Remerciements

Les résultats exposés dans cette thèse sont en majeure partie le fruit d'une collaboration constante avec Vincent Danos. Je crois qu'ensemble nous avons fourni un effort global plus grand que si nous avions travaillé individuellement. En ce qui me concerne c'est une certitude. Pour cela je le remercie ici.
Je remercie Jean-Yves Girard d'avoir bien voulu être mon directeur de thèse, de ne m'avoir jamais fait défaut ni marchandé son soutien quand les circonstances le demandaient et de nous avoir laissé puiser dans le réservoir à idées que ses travaux représentaient pour nous.
Je remercie Michel Parigot pour avoir su trouver des solutions efficaces à mes problèmes financiers et pour son infinie disponibilité malgré ses nombreuses occupations.
Je remercie Jean-Louis Krivine pour toute la sympathie qu'il a manifestée à mon égard, et pour avoir été à travers son livre un modèle (petitement suivi) de rédaction.
A ma connaissance, Eric Duquesne est le premier à avoir remarqué l'isomorphisme entre les types du systèmes **D** et les réseaux. Je tiens à le remercier de m'avoir laissé le libre usage de cette idée qui est devenue le centre de la seconde partie de la thèse.
Je remercie Paul Rozière qui sait prouver les lemmes de lambda-calcul sur lesquels je bute.
Je remercie Gilles Amiot, Loic Colson, Thomas Ehrhard, Jean-Baptiste Joinet, Pasquale Malacaria, Sophie Malecki, Pascal Manoury, Christophe Raffalli, Marianne Simonot pour les nombreuses discussions mathématiques ou autres que j'ai pu avoir avec eux.
Je remercie l'équipe de logique de Paris 7 en entier, pour l'accueil chaleureux que j'y trouvai.
Je remercie l'université Paris 11 et le L.R.I. de m'avoir accueilli et permis de terminer cette thèse ces deux dernières années.

Introduction

Le lambda-calcul est un langage simple modélisant l'exécution des programmes, un "modèle de la calculabilité". De plus il possède des propriétés structurelles qui se manifestent par la définition des systèmes de typage des termes par différentes logiques. C'est ce qui en fait un bon outil théorique pour l'étude de l'exécution des programmes.
En dépit du fait que le mot "réseau" apparaisse plus souvent que celui de "terme", le sujet de ce travail est bien le lambda-calcul. Le projet est de développer de nouvelles méthodes de réduction, d'exécution des lambda-termes. Pour cela on va invoquer les résultats récents de logique linéaire ([Gir87]) et de géomètrie de l'interaction ([Gir89]), mais dans le cadre non typé du lambda-calcul et des *réseaux*. Ceux-ci sont des objets issus de l'analyse de la traduction de la logique intuitionniste dans la logique linéaire, permettant de coder les lambda-termes dans un formalisme proche des *réseaux de preuve* de Girard.
Toutefois réseaux et lambda-termes ne sont pas des structures isomorphes. En général un même réseau code plusieurs lambda-termes. Ceci engendre donc une relation d'équivalence sur les termes appelée *équivalence opérationnelle*, i.e., deux termes sont équivalents s'ils sont codés par le même réseau. La première partie de la thèse est destinée à convaincre le lecteur que, du point du vue du lambda-calcul, c'est à dire sans parler encore de réseaux, cette identification est naturelle et compatible avec la beta-réduction. Notons que dans sa thèse, Vidal qui s'intéresse aussi à l'étude des méthodes d'exécutions des termes, a introduit une relation d'équivalence très proche de l'équivalence opérationnelle ([Vid89]).
Afin de mettre en lumière le lien profond entre les réseaux et le lambda-calcul, on présente la traduction des lambda-termes par l'intermédiaire du système **D** ou *intersection type discipline* [CoDe86]. Eric Dusquesne a montré que les types principaux de **D**, qui par ailleurs caractérisent les formes normales des termes ([Kri90]), sont isomorphes aux réseaux normaux ([Duq90]). On généralise ce résultat en introduisant un système de calcul des types principaux appelé **De**, et en montrant que **De** est isomorphe aux réseaux.
La troisième partie s'attache à l'étude des *traces*. Les traces sont issues de l'interprétation géométrique de la formule d'exécution de Girard ([Gir89]), mais on en donne ici une définition interne aux réseaux i.e., sans mention de l'algèbre : ce sont des chemins particuliers dans un réseau, pour lesquels on peut définir une notion de réduction. Disons tout de suite que cette partie bien que longue est incomplète. On s'est contenté de poser les bases de la théorie mais des questions fines ne sont pas traitées. En particulier la géométrie des traces devrait induire une classification des réseaux basés sur de nouveaux critères ; une question naturelle est celle du rapport entre la longueur des traces maximales d'un réseau fortement normalisable et la longueur de ses réductions. On sait que la différence peut être exponentielle, mais ce n'est pas toujours le cas. La troisième partie s'achève sur la réduction *étiquetée* des traces ainsi nommée par référence à la réduction étiquetée de Lévy ([Lev80]).
C'est dans la dernière partie que l'on relie les traces à l'algèbre. Une trace *dérivable* est une trace dont

