Réductions closes

Correction.

Documents autorisés (en particulier le poly).

Le but de ce problème est d'étudier une forme faible de réduction où l'α-conversion n'est pas nécessaire pour éviter les captures de variables. Ceci est dû à Maribel Fernández et Ian Mackie.

Le calcul que nous allons étudier est appelé le *λ_c-calcul.* Il s'agit d'un calcul à substitution explicites, dans lequel les variables ont des noms x, y, z, $\ldots$, comme dans le λ-calcul, et contrairement au $\lambda\sigma$-calcul.

Pour distinguer la substitution usuelle $s[x := t]$ des substitutions explicites, nous noterons ces dernières $M\{x := t\}$. L'opérateur $\_\{\_ := \_\}$ sera un constructeur du λ_c-calcul.

Une autre caractéristique du λ_c-calcul est qu'il est très fortement contraint. Par exemple, la λ-abstraction $\lambda x \cdot M$ ne sera un λ_c-terme valide que si x est libre dans M, et l'application MN ne sera un λ_c-terme valide que si M et N n'ont aucune variable libre en commun. En fait, toute variable apparaîtra exactement une fois dans tout λ_c-terme.

Si on s'en tenait là, on ne pourrait pas écrire grand-chose en λ_c-calcul. On a donc deux constructions supplémentaires:

- lorsque x n'est pas libre dans M, l'*effaceur* $E_x(M)$ est un terme où x est libre et dont la sémantique intuitive est: "effacer x, puis évaluer M"; si x n'est pas libre dans M, on ne peut pas écrire $\lambda x \cdot M$, mais on pourra écrire $\lambda x \cdot E_x(M)$.
- lorsque y et z sont deux variables distinctes, toutes deux libres dans M, le *copieur* $C_x^{y,z}(M)$ est un terme dans lequel x est libre mais y et z ne le sont plus. Sa sémantique intuitive est: "faire deux copies de x, mettre la première dans y, la seconde dans z, puis évaluer M". On ne peut pas écrire xx en λ_c-calcul, mais on peut à la place écrire $C_x^{y,z}(yz)$.

On définit la syntaxe des λ_c-termes M, N, P, $\ldots$, en même temps que l'ensemble des variables libres $\mathrm{fv}(M)$ dans M comme suit. L'ensemble des λ_c-termes est le plus petit tel que:

(a) Toute variable x est un λ_c-terme; on a $\mathrm{fv}(x) = \{x\}$;

(b) l'*abstraction* $\lambda x \cdot M$ est un λ_c-terme pour tout λ_c-terme M tel que $x \in \mathrm{fv}(M)$; on a $\mathrm{fv}(\lambda x \cdot M) = \mathrm{fv}(M) \setminus \{x\}$;

(c) l'*application* MN est un λ_c-terme pour tous λ_c-termes M et N tels que $\mathrm{fv}(M) \cap \mathrm{fv}(N) = \emptyset$; on a $\mathrm{fv}(MN) = \mathrm{fv}(M) \cup \mathrm{fv}(N)$;

(d) l'*effaceur* $E_x(M)$ est un λ_c-terme pour tout λ_c-terme M tel que $x \not\in \text{fv}(M)$; on a $\text{fv}(E_x(M)) = \text{fv}(M) \cup \{x\}$;

(e) le *copieur* $C_x^{y,z}(M)$ est un λ_c-terme pour tout λ_c-terme M et pour toutes variables $x \not\in \text{fv}(M)$, $y, z \in \text{fv}(M)$, telles que $y \neq z$; on a $\text{fv}(C_x^{y,z}(M)) = (\text{fv}(M) \setminus \{y, z\}) \cup \{x\}$;

(f) la *substitution* $M\{x := N\}$ est un λ_c-terme pour tous λ_c-termes M et N et pour toute variable $x \in \text{fv}(M)$ où $(\text{fv}(M)\setminus\{x\})\cap\text{fv}(N) = \emptyset$; on a $\text{fv}(M\{x := N\}) = (\text{fv}(M)\setminus\{x\})\cup\text{fv}(N)$.

On rappelle qu'il n'y a *pas* de règle de α-renommage.

