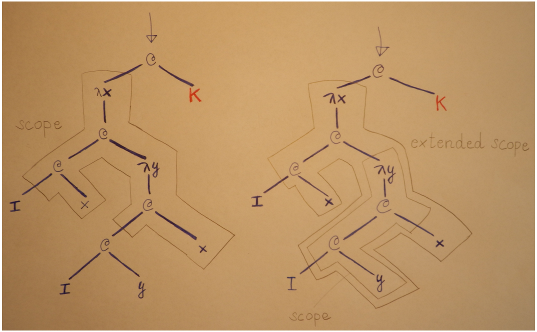
weak β-step

$$(\lambda x.I\,x\,(\lambda y.I\,y\,x))\,{\color{red}K} \;\;\to_{w\beta}\;\; I\,{\color{red}K}\,(\lambda y.I\,{\color{red}K})$$



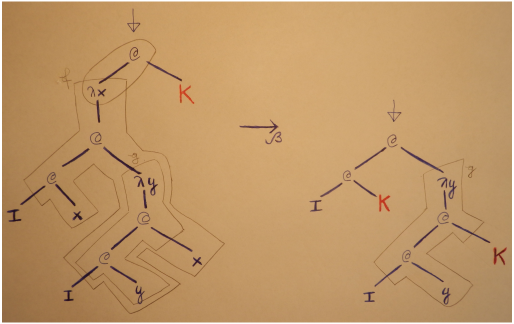
scope versus extended scope

$$(\lambda x.I\,x\,(\lambda y.I\,y\,x))\,K$$



oTRS vertaling van λ-calculus [VvO, review, 2005]

$$(\lambda x.I\,x\,(\lambda y.I\,y\,x))\,K \;\to_{w\beta}\; I\,K\,(\lambda y.I\,K)$$

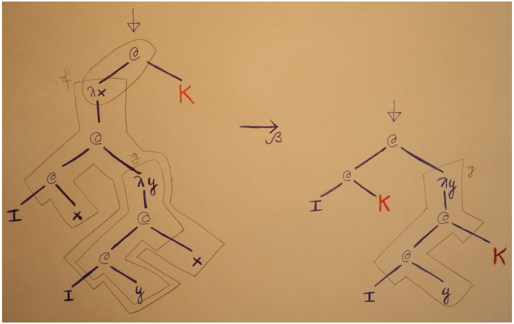


$f(w_1,w_2)\,X \to x_1\,X\,g(w_2,X)$ $g(z_1,z_2)\,Y \to z_1\,Y\,z_2$ $f(I,I)\,K$	$\to$	$f(w_1,w_2)\,X \to x_1\,X\,g(w_2,X)$ $g(z_1,z_2)\,Y \to z_1\,Y\,z_2$ $I\,K\,g(I,K)$

orthogonal term rewrite system (applicative notation)

oTRS vertaling van λ-calculus [VvO, review, 2005]

$$(\lambda x.I\,x\,(\lambda y.I\,y\,x))\,K \;\to_{w\beta}\; I\,K\,(\lambda y.I\,K)$$



$@(f(w_1,w_2),X) \to @(@(x_1,X),g(w_2,X))$ $@(g(z_1,z_2),Y) \to @(@(z_1,Y),z_2)$
$@(f(I,I),K)$

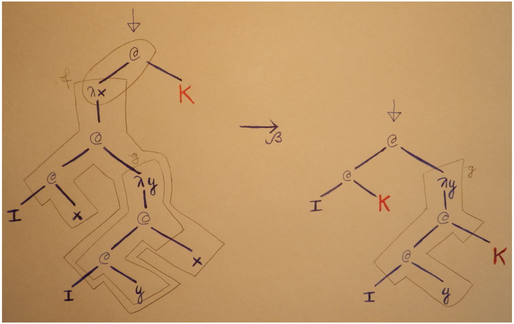
$\to$

$@(f(w_1,w_2),X) \to @(@(x_1,X),g(w_2,X))$ $@(g(z_1,z_2),Y) \to @(@(z_1,Y),z_2)$
$@(@(I,K),g(I,K))$

orthogonal term rewrite system

oTRS vertaling van λ-calculus [VvO, review, 2005]

$$(\lambda x. I\, x\, (\lambda y. I\, y\, x))\, K \;\to_{w\beta}\; I\, K\, (\lambda y. I\, K)$$



$F\, w_1\, w_2\, x \;\to\; w_1\, x\, (g\, w_2\, x)$ $G\, z_1\, z_2\, y \;\to\; z_1\, y\, z_2$		$F\, w_1\, w_2\, x \;\to\; w_1\, x\, (g\, w_2\, x)$ $G\, z_1\, z_2\, y \;\to\; z_1\, y\, z_2$
$F\, I\, I\, K$	$\to$	$I\, K\, (G\, I\, K)$

super combinator system (result of fully-lazy lambda-lifting)

gradaties van optimale reductie

calculus (rewrite relation)	labelling	graph rewriting implementation	sharing notion
λ-calculus (weak-β red. $\rightarrow_{w\beta}$)	Blanc/Lévy/ Maranget '05/'07	Wadsworth '71 Shivers/Wand '04	extended-scope sharing
orthogonal TRS (induced rewrite relation $\rightarrow$)	VvO '03 (Terese)	Staples '80 VvO '03 (Terese)	subterm sharing

diep

- ‘ik begrijp echt niet wat mensen bedoelen met diepe resultaten’
- zijn deze diep omdat:
  - geloof mij maar?
  - ik heb zo een gevoel?
- acceptabel:
  ‘A verklaart B, en is daarom dieper of verklarender dan B’
- ‘Was sich überhaupt sagen lässt, lässt sich klar sagen; [. . .]’
  (Wittgenstein, TLP)

gewoon

- ‘let maar eens op als mensen *gewoon* zeggen

  — meestal klopt er dan iets niet’

