De λ à π : encodages et dualité

Stage supervisé par Daniel Hirschkoff et Davide Sangiorgi

Jean-Marie Madiot

LIP, ENS de Lyon

Vendredi 2 septembre 2011

Algèbre de processus avec *passage de nom*

$$P ::= 0 \mid P|P \mid a(x).P \mid \bar{a}b \mid (\nu a)P \mid !a(x).P$$

Sémantique : réduction ($\rightarrow$ et $\equiv$) ou LTS ($\xrightarrow{\mu}$) :

$$\frac{P \xrightarrow{(\nu\tilde{c})\bar{a}\langle\tilde{b}\rangle} P' \quad Q \xrightarrow{a(\tilde{x})} Q'}{P \mid Q \xrightarrow{\tau} (\nu\tilde{c})(P' \mid Q'[\tilde{b}/\tilde{x}])}\ \mathit{comm}_l \qquad \frac{}{\alpha.P \xrightarrow{\alpha} P}\ \mathit{pre}$$

$$\frac{P \xrightarrow{\mu} P' \quad \mathrm{bn}(\mu) \cap \mathrm{fn}(R) = \emptyset}{P \mid R \xrightarrow{\mu} P' \mid R}\ \mathit{par}_l \qquad \frac{P \xrightarrow{\mu} P' \quad x \notin \mathrm{n}(\mu)}{(\nu x)P \xrightarrow{\mu} (\nu x)P'}\ \mathit{new}$$

$$\frac{P \xrightarrow{(\nu\tilde{c})\bar{a}(\tilde{b})} P' \quad a \neq d, c \in \tilde{b} - \tilde{c}}{(\nu d)P \xrightarrow{(\nu\tilde{c}d)\bar{a}(\tilde{b})} P'}\ \mathit{open} \qquad \frac{}{!\alpha.P \xrightarrow{\alpha} P \mid !\alpha.P}\ \mathit{bang}$$

Relations sur les processus :

- $\sim$ bisimilarité forte
- $\approx$ bisimilarité faible (insensible aux $\xrightarrow{\tau}$)
- $\lesssim$ expansion (davantage de $\xrightarrow{\tau}$ à droite)

De λ à π

- λ : substitution de termes
- π : substitution de noms

[Milner, 1992] : encodage du λ-calcul en *appel par nom* (cbn)

$$
\begin{aligned}
[\![x]\!]_p &= \overline{x}p \\
[\![\lambda x.M]\!]_p &= p(x,q).[\![M]\!]_q \\
[\![MN]\!]_p &= (\nu q)([\![M]\!]_q \mid (\nu x)(\overline{q}\langle x,p\rangle.!x(r).[\![N]\!]_r))
\end{aligned}
$$

(d'autres existent)

De λ à π : nouvel encodage

[van Bakel and Vigliotti, 2009] a proposé, pour le λ-calcul en appel par nom et réduction sous les λ *(strong cbn)* :

$$
\begin{aligned}
[\![x]\!]_p &= x(p').p' \rightarrow p \\
[\![\lambda x.M]\!]_p &= \overline{p}(x,q):[\![M]\!]_q \\
[\![MN]\!]_p &= (\nu q)([\![M]\!]_q \mid q(x,p').(p' \rightarrow p \mid !\overline{x}(r).[\![N]\!]_r))
\end{aligned}
$$

$$
\begin{aligned}
[\![x]\!]_p &= \overline{x}p \\
[\![\lambda x.M]\!]_p &= p(x,q).[\![M]\!]_q \\
[\![MN]\!]_p &= (\nu q)([\![M]\!]_q \mid (\nu x)(\overline{q}\langle x,p\rangle. \\
&\qquad !x(r).[\![N]\!]_r))
\end{aligned}
$$

Différences

$$\llbracket \lambda x.M \rrbracket_p = p(x,q).\llbracket M \rrbracket_q \qquad \text{Milner}$$
$$\llbracket \lambda x.M \rrbracket_p = \overline{p}(x,q){:}\llbracket M \rrbracket_q \qquad \text{van Bakel}$$

- *strong cbn* : passer de $a(x).$ à $a(x){:}$
- Les entrées dans $\llbracket \cdot \rrbracket_p$ sont des sorties dans $\llbracket \cdot \rrbracket_p$ (et vice versa)

. . . Dualité entrée/sortie ?