Les règles de réduction sont:

$$
\begin{array}{lrcll}
(\beta) & (\lambda x \cdot M)N & \to & M\{x := N\} & \text{si } \text{fv}(\lambda x \cdot M) = \emptyset \\
(Var) & x\{x := N\} & \to & N & \\
(App_1) & (M_1M_2)\{x := N\} & \to & (M_1\{x := N\})M_2 & \text{si } x \in \text{fv}(M_1) \\
(App_2) & (M_1M_2)\{x := N\} & \to & M_1(M_2\{x := N\}) & \text{si } x \in \text{fv}(M_2) \\
(Lam) & (\lambda x \cdot M)\{y := N\} & \to & \lambda x \cdot (M\{y := N\}) & \text{si } x \neq y, \text{fv}(N) = \emptyset \\
(Copy_1) & C_x^{y,z}(M)\{x := N\} & \to & M\{y := N\}\{z := N\} & \text{si } \text{fv}(N) = \emptyset \\
(Copy_2) & C_x^{y,z}(M)\{x' := N\} & \to & C_x^{y,z}(M\{x' := N\}) & \text{si } x' \neq x, x' \neq y, x' \neq z, \text{fv}(N) = \emptyset \\
(Erase_1) & E_x(M)\{x := N\} & \to & M & \text{si } \text{fv}(N) = \emptyset \\
(Erase_2) & E_x(M)\{y := N\} & \to & E_x(M\{y := N\}) & \text{si } x \neq y, \text{fv}(N) = \emptyset \\
(Comp) & M\{x := N\}\{y := P\} & \to & M\{x := N\{y := P\}\} & \text{si } y \in \text{fv}(N)
\end{array}
$$

On notera que $(\lambda x \cdot M)N$ n'est pas toujours un β-rédex: il ne l'est que si $\lambda x \cdot M$ est *clos*, c'est-à-dire n'a aucune variable libre. Les règles (Lam), $(Copy_1)$, $(Copy_2)$, $(Erase_1)$, $(Erase_2)$ ont des conditions similaires. A cause des conditions de formation des applications, $(M_1M_2)\{x := N\}$ est toujours soit un App_1-rédex, soit un App_2-rédex, et ne peut pas être les deux en même temps.

I. Réduction, machines.

1. On rappelle que le combinateur de point fixe Y de Church est $Y = \lambda f \cdot (\lambda x \cdot f(xx))(\lambda x \cdot f(xx))$ en λ-calcul. Le combinateur correspondant en λ_c-calcul est

$$Y_c = \lambda f \cdot C_f^{g,h}((\lambda x \cdot g(C_x^{y,z}(yz)))(\lambda x \cdot h(C_x^{y,z}(yz))))$$

   Y est-il fortement normalisant en λ-calcul? faiblement normalisant en λ-calcul? Mêmes questions pour Y_c en λ_c-calcul.

   > *Y a un seul rédex, et se réduit en $\lambda f \cdot f((\lambda x \cdot f(xx))(\lambda x \cdot f(xx)))$, qui se reréduit en $\lambda f \cdot f(f((\lambda x \cdot f(xx))(\lambda x \cdot f(xx))))$, etc. Y n'est donc ni fortement ni faiblement normalisant.*
   >
   > *En revanche, Y_c est normal en λ_c-calcul, parce que $(\lambda x \cdot g(C_x^{y,z}(yz)))$ n'est pas clos.*

2. Montrer que si M se réduit en N en λ_c, alors $\text{fv}(M) = \text{fv}(N)$. (On ne traitera que les règles (β) et $(Erase_1)$ pour aller plus vite, et on admettra les autres cas.)

Par récurrence sur le nombre de réductions, puis sur la profondeur du rédex contracté. Les cas de récurrence étant évidents, on se concentre sur les cas de base. Traitons des deux cas de l'énoncé:

- (β) $\mathrm{fv}((\lambda x \cdot M)N) = \mathrm{fv}(\lambda x \cdot M) \cup \mathrm{fv}(N) = \mathrm{fv}(N)$ *puisque* $\lambda x \cdot M$ *est supposé clos. En particulier* $\mathrm{fv}(M) \subseteq \{x\}$*, donc* $\mathrm{fv}(M\{x := N\}) = (\mathrm{fv}(M) \setminus \{x\}) \cup \mathrm{fv}(N) = \mathrm{fv}(N)$.
- $(Erase_1)$ $\mathrm{fv}(E_x(M)\{x := N\}) = (\mathrm{fv}(E_x(M))\setminus\{x\}) \cup \mathrm{fv}(N) = \mathrm{fv}(M) \cup \mathrm{fv}(N)$, *puisque* x *n'est pas libre dans* M *par construction; or ceci vaut* $\mathrm{fv}(M)$ *puisque* $E_x(M)\{x := N\}$ *n'est un rédex que si* $\mathrm{fv}(N) = \emptyset$.