le poids dans l'algèbre $\mathbf{\Lambda}^*$ est non nul. C'est le théorème principal ; il complète la partie précédente en fournissant un moyen simple de calculer les traces dérivables. La combinaison du théorème des réductions étiquetées et du calcul des poids de traces dans l'algèbre devrait déboucher sur la définition d'une réduction partagée du lambda-calcul.

Première partie

Préliminaires lambda-calculesques

Chapitre 1

Le lambda-calcul

1.1 Terminologie

On appelle **V** l'ensemble des (noms des) variables du lambda-calcul et Λ l'ensemble des lambda-termes quotienté par alpha-équivalence. On note $(\ldots(U)V_1\ldots)V_n$ l'application des V_i à U ou, lorsque ce n'est pas ambigu: $(U)V_1\ldots V_n$. On dira que U est la *fonction* et les V_i les *arguments*. On utilise la notation $T \equiv U$ pour T est alpha-équivalent à U et $x \in T$ pour "x est variable libre de T".
On note $T[x/U]$ le terme T dans lequel la variable libre x est substituée par U après renommage si nécessaire des variables liées de T.
Un *contexte* C est un terme dans lequel on a laissé un *trou* i.e., un emplacement susceptible d'être rempli par une terme T quelconque, ce que l'on note $C[T]$. C'est une notion purement lexicale, et en particulier si C a des lambda dont la portée couvre le trou, ceux-ci lient les variables de T. On définit les contextes par induction :

- $C = []$ est un contexte ; pour tout terme T, $C[T] = T$;
- si U est un terme et $C[]$ un contexte alors $(U)C[]$ et $(C[])U$ sont des contextes ;
- Si $C[]$ est un contexte alors $\lambda x C[]$ en est un. Comme on l'a dit plus haut, si T est un terme possédant la variable libre x alors $\lambda x C[T]$ est un terme dans lequel la variable x est liée.

Exemple : $C \equiv \lambda x(x)[]$, $T \equiv \lambda y(y)x$ et $C[T] \equiv \lambda x(x)\lambda y(y)x$
On rappelle qu'une relation binaire $\mathcal{R}$ sur les termes *passe au contexte* quand :

$$U \,\mathcal{R}\, U' \;\Rightarrow\; \lambda x U \,\mathcal{R}\, \lambda x U' \tag{1.1}$$
$$U \,\mathcal{R}\, U', V \,\mathcal{R}\, V' \;\Rightarrow\; (U)V \,\mathcal{R}\, (U')V' \tag{1.2}$$
$$\tag{1.3}$$

ce qui peut aussi s'écrire : pour tout contexte C, si $T \,\mathcal{R}\, T'$ alors $C[T] \,\mathcal{R}\, C[T']$. Remarquons que si $\mathcal{R}$ passe au contexte alors pour tout T_1, T_2, U_1, U_2 tels que $T_1 \,\mathcal{R}\, T_2$ et $U_1 \,\mathcal{R}\, U_2$ on a $T_1[U_1/x] \,\mathcal{R}\, T_2[U_2/x]$.
La relation d'équivalence $\mathcal{R}$ *associée* à la relation ρ est la plus petite relation d'équivalence contenant ρ, soit $\mathcal{R}$ est définie par : $U \,\mathcal{R}\, U'$ ssi $\exists U_1, \ldots, U_n$ tels que $U \rho U_1$, $U_1 \rho U_2$, $\ldots$, $U_n \rho U'$.