1. Deux encodages de λ vers π : sont-ils duaux ?
2. Dualité en mobilité interne : πI
3. Solution 1 : Lπ symétrisé, π_{io}
4. Solution 2 : π avec un opérateur de fil, πw

πI, le fragment dualisant

Communications en π-calcul :

$$\frac{P \xrightarrow{\bar{a}b} P' \quad Q \xrightarrow{a(x)} Q'}{P \mid Q \xrightarrow{\tau} P' \mid Q'[b/x]} \qquad \frac{P \xrightarrow{\bar{a}(b)} P' \quad Q \xrightarrow{a(x)} Q'}{P \mid Q \xrightarrow{\tau} (\nu b)(P' \mid Q'[b/x])}$$

Dualité :

- $a(x).P \rightsquigarrow (\nu x)\bar{a}x.P$
- $\bar{a}b \quad \rightsquigarrow \quad ?$

[Sangiorgi, 1996] π**I** : π-calcul à **mobilité interne** : pas d'émission de nom libre $\bar{a}b$, seulement des émissions de noms frais $(\nu b)\bar{a}b$ qu'on note $\bar{a}(b)$.

Dualité en π**I** :

$$P \xrightarrow{\mu} P' \Leftrightarrow \overline{P} \xrightarrow{\bar{\mu}} \overline{P'}$$

Traduire π vers πI

[Boreale, 1998], [Merro and Sangiorgi, 2004] : $\pi \to \mathrm{L}\pi \to \pi\mathrm{I}$

- seulement à partir d'un fragment de π:
  Lπ : π-calcul localisé:

$$a(x).P \in \mathrm{L}\pi \quad \Rightarrow \quad x \text{ non récepteur dans } P$$

par exemple, $x(y).y(z).0 \notin \mathrm{L}\pi$

- besoin de *liens* dans πI
  Liens : cas particulier de définitions récursives

$$a \hookrightarrow b := !a(x).\overline{b}(y).y \hookrightarrow x$$

- $(\mathrm{L}\pi \to \pi\mathrm{I})$: $[\![\bar{a}b]\!] := \bar{a}(x).x \hookrightarrow b$

De Lπ à πI

Problèmes :

- Lπ ne suffit pas pour $[\![\cdot]\!]_p$ et $[\![\cdot]\!]_p$
  $[\![\lambda x.\lambda y.y]\!]_p = p(x,q).q(y,r).\overline{x}r \notin \mathrm{L}\pi$
- π vers Lπ existe, mais trop difficile à analyser dans la dualité :

$$[\![a(x).P]\!]_\rho = \begin{cases} (\nu h)(\overline{z}h \mid h(x,y).[\![P]\!]_{\rho[y/x]}) & \text{si } \rho(a) = z \\ a(x,z).[\![P]\!]_{\rho[z/x]} & \text{sinon} \end{cases}$$

Pour la dualité on veut donc une traduction ((plus que Lπ) $\to$ πI)
(mais pas forcément $\pi \to \pi$I)

1. Deux encodages de λ vers π : sont-ils duaux ?
2. Dualité en mobilité interne : πI
3. Solution 1 : Lπ symétrisé, π_{io}
4. Solution 2 : π avec un opérateur de fil, πw

Solution 1

Première approche :

- relaxer Lπ pour prendre en compte la “capacité de réception” : π_{io}
- on garde des contraintes, pour mieux faire les liens
- adapter la traduction : $(\mathsf{L}\pi \to \pi\mathsf{I}) \leadsto (\pi_{io} \to \pi\mathsf{I})$

Avec des types d'entrée-sortie : π_{io}
(une restriction de [Pierce and Sangiorgi, 1996])

Types stricts : $T ::= iT \mid oT \mid \mu x.T$
Types hybrides : $\#T$

Pour les noms : $\dfrac{\Gamma(x) \in \{iT, \#T\}}{\Gamma \vdash x : iT}$ $\dfrac{\Gamma(x) \in \{oT, \#T\}}{\Gamma \vdash x : oT}$