3. Montrer que si M se réduit en N par l'une quelconque des règles de réécriture de λ_c, et M est un λ_c-terme, alors N est aussi un λ_c-terme. (On ne traitera que les règles (App_1), $(Copy_1)$ et $(Comp)$ pour aller plus vite, et on admettra les autres cas.)

Encore une fois par récurrence sur la profondeur du rédex contracté. Les cas de récurrence sont des conséquences triviales de la question I.1: par exemple si MN *est un* λ_c*-terme, et* $M \to M'$*, alors* $FV(M) = \mathrm{fv}(M')$*, donc* $\mathrm{fv}(M') \cap \mathrm{fv}(N) = \mathrm{fv}(M) \cap \mathrm{fv}(N) = \emptyset$*, donc* $M'N$ *est un* λ_c*-terme. Les autres cas sont similaires, les règles de bonne formation des* λ_c*-termes ne dépendant que des ensembles de variables libres des sous-termes, qui ne changent pas par réduction.*

Il ne reste qu'à traiter le cas de base, où le rédex est contracté. On considère les trois cas de l'énoncé:

- (App_1)*:* $(M_1M_2)\{x := N\}$ *est un rédex, donc* x *est libre dans* M_1*. De plus, les ensembles de variables libres de* M_1 *et* M_2 *ne se recontrent pas, et ceux de* M_i *et* N *(*$i = 1, 2$*) se rencontrent en au plus* x *(fait* $(*)$*). En particulier,* $M_1\{x := N\}$ *est un* λ_c*-terme. Ses variables libres sont celles de* M_1 *sauf* x*, et celles de* N*; supposons que* y *soit libre à la fois dans* $M_1\{x := N\}$ *et dans* M_2*. Comme* y *est libre dans* M_2*, elle ne l'est pas dans* M_1*, donc elle l'est dans* N*. Donc* $y = x$ *par* $(*)$*. Or* x *n'est pas libre dans* M_2*, puisque* x *est libre dans* M_1 *mais* M_1 *et* M_2 *n'ont aucune variable libre en commun. Donc* $M_1\{x := N\}$ *et* M_2 *n'ont aucune variable libre en commun, d'où* $(M_1\{x := N\})M_2$ *est un* λ_c*-terme.*
- $(Copy_1)$*:* N *est clos, et* y *et* z *sont libres dans* M*. Donc* $M\{y := N\}$ *est un* λ_c*-terme (trivialement, car* N *est clos, et* y *est libre dans* M*). Trivialement encore, comme* N *est clos, il suffit de vérifier que* z *est libre dans* $M\{y := N\}$ *pour montrer que le côté droit est un* λ_c*-terme. Or* z *est libre dans* M *et est différent de* y *par construction, d'où le résultat.*
- $(Comp)$*:* y *est libre dans* N*, donc pas dans* M *à moins que* $y = x$*.*
  Montrons d'abord que $N\{y := P\}$ *est un* λ_c*-terme:* y *est libre dans* N*, et il reste à montrer qu'aucune variable n'est libre à la fois dans* N *et dans* P*. En effet, sinon une telle variable serait libre à la fois dans* $M\{x := N\}$ *et dans* P*, ce qui voudrait dire que le côté gauche ne serait pas un* λ_c*-terme.*

Montrons ensuite que le côté droit tout entier est un λ_c-terme. Pour ceci, on vérifie que x est libre dans M (car $M\{x := N\}$, du côté gauche, est un λ_c-terme); et qu'aucune variable z libre dans M et différente de x ne peut être libre dans $N\{y := P\}$. En effet, une telle variable z devrait être soit libre dans N et différente de y, soit libre dans P.
Dans le premier cas, z serait à la fois libre dans M et dans N, contredisant le fait que $M\{x := N\}$, du côté gauche, est un λ_c-terme.
Dans le second cas, z serait libre dans M et dans P. Comme le côté gauche est un λ_c-terme, et que z est libre dans P, z est soit libre dans $M\{x := N\}$ soit égale à y.