La *beta-réduction* est la plus petite relation d'ordre passant au contexte contenant : si $x \notin U$ alors $(\lambda x U)V \to_\beta U[x/V]$. Si $T \twoheadrightarrow_\beta T'$ on dit que T' est un *beta-réduit* de T. La *êta-réduction* est la plus petite relation d'ordre passant au contexte contenant : $\lambda x(T)x \to_\eta T$. Si $T \twoheadrightarrow_\eta T'$ on dit que T' est un *êta-réduit* de T.
Un terme est en *forme normale* s'il ne contient aucun *redex* i.e., aucun sous-terme de la forme : $(\lambda x U)V$.
Un terme est en *forme normale de tête* s'il s'écrit : $\lambda x_1 \dots \lambda x_n(h)U_1 \dots U_p$ où h peut éventuellement être égale à l'un des x_i.
Une *stratégie* est une fonction associant à un terme T un redex de T. La réduction *associée* à une stratégie S est celle qui choisit le redex $S(T)$ à chaque étape de réduction, jusqu'à obtention de la forme normale. On appelle *stratégie gauche* la stratégie qui choisit le redex de gauche, et *réduction gauche* la beta-réduction associée à cette stratégie. On note $T \to_{\mathrm{G}} T'$ pour "T se réduit en T' par contraction du redex de gauche" et $T \overset{n}{\twoheadrightarrow}_{\mathrm{G}} T'$ pour "T se réduit à gauche en T' en n étapes". On appelle réduction de tête la réduction qui choisit toujours le redex de gauche lorsque celui-ci est à gauche de la première occurrence de variable du terme ; donc la réduction de tête s'arrête dès l'obtention d'une forme normale de tête du terme. Un terme dont la réduction de tête termine est *solvable* ou *résoluble*. Un terme dont la réduction gauche termine est *normalisable*. On appelle *norme gauche* la longueur de la réduction gauche de T que l'on note note $\|T\|_{\mathrm{G}}$, et *norme de tête* (notée $\|T\|_{\mathrm{h}}$) celle de sa réduction de tête. Un terme dont toute réduction termine est *fortement normalisable*. On appelle *norme* la longueur de la plus longue réduction de T que l'on note $\|T\|$.
On appelle *stratégie perpétuelle* ou *stratégie de Barendregt* la stratégie suivante :

$$
\begin{array}{lcll}
B(\lambda x T) & = & B(T) & \\
B\big((U)V\big) & = & B(U) & \text{si } U \text{ n'est pas normal et ne commence pas par } \lambda \\
B\big((U)V\big) & = & B(V) & \text{si } U \text{ est normal et ne commence pas par } \lambda \\
B\big((\lambda x U)V\big) & = & (\lambda x U)V & \text{si } x \in U \text{, ou si } V \text{ est normal} \\
B\big((\lambda x U)V\big) & = & B(V) & \text{si } x \notin U \text{ et } V \text{ n'est pas normal.}
\end{array}
$$

Barendregt a démontré dans [Bar89] que si la réduction perpétuelle de T termine alors T est fortement normalisable. On verra plus loin une généralisation de ce résultat. On appelle *norme Barendregt*, et on note $\|T\|_{\mathrm{B}}$ la longueur de la réduction par la stratégie B de T ; $T \to_{\mathrm{B}} T'$, respectivement $T \overset{n}{\twoheadrightarrow}_{\mathrm{B}} T'$ pour "T se réduit par contraction du redex $B(T)$ (respectivement par n contractions successives des redex $B(T)$) en T'".