- empirisch feit

- mogelijke oorzaken? — behoefte van mensen om
  - niet verder na te denken
  - ergens bij te horen

het Vincents

- variant[1] van het Nederlands

[1] *van Dale*: variant = 'vorm die enigszins afwijkt van de gewone'

het Vincents

- variant[1] van het Nederlands
- er is maar een enkeling die het correct spreekt
- correctheid wordt duidelijk beargumenteerd
- Nederlands als daarover maar goed genoeg was nagedacht
  - bijv.: gegevens worden ‘aangehecht’ bij een e-mail niet ‘bijgesloten’
  - fout: ‘per omgaande’ voor situaties zonder analogie met de postbode
- gekenmerkt door grote kennis van en groot interesse voor streektalen
- altijd inspirerend:
  speurt betekenissen van uitdrukkingen nauwkeurig na

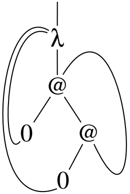
[1] *van Dale*: variant = ‘vorm die enigszins afwijkt van de gewone’

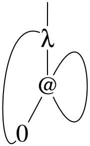
moeten?

- echt?
- waarom?
- van wie? wie zegt dat?
- wie gaat over de deadline? is deze redelijk?
- woord beter niet gebruiken?
- mogelijk om een ander woord te gebruiken?
  soms ja:
  - 'dienen te'
  - A 'zou het graag zien' als N gebeurt

maximal sharing [G/Rochel, ICFP'14]

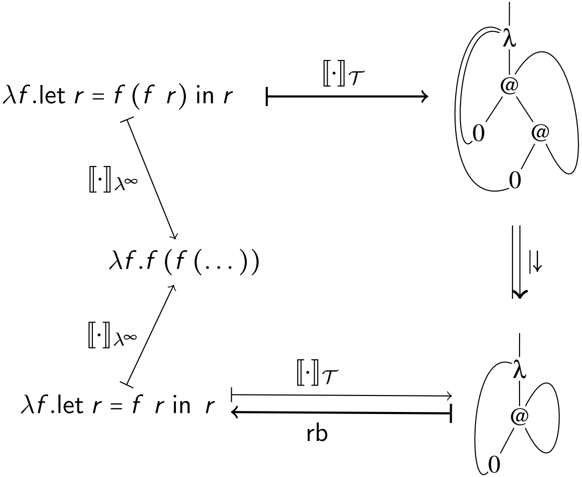
$$\begin{tikzcd}[column sep=huge]
\lambda f.\text{let } r = f\,(f\ r)\text{ in } r \arrow[d, mapsto, "{[\![\cdot]\!]_{\lambda^\infty}}"'] \arrow[r, mapsto, "{[\![\cdot]\!]_{\mathcal{T}}}"] & G_1 \arrow[dd, Rightarrow, "|\downarrow"] \\
\lambda f.f\,(f\,(\dots)) & \\
\lambda f.\text{let } r = f\ r\text{ in } r \arrow[u, mapsto, "{[\![\cdot]\!]_{\lambda^\infty}}"] \arrow[r, mapsto, shift left, "{[\![\cdot]\!]_{\mathcal{T}}}"] & G_2 \arrow[l, mapsto, shift left, "\text{rb}"]
\end{tikzcd}$$



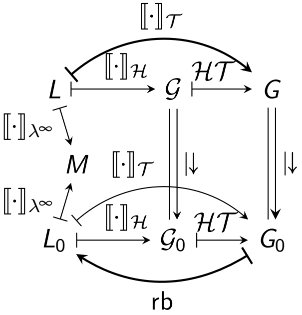


maximal sharing [G/Rochel, ICFP'14]

$$
\begin{tikzcd}
L \arrow[r, mapsto, "\llbracket\cdot\rrbracket_{\mathcal{H}}"] \arrow[rr, mapsto, bend left, "\llbracket\cdot\rrbracket_{\mathcal{T}}"] \arrow[d, mapsto, "\llbracket\cdot\rrbracket_{\lambda^\infty}"'] & \mathcal{G} \arrow[r, mapsto, "\mathcal{HT}"] \arrow[dd, Rightarrow, "|\!\downarrow"] & G \arrow[dd, Rightarrow, "|\!\downarrow"] \\
M & & \\
L_0 \arrow[u, mapsto, "\llbracket\cdot\rrbracket_{\lambda^\infty}"] \arrow[r, mapsto, "\llbracket\cdot\rrbracket_{\mathcal{H}}"] \arrow[rr, mapsto, bend left, "\llbracket\cdot\rrbracket_{\mathcal{T}}"] & \mathcal{G}_0 \arrow[r, mapsto, "\mathcal{HT}"] & G_0 \arrow[ll, mapsto, bend left, "\mathsf{rb}"]
\end{tikzcd}
$$



maximal sharing [G/Rochel, ICFP'14]

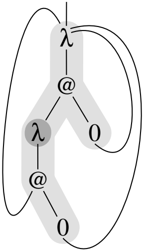


1. term graph translation $[\![\cdot]\!]$.
   of program L into:
   a. higher-order term graph $\mathcal{G} = [\![L]\!]_{\mathcal{H}}$
   b. first-order term graph $G = [\![L]\!]_{\mathcal{T}}$
2. bisimulation collapse $|\!\downarrow$
   of f-o term graph G into G_0
3. readback rb
   of f-o term graph G_0
   into program $L_0 = \mathrm{rb}(G_0)$.