- *Si $z = y$ (deuxième sous-cas), comme y est libre dans N par hypothèse, y n'est pas libre dans M ou bien $y = x$. Or $y = z$ est libre dans M par hypothèse, donc $y = x$. Mais ceci implique $z = x$, contradiction (on a supposé $z \neq x$).Pour résumer, on se retrouve dans le cas d'une réduction $M\{x := N\}\{x := P\} \to M\{x := N\{x := P\}\}$ avec x libre dans P, N, et M, qui est parfaitement valide.*
- *Si z est libre dans $M\{x := N\}$, comme z est aussi libre dans P, et que l'on peut supposer $z \neq y$ sans perte de généralité (voir premier sous-cas), le côté gauche tout entier ne serait pas un λ_c-terme, contradiction.*

4. On construit une machine de Krivine pour évaluer les λ_c-termes. Un *état* de la machine est un couple M, R où M est un λ_c-terme, et R est une liste d'objets qui peuvent être soit des λ_c-termes soit des *associations* $\{x := N\}$, où x est une variable et N un λ_c-terme. La sémantique d'un tel couple est donné par la fonction γ:

$$\gamma(M, []) = M \quad \gamma(M, N :: R) = \gamma(MN, R) \quad \gamma(M, \{x := N\} :: R) = \gamma(M\{x := N\}, R)$$

Un état M, R est *légal* ssi $\gamma(M, R)$ est un λ_c-terme. Les transitions de la machine sont:

$$\begin{array}{rcl} MN, R & \rightsquigarrow & M, N :: R \\ M\{x := N\}, R & \rightsquigarrow & M, \{x := N\} :: R \\ \lambda x \cdot M, N :: R & \rightsquigarrow & M, \{x := N\} :: R \quad \textit{Errata: ajouter “si } \mathrm{fv}(\lambda x \cdot M) = \emptyset\text{”} \\ & & \ldots \end{array}$$

Compléter. Les deux premières transitions servent à chercher le rédex de tête. La troisième réalise la règle (β). On demande de compléter par les transitions correspondant aux autres règles de réduction. On devra avoir les trois *(Errata: les deux)* propriétés:

(a) si $M, R \rightsquigarrow M', R'$, et M, R est un état légal, alors $\gamma(M, R) \to^* \gamma(M', R')$ dans le λ_c-calcul et M', R' est un état légal;

(b) si M, R est un état légal *et final* (i.e., aucune transition ne s'applique), alors $\gamma(M, R)$ est une forme *normale de tête faible*, c'est-à-dire engendré par la grammaire:

$$\begin{array}{rcl} whnf & ::= & \lambda x \cdot N \quad | \quad xN_1 \ldots N_k \\ & | & C_x^{y,z}(whnf) \quad | \quad E_x(whnf) \end{array}$$

On ne demande pas une preuve de ces propriétés, mais, au moins pour la propriété (b), un argumentaire rapide montrant pourquoi $\gamma(M, R)$ ne peut pas être d'une autre forme.

Errata: les formes normales de tête faibles du λ_c-calcul sont beaucoup plus compliquées. On demande à la place de coder une machine de Krivine raisonnable, comme en cours et en TD, et on ne cherchera pas à montrer la propriété (b) ou une propriété analogue. Quoique quiconque qui aura réussi ce tour de force aura des points en plus.

La machine est juste:

$$\begin{array}{rcll}
MN, R & \leadsto & M, N :: R & \\
M\{x := N\}, R & \leadsto & M, \{x := N\} :: R & \\
\lambda x \cdot M, N :: R & \leadsto & M, \{x := N\} :: R & \textit{si } \mathrm{fv}(\lambda x \cdot M) = \emptyset \\
x, \{x := N\} :: R & \leadsto & N, R & \\
M_1 M_2, \{x := N\} :: R & \leadsto & M_1, \{x := N\} :: M_2 :: R & \textit{si } x \in \mathrm{fv}(M_1) \\
M_1 M_2, \{x := N\} :: R & \leadsto & M_1, (M_2\{x := N\}) :: R & \textit{si } x \in \mathrm{fv}(M_2) \\
\lambda x \cdot M, \{y := N\} :: R & \leadsto & \lambda x \cdot M\{y := N\}, R & (x \neq y) \textit{si } N \textit{ clos} \\
C_x^{y,z}(M), \{x := N\} :: R & \leadsto & M, \{y := N\} :: \{z := N\} :: R & \textit{si } N \textit{ clos} \\
C_x^{y,z}(M), \{x' := N\} :: R & \leadsto & C_x^{y,z}(M\{x' := N\}), R & (x' \neq x) \textit{si } N \textit{ clos} \\
E_x(M), \{x := N\} :: R & \leadsto & M, R & \textit{si } N \textit{ clos} \\
E_x(M), \{y := N\} :: R & \leadsto & E_x(M\{y := N\}), R & (x \neq y) \textit{si } N \textit{ clos} \\
M, \{x := N\} :: \{y := P\} :: R & \leadsto & M, \{x := N\{y := P\}\} :: R & \textit{si } y \in \mathrm{fv}(N)
\end{array}$$

