λ-calcul différentiel et logique classique

interactions calculatoires

Lionel Vaux

Institut de Mathématiques de Luminy
Université de la Méditerrannée Aix-Marseille 2

Thèse de Doctorat
vendredi 23 novembre 2007

Logique de la programmation

Étudier les propriétés mathématiques des démonstrations

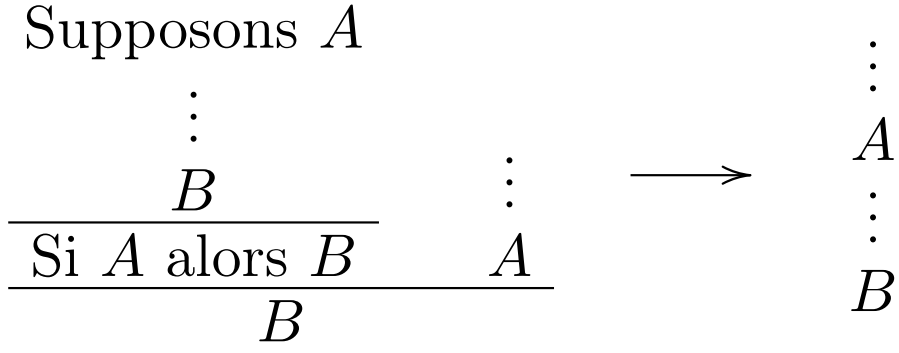
Logique de la programmation

Étudier les propriétés mathématiques des démonstrations en particulier leur dynamique.

$$\dfrac{\dfrac{\begin{array}{c}\text{Supposons } A\\ \vdots\\ B\end{array}}{\text{Si } A \text{ alors } B} \qquad \begin{array}{c}\vdots\\ A\end{array}}{B}$$

Logique de la programmation

Étudier les propriétés mathématiques des démonstrations en particulier leur dynamique.



Logique de la programmation

Étudier les propriétés mathématiques des démonstrations en particulier leur dynamique.

$$\dfrac{\dfrac{\begin{array}{c}\text{Supposons } A\\ \vdots f\\ B\end{array}}{\text{Si } A \text{ alors } B} \qquad \begin{array}{c}\vdots x\\ A\end{array}}{B} \longrightarrow \begin{array}{c}\vdots x\\ A\\ \vdots f\\ B\end{array}$$

Logique de la programmation

Étudier les propriétés mathématiques des démonstrations en particulier leur dynamique.

$$\dfrac{\dfrac{\begin{array}{c}\text{Supposons } A\\ \vdots\, f\\ B\end{array}}{\text{Si } A \text{ alors } B} \qquad \begin{array}{c}\vdots\, x\\ A\end{array}}{B} \quad \longrightarrow \quad \begin{array}{c}\vdots\, x\\ A\\ \vdots\, f\\ B\end{array}$$

Représente le calcul d'une fonction : λ-calcul.

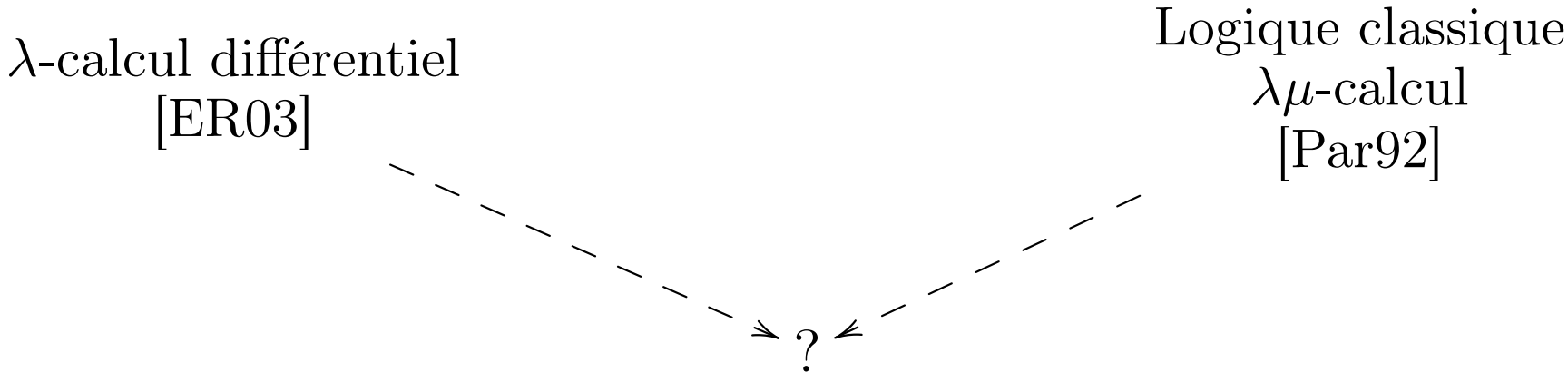
Motivations, Résultats

λ-calcul différentiel
[ER03]

Logique classique

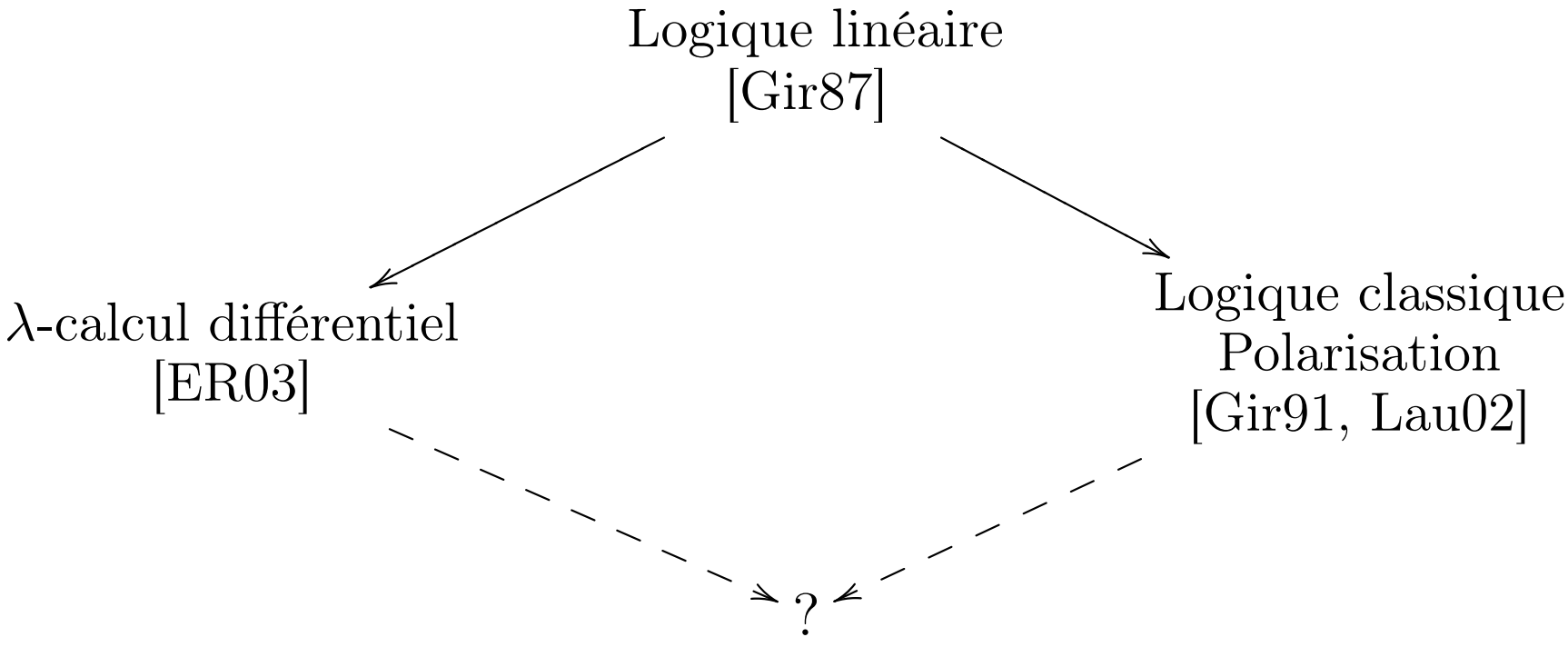
Motivations, Résultats

Motivations, Résultats



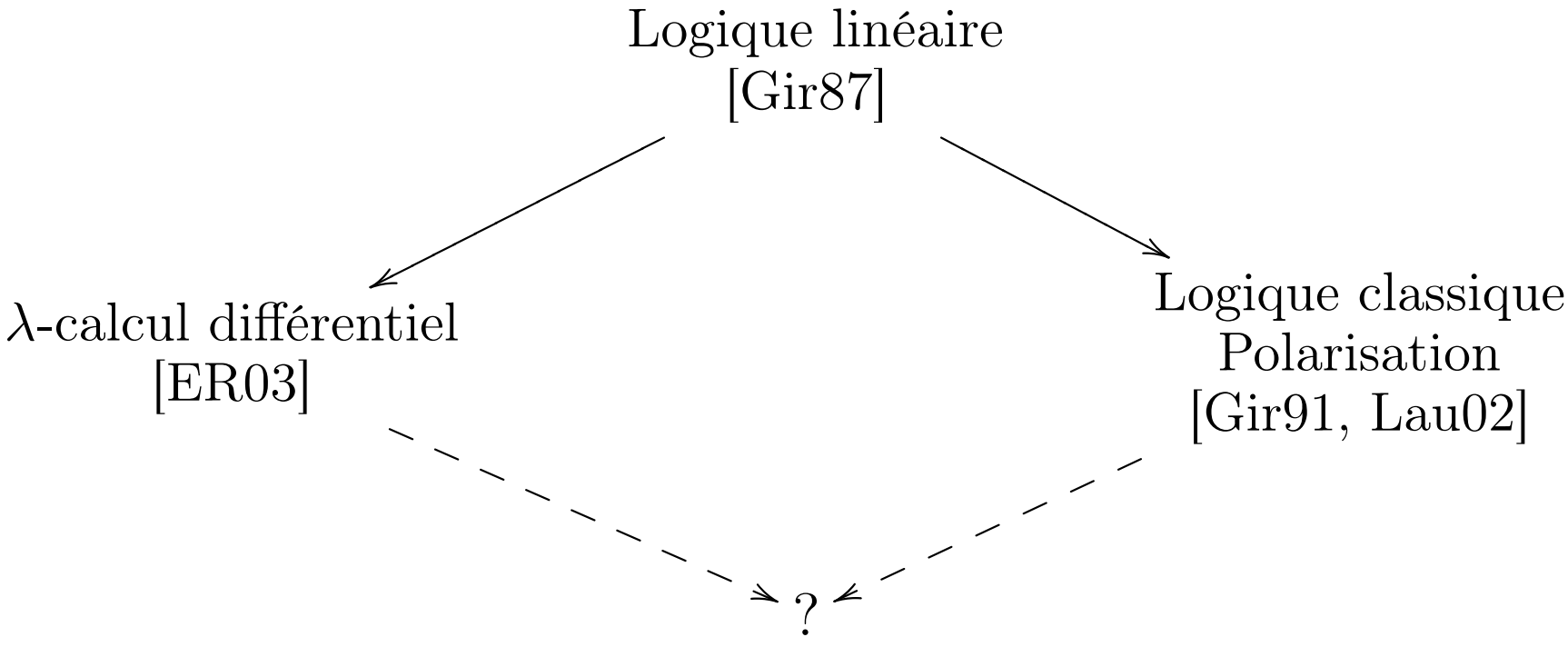
- Compatibilité : $\lambda\mu$-calcul différentiel [Vau07b].

Motivations, Résultats



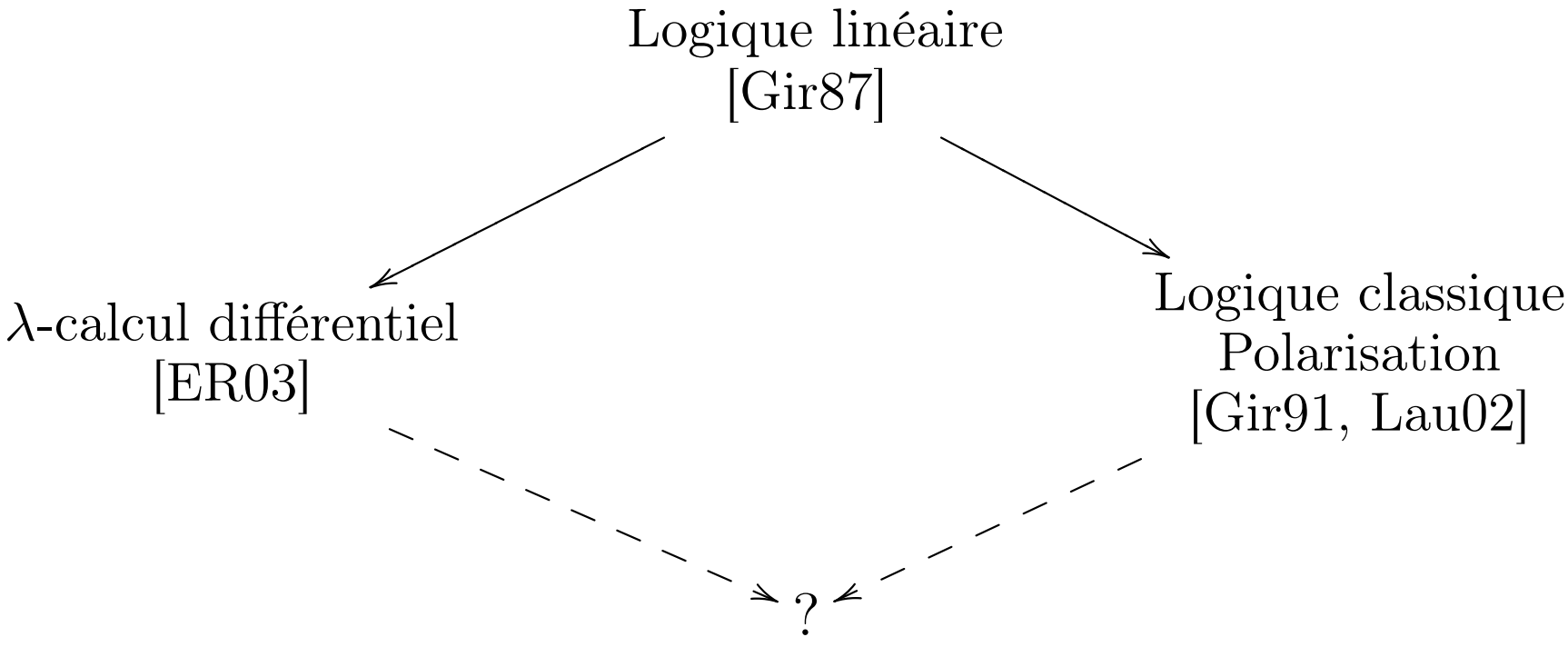
- Compatibilité : $\lambda\mu$-calcul différentiel [Vau07b].