1.2 Une généralisation de la réduction

On appellera désormais *redex visibles* les redex du lambda-calcul. On dira que λx est le *lambda de gauche* de T si :

- $T \equiv \lambda x U$ ou
- $T \equiv (\lambda y U)V V_1 \dots V_n$ et λx est le lambda de gauche de $(U)V_1 \dots V_n$

Plus généralement on dira que λx est le *kième lambda de gauche* de T si λx est le lambda de gauche de $(T)x_1 \cdots x_{k-1}$.
On appelle *occurrence de tête* ou *variable de tête* d'un terme T l'occurrence de variable la plus à gauche dans T. On la définit par induction comme suit : h est l'occurrence de tête de T si :

- $T \equiv h$;

– $T \equiv \lambda x U$ (avec éventuellement $x = h$) et h est la variable de tête de U ;

– $T \equiv (U)V$ et h est la variable de tête de V.

Remarquons que la variable de tête de T n'est pas en général la variable de tête de la forme normale de tête de T.
On définit les *arguments de la variable de tête* d'un terme T par induction :

– si $T \equiv (h)V_1 \cdots V_k$ alors les V_i sont les arguments de la variable de tête de T ;

– si $T \equiv \lambda x U$ ou si $T \equiv (U)V$ et U a un lambda de gauche, alors les arguments de la variable de tête de T sont les mêmes que les arguments de la variable de tête de U ;

– si $T \equiv (U)V$, U n'a pas de lambda de gauche et la variable de tête de U a k arguments $V_1, \ldots, V_k$ alors les arguments de la variable de tête de T sont $V_1, \ldots, V_k$, V ;

Un couple $(\lambda x, V)$ est un *redex premier* d'un terme T si T a un sous-terme de la forme $(U)V$ et λx est le lambda de gauche de U. On dira alors que V est *l'argument* de λx dans T.
Soit U un terme dont le lambda de gauche est λx de façon à ce que $(U)V$ soit un redex premier : on a alors

$$U \equiv \overbrace{(\cdots(}^{k_1} \lambda x_1 \cdots \lambda x_{l_1} \overbrace{(\cdots(}^{k_2} \lambda x_{l_1+1} \cdots \lambda x_{l_2} \cdots \overbrace{(\cdots(}^{k_n} \lambda x_{l_n+1} \cdots \lambda x_{l_{n+1}} \lambda x U_0)V_1) \cdots)V_m$$

avec $\forall p \leq n$, $\sum_{i=1}^{p} k_i \geq l_p$ et $\sum_{i=1}^{n} k_i = l_{n+1} = m$. Remarquons que cette condition implique que à chaque λx_i correspond un V_j tel que $(\lambda x_i, V_j)$ soit un redex premier de U.
On définit la réduction du redex $(\lambda x, V)$ dans $(U)V$ comme produisant le terme :

$$U' \equiv \overbrace{(\cdots(}^{k_1} \lambda x_1 \cdots \lambda x_{l_1} \overbrace{(\cdots(}^{k_2} \lambda x_{l_1+1} \cdots \lambda x_{l_2} \cdots \overbrace{(\cdots(}^{k_n} \lambda x_{l_n+1} \cdots \lambda x_{l_{n+1}} U_0[V/x])V_1) \cdots)V_m$$

en supposant que V n'a aucune des variables x_i libre et que les V_k n'ont pas la variable x libre ce que l'on peut toujours faire car on raisonne à alpha-conversion près.
Si on réduit m fois U à gauche on obtient $U'' \equiv \lambda x U_0[V_{f(i)}/x_i]$ où $f(i)$ est définie par : $f(i) = j$ ssi $(\lambda x_i, V_j)$ est un redex premier de U.
Donc on a :

$$(U)V =_\beta U_0([V_{f(i)}/x_i])_{1 \leq i \leq m}[V/x]$$

Comme, par le même raisonnement :

$$U' =_\beta U_0[V/x]([V_{f(i)}/x_i])_{1 \leq i \leq m}$$

et que x n'est libre dans aucun V_k (puisque λx ne lie que U_0), non plus que les x_i dans V (puisque V est en dehors de la portée des λx_i) on obtient $U =_\beta U'$.
Exemple : considérons le terme :

$$T \equiv ((\lambda x_1(\lambda x_2 \lambda x_3 \lambda x_4((\lambda x_5(x)U)V_5)W)V_2)V_1)V_3$$

T a quatre redex premiers $(\lambda x_i, V_i), i = 1, 2, 3, 5$ et un lambda de gauche λx_4 ; sa variable de tête x a deux arguments U et W.