II. Confluence du λ_c-calcul.

1. Soit σ_c le système de réécriture composé de toutes les règles de λ_c sauf (β).

   Pour tout λ_c-terme M où x est libre, on définit $|M|_x$ par:

$$|x|_x = 1 \quad |\lambda y \cdot M|_x = 1 + |M|_x \ (x \neq y) \quad |MN|_x = \begin{cases} 1 + |M|_x & \text{si } x \in \mathrm{fv}(M) \\ 1 + |N|_x & \text{si } x \in \mathrm{fv}(N) \end{cases}$$

$$|E_x(M)|_x = 1 \quad |E_y(M)|_x = 1 + |M|_x \ (x \neq y) \quad |M\{y := N\}|_x = \begin{cases} 1 + |M|_x & \text{si } x \in \mathrm{fv}(M) \\ & \textit{Errata: ajouter} \\ & \textit{“et si } x \notin \mathrm{fv}(N \\ 1 + |N|_x + |M|_y & \text{si } x \in \mathrm{fv}(N) \end{cases}$$

$$|C_x^{y,z}(M)|_x = 1 + |M|_y + |M|_z \quad |C_x^{y,z}(M)|_{x'} = 1 + |M|_{x'} \quad (x \neq x')$$

   $|M|_x$ est, en gros, la profondeur de l'(unique) occurrence de x dans M. Montrer que pour tout λ_c-terme M, pour toute variable $z \in \mathrm{fv}(M)$, si $M \to N$ en σ_c, alors $|M|_z \geq |N|_z$.

   Par récurrence sur la profondeur du rédex contracté dans M. Comme la réduction préserve les ensembles de variables libres, par la question 1, le cas de récurrence est évident. Traitons les cas de base:

(Var) *si z est libre dans $x\{x := N\}$, alors z est libre dans N, et $|x\{x := N\}|_z = 1 + |N|_z + |x|_x = 2 + |N|_z > |N|_z$.*

(App_1) *si $x \in \text{fv}(M_1)$, on a trois sous-cas:*

- *si $z \in \text{fv}(N)$, $|(M_1M_2)\{x := N\}|_z = 1 + |N|_z + 1 + |M_1|_x$ et $|(M_1\{x := N\})M_2|_z = 1 + 1 + |N|_z + |M_1|_x$: on a l'égalité;*
- *si $z \in \text{fv}(M_1)$, $|(M_1M_2)\{x := N\}|_z = 1 + 1 + |M_1|_z$ et $|(M_1\{x := N\})M_2|_z = 1 + 1 + |M_1|_z$: on a encore l'égalité;*
- *si $z \in \text{fv}(M_2)$, $|(M_1M_2)\{x := N\}|_z = 1 + 1 + |M_2|_z$ et $|(M_1\{x := N\})M_2|_z = 1 + |M_2|_z$: ici on a l'inégalité stricte.*

(App_2) *Similaire.*

(Lam) *Soit N clos. Si z est libre dans $(\lambda x \cdot M)\{y := N\}$, alors z est libre dans M. Donc $|(\lambda x{\cdot}M)\{y := N\}| = 1{+}1{+}|M|_z$, et $|\lambda x{\cdot}(M\{y := N\})|_z = 1{+}1{+}|M|_z$: on a l'égalité.*

$(Copy_1)$ *Soit N clos. Si z' est libre dans $C_x^{y,z}(M)\{x := N\}$, z' est libre dans M, et $|C_x^{y,z}(M)\{x := N\}|_{z'} = 1{+}1{+}|M|_{z'}$, $|M\{y := N\}\{z := N\}|_{z'} = 1{+}1{+}|M|_{z'}$.*

$(Copy_2)$ *Soit N clos. Si z' est libre dans $C_x^{y,z}(M)\{x' := N\}$, alors z' est libre dans M, donc $|C_x^{y,z}(M)\{x' := N\}|_{z'} = 1 + 1 + |M|_{z'}$, et $|C_x^{y,z}(M\{x' := N\})|_{z'} = 1 + 1 + |M|_{z'}$: on a l'égalité.*