Motivations, Résultats



- Compatibilité : $\lambda\mu$-calcul différentiel [Vau07b].
- Extension polarisée des règles costructurelles.

Motivations, Résultats



- Compatibilité : $\lambda\mu$-calcul différentiel [Vau07b].
- Extension polarisée des règles costructurelles.
- Règles costructurelles explicites dans la syntaxe : $\bar{\lambda}\mu$-calcul avec produit de convolution [Vau07a].

Sommes et linéarité

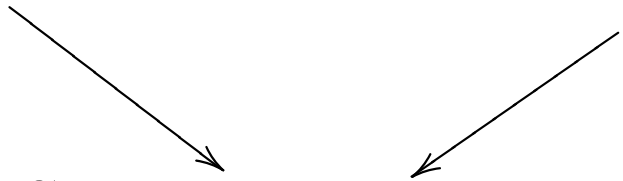
Linéarité en algèbre :
commutation aux sommes
$(f + g)[x] = f[x] + g[x]$

Linéarité en programmation :
positions évaluées
exactement une fois
code

Sommes et linéarité

Linéarité en algèbre :
commutation aux sommes
$(f + g)[x] = f[x] + g[x]$

Linéarité en programmation :
positions évaluées
exactement une fois
code

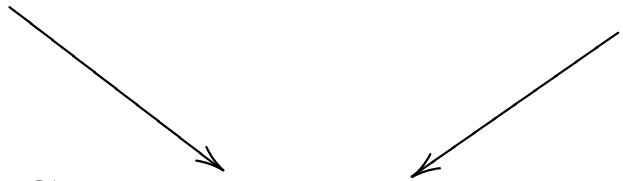


Coïncident en λ-calcul :
$\lambda x\, \underline{s}$ $\quad$ $(\underline{s})\, t$

Sommes et linéarité

Linéarité en algèbre :
commutation aux sommes
$(f+g)[x] = f[x] + g[x]$

Linéarité en programmation :
positions évaluées
exactement une fois
code



Coïncident en λ-calcul :
$\lambda x\, \underline{s} \qquad (\underline{s})\, t$

$$s, t ::= x \mid \lambda x\, s \mid (s)\, t \mid \mathbf{0} \mid s + t$$

Sommes et linéarité

Linéarité en algèbre :
commutation aux sommes
$(f+g)[x] = f[x] + g[x]$

Linéarité en programmation :
positions évaluées
exactement une fois
code

Coïncident en λ-calcul :
$\lambda x\, \underline{s}$ $\qquad$ $(\underline{s})\, t$

$$s, t ::= x \mid \lambda x\, s \mid (s)\, t \mid \mathbf{0} \mid s + t$$

$$\begin{aligned} \lambda x\, (s+t) &= \lambda x\, (s) + \lambda x\, (t) \\ (s+t)\, u &= (s)\, u + (t)\, u \end{aligned}$$

Sommes et linéarité

Linéarité en algèbre :
commutation aux sommes
$(f+g)[x] = f[x] + g[x]$

Linéarité en programmation :
positions évaluées
exactement une fois
code

Coïncident en λ-calcul :
$\lambda x\, \underline{s} \qquad (\underline{s})\, t$

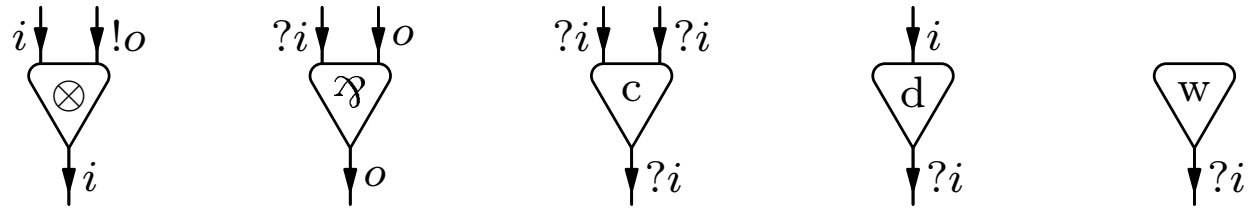
$$s, t ::= x \mid \lambda x\, s \mid (s)\, t \mid \mathbf{0} \mid s + t \mid a.s$$

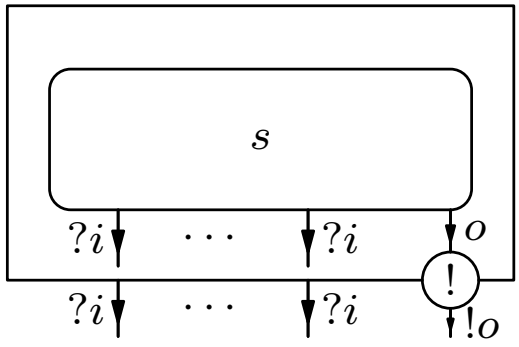
Problèmes... [ER03, Vau07c]

$$\begin{aligned} \lambda x\, (s+t) &= \lambda x\, (s) + \lambda x\, (t) \\ (s+t)\, u &= (s)\, u + (t)\, u \end{aligned}$$

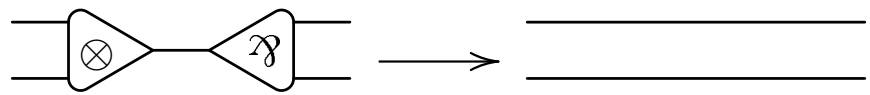
Réseaux purs

$$o = ?o^{\perp} \parr o$$
$$i = o^{\perp}$$

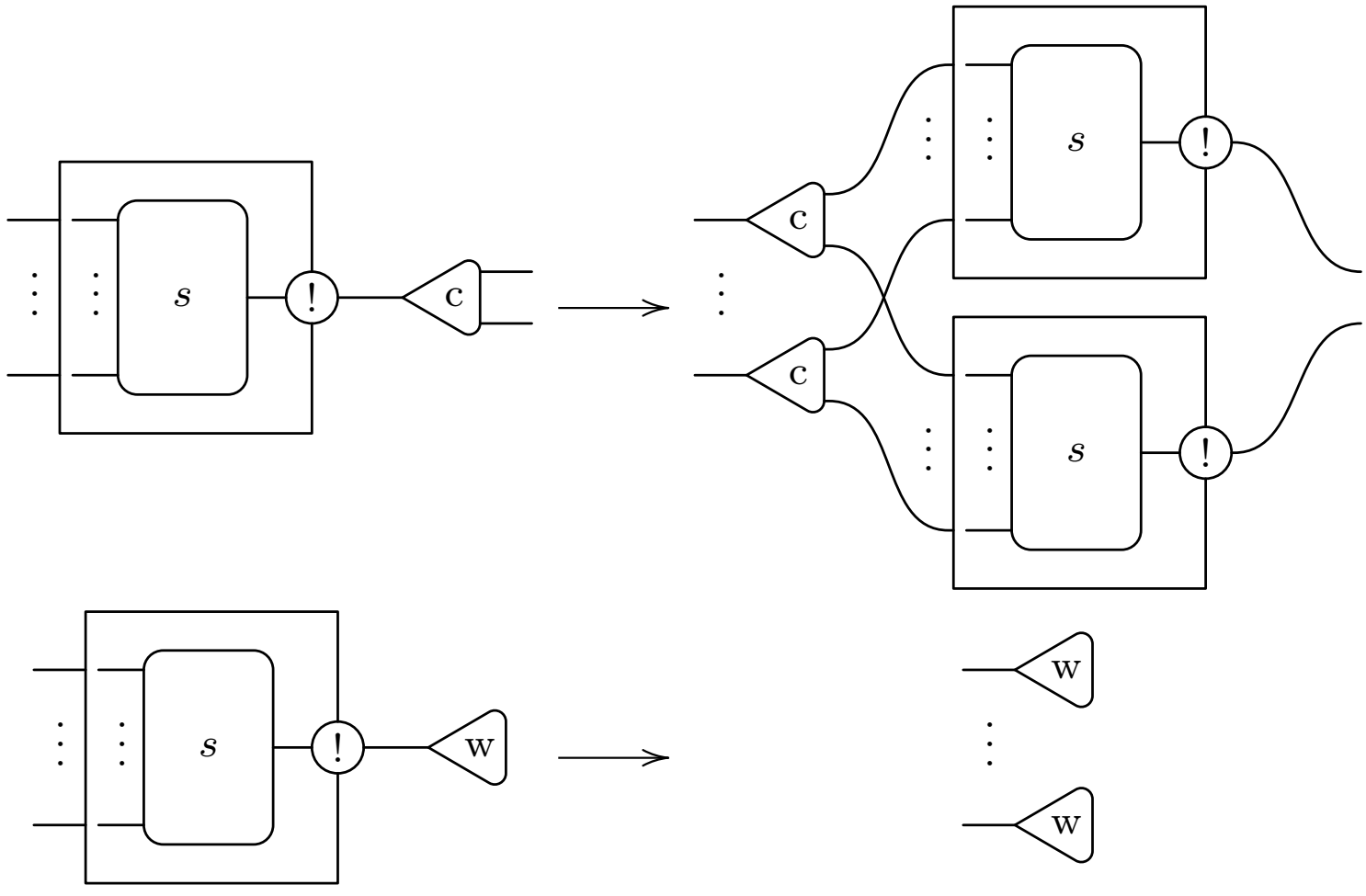




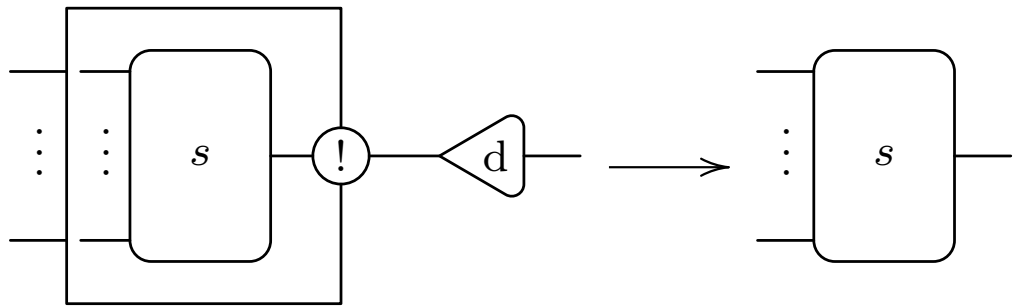
Réduction multiplicative $\langle \parr , \otimes \rangle$



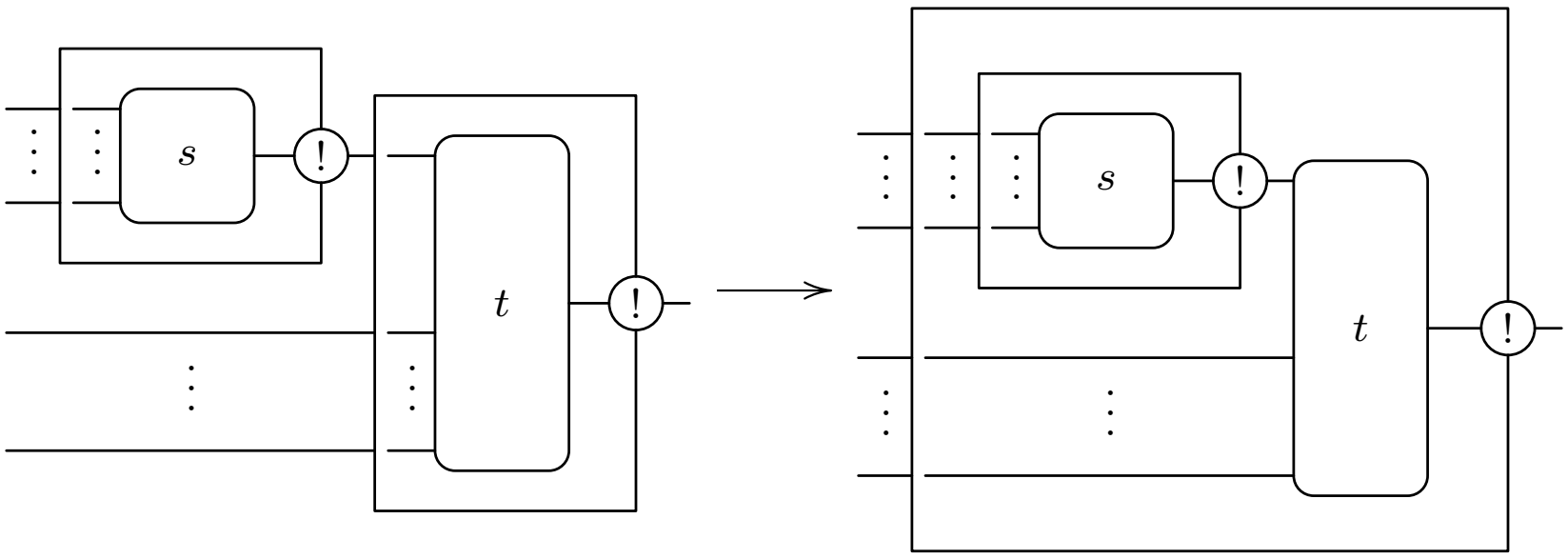
Duplication $\langle !\,,\,\mathrm{c}\rangle$ et effacement $\langle !\,,\,\mathrm{w}\rangle$



Ouverture $\langle !\ , \mathrm{d} \rangle$

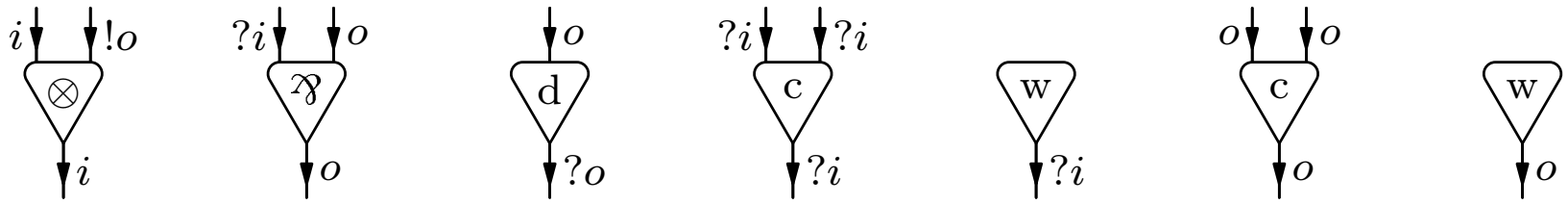


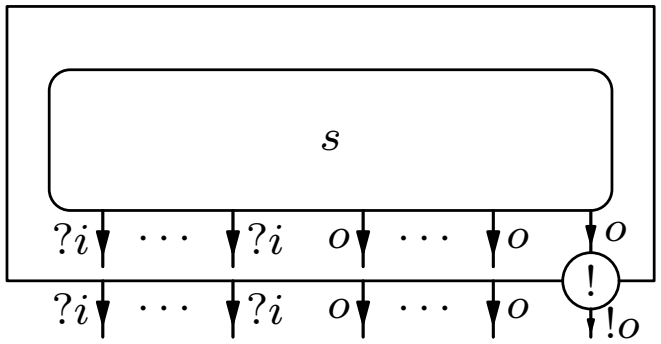
Commutation $\langle !\ ,\ ! \rangle$



Logique linéaire polarisée

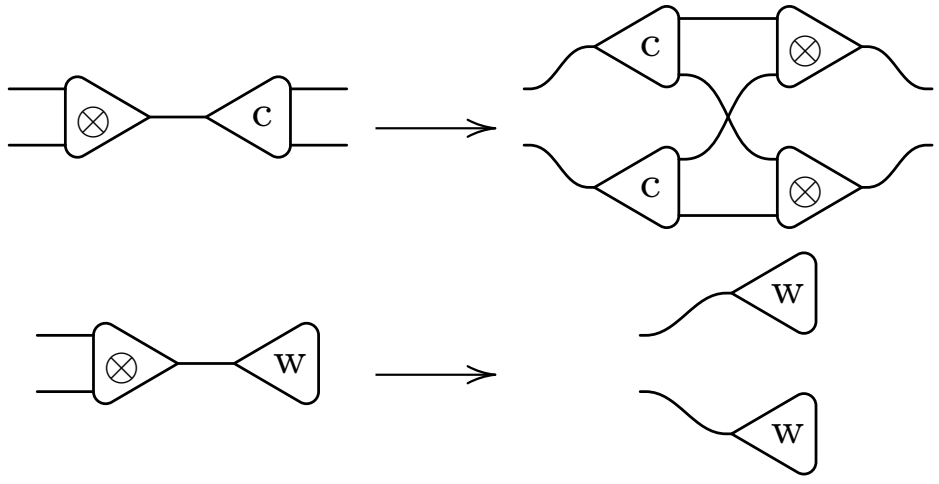
On relâche les règles structurelles.