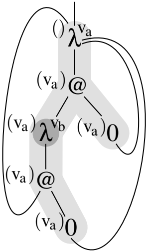
- tool on Hackage:
  `http://hackage.haskell.org/package/maxsharing/`
- paper (ICFP'14) and report (arXiv):
  G/Rochel: *Maximal Sharing in the λ-calculus with letrec*.

higher-order and first-order term graph interpretations

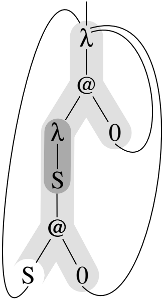
let $f = \lambda x.(\lambda y.f\ x)\, x$ in f



naive term graph interpretation



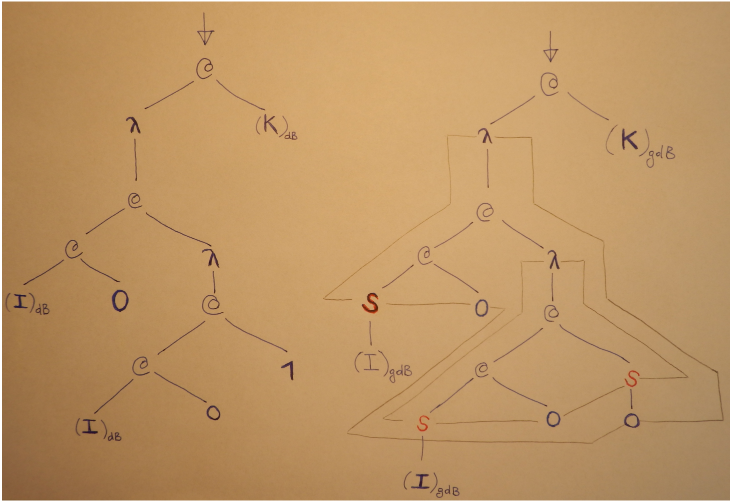
λ-h-o-term-graph interpretation



λ-term-graph interpretation

generalized de Bruijn index form [Patterson/Bird, VvO/Hendriks]

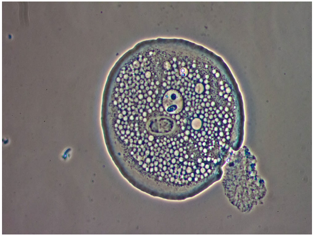
$$(\lambda x. I\, x\, (\lambda y. I\, y\, x))\, K$$



$\lambda\, (I)_{\mathrm{dB}}\, 0\, (\lambda\, (I)_{\mathrm{dB}}\, 0\, 1)$ $\lambda\, (S\, (I)_{\mathrm{gdB}})\, 0\, (\lambda\, (S\, (I)_{\mathrm{gdB}})\, 0\, (S\, 0))$

nested

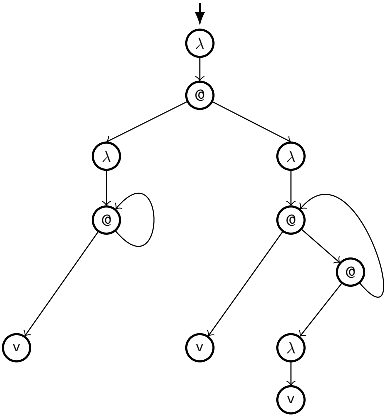
'a group of objects made to fit close together or one within another'



$$x = \sqrt{2 + \sqrt{2 + \sqrt{2 + \sqrt{2 + \ldots}}}}$$

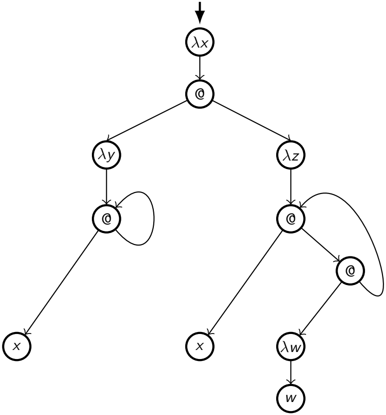
```
for i = 0 to 9 do
   for j = 0 to 9 do
     for k = 0 to 9 do
       sum = sum + i*100
                 + j*10 + k + 1;
```

nested scopes



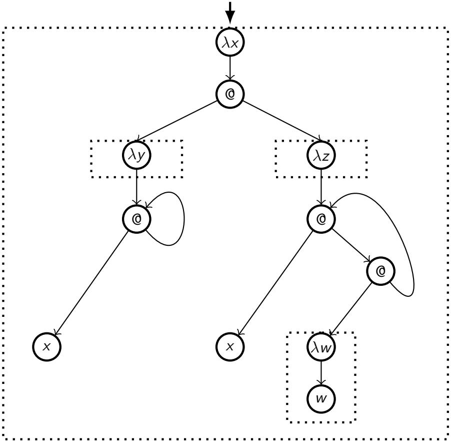
term graph with $\Sigma = \{\lambda/1, @/2, \mathsf{v}/0\}$

nested scopes



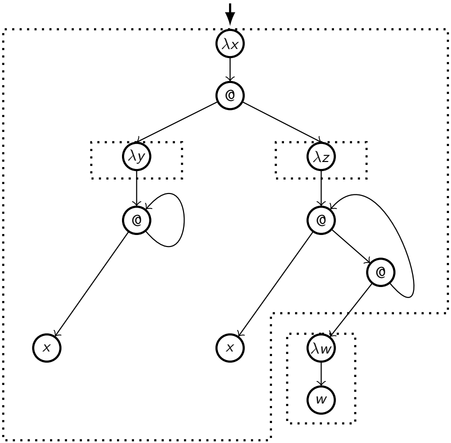
$$\lambda x.(\lambda y.\mathrm{let}\ \alpha = x\,\alpha\ \mathrm{in}\ \alpha)\,(\lambda z.\mathrm{let}\ \beta = x\,(\lambda w.w)\,\beta\ \mathrm{in}\ \beta)$$

nested scopes



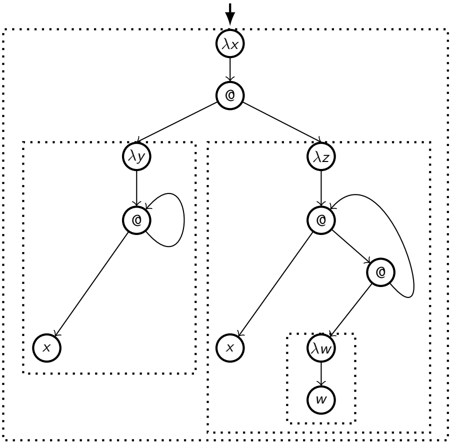
$$\lambda x.(\lambda y.\text{let } \alpha = x\,\alpha \text{ in } \alpha)\,(\lambda z.\text{let } \beta = x\,(\lambda w.w)\,\beta \text{ in } \beta)$$