$(Erase_1)$ *Soit N clos. Si z est libre dans $E_x(M)\{x := N\}$, alors z est libre dans M et $z \neq x$, donc $|E_x(M)\{x := N\}|_z = 1 + 1 + |M|_z > |M|_z$.*

$(Erase_2)$ *Soit N clos. Si z est libre dans $E_x(M)\{y := N\}$, alors z est libre dans M et $z \neq x$, donc $|E_x(M)\{y := N\}|_z = 1 + 1 + |M|_z$, et $|E_x(M\{y := N\})|_z = 1 + 1 + |M|_z$, d'où l'égalité.*

$(Comp)$ *Soit $y \in \text{fv}(N)$. Si z est libre dans $M\{x := N\}\{y := P\}$, on a trois sous-cas:*

- *Si z est libre dans P, $|M\{x := N\}\{y := P\}|_z = 1 + |P|_z + 1 + |N|_y + 1 + |M|_x$, et $|M\{x := N\{y := P\}\}|_z = 1 + |N\{y := P\}|_z + |M|_x = 1 + 1 + |P|_z + |N|_y + |M|_x$, d'où l'inégalité stricte;*
- *Si z est libre dans N, $|M\{x := N\}\{y := P\}|_z = 1 + |M\{x := N\}|_z = 1 + 1 + |N|_z + |M|_x$, et $|M\{x := N\{y := P\}\}|_z = 1 + |N\{y := P\}|_z + |M|_x = 1 + 1 + |N|_z + |M|_x$, d'où l'égalité;*
- *Si z est libre dans M, $|M\{x := N\}\{y := P\}|_z = 1 + 1 + |M|_z$, et $|M\{x := N\{y := P\}\}|_z = 1 + |M|_z$, d'où l'inégalité stricte.*

2. En utilisant cette construction, on définit la traduction suivante:

$$[x] = x \quad [\lambda x \cdot M] = [M] \quad [MN] = app([M], [N])$$
$$[E_x(M)] = [M] \quad [C_x^{y,z}(M)] = [M] \quad [M\{x := N\}] = f_{|M|_x}([M], [N])$$

vers les termes du premier ordre sur la signature app (d'arité 2), f_n, $n \geq 1$ (d'arité 2).

Montrer que si $M \to N$ en σ_c, alors $[M] \to^+ [N]$ dans un système de réécriture σ'_c que l'on déterminera. Montrer que σ'_c termine, et en déduire que σ_c termine. (Ceci devrait vous rappeler la preuve de terminaison de σ du poly.)

Posons σ'_c le système de réécriture:

$$
\begin{array}{rccl}
(Var') & f_1(x,u) & \to & u \\
(App'_1) & f_{1+n}(app(t_1,t_2),u) & \to & app(f_n(t_1,u),t_2) \\
(App'_2) & f_{1+n}(app(t_1,t_2),u) & \to & app(t_1,f_n(t_2,u)) \\
(Lam') & f_{1+n}(t,u) & \to & f_n(t,u) \\
(Copy'_1) & f_{1+m+n}(t,u) & \to & f_{1+n}(f_m(t,u),u) \\
(Copy'_2) & f_{1+n}(t,u) & \to & f_n(t,u) \\
(Erase'_1) & f_1(t,u) & \to & t \\
(Erase'_2) & f_{1+n}(t,u) & \to & f_n(t,u) \\
(Comp') & f_{1+n+m}(f_m(t,u),v) & \to & f_m(t,f_n(u,v))
\end{array}
$$

Examinons chacun des rédexes de σ_c.

(Var) *On a* $[x\{x := N\}] = f_1(x,[N]) \to [N]$ *par* (Var')*;*

(App_1) *Si* $x \in \mathrm{fv}(M_1)$*, on a* $[(M_1M_2)\{x := N\}] = f_{1+|M_1|_x}(app([M_1],[M_2]),[N]) \to app(f_{|M_1|_x}([M_1],[N]),[M_2])$ *(par* (App'_1)*)* $= [(M_1\{x := N\})M_2]$*;*

(App_2) *Si* $x \in \mathrm{fv}(M_2)$*, on a* $[(M_1M_2)\{x := N\}] = f_{1+|M_2|_x}(app([M_1],[M_2]),[N]) \to app([M_1],f_{|M_2|_x}([M_2],[N]))$ *(par* (App'_2)*)* $= [M_1(M_2\{x := N\})]$*;*