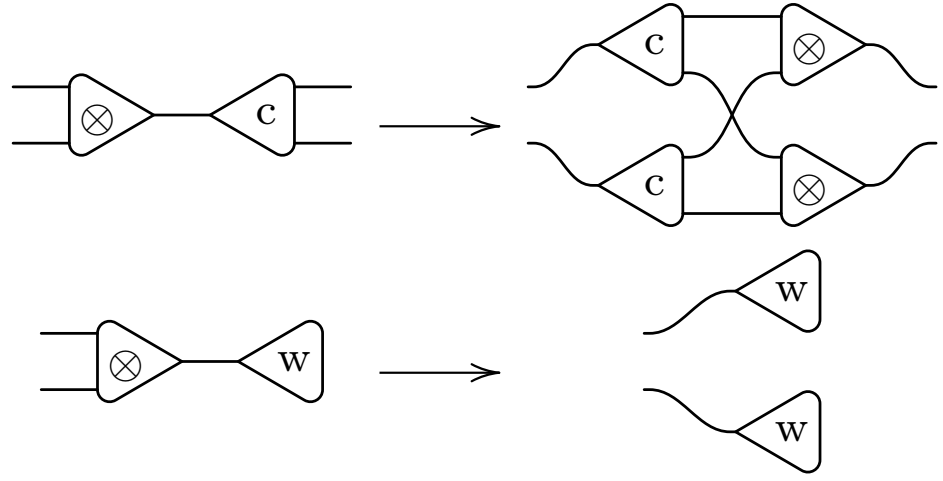
Logique linéaire polarisée

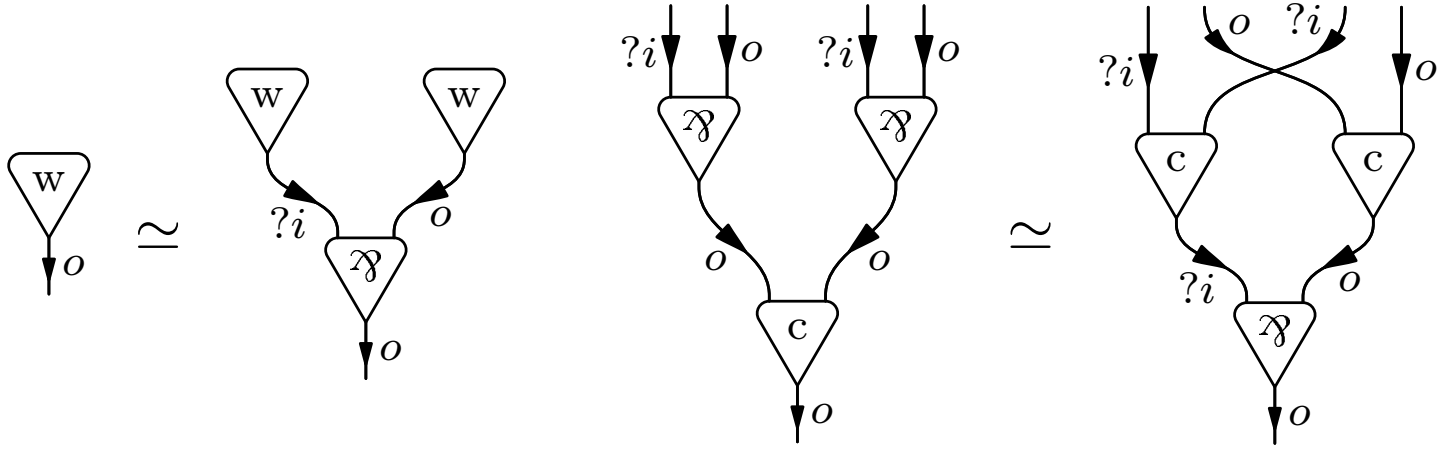
Nouveaux redex typables :



Logique linéaire polarisée

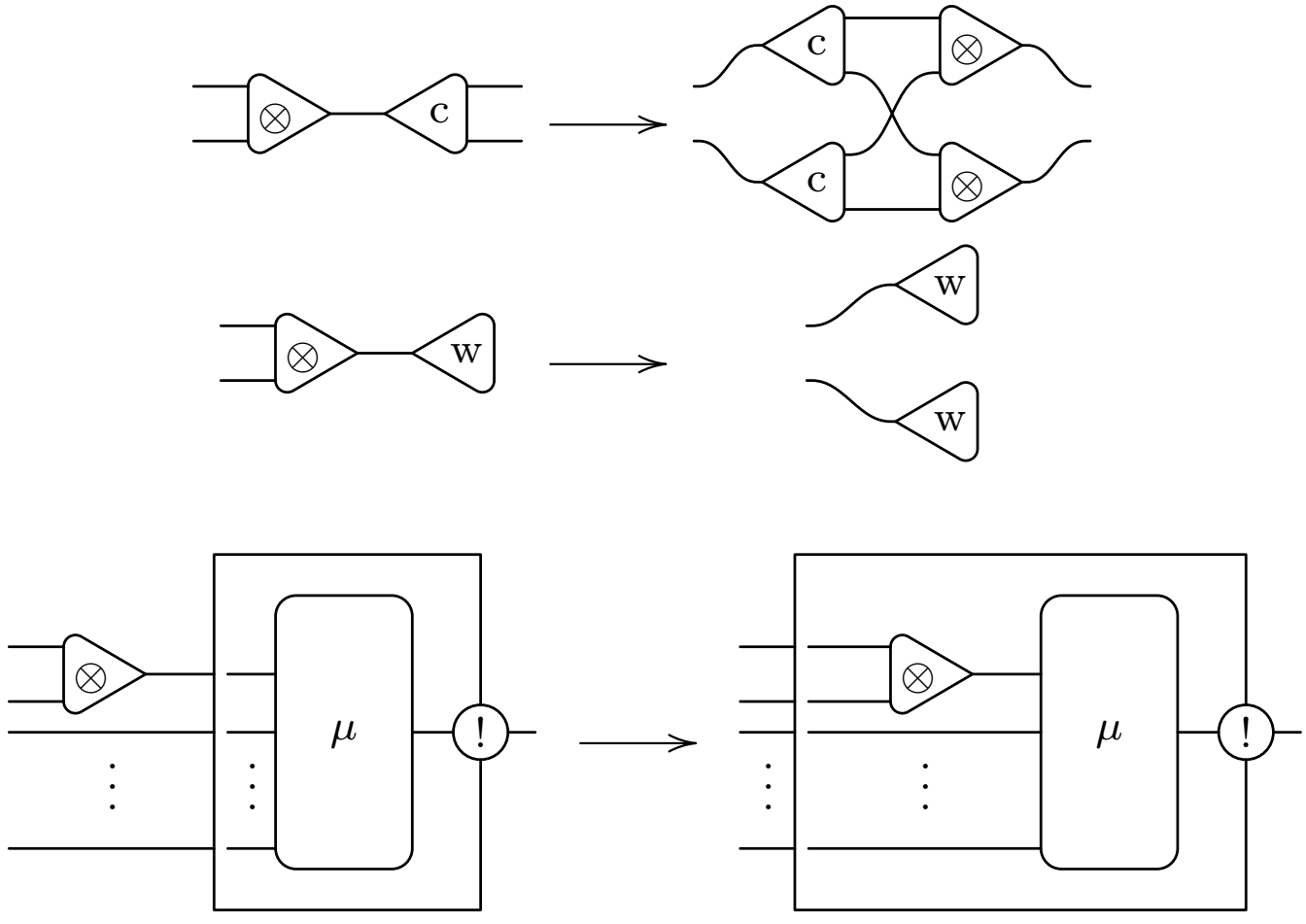
Nouveaux redex typables :





Logique linéaire polarisée

Nouveaux redex typables :

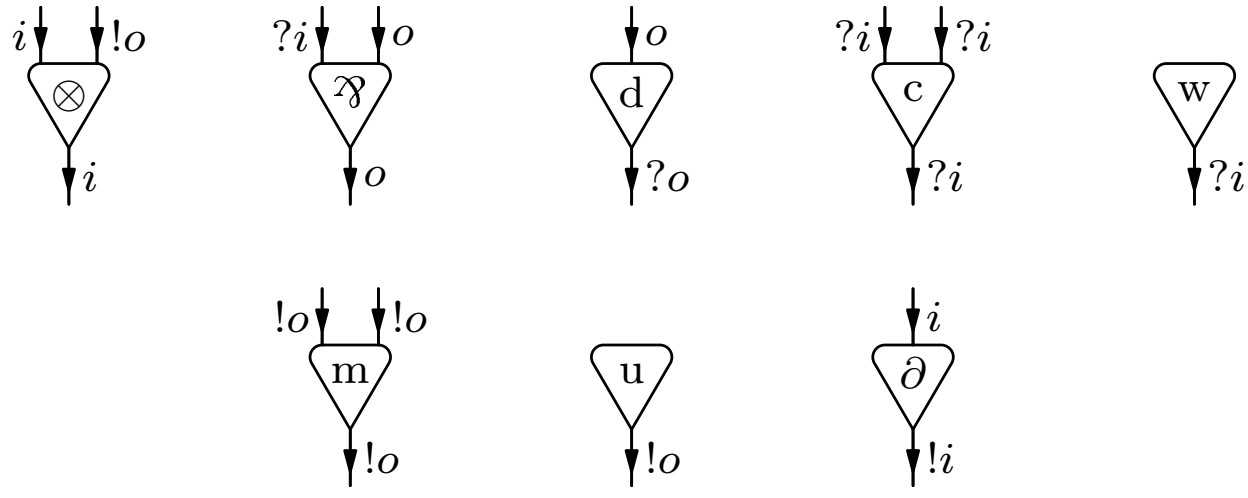


Réseaux différentiels

Des réseaux pour le λ-calcul différentiel [ER05].

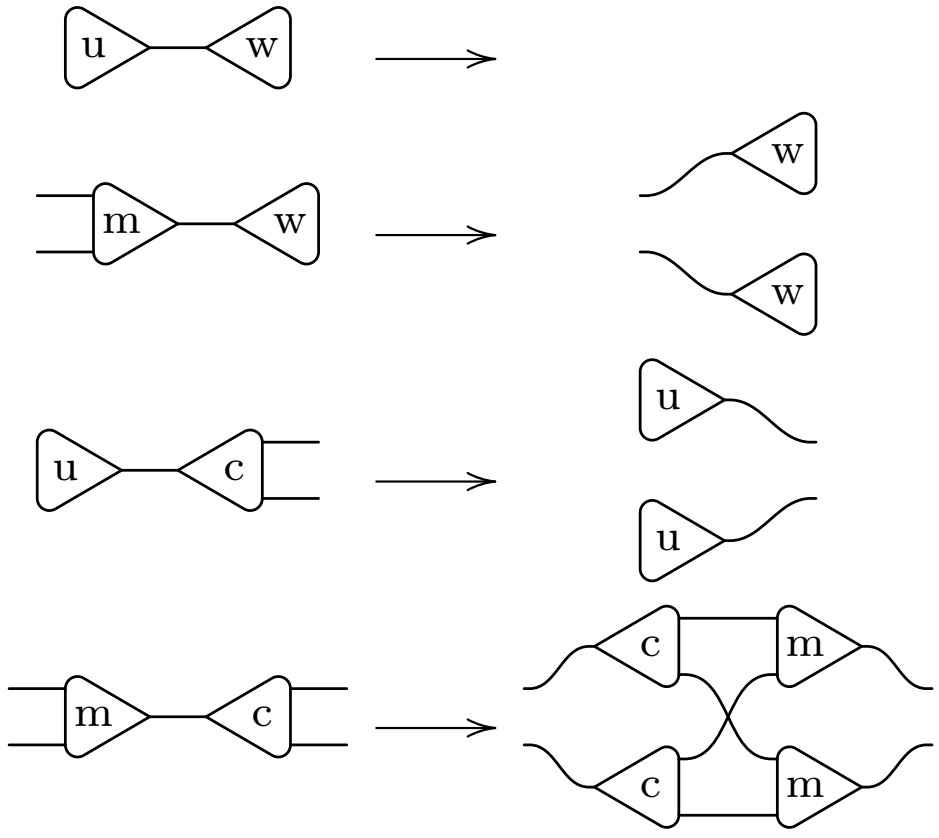
Réseaux différentiels

Des réseaux pour le λ-calcul différentiel [ER05].