nested scopes



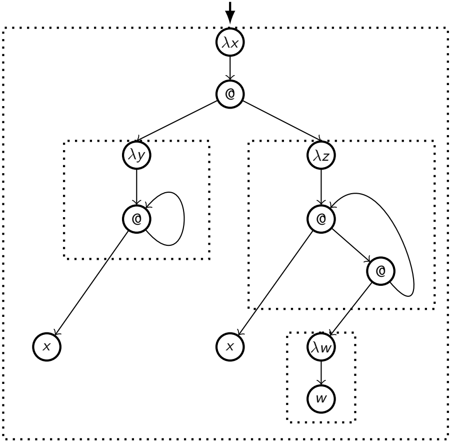
$$\lambda x.(\lambda y.\text{let } \alpha = x\,\alpha \text{ in } \alpha)\,(\lambda z.\text{let } \beta = x\,(\lambda w.w)\,\beta \text{ in } \beta)$$

nested scopes



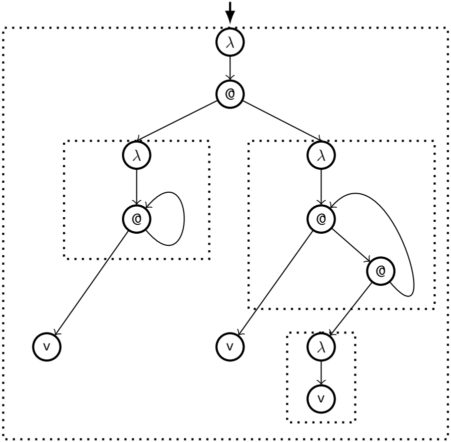
$$\lambda x.(\lambda y.\text{let } \alpha = x\,\alpha \text{ in } \alpha)\,(\lambda z.\text{let } \beta = x\,(\lambda w.w)\,\beta \text{ in } \beta)$$

nested scopes



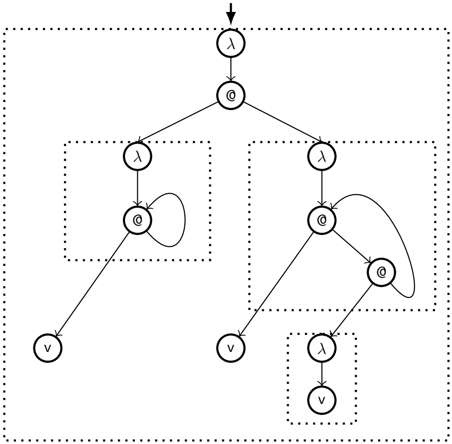
$$\lambda x.(\lambda y.\text{let } \alpha = x\,\alpha \text{ in } \alpha)\,(\lambda z.\text{let } \beta = x\,(\lambda w.w)\,\beta \text{ in } \beta)$$

nested scopes

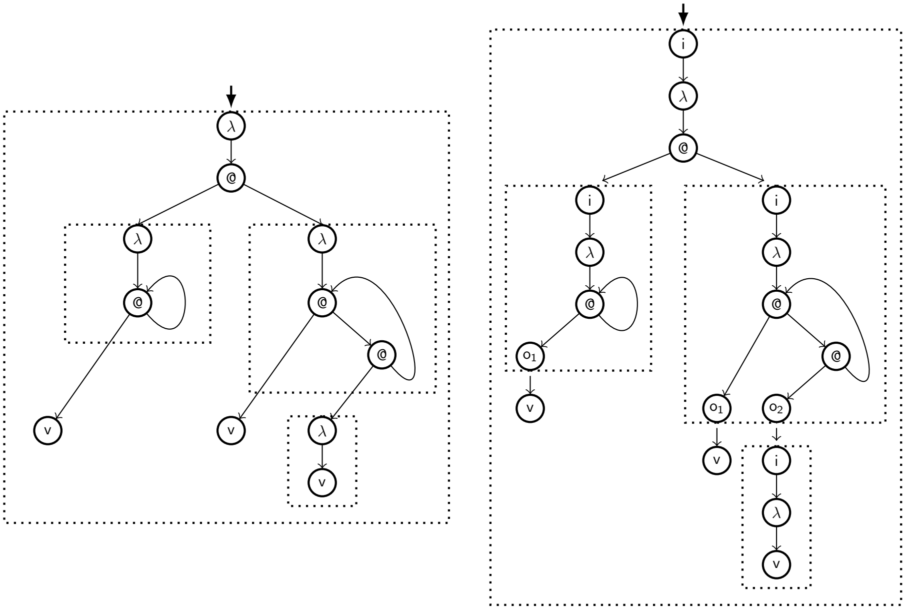


$$\lambda x.(\lambda y.\text{let } \alpha = x\,\alpha \text{ in } \alpha)\,(\lambda z.\text{let } \beta = x\,(\lambda w.w)\,\beta \text{ in } \beta)$$