(Lam) $[(\lambda x \cdot M)\{y := N\}] = f_{1+|M|_y}([M],[N]) \to f_{|M|_y}([M],[N])$ *(par* (Lam')*)* $= [\lambda x \cdot M\{y := N\}]$*;*

$(Copy_1)$ $[C^{y,z}_x(M)\{x := N\}] = f_{1+|M|_y+|M|_z}([M],[N]) \to f_{1+|M|_z}(f_{|M|_y}([M],[N]),[N])$ *(par* $(Copy'_1)$*)* $= f_{1+|M|_z}([M\{y := N\}],[N]) = f_{|M\{y:=N\}|_z}([M\{y := N\}],[N])$ *(car* $z \in \mathrm{fv}(M)$*)* $= [M\{y := N\}\{z := N\}]$*;*

$(Copy_2)$ $[C^{y,z}_x(M)\{x' := N\}] = f_{1+|M|_{x'}}([M],[N]) \to f_{|M|_{x'}}([M],[N])$ *(par* $(Copy'_2)$*)* $= [C^{y,z}_x(M\{x' := N\})]$*;*

$(Erase_1)$ $[E_x(M)\{x := N\}] = f_1([M],[N]) \to [M]$ *par* $(Erase'_1)$*;*

$(Erase_2)$ $[E_x(M)\{y := N\}] = f_{1+|M|_y}([M],[N]) \to f_{|M|_y}([M],[N])$ *(par* $(Erase'_2)$*)* $= E_x(M\{y := N\})$*;*

$(Comp)$ *Comme* $y \in \mathrm{fv}(N)$*,* $[M\{x := N\}\{y := P\}] = f_{1+|N|_y+|M|_x}(f_{|M|_x}([M],[N]),[P]) \to f_{|M|_x}([M],f_{|N|_y}([N],[P]))$ *(par* $(Comp')$*)* $= [M\{x := N\{y := P\}\}]$.

En utilisant la règle (Lam') ou $(Erase'_2)$ (c'est la même), on montre comme dans le cours que si $M \to N$ en contractant un σ_c-rédex à une profondeur quelconque, alors $[M] \to^+ [N]$. C'est parce que les profondeurs des variables dans les termes ne peuvent que décroître, par la question précédente.

En choisissant la précédence $\ldots \succ f_{1+n} \succ f_n \succ \ldots \succ f_1 \succ f_0 \succ app$, qui est bien fondée, on voit que $l \succ^{lpo} r$ pour toute règle $l \to r$ de σ'_c. Donc σ'_c termine. Donc σ_c termine: sinon une réduction infinie dans σ_c fournirait une réduction infinie dans σ'_c par traduction.

3. On admettra que σ_c est localement confluent. (Comme on l'a dit en cours, les machines sont bien meilleures à montrer ce genre de résultat que les êtres humains.) Montrer que σ_c est confluent. On demande en particulier le nom du théorème qui permet de conclure.

   C'est par le lemme de Newman: tout système de réécriture localement confluent et terminant est confluent.

4. On définit le λ_c^*-calcul sur le langage des λ_c-termes, augmenté de la construction $(\lambda^* x \cdot M)N$. Formellement, l'ensemble des λ_c^*-termes est le plus petit vérifiant les conditions **(a)–(f)** de la page 1 (avec "λ_c-terme" remplacé par "λ_c^*-terme" partout) et tel que

   (g) $(\lambda^* x \cdot M)N$ est un λ_c^*-terme pour tous λ_c^*-termes M et N tels que $\mathrm{fv}(M) = \{x\}$; on a $\mathrm{fv}((\lambda^* x \cdot M)N) = \mathrm{fv}(N)$.

   Les règles de réduction du λ_c^* sont celles de σ_c, plus les règles

$$\begin{array}{lccc} (\beta^*) & (\lambda^* x \cdot M)N & \to & M\{x := N\} \\ (App^*) & ((\lambda^* x \cdot M)N)\{x := P\} & \to & (\lambda^* x \cdot M)(N\{x := P\}) \end{array}$$

   qui remplacent la règle (β) du λ_c-calcul. Montrer que le λ_c^*-calcul termine. Indication: on pourra exhiber une traduction simple du λ_c^*-calcul dans le λ_c-calcul qui envoie toute réduction du λ_c^*-calcul vers une réduction dans σ_c. (La définition du λ_c^*-calcul devrait vous rappeler le théorème des développements finis présenté en cours — et qui n'est pas dans le poly. La démonstration de terminaison est infiniment plus simple.)