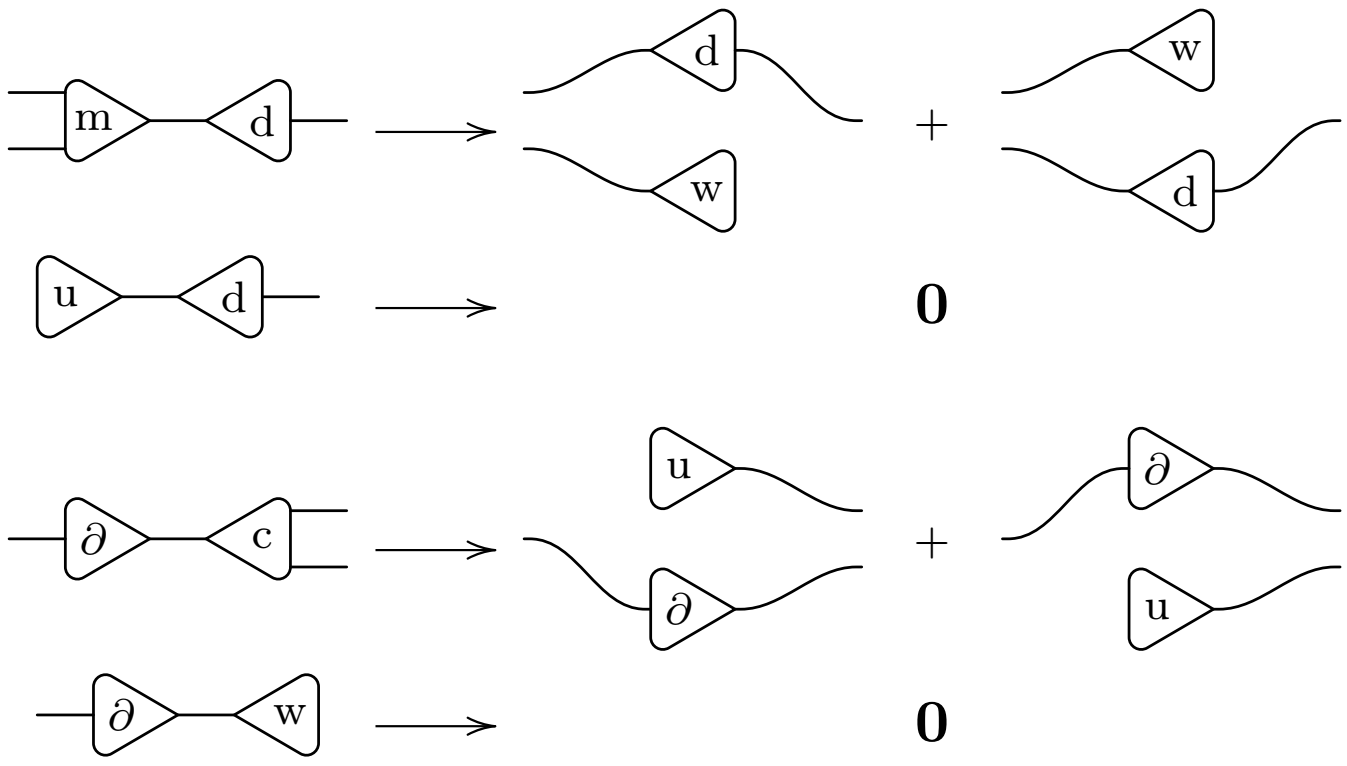


Représentation graphique de [Ehr01, Ehr05].