Restriction : $\dfrac{\Gamma, a : \#T \vdash P}{\Gamma \vdash (\nu a)P}$

Entrées/sorties : $\dfrac{\Gamma \vdash a : iT \quad \Gamma, x : T \vdash P}{\Gamma \vdash a(x).P}$ $\dfrac{\Gamma \vdash a : oT \quad \Gamma \vdash v : T \quad \Gamma \vdash P}{\Gamma \vdash \overline{a}\langle v \rangle.P}$

Lois pour $\mid$: $\dfrac{}{\Gamma \vdash 0}$ $\dfrac{\Gamma \vdash \alpha.P}{\Gamma \vdash !\alpha.P}$ $\dfrac{\Gamma \vdash P_1 \quad \Gamma \vdash P_2}{\Gamma \vdash P_1 \mid P_2}$

Traduction de π_{io} vers πI

Lπ vers πI, c'était :

$$[\![\bar{a}b]\!] = \bar{a}(x).x \hookrightarrow b$$

Mais le lien dépend de l'utilisation de x dans le processus qui le reçoit...

Dans π_{io} on retrouve le type de a :

$$[\![\bar{a}b]\!]_\Gamma = \bar{a}(x).x \overset{T}{\hookrightarrow} b$$

où T est tel que $\Gamma(a) = \#T$ ou oT

Il reste à construire les liens $x \overset{T}{\hookrightarrow} b$

Liens pour les deux capacités

Liens à capacité d'émission ou de réception :

$$a \overset{oT}{\hookrightarrow} b := !a(x).\overline{b}(y).y \overset{T}{\hookrightarrow} x$$

$$a \overset{iT}{\hookrightarrow} b := !\overline{a}(x).b(y).x \overset{T}{\hookrightarrow} y$$

La traduction ($\mathsf{L}\pi \to \pi\mathsf{I}$) est un cas particulier de celle-ci :

- $\mathsf{L}\pi$, c'est π_{io} avec des types "*ooooo...*" (o^ω)
- $a \hookrightarrow b = a \overset{o^\omega}{\hookrightarrow} b$

Propriétés : les liens comme substitutions

- transitivité : $a \overset{T}{\hookrightarrow} c \;\lesssim\; (\nu b)(a \overset{T}{\hookrightarrow} b \mid b \overset{T}{\hookrightarrow} c)$
- substitution* : $[\![P]\!][b/a] \;\lesssim\; (\nu a)([\![P]\!] \mid a \overset{T}{\hookrightarrow} b)$
- *full abstraction** : $P \approx_l Q$ ssi $[\![P]\!] \asymp_a [\![Q]\!]$

$\approx_l$: bisimilarité de liens
$\asymp_a$ bisimilarité asynchrone *ground*
[Merro and Sangiorgi, 2004]

* : seulement prouvés dans le cas linéaire, c-à-d. les liens substituent des noms qui ne sont utilisés qu'une seule fois. C'est assez pour $[\![\,\cdot\,]\!]$ et $[\![\,\cdot\,]\!]$

Application : retour aux encodages

On étend la dualité de πI aux liens :

$$\overline{a \overset{oT}{\hookrightarrow} b} := a \overset{iT}{\hookrightarrow} b \qquad \overline{a \overset{iT}{\hookrightarrow} b} := a \overset{oT}{\hookrightarrow} b$$

Et alors on a pour tout λ-terme M, $[\![[\![M]\!]_p]\!] = \overline{[\![[\![M]\!]_p]\!]}$:

$$\begin{aligned}
[\![[\![x]\!]_p]\!] &= \overline{x}(p').p' \overset{i\gamma}{\hookrightarrow} p \\
[\![[\![\lambda x.M]\!]_p]\!] &= p(x,q){:}[\![[\![M]\!]_q]\!] \\
[\![[\![MN]\!]_p]\!] &= (\nu q)([\![[\![M]\!]_q]\!] \mid (\overline{q}(x,p'). \\
&\quad (p' \overset{i\gamma}{\hookrightarrow} p \mid !x(r).[\![[\![N]\!]_r]\!]))
\end{aligned}
\qquad
\begin{aligned}
[\![[\![x]\!]_p]\!] &= x(p').p' \overset{o\gamma}{\hookrightarrow} p \\
[\![[\![\lambda x.M]\!]_p]\!] &= \overline{p}(x,q){:}[\![[\![M]\!]_q]\!] \\
[\![[\![MN]\!]_p]\!] &= (\nu q)([\![[\![M]\!]_q]\!] \mid q(x,p'). \\
&\quad (p' \overset{o\gamma}{\hookrightarrow} p \mid !\overline{x}(r).[\![[\![N]\!]_r]\!]))
\end{aligned}$$