   Il suffit de traduire $(\lambda^* x \cdot M)N$ en $z\{z := M\{x := N\}\}$, où z est une variable fraîche. On ne peut pas traduire directement en $M\{x := N\}$, sinon la β^*-réduction se traduit en une réduction vide (à 0 étapes). Dans ce dernier cas, on peut arguer du fait que lorsque la réduction est vide, le nombre de symboles λ^* a diminué d'au moins 1.

5. Si M est un λ_c^*-terme, son effacement $E(M)$ est le λ_c-terme obtenu en effaçant toutes les étoiles; autrement dit $E((\lambda^* x \cdot M)N) = (\lambda x \cdot E(M))(E(N))$, les autres cas étant évidents.

   Une *décoration* d'un λ_c-terme M est un λ_c^*-terme M' quelconque tel que $E(M') = M$. Autrement dit, on obtient les décorations de M en décidant d'ajouter ou pas une étoile au-dessus de chaque λ qui débute un β-rédex dans M.

   On notera que tout λ_c^*-terme sans étoile est β^*-normal (n'a pas de rédex pour la règle (β^*)). Réciproquement, tout λ_c^*-terme qui est β^*-normal est sans étoile, et est donc un λ_c-terme.

Soit $\to_1$ la relation de réduction définie sur les λ_c-termes par $M \to_1 N$ si et seulement s'il existe une décoration M' de M tel que $M' \to^* N$ dans λ_c^*. (N.B.: comme N est un λ_c-terme, il est sans étoile.)

Montrer que si $M \to N$ dans le λ_c-calcul, alors $M \to_1 N$, et que si $M \to_1 N$ alors $M \to^* N$ dans le λ_c-calcul.

> *Si $M \to N$ dans le λ_c-calcul, alors soit c'est par une règle de σ_c et on a clairement $M \to N$ dans λ_c^*, donc $M \to_1 N$ puisque M est une décoration (triviale) de lui-même; soit c'est par une règle (β), auquel cas $M' \to N$ dans λ_c, où M' est obtenu en décorant l'unique λ qui démarre le (β)-rédex contracté. Dans tous les cas $M \to_1 N$.*
>
> *Si $M \to_1 N$ alors $M' \to^* N$ en λ_c pour une certaine décoration M' de M. Or si $P \to Q$ en λ_c^*, clairement $E(P) \to E(Q)$ en λ_c, donc $M' \to^* N$ en λ_c.*

6. On admettra que λ_c^* est localement confluent. Montrer que $\to_1$ est fortement confluente.

> *Supposons $M \to_1 N_1$ et $M \to_2 N_2$. Donc $M_1 \to^* N_1$ et $M_2 \to^* N_2$ pour deux décorations M_1 et M_2 de M, et où les réductions sont effectuées dans λ_c^*. Il est clair que, partant d'une réduction de M_i ($i = 1, 2$) vers N_i en λ_c^*, si on ajoute des étoiles sur des β-rédexes de M_i, on obtient encore une réduction de même longueur dans λ_c^*, et qui aboutit à une décoration N_i' de N_i. Soit donc M' la décoration de M obtenu en mettant des étoiles à toutes les positions où on en a rajouté dans M_1 ou dans M_2. On a $M' \to^* N_1'$ et $M' \to^* N_2'$ en λ_c^*, donc comme λ_c^* est localement confluent, on peut trouver un λ_c^*-terme P tel que $N_1' \to^* P$ et $N_2' \to^* P$ en λ_c^*. Comme λ_c^* terme, on peut sans perte de généralité supposer que P est λ_c^*-normal, donc sans étoile, donc que P est un λ_c-terme. Mais alors $N_i' \to^* P$ implique $N_i \to_1 P$, $i = 1, 2$. D'où la confluence forte.*

7. En déduire que le λ_c-calcul est confluent.

> *Supposons $M \to^* N_1$ et $M \to^* N_2$ en λ_c. Alors $M \to_1^* N_1$ et $M \to_1^* N_2$ par la question II.5 ($\to \subseteq \to_1$). Comme $\to_1$ est fortement confluent, il est confluent, donc il existe P tel que $N_1 \to_1^* P$ et $N_2 \to_1^* P$. Par la question II.5 ($\to_1 \subseteq \to^*$), $N_1 \to^* P$ et $N_2 \to^* P$, d'où le résultat.*