nested scopes



nested scopes $\rightharpoonup$ nested term graph



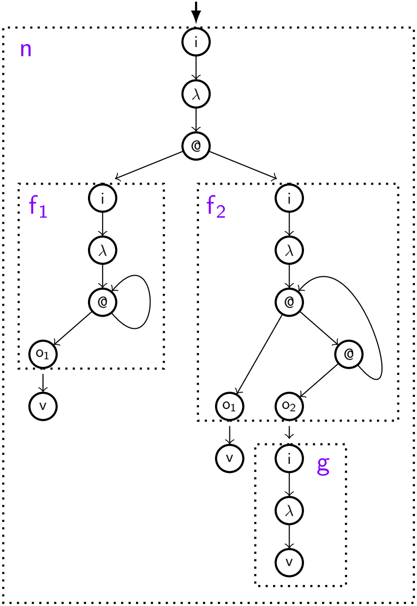
nested term graph

gletrec

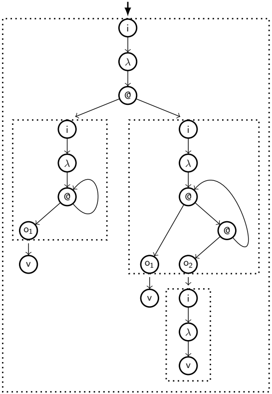
$$\begin{array}{rcl}
\mathsf{n}() & = & \lambda x.\mathsf{f}_1(x)\mathsf{f}_2(x, \mathsf{g}()) \\
\mathsf{f}_1(X_1) & = & \lambda x.\mathrm{let}\,\alpha = X_1\alpha\,\mathrm{in}\,\alpha \\
\mathsf{f}_2(X_1, X_2) & = & \lambda y.\mathrm{let}\,\beta = X_1(X_2\beta)\,\mathrm{in}\,\beta \\
\mathsf{g}() & = & \lambda z.z
\end{array}$$

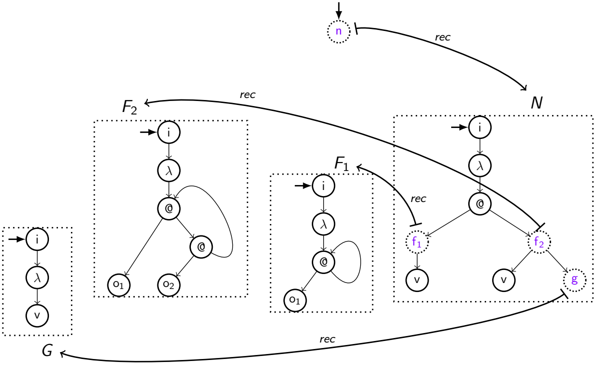
in

$$\mathsf{n}()$$



nested term graph



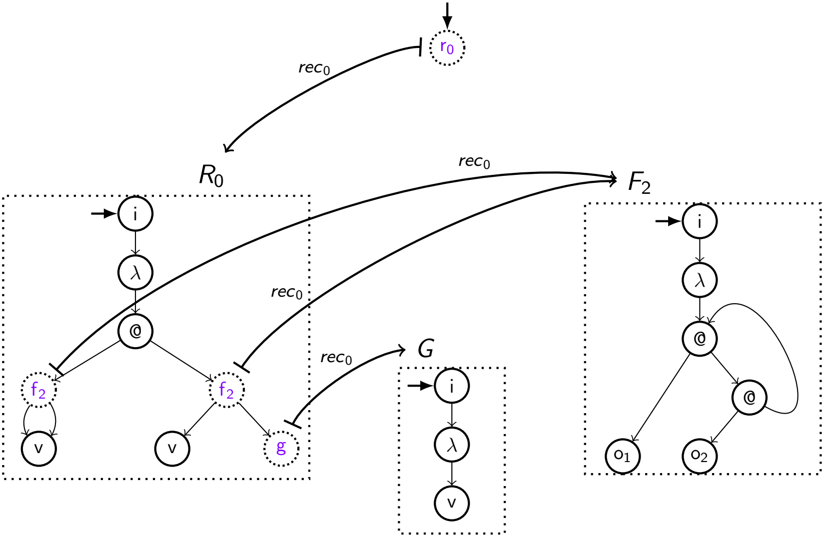


signature

A *signature for nested term graphs* (*ntg-signature*) is a signature Σ that is partitioned into:

- *atomic* symbol alphabet Σ_{at}
- *nested* symbol alphabet Σ_{ne}
- *interface* symbols alphabet $IO = I \cup O$
  - $I = \{\mathrm{i}\}$ with i unary
  - $O = \{\mathrm{o}_1, \mathrm{o}_2, \mathrm{o}_3, \ldots\}$ with o_i nullary

recursive graph specification



$\Sigma_{at} = \{\lambda/1,\ @/2,\ v/0\},\ \Sigma_{ne} = \{r_0/0,\ f_2/2,\ g/0\},\ I = \{i/1\},\ O = \{o_1/0, o_2/0, \ldots\}.$

recursive graph specification

Definition

Let Σ be an ntg-signature.
A *recursive graph specification* (a *rgs*) $\mathcal{R} = \langle rec, r\rangle$ consists of:

- *specification function*

$$rec : \Sigma_{\mathrm{ne}} \longrightarrow \mathrm{TG}(\Sigma \cup IO)$$
$$f \longmapsto rec(f_{/k}) = F \in \mathrm{TG}(\Sigma \cup \{\mathrm{i}, \mathrm{o}_1, \ldots, \mathrm{o}_k\})$$

  where F contains precisely one vertex labeled by i, the root, and one vertex each labeled by o_i, for $i \in \{1, \ldots, k\}$;
- nullary *root symbol* $r \in \Sigma_{\mathrm{ne}}$.

recursive graph specification

Definition

Let Σ be an ntg-signature.
A *recursive graph specification* (a *rgs*) $\mathcal{R} = \langle rec, r \rangle$ consists of:

- *specification function*

$$\begin{aligned} rec : \Sigma_{\mathrm{ne}} &\longrightarrow \mathrm{TG}(\Sigma \cup IO) \\ f &\longmapsto rec(f_{/k}) = F \in \mathrm{TG}(\Sigma \cup \{\mathrm{i}, \mathrm{o}_1, \ldots, \mathrm{o}_k\}) \end{aligned}$$