On appelle *redex spinaux* de T les redex premiers dont les lambda sont à gauche de la variable de tête de T. On appelle *réduction spinale* une réduction choisisant un redex spinal à chaque étape. La réduction de tête est une réduction spinale, mais pas la réduction perpétuelle. On note $T \rightarrow_{\mathrm{S}} T'$ pour "T se réduit en T' par contraction d'un redex spinal" et $T \overset{n}{\twoheadrightarrow}_S T'$ pour "il y a une réduction spinale de longueur n menant de T à T'".
On appelle *lambda de tête* de T le lambda s'il existe liant la variable de tête de T et *redex de tête* de T le redex spinal dont le lambda est le lambda de tête de T. On dira que T est en *quasi forme normale de tête* si T n'a pas de redex de tête. Remarquons que si T est en quasi forme normale de tête alors n réductions spinales de T, où n est le nombre de redex spinaux de T, mènent à une forme normale de tête de T ; ou encore que le seul moyen de créer de nouveaux redex spinaux est de réduire le redex de tête.
On appelle *réduction par variable de tête* la réduction spinale qui choisit le redex de tête à chaque étape et s'arrête sur une quasi forme normale tête. On note $T \rightarrow_{\mathrm{T}} T'$ pour "T se réduit en T' par une contraction du redex de tête" et $T \overset{n}{\twoheadrightarrow}_T T'$ pour "T se réduit en T' par n contractions des redex de tête successifs".

Proposition 1.2.1 *Les réductions spinales sont* fortement confluentes *i.e., si* $T \rightarrow_{\mathrm{S}} T_i$ *où* i *est* 1 *ou* 2 *alors il existe* T' *tel que* $T_i \rightarrow_{\mathrm{S}} T'$

Preuve : on suppose que $T \rightarrow_{\mathrm{S}} T_1$ et $T \rightarrow_{\mathrm{S}} T_2$. Soit $(\lambda x_1, V_1)$ le redex spinal menant à T_1 et $(\lambda x_2, V_2)$ celui menant à T_2. Donc $R_2 \equiv (\lambda x_2, V_2[V_1/x_1])$ est un redex spinal de T_1 et de même $R_1 \equiv (\lambda x_1, V_1[V_2/x_2])$ en est un de T_1. Si on réduit respectivement R_1 dans T_2 et R_2 dans T_1 on obtient alors le même terme T'. •

Chapitre 2

L'équivalence opérationnelle

2.1 Définition

Comme on a vu plus haut, la forme générale d'un terme T possédant le redex premier $(\lambda x, V)$ est relativement compliquée, et ce bien que l'on se placait dans le cas simple où $T \equiv (U)V$. C'est pourquoi on va se donner une relation d'équivalence sur les termes qui égalisera ceux d'entre eux qui possédent exactement les mêmes redex premiers. En particulier, dans chaque classe on trouvera des termes pour lesquels tous les redex premiers sont visibles.

Définition 2.1.1 *On appelle* réduction opérationnelle *la plus petite relation passant au contexte contenant :*

$$((\lambda x U)V)W \rightarrow_{\text{op}} (\lambda x(U)W)V \ (x \notin W) \tag{2.1}$$
$$(\lambda x \lambda y U)V \rightarrow_{\text{op}} \lambda y(\lambda x U)V \ (y \notin V) \tag{2.2}$$

On notera $T \overset{n}{\twoheadrightarrow}_{\text{op}} T'$ quand T se réduit en T' par n applications de op, $T =_{\text{op}} T'$ quand il existe $T_1, \ldots, T_n$ tels que : $T \twoheadrightarrow_{\text{op}} T_1, T_2 \twoheadrightarrow_{\text{op}} T_1, \ldots T_{n-1} \twoheadrightarrow_{\text{op}} T_n, T' \twoheadrightarrow_{\text{op}} T_n$