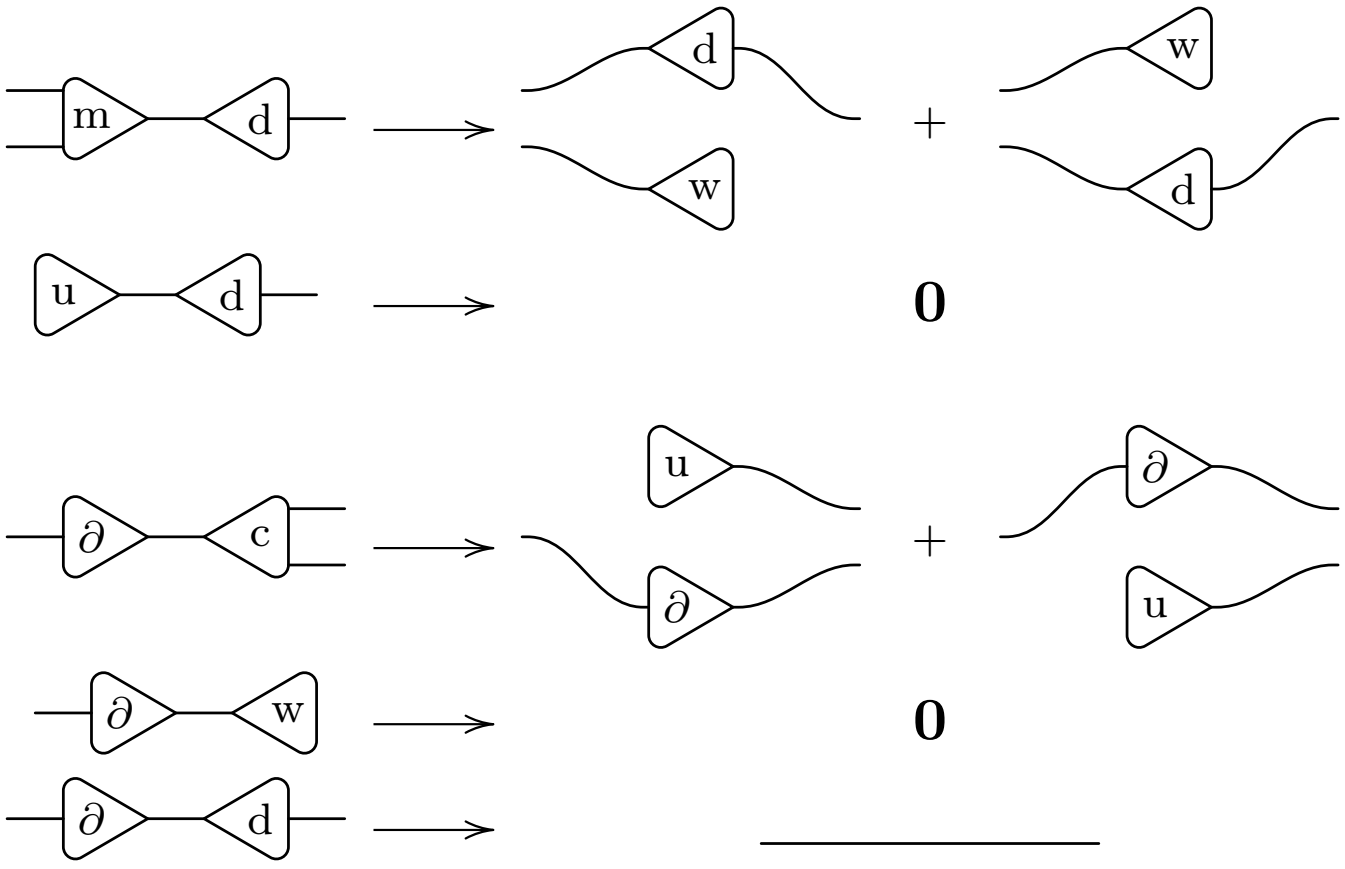
Bigèbres



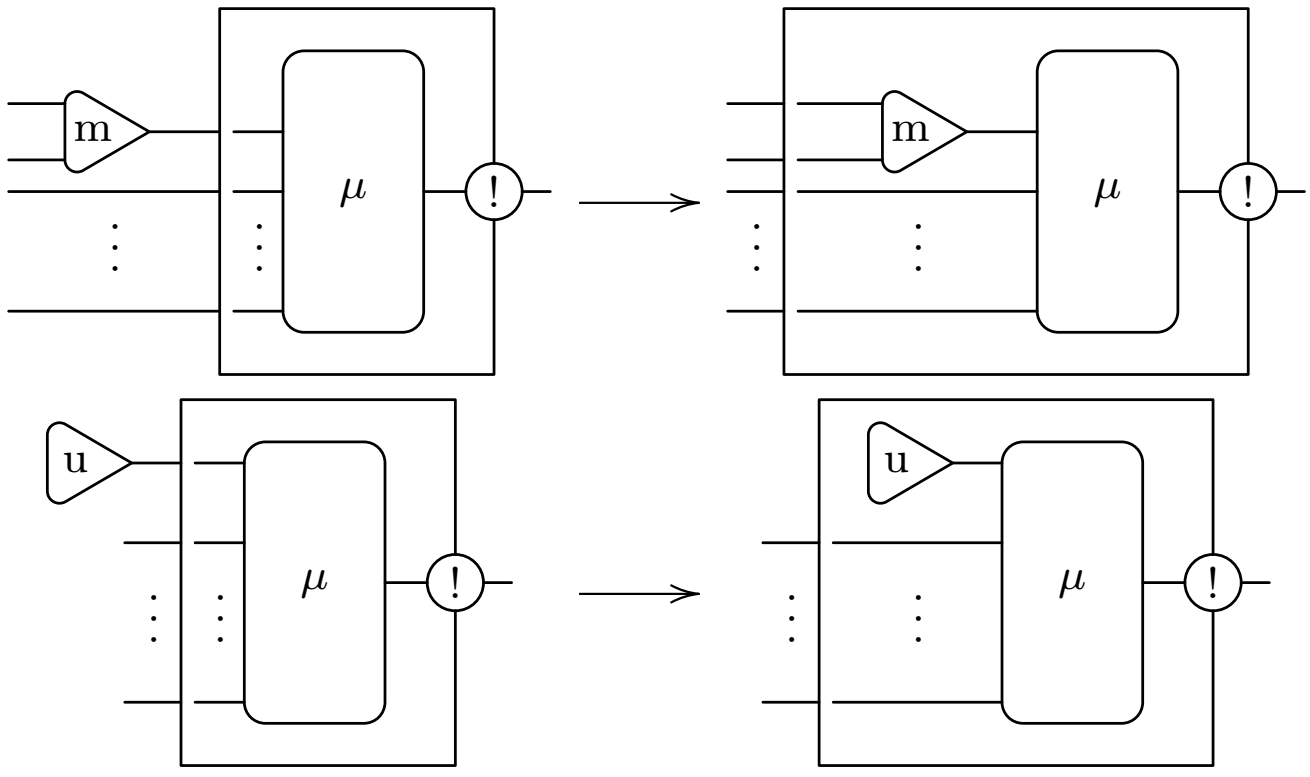
Routage



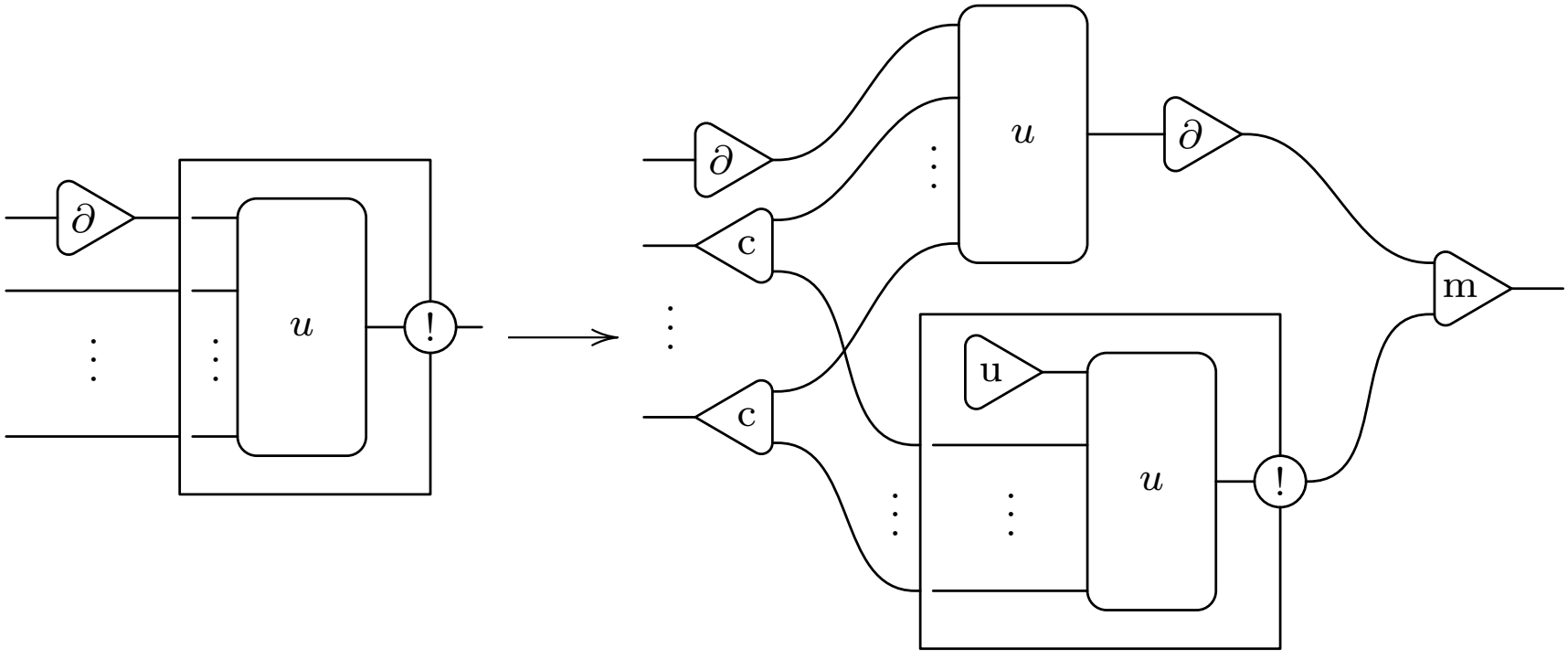
Routage et communication



Avec boîtes

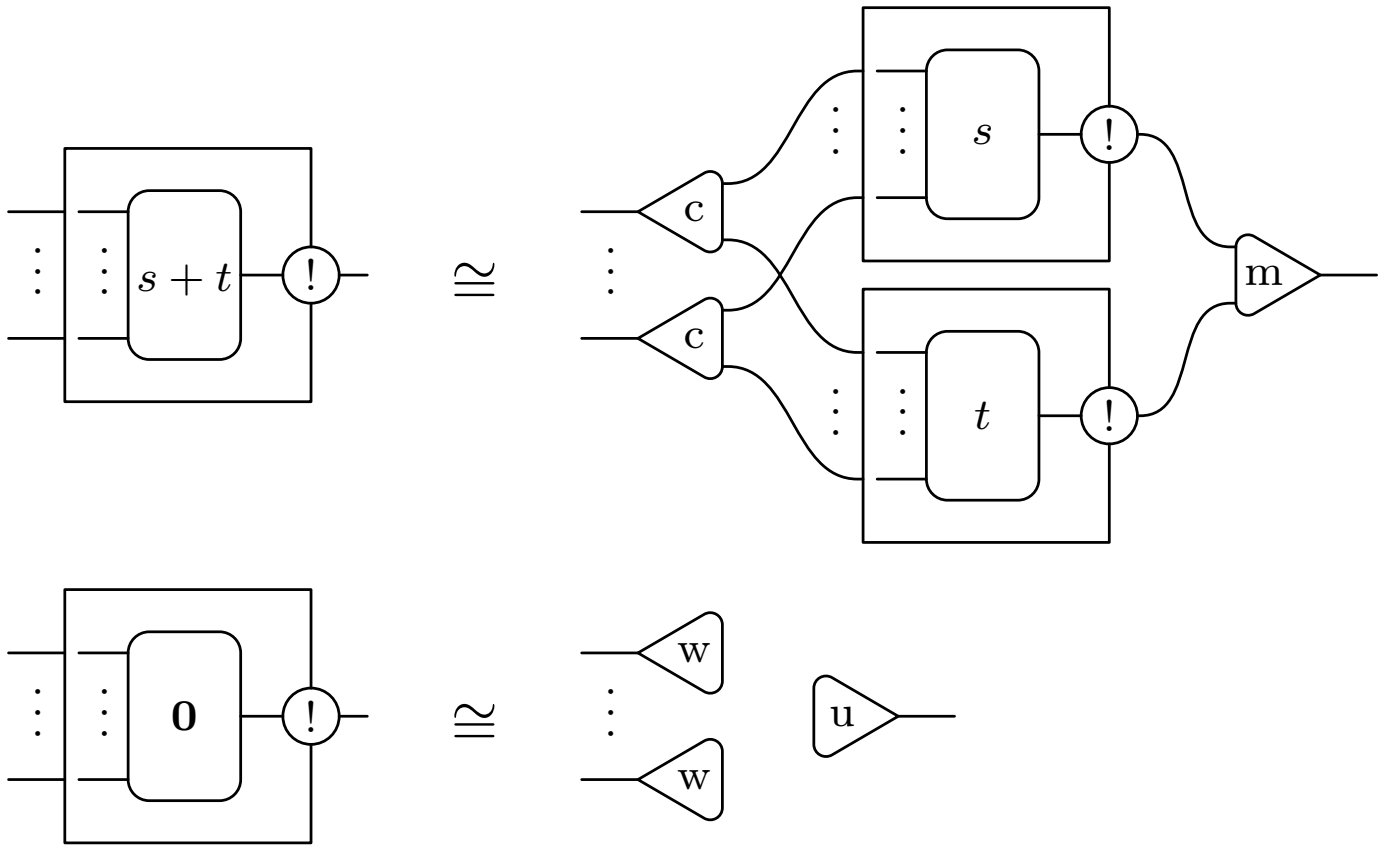


Loi de la chaîne



$$(f[u])' = f'[u] \times u' = (\mathrm{D}f \cdot u')[u]$$

Convolution



Réunion

Réseaux différentiels avec boîtes

Réunion

Réseaux différentiels avec boîtes

+ règles structurelles sur les négatifs.

Réunion

Réseaux différentiels avec boîtes

+ règles structurelles sur les négatifs.

- Pas de nouvelle interaction.

Réunion

Réseaux différentiels avec boîtes

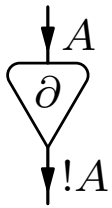
+ règles structurelles sur les négatifs.

- Pas de nouvelle interaction.
- Correspond au $\lambda\mu$-calcul différentiel [Vau07b].

Réunion

Réseaux différentiels avec boîtes

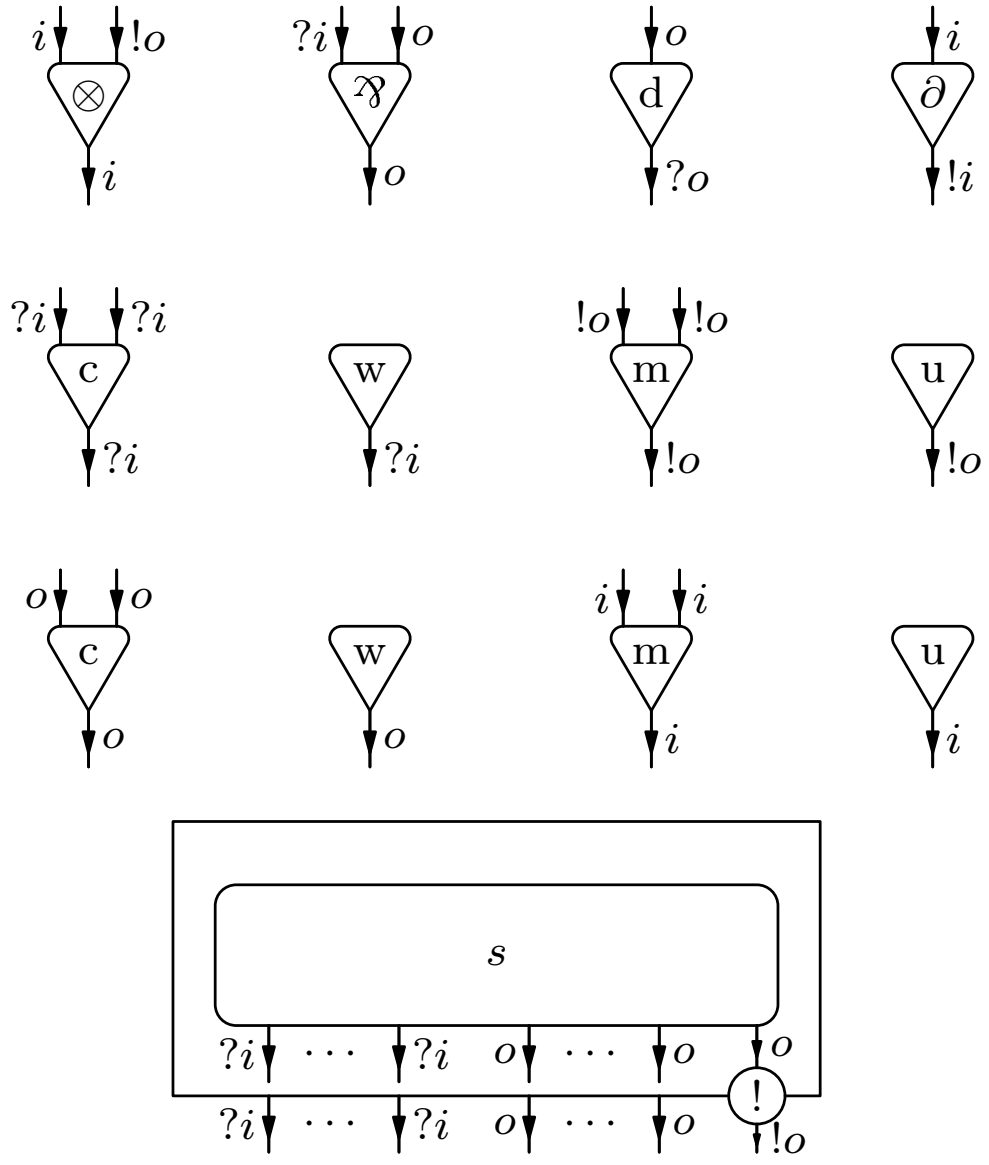
+ règles structurelles sur les négatifs.

- Pas de nouvelle interaction.
- Correspond au $\lambda\mu$-calcul différentiel [Vau07b].
- 

met à mal la polarisation.

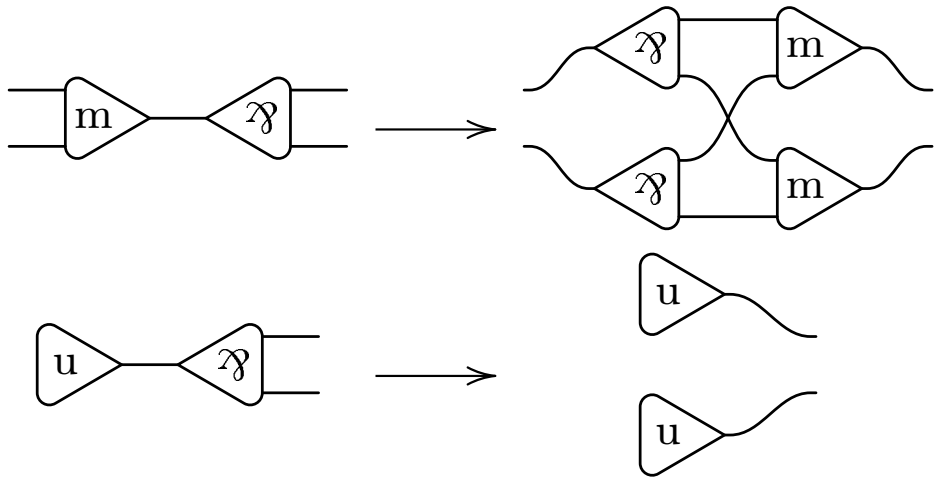
Extension polarisée des réseaux différentiels

Idée : on étend les règles (co)structurelles.



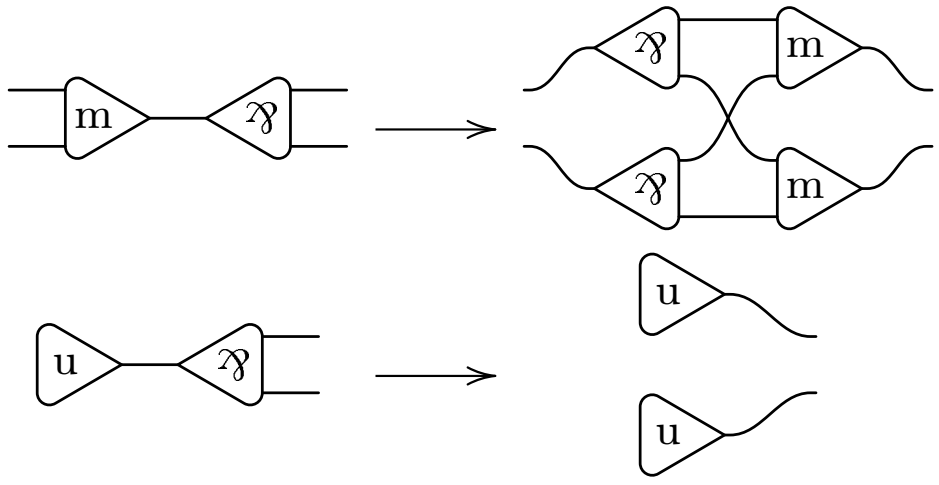
Réseaux différentiels polarisés

Nouveaux redex typables :



Réseaux différentiels polarisés

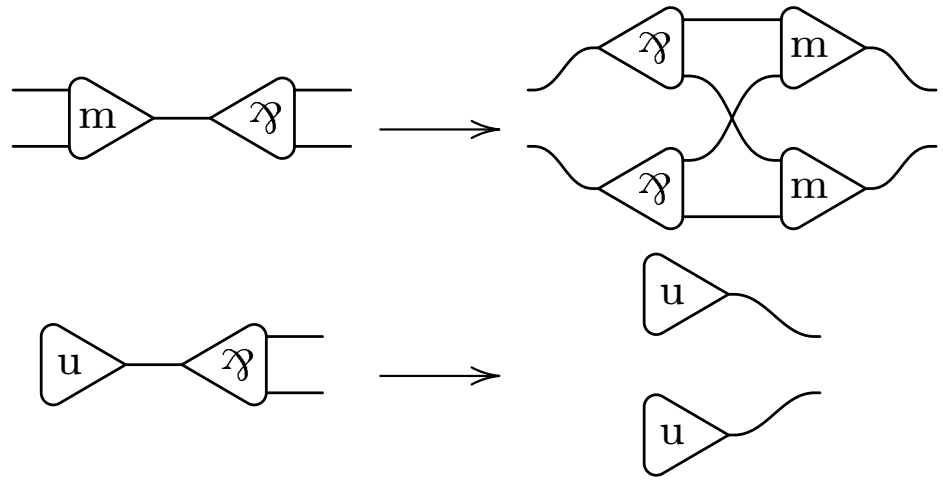
Nouveaux redex typables :



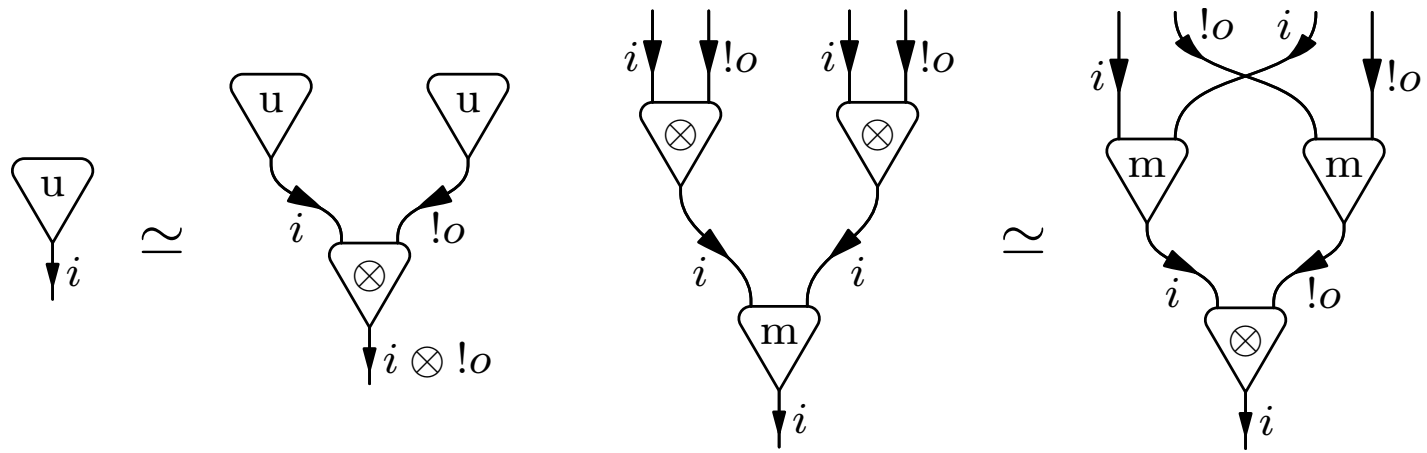
En accord avec une sémantique dans le modèle relationnel multiensembliste.

Réseaux différentiels polarisés

Nouveaux redex typables :



En accord avec une sémantique dans le modèle relationnel multiensembliste.



$\bar{\lambda}\mu$-calcul [Her95, CH00].

Morphologie :

$$
\begin{array}{lrcl}
\text{Termes :} & s, t & ::= & x \mid \lambda x\, s \mid \mu\alpha\, c \\
\text{Piles :} & e & ::= & \alpha \mid s \cdot e \\
\text{Commandes :} & c & ::= & \langle s\, , e \rangle
\end{array}
$$

$\bar{\lambda}\mu$-calcul [Her95, CH00] : un calcul pour les séquents.