- $[\![\cdot]\!]$ et $[\![\cdot]\!]$ vérifient les conditions de la propriété de *full abstraction*.
- il apparaît que $[\![P]\!] \asymp_a [\![Q]\!]$ ssi $\overline{[\![P]\!]} \asymp_a \overline{[\![Q]\!]}$
  problèmes dans la preuve : permutation dans les traces

On obtient :

- Les encodages $[\![\cdot]\!]$ et $[\![\cdot]\!]$ sont duaux
- Ils induisent la même équivalence sur les λ-termes :

$$[\![[\![M]\!]_p]\!] \asymp_a [\![[\![N]\!]_p]\!] \iff [\![[\![M]\!]_p]\!] \asymp_a [\![[\![N]\!]_p]\!]$$

$$[\![M]\!]_p \approx_l [\![N]\!]_p \iff [\![M]\!]_p \approx_l [\![N]\!]_p$$

1. Deux encodages de λ vers π : sont-ils duaux ?

2. Dualité en mobilité interne : πI

3. Solution 1 : Lπ symétrisé, π_{io}

4. Solution 2 : π avec un opérateur de fil, πw

Solution 2

Deuxième approche : introduire un nouveau calcul : $\pi \to \pi\mathsf{w} \to \pi\mathsf{wl}$

- πw $=$ π-calcul $+$ un opérateur, le *fil*
- *fil* $=$ lien avec les capacités d'émission et de réception.
- …traduction et dualité plus faciles que πl+liens

πw: les fils

Idée :

- fil, $\{a = b\}$: nouvel opérateur, au lieu de $a \overset{T}{\hookrightarrow} b$
- $$\frac{}{\{a = b\} \xrightarrow{a=b} \{a = b\}} \qquad \frac{P \xrightarrow{\overline{a}c} P' \quad Q \xrightarrow{a=b} Q'}{P \mid Q \xrightarrow{\overline{b}c} P' \mid Q'}$$

Quelques conséquences, si on veut un calcul avec de bonnes propriétés :

- LTS : $\xrightarrow{\beta,E}$ où β est une action et E une classe d'équivalence de noms
- une action τ peut *attendre/nécessiter* un fil: $\tau/a = b$

πw: les fils

Exemples de règles du LTS :

$$\frac{P \xrightarrow{\bar{a}c} P' \quad Q \xrightarrow{b(x)} Q'}{P \mid Q \xrightarrow{\tau / a=b} P' \mid Q'} \qquad \frac{P \xrightarrow{\tau / a=b} P' \quad Q \xrightarrow{a=b} Q'}{P \mid Q \xrightarrow{\tau} P' \mid Q'}$$

Règle principale (communication) :

$$\frac{P \xrightarrow{\mu_1} P' \quad Q \xrightarrow{\mu_2} Q' \quad (\ell, \mathcal{C}) = \mu_1 \circ \mu_2}{P \mid Q \xrightarrow{\ell} \mathcal{C}[P', Q']}$$

où ℓ est :

- l'unification des classes d'équivalences de μ_1 et μ_2
- la composition des actions ($\bar{a}b \mid a(x) \to \tau, \dots$)

et $\mathcal{C}[P', Q']$ est selon les cas $P' \mid Q'[c/x]$ ou $(\nu c)(P' \mid Q'[c/x])$ ..