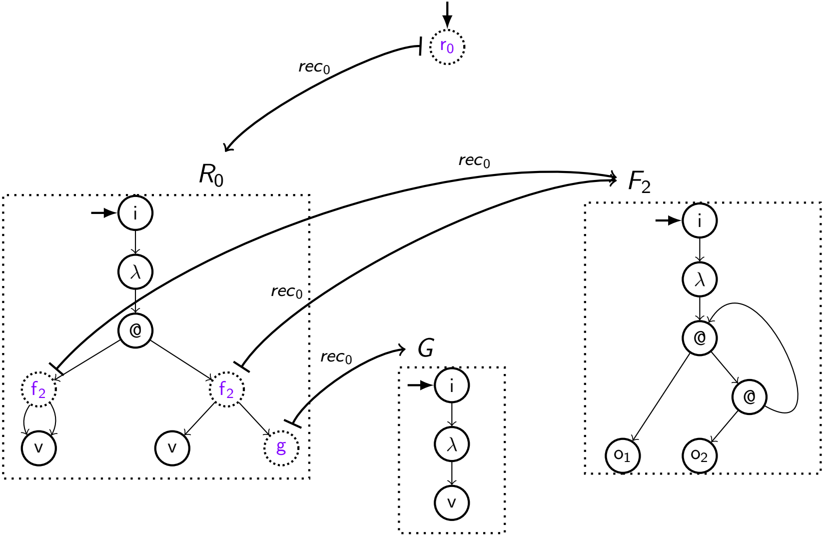
  where F contains precisely one vertex labeled by i, the root, and one vertex each labeled by o_i, for $i \in \{1, \ldots, k\}$;
- nullary *root symbol* $r \in \Sigma_{\mathrm{ne}}$.

rooted *dependency* ARS $\multimapinv$ of $\mathcal{R}$:

- objects: nested symbols in Σ_{ne}
- steps: for all $f, g \in \Sigma_{\mathrm{ne}}$:

$$p : f \multimapinv g \iff g \text{ occurs in the term graph } rec(f) \text{ at position } p$$

recursive graph specification



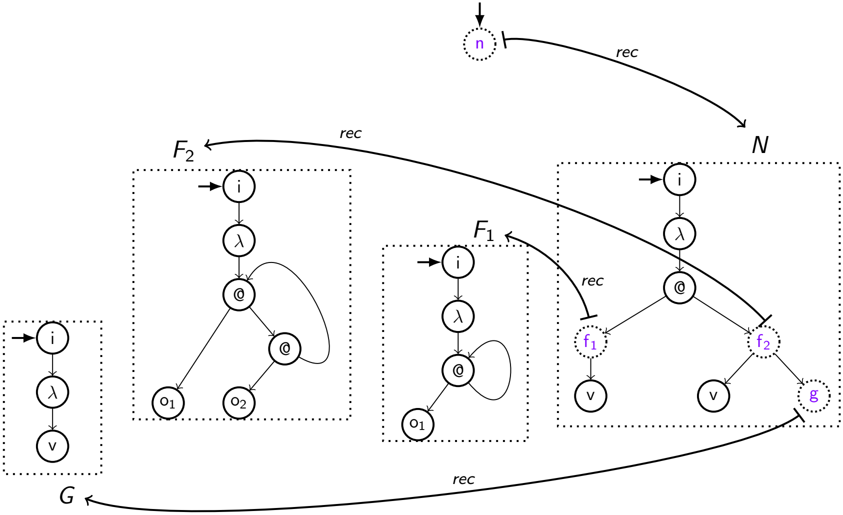
dependency ARS: $\mathsf{f}_2 \rightrightarrows \mathsf{r}_0 \leftarrow \mathsf{g}$ is a dag (but not a tree).

nested term graph

Definition

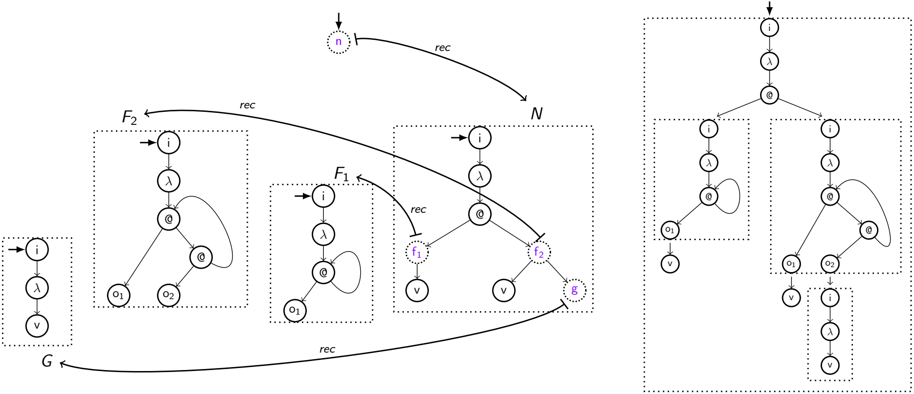
Let Σ be an ntg-signature.
A *nested term graph* over Σ is an rgs $\mathcal{N} = \langle rec, r \rangle$ such that the rooted dependency ARS $\multimapinv$ is a tree.

nested term graph



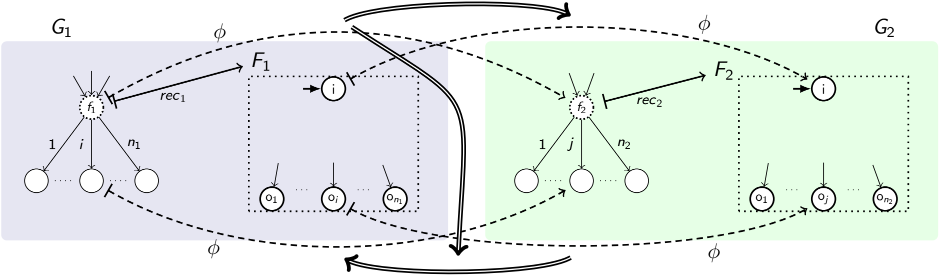
dependency ARS: $f_1 \multimap r_0 \begin{array}{l} \circ\!- f_2 \\ \circ\!- g \end{array}$ is a tree.