Morphologie :

Termes :	s, t	$::=$	$x \mid \lambda x\, s \mid \mu\alpha\, c$	*(conclusions)*
Piles :	e	$::=$	$\alpha \mid s \cdot e$	*(hypothèses)*
Commandes :	c	$::=$	$\langle s\, , e \rangle$	*(coupures)*

$\bar{\lambda}\mu$-calcul [Her95, CH00] : un calcul pour les séquents.

Morphologie :

$$\begin{array}{lrcll} \text{Termes :} & s, t & ::= & x \mid \lambda x\, s \mid \mu\alpha\, c & \textit{(conclusions)} \\ \text{Piles :} & e & ::= & \alpha \mid s \cdot e & \textit{(hypothèses)} \\ \text{Commandes :} & c & ::= & \langle s\, , e \rangle & \textit{(coupures)} \end{array}$$

Réductions :

$$\begin{array}{rcl} \langle \lambda x\, s\, , t \cdot e \rangle & \rightarrow & \langle s\, [x := t]\, , e \rangle \\ \langle \mu\alpha\, c\, , e \rangle & \rightarrow & c\, [\alpha := e] \end{array}$$

$\bar{\lambda}\mu$-calcul [Her95, CH00] : un calcul pour les séquents.

Morphologie :

$$\begin{array}{lrcll} \text{Termes :} & s, t & ::= & x \mid \lambda x\, s \mid \mu\alpha\, c & \textit{(conclusions)} \\ \text{Piles :} & e & ::= & \alpha \mid s \cdot e & \textit{(hypothèses)} \\ \text{Commandes :} & c & ::= & \langle s\, , e \rangle & \textit{(coupures)} \end{array}$$

Réductions :

$$\begin{array}{rcl} \langle \lambda x\, s\, , t \cdot e \rangle & \rightarrow & \langle s\, [x := t]\, , e \rangle \\ \langle \mu\alpha\, c\, , e \rangle & \rightarrow & c\, [\alpha := e] \end{array}$$

Codage de l'application :

$$(s)\, t = \mu\alpha\, \langle s\, , t \cdot \alpha \rangle$$

$\bar{\lambda}\mu$-calcul [Her95, CH00] : un calcul pour les séquents.

Morphologie :

$$\begin{array}{lrcll} \text{Termes :} & s, t & ::= & x \mid \lambda x\, s \mid \mu\alpha\, c & \textit{(conclusions)} \\ \text{Piles :} & e & ::= & \alpha \mid s \cdot e & \textit{(hypothèses)} \\ \text{Commandes :} & c & ::= & \langle s\, , e \rangle & \textit{(coupures)} \end{array}$$

Réductions :

$$\begin{array}{rcl} \langle \lambda x\, s\, , t \cdot e \rangle & \to & \langle s\, [x := t]\, , e \rangle \\ \langle \mu\alpha\, c\, , e \rangle & \to & c\, [\alpha := e] \end{array}$$

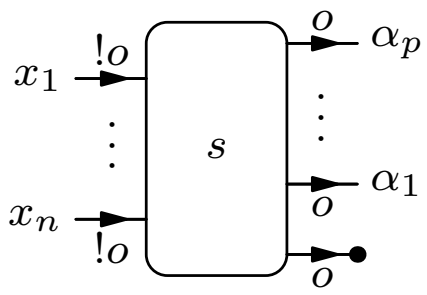
Codage de l'application :

$$(s)\, t = \mu\alpha\, \langle s\, , t \cdot \alpha \rangle$$

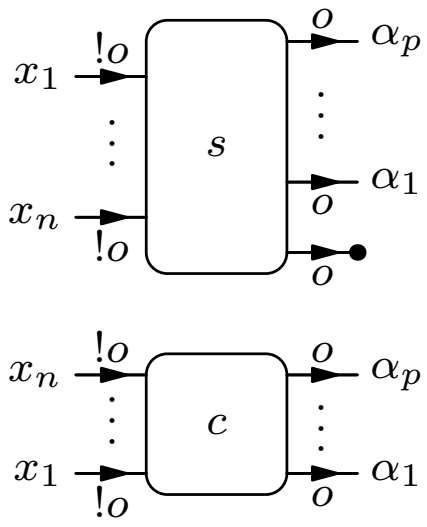
η-expansion :

$$s \leftarrow_\eta \lambda x\, \mu\alpha\, \langle s\, , x \cdot \alpha \rangle$$

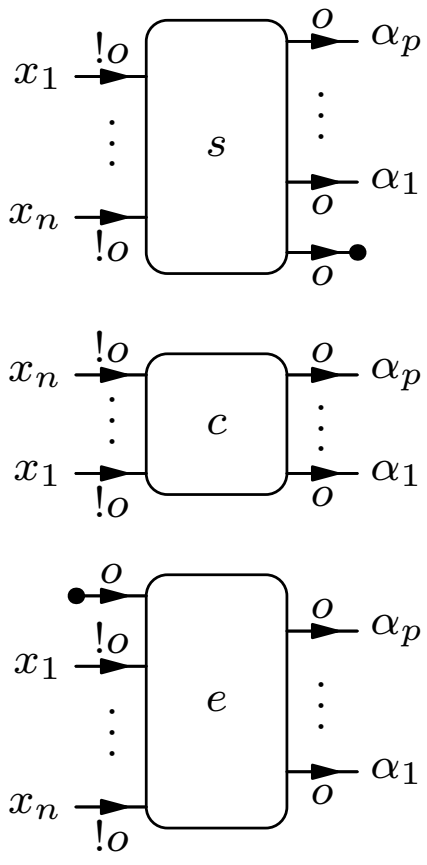
Traduction dans LLP



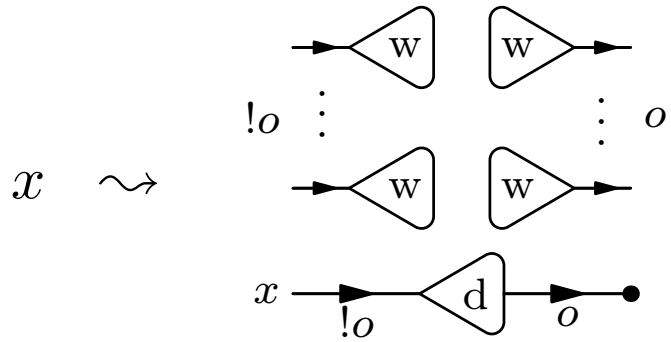
Traduction dans LLP



Traduction dans LLP

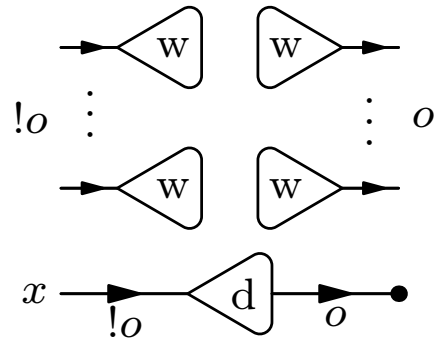


Traduction : termes

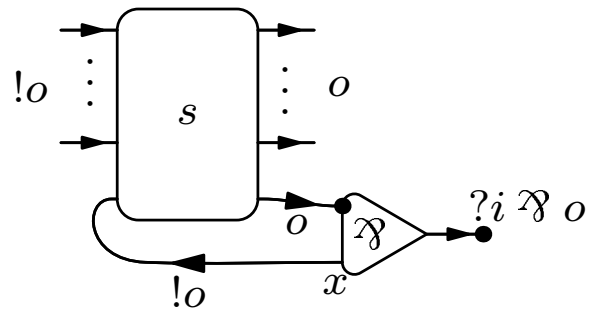


Traduction : termes

$x \quad \rightsquigarrow$

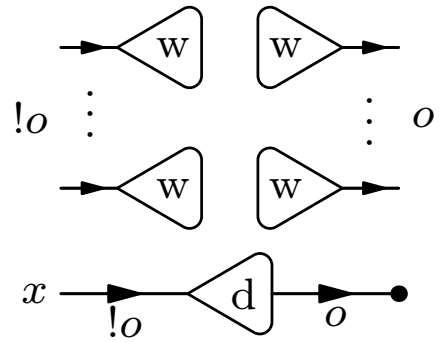


$\lambda x\, s \quad \rightsquigarrow$

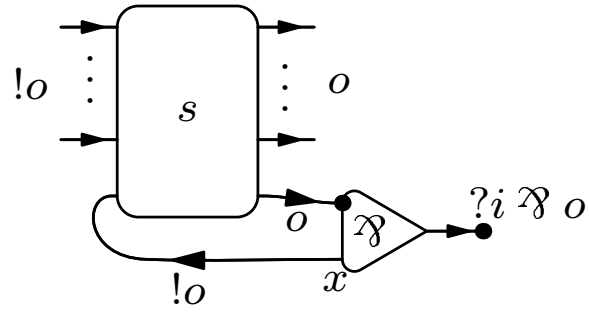


Traduction : termes

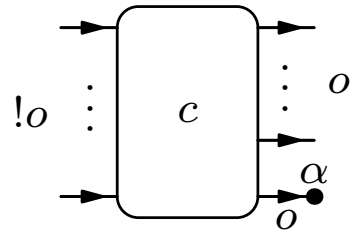
$x \rightsquigarrow$



$\lambda x\, s \rightsquigarrow$

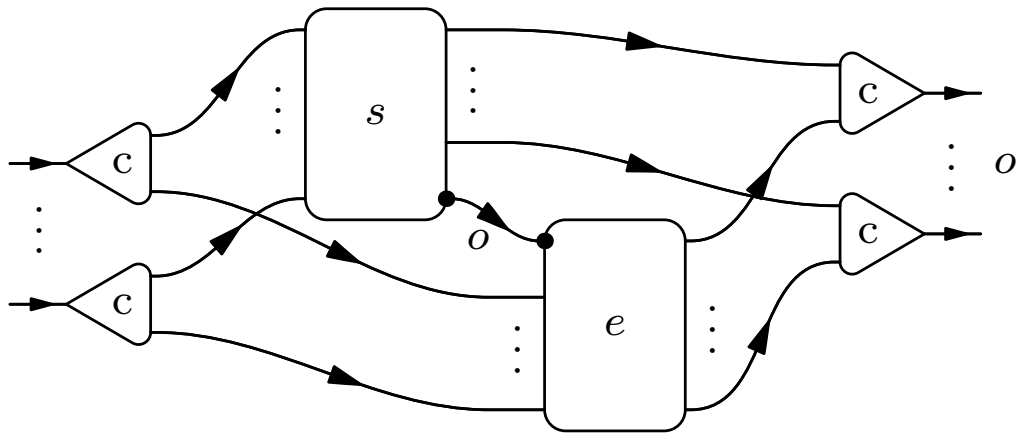


$\mu\alpha\, c \rightsquigarrow$



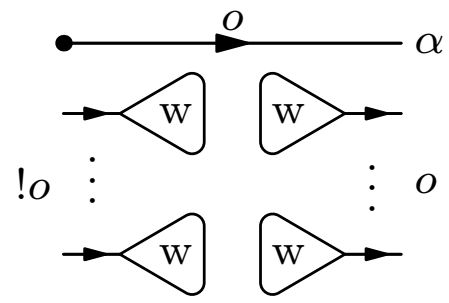
Traduction : commandes

$\langle s\,,\,e\rangle \quad \rightsquigarrow$

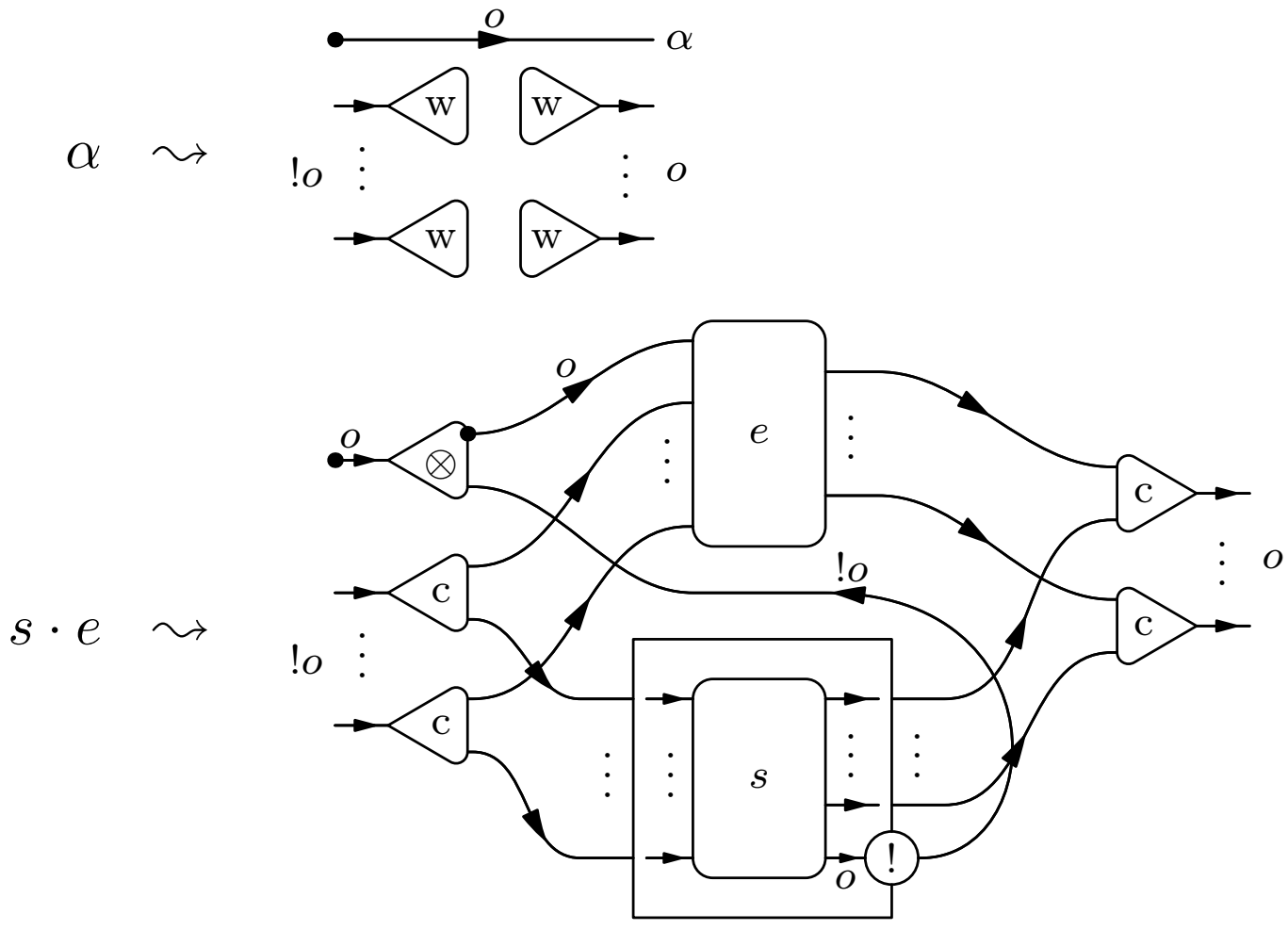


Traduction : piles

$\alpha \quad \rightsquigarrow$



Traduction : piles



$\bar{\lambda}\mu$-calcul avec produit de convolution

Interprétation calculatoire de m et u sur i ?