Propriétés de πw

Ce qui change :

- bisimilarité ouverte $\sim_o$ (modulo substitutions ; congruence)
- de πw à πwl :

$$[\![\bar{a}b.P]\!] = \bar{a}(b').(\{b' = b\} \mid [\![P]\!])$$

- correspondences de bisimulations fortes (et non $\approx$ / $\lesssim$)

$$P \sim_o Q \Leftrightarrow [\![P]\!] \sim_o [\![Q]\!]$$

- preuve de substitution plus facile, et énoncé plus fort:

$$P[b/a] \sim_o (\nu a)(\{a = b\} \mid P)$$

  - bisimilarité forte
  - pas besoin de typage pour a
  - pas seulement vrai pour $P = [\![Q]\!]$ pour un certain Q

Dualité dans πwl

πwl peut être dualisé au même titre que πl. (le dual de $\{a = b\}$ est $\{a = b\}$)

Les traductions en πwl de $[\![\cdot]\!]$ et $[\![\cdot]\!]$ sont duales à partir du moment on traduit les liens finis de $[\![\cdot]\!]$ par des fils.

πw et πwl sont plus faciles à manipuler que les liens de πl en général.

Conclusion

- Étude de la dualité, motivée par un encodage du λ-calcul dans le π-calcul
- Développement des liens à capacité de réception
- Étude d'un nouveau calcul, πw, qui semble intéressant en soi.

Travaux à suivre

De manière générale

- autres stratégies que l'appel par nom + réduction sous les λ

Avec $\pi_{io} \to \pi\mathbf{l}$:

- travailler sans l'hypothèse de linéarité ?
- des liens pour $\#T$? (*equators* de Honda : solution partielle)

Avec πw et πw**l** :

- théories de πw et πw**l** (notamment, πw**l** contre π)
- lien avec le calcul des fusions explicites (ici, pas de substitution des objets, pas de *datum*)

Point techniques non détaillés

- **Optimisations :** la traduction de π_{io} vers πI a des contraintes et des optimisations pour éviter de s'occuper de l'inutile.

$$[\![(\nu b)\overline{a}b.P]\!] = \overline{a}(b).[\![P]\!]$$

$$[\![(\nu b)\overline{a}b.P]\!] \neq (\nu b)\overline{a}(b').(b' \overset{T}{\hookrightarrow} b \mid [\![P]\!])$$

(c'est obligatoire pour que $[\![\cdot]\!]$ et $[\![\cdot]\!]$ valident les contraintes de la *full abstraction*)

- **Linéarité :** le lemme de substitution n'est prouvé que dans le cas linéaire, dans le sens ou seules les variables utilisées à un endroit seulement sont substituées (mais les encodages respectent cette contrainte)

Point techniques non détaillés, suite

- **Dualité sur les liens** : elle n'est pas complètement satisfaisante d'un point de vue opérationnel : on n'a pas $P \xrightarrow{\mu} P' \Leftrightarrow \overline{P} \xrightarrow{\overline{\mu}} \overline{P'}$ car les traces subissent des permutations. $[\![P]\!] \asymp_a [\![Q]\!]$ ssi $\overline{[\![P]\!]} \asymp_a \overline{[\![Q]\!]}$ n'est pas prouvé.
- **Full abstraction**: on a besoin de bisimilarités compliquées pour l'établir. $P \approx_l Q$ ssi $[\![P]\!] \asymp_a [\![Q]\!]$ mais $\approx_l$ et $\asymp_a$ ont des définitions assez *ad-hoc*. (adaptée de [Merro and Sangiorgi, 2004])

Avez-vous des questions ?

Boreale, M. (1998).
On the expressiveness of internal mobility in name-passing calculi.
Theoretical Computer Science, 195:205–226.

Merro, M. and Sangiorgi, D. (2004).
On asynchrony in name-passing calculi.
Mathematical Structures in Computer Science, 14(5):715–767.

Milner, R. (1992).
Functions as processes.
Mathematical Structures in Computer Science, 2(2):119–141.

Pierce, B. C. and Sangiorgi, D. (1996).
Typing and subtyping for mobile processes.
Mathematical Structures in Computer Science, 6(5):409–453.

Sangiorgi, D. (1996).
π-calculus, internal mobility and agent-passing calculi.
Theoretical Computer Science, 167(2):235–274.

van Bakel, S. and Vigliotti, M. G. (2009).
A logical interpretation of the λ-calculus into the π-calculus, preserving spine reduction and types.
In *CONCUR*, pages 84–98.