nested term graph



dependency ARS: $\mathsf{f}_1 \multimap \mathsf{r}_0 \begin{array}{l} \circ\!\!- \ \mathsf{f}_2 \\ \circ\!\!- \ \mathsf{g} \end{array}$ is a tree.

bisimulation clause



implementation by first-order term graphs [G/VvO, Termgraph'14]

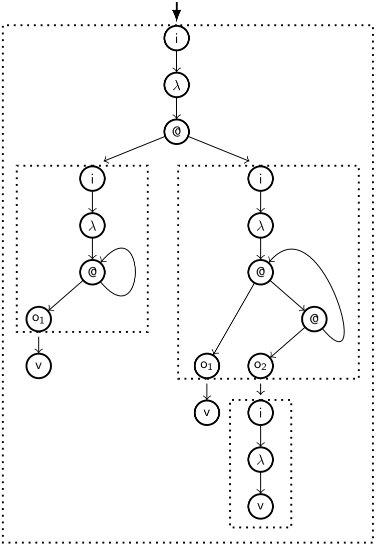
Theorem

Let Σ *be an ntg-signature, and* $\Sigma' = \Sigma \cup I \cup \{\mathsf{o}/2, \mathsf{i_r}/1, \mathsf{o_r}/1\}$.
There are functions

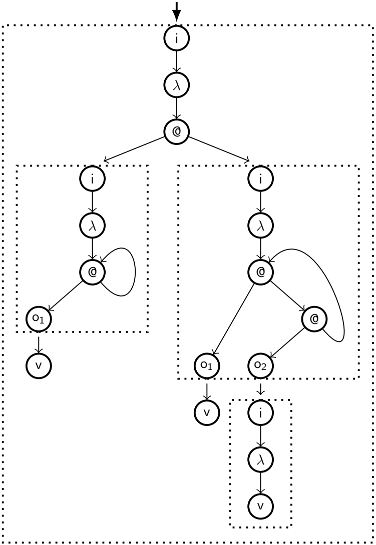
$$T : \mathrm{NG}(\Sigma) \to \mathrm{TG}(\Sigma') \quad \textit{and} \quad \mathcal{N} : \mathrm{TG}(\Sigma') \to \mathrm{NG}(\Sigma) \quad \textit{such that:}$$

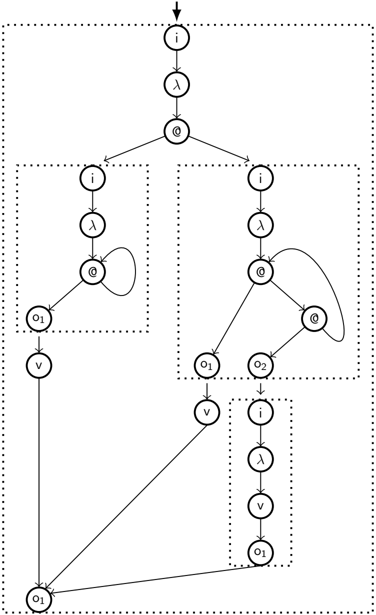
1. $\mathcal{N} \circ T = \mathrm{id}_{\mathrm{NG}(\Sigma)}$ *(i.e.* T *is a retraction of* $\mathcal{N}$*,* $\mathcal{N}$ *is a section of* T*)*
2. T *and* $\mathcal{N}$ *preserve and reflect* $\trianglerighteq$*.*
3. T *and* $\mathcal{N}$ *are efficiently computable.*

implementation by first-order term graph

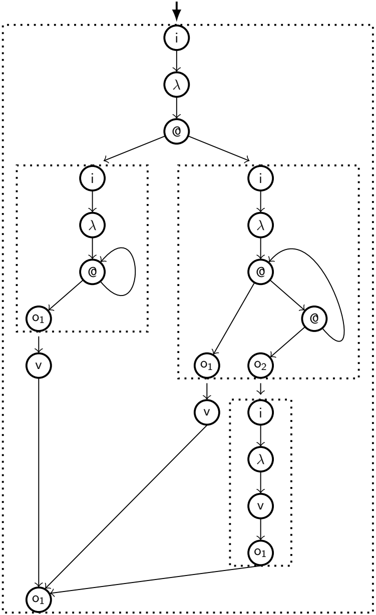


implementation by first-order term graph

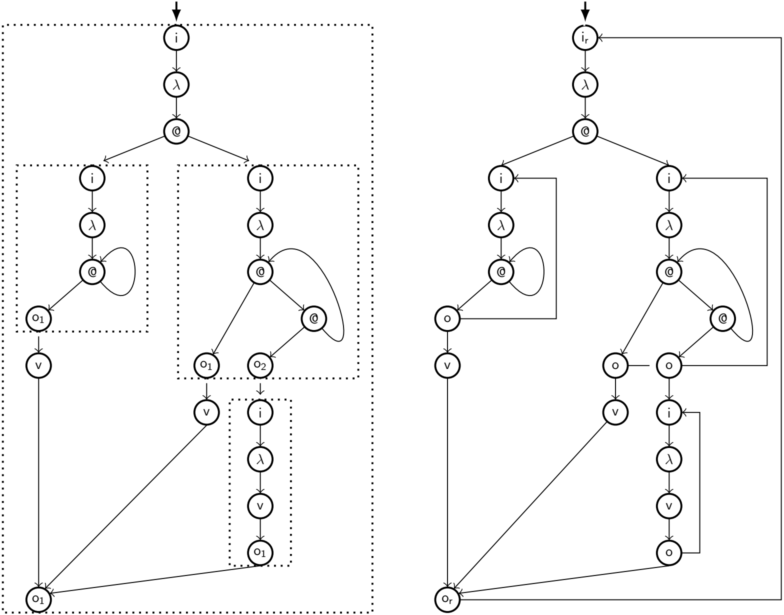




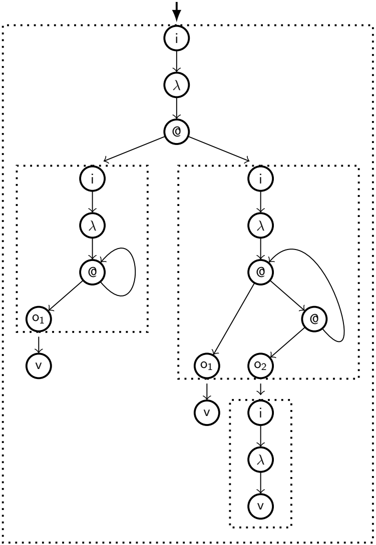
implementation by first-order term graph