$\bar{\lambda}\mu$-calcul avec produit de convolution

Interprétation calculatoire de m et u sur i ?
i : type des piles.

$\bar{\lambda}\mu$-calcul avec produit de convolution

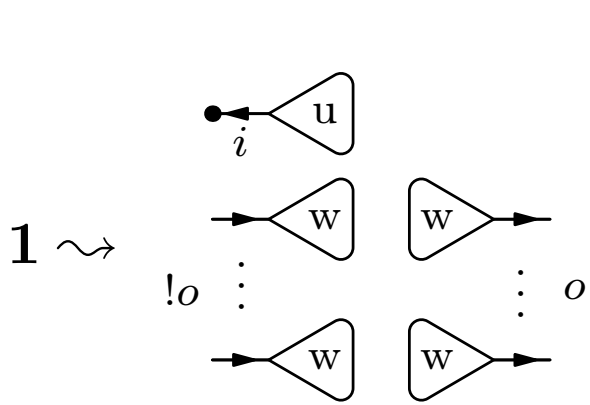
Interprétation calculatoire de m et u sur i ?
i : type des piles.

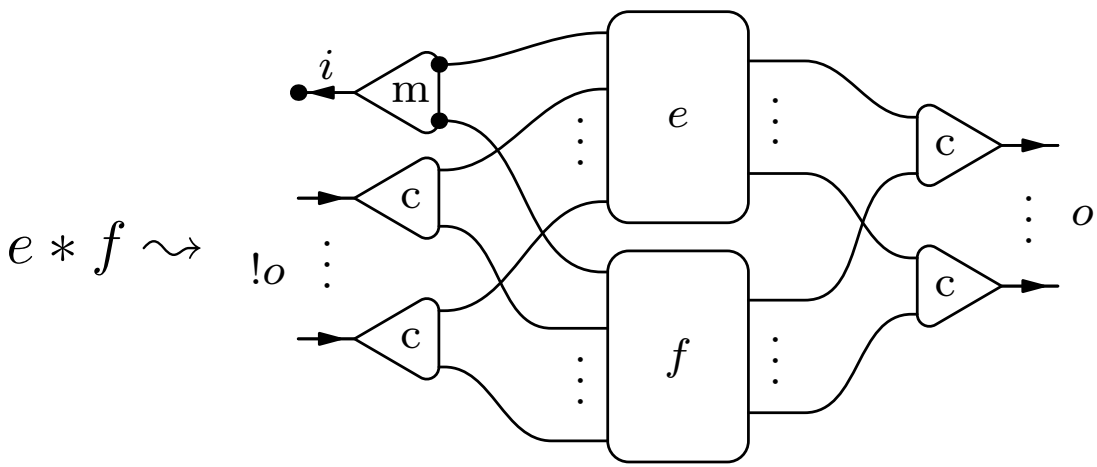
$$
\begin{array}{rcl}
s, t & ::= & x \mid \lambda x\, s \mid \mu\alpha\, c \\
e, f & ::= & \alpha \mid s \cdot e \mid \mathbf{1} \mid e * f \\
c & ::= & \langle s\, , e \rangle
\end{array}
$$

$\bar{\lambda}\mu$-calcul avec produit de convolution

Interprétation calculatoire de m et u sur i ?
i : type des piles.

$$
\begin{array}{rcl}
s, t & ::= & x \mid \lambda x\, s \mid \mu\alpha\, c \\
e, f & ::= & \alpha \mid s \cdot e \mid \mathbf{1} \mid e * f \\
c & ::= & \langle s\, , e \rangle
\end{array}
$$



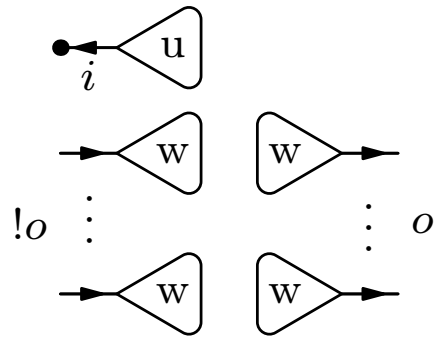


$\bar{\lambda}\mu$-calcul avec produit de convolution

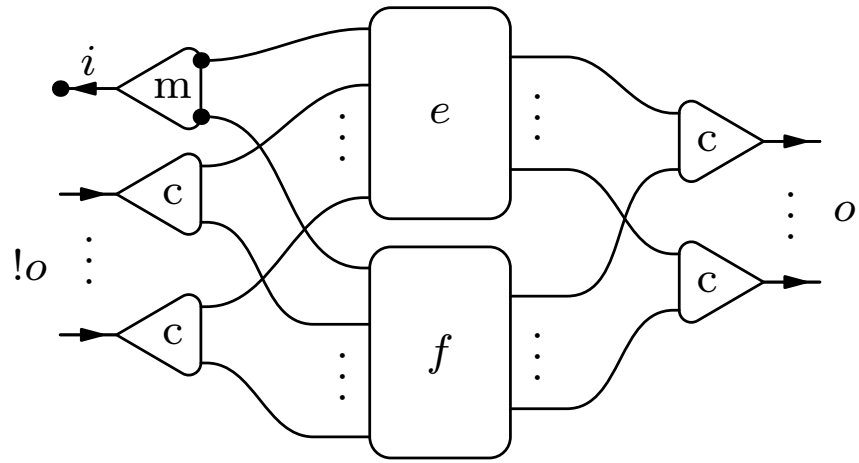
Interprétation calculatoire de m et u sur i ?
i : type des piles.

$$
\begin{array}{rcl}
s, t & ::= & x \mid \lambda x\, s \mid \mu\alpha\, c \mid \mathbf{0} \mid s + t \\
e, f & ::= & \alpha \mid s \cdot e \mid \mathbf{1} \mid e * f \mid \mathbf{0} \mid e + f \\
c & ::= & \langle s\, , e \rangle \mid \mathbf{0} \mid c + c'
\end{array}
$$

$\mathbf{1} \rightsquigarrow$

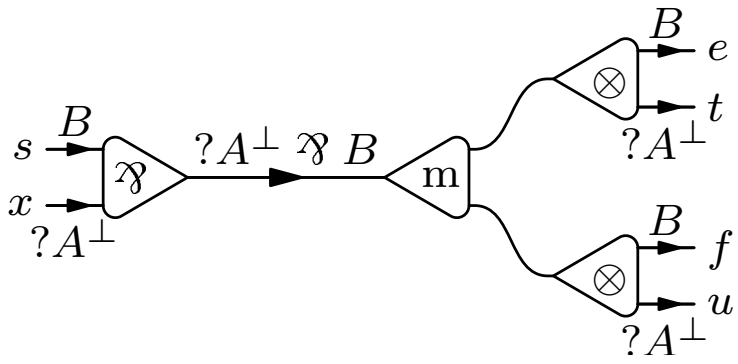


$e * f \rightsquigarrow$



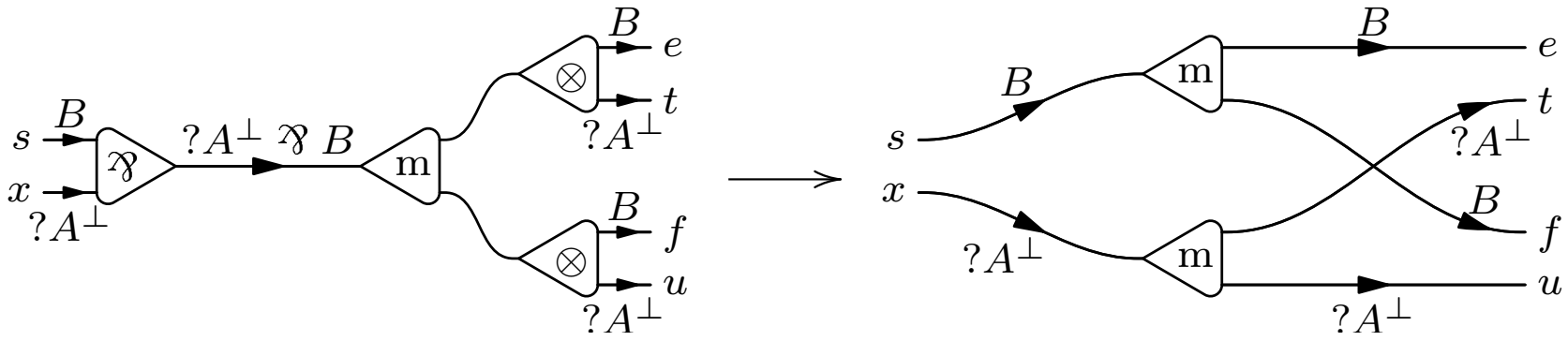
Réduction de routage $\rightsquigarrow$ sommes.

Réduction



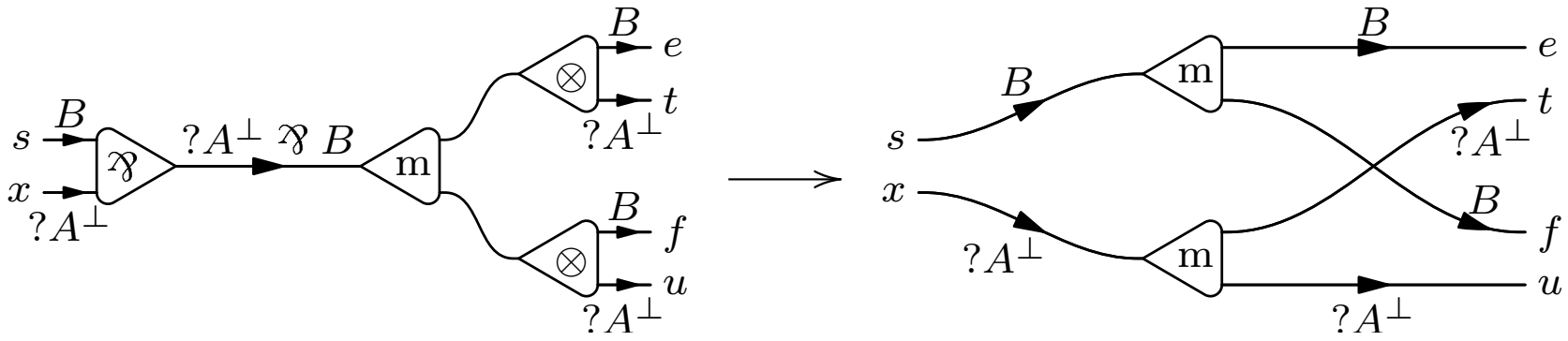
$$\langle \lambda x\, s\ ,\ (t \cdot e) * (u \cdot f) \rangle$$

Réduction



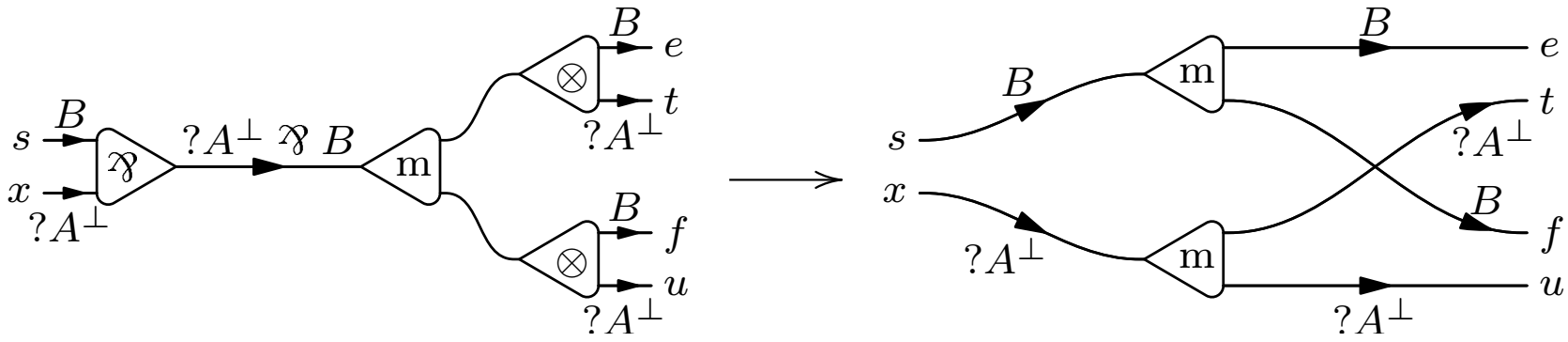
$$\langle \lambda x\, s\, ,\, (t \cdot e) * (u \cdot f) \rangle$$

Réduction



$$\langle \lambda x\, s\ ,\ (t \cdot e) * (u \cdot f) \rangle \ \rightarrow \langle s\,[x := t + u]\ ,\ e * f \rangle$$

Réduction



$$\langle \lambda x\, s\ ,\ (t \cdot e) * (u \cdot f) \rangle \ \rightarrow \langle s\,[x := t + u]\ ,\ e * f \rangle$$

$$(u \cdot e) * (v \cdot f) \simeq (t + u) \cdot (e * f)$$

Réduction

$$
\begin{array}{rcl}
\langle \mu\alpha\, c \,,\, e \rangle & \to & c\,[\alpha := e] \\
\langle \lambda x\, s \,,\, (t \cdot e) * (u \cdot f) \rangle & \to & \langle s\,[x := t + u] \,,\, e * f \rangle \\
\langle \lambda x\, s \,,\, \mathbf{1} \rangle & \to & \langle s\,[x := \mathbf{0}] \,,\, \mathbf{1} \rangle
\end{array}
$$

Réduction

$$
\begin{array}{rcl}
\langle \mu\alpha\, c\, ,\, e\rangle & \rightarrow & c\,[\alpha := e] \\
\langle \lambda x\, s\, ,\, (t \cdot e) * f\rangle & \rightarrow & \langle \lambda y\, \mu\beta\, \langle s\,[x := y + t]\, ,\, e * \beta\rangle\, ,\, f\rangle \\
\langle \lambda x\, s\, ,\, \mathbf{1}\rangle & \rightarrow & \langle s\,[x := \mathbf{0}]\, ,\, \mathbf{1}\rangle
\end{array}
$$

Réduction

$$\begin{array}{rcl} \langle \mu\alpha\, c \,,\, e\rangle & \to & c\,[\alpha := e] \\ \langle \lambda x\, s \,,\, (t \cdot e) * f\rangle & \to & \langle \lambda y\, \mu\beta\, \langle s\,[x := y + t] \,,\, e * \beta\rangle \,,\, f\rangle \\ \langle \lambda x\, s \,,\, \mathbf{1}\rangle & \to & \langle s\,[x := \mathbf{0}] \,,\, \mathbf{1}\rangle \end{array}$$

Formulations équivalentes, modulo :

$$\langle s \,,\, e * f\rangle \leftarrow_{\eta'} \langle \lambda x\, \mu\alpha\, \langle s \,,\, e * (x \cdot \alpha)\rangle \,,\, f\rangle\,.$$

Produit de convolution sur les distributions

[Sch66]
Si e et f sont des distributions à supports compacts et φ est une fonction de test :

$$\langle \lambda z \, \varphi(z) \, , e * f \rangle = \langle \lambda y \, \langle \lambda x \, \varphi(x+y) \, , e \rangle \, , f \rangle$$

Produit de convolution sur les piles

On a

$$\begin{array}{lll} \langle \lambda z\, s\, , e * f \rangle & \leftarrow_{\eta'} & \langle \lambda y\, \mu\beta\, \langle \lambda z\, s\, , e * (y \cdot \beta) \rangle\, , f \rangle \\ & \rightarrow & \langle \lambda y\, \mu\beta\, \langle \lambda x\, \mu\alpha\, \langle s\, [z := x + y]\, , \alpha * \beta \rangle\, , e \rangle\, , f \rangle \end{array}$$

Produit de convolution sur les piles

On a

$$\begin{array}{rcl} \langle \lambda z\, s\, , e * f \rangle & \leftarrow_{\eta'} & \langle \lambda y\, \mu\beta\, \langle \lambda z\, s\, , e * (y \cdot \beta) \rangle\, , f \rangle \\ & \rightarrow & \langle \lambda y\, \mu\beta\, \langle \lambda x\, \mu\alpha\, \langle s\, [z := x + y]\, , \alpha * \beta \rangle\, , e \rangle\, , f \rangle \end{array}$$

À comparer avec :

$$\langle \lambda z\, \varphi(z)\, , e * f \rangle = \langle \lambda y\, \langle \lambda x\, \varphi(x + y)\, , e \rangle\, , f \rangle$$

Produit de convolution sur les piles

On a

$$\begin{array}{rcl} \langle \lambda z\, s\, , e * f \rangle & \leftarrow_{\eta'} & \langle \lambda y\, \mu\beta\, \langle \lambda z\, s\, , e * (y \cdot \beta) \rangle\, , f \rangle \\ & \rightarrow & \langle \lambda y\, \mu\beta\, \langle \lambda x\, \mu\alpha\, \langle s\, [z := x + y]\, , \alpha * \beta \rangle\, , e \rangle\, , f \rangle \end{array}$$

À comparer avec :

$$\langle \lambda z\, \varphi(z)\, , e * f \rangle = \langle \lambda y\, \langle \lambda x\, \varphi(x + y)\, , e \rangle\, , f \rangle$$

φ est à valeurs scalaires ($\perp$) alors que s a un type de sortie.

Désynchronisation

On pose :

$$\begin{aligned} s^{!} &= s \cdot \mathbf{1} \\ \uparrow e &= \mathbf{0} \cdot e \end{aligned}$$

Désynchronisation

On pose :

$$\begin{aligned} s^{!} &= s \cdot \mathbf{1} \\ \uparrow e &= \mathbf{0} \cdot e \end{aligned}$$

Alors

$$s \cdot e \simeq s^{!} * \uparrow e$$

Désynchronisation

On pose :

$$\begin{aligned} s^{!} &= s \cdot \mathbf{1} \\ \uparrow e &= \mathbf{0} \cdot e \end{aligned}$$

Alors

$$s \cdot e \simeq s^{!} * \uparrow e$$

$$\begin{aligned} \left\langle \lambda x\, s\, ,\, t^{!} * f \right\rangle &\rightarrow \langle \lambda x\, s\, [x := x + t]\, ,\, f \rangle \\ \langle \lambda x\, s\, ,\, \uparrow e * f \rangle &\rightarrow \langle \lambda x\, \mu\alpha\, \langle s\, ,\, e * \alpha \rangle\, ,\, f \rangle\, . \end{aligned}$$

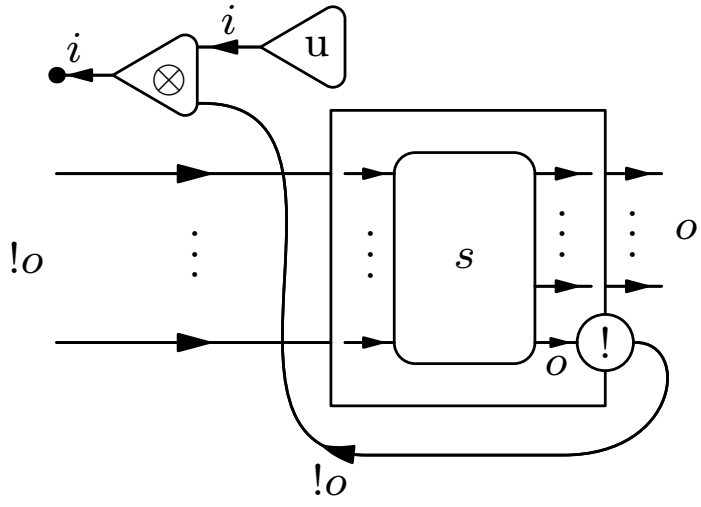
Désynchronisation

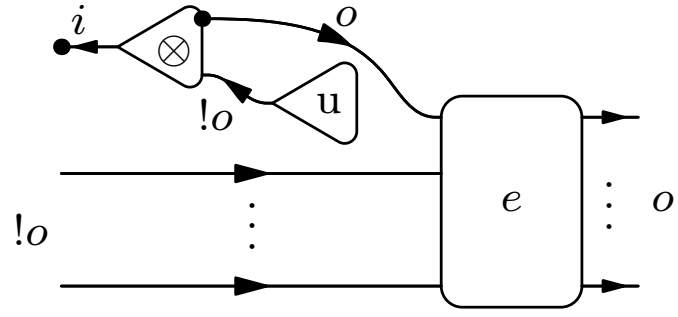
On pose :

$$\begin{aligned} s^{!} &= s \cdot \mathbf{1} \\ \uparrow e &= \mathbf{0} \cdot e \end{aligned}$$

Alors

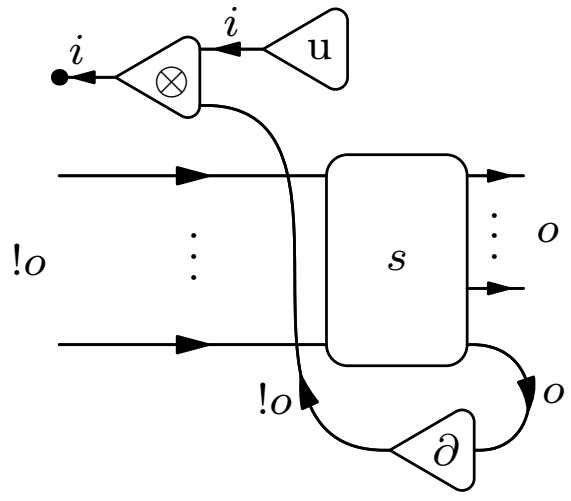
$$s \cdot e \simeq s^{!} * \uparrow e$$





$\bar{\lambda}\mu$-calcul différentiel

On ajoute la dérivée : $[s]$



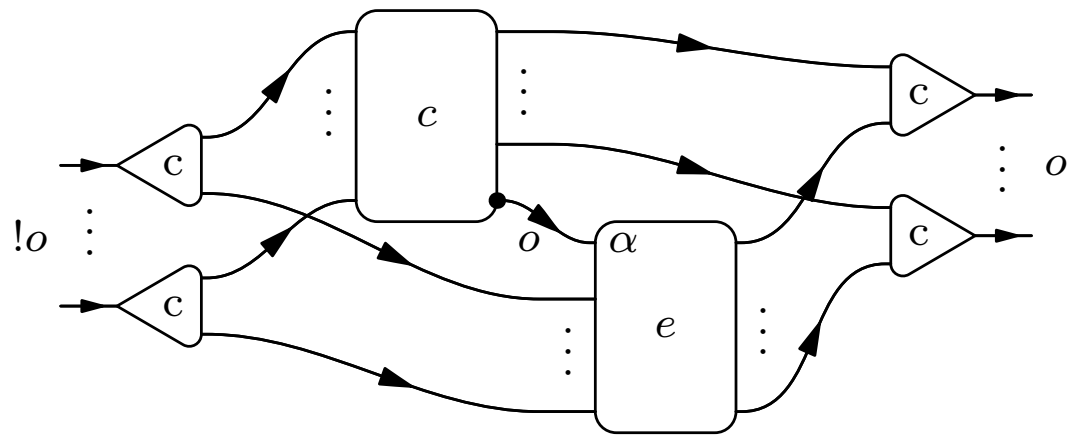
Réduction

$$
\begin{array}{rcl}
\langle \lambda x\, s\, ,\, \mathbf{1} \rangle & \to & \langle s\, [x := \mathbf{0}]\, ,\, \mathbf{1} \rangle \\
\left\langle \lambda x\, s\, ,\, t^{!} * f \right\rangle & \to & \langle \lambda x\, s\, [x := t + x]\, ,\, f \rangle \\
\langle \lambda x\, s\, ,\, \uparrow e * f \rangle & \to & \langle \lambda x\, \mu\alpha\, \langle s\, ,\, e * \alpha \rangle\, ,\, f \rangle \\
\langle \lambda x\, s\, ,\, [t] * f \rangle & \to & \left\langle \lambda x \left(\dfrac{\partial\, s}{\partial\, x} \cdot t \right) ,\, f \right\rangle
\end{array}
$$

Réduction

$$
\begin{array}{rcl}
\langle \lambda x\, s\, ,\, \mathbf{1} \rangle & \to & \langle s\, [x := \mathbf{0}]\, ,\, \mathbf{1} \rangle \\
\left\langle \lambda x\, s\, ,\, t^{!} * f \right\rangle & \to & \langle \lambda x\, s\, [x := t + x]\, ,\, f \rangle \\
\langle \lambda x\, s\, ,\, \uparrow e * f \rangle & \to & \langle \lambda x\, \mu\alpha\, \langle s\, ,\, e * \alpha \rangle\, ,\, f \rangle \\
\langle \lambda x\, s\, ,\, [t] * f \rangle & \to & \left\langle \lambda x \left(\dfrac{\partial\, s}{\partial\, x} \cdot t \right) ,\, f \right\rangle \\
\langle \mu\alpha\, c\, ,\, e \rangle & \to & \langle c\, ,\, e \rangle_{\alpha}
\end{array}
$$

Coupure nommée



Coupure nommée

$$
\begin{aligned}
\langle \_ \,, \beta \rangle_\alpha &= \_\,[\alpha := \beta] \\
\langle \_ \,, s^! \rangle_\alpha &= \_\,\left[\alpha := s^!\right] \\
\langle \_ \,, \mathbf{1} \rangle_\alpha &= \_\,[\alpha := \mathbf{1}] \\
\langle \_ \,, \uparrow e \rangle_\alpha &= \langle \_\,[\alpha := \uparrow\alpha] \,, e \rangle_\alpha \\
\langle \_ \,, [s] \rangle_\alpha &= (\mathrm{D}_\alpha \_ \cdot s)\,[\alpha := \mathbf{1}] \\
\langle \_ \,, (e * f) \rangle_\alpha &= \left\langle \langle \_\,[\alpha := \beta * \alpha] \,, e \rangle_\beta \,, f \right\rangle_\alpha
\end{aligned}
$$

À suivre. . .

- Concurrence [EL07].
- Formule de Taylor [ER06, dC07] :

$$s^! \stackrel{?}{=} \sum_{n=0}^{\infty} \frac{1}{n!} [s]^n .$$

- Appel par valeur ?

Fin

Biblio


Pierre-Louis Curien and Hugo Herbelin.
The duality of computation.
ACM SIGPLAN Notices, 35(9) :233–243, 2000.


Daniel de Carvalho.
Sémantiques de la logique linéaire et temps de calcul.
Thèse de doctorat, Université Aix-Marseille II, 2007.


Thomas Ehrhard.
On Köthe sequence spaces and linear logic.
Mathematical Structures in Computer Science, 12 :579–623, 2001.


Thomas Ehrhard.
Finiteness spaces.
Mathematical. Structures in Comp. Sci., 15(4) :615–646, 2005.


Thomas Ehrhard and Olivier Laurent.
Interpreting a finitary pi-calculus in differential interaction nets.
In Luis Caires and Vasco T. Vasconcelos, editors, *Concurrency Theory (CONCUR '07)*, volume 4703 of *Lecture Notes in Computer Science*, pages 333–348. Springer, September 2007.


Thomas Ehrhard and Laurent Regnier.
The differential lambda-calculus.
Theoretical Computer Science, 309 :1–41, 2003.


Thomas Ehrhard and Laurent Regnier.
Differential interaction nets.

Electr. Notes Theor. Comput. Sci., 123 :35–74, 2005.


Thomas Ehrhard and Laurent Regnier.
Uniformity and the taylor expansion of ordinary lambda-terms.
Archives ouvertes `oai:hal.archives-ouvertes.fr:hal-00150275_v1`, à paraître dans Theoretical Computer Science, 2006.


Jean-Yves Girard.
Linear logic.
Theor. Comput. Sci., 50 :1–102, 1987.


Jean-Yves Girard.
A new constructive logic : Classical logic.
Mathematical Structures in Computer Science, 1(3) :255–296, 1991.


Hugo Herbelin.
Séquents qu'on calcule.
Thèse d'université, Université Paris 7, 1995.


Olivier Laurent.
Etude de la polarisation en logique.
Thèse de doctorat, Université Aix-Marseille II, March 2002.


Michel Parigot.
$\lambda\mu$-calculus : An algorithmic interpretation of classical natural deduction.
In *LPAR '92 : Proceedings of the International Conference on Logic Programming and Automated Reasoning*, pages 190–201, London, UK, 1992. Springer-Verlag.


Laurent Schwartz.

Théorie des distributions.
Hermann, 1966.


Lionel Vaux.
Convolution $\bar{\lambda}\mu$-calculus.
In Simona Ronchi Della Rocca, editor, *TLCA*, volume 4583 of *Lecture Notes in Computer Science*, pages 381–395. Springer, 2007.


Lionel Vaux.
The differential $\lambda\mu$-calculus.
Theor. Comput. Sci., 379(1-2) :166–209, 2007.


Lionel Vaux.
On linear combinations of λ-terms.
In Franz Baader, editor, *RTA*, volume 4533 of *Lecture Notes in Computer Science*, pages 374–388. Springer, 2007.