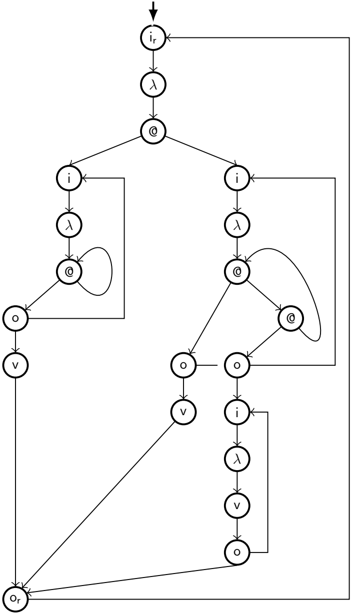


implementation by first-order term graph

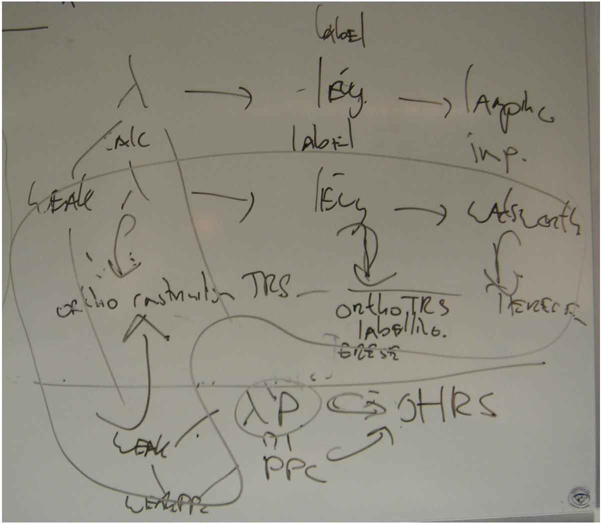


implementation by first-order term graph





gradaties van optimale reductie [VvO, plaatje 2008/10]



gradaties van optimale reductie

calculus (rewrite relation)	labelling	graph rewriting implementation	sharing notion
λ-calculus (β-reduction $\to_\beta$)	Lévy labelling '78	Lamping '89 Kathail '90 Abdadi/Gonthier/ /Levy '92 Asperti/Guerrini '93 VvO '03 (Terese)	context sharing
λ-calculus (weak-β red. $\to_{w\beta}$)	Blanc/Lévy/ Maranget '05/'07	Wadsworth '71 Shivers/Wand '04	extended-scope sharing
orthogonal TRS (induced rewrite relation $\to$)	VvO '03 (Terese)	Staples '80 VvO '03 (Terese)	subterm sharing

bijkomende gradaties van optimale reductie

calculus (rewrite relation)	labelling	graph rewriting implementation	sharing notion
λ-calculus (β-reduction $\to_\beta$)	Lévy labelling '78	Lamping '89 Kathail '90 Abdadi/Gonthier/ /Levy '92 Asperti/Guerrini '93 VvO '03 (Terese)	context sharing
	?	term/port graph implementation	scope sharing
	?	nested term graph implementation	extended scope sharing
λ-calculus (weak-β red. $\to_{w\beta}$)	Blanc/Lévy/ Maranget '05/'07	Wadsworth '71 Shivers/Wand '04	extended-scope sharing
orthogonal TRS (induced rewrite relation $\to$)	VvO '03 (Terese)	Staples '80 VvO '03 (Terese)	subterm sharing

university management

'Traditionally, a university was regarded as an institution whose primary function was the furtherance of learning and knowledge. Money was needed to maintain the infrastructure and pay the staff, but the money was a means to an end, not an end in itself.'

'However, it seems that this quaint notion is now rejected in favour of a model of a university whose success is measured in terms of its income, not in terms of its intellectual capital.'

'Management [. . .] just doesn't seem to get a very basic fact about running a university: Its academic staff are vital for the university's goal of achieving academic excellence. They need to be fostered, not bullied.'

Dorothy Bishop
'The university as Big Business'
22–06–2014

voetbal

Alles wat ik zeker weet over moraal en verplichtingen van de mens, heb ik aan voetbal te danken.

Albert Camus

voetbal

Alles wat ik zeker weet over moraal en verplichtingen van de mens, heb ik aan voetbal te danken.

Albert Camus

voetbal

Alles wat ik zeker weet over moraal en verplichtingen van de mens, heb ik aan voetbal te danken.

Albert Camus

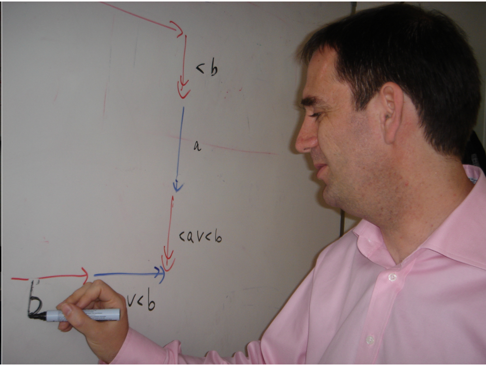
voetbal

Alles wat ik zeker weet over moraal en verplichtingen van de mens, heb ik aan voetbal te danken.

Albert Camus

mijn dank & beste wensen voor de toekomst

- aan Vincent



- aan de disciplinegroep Theoretische Filosofie
- aan de studenten CKI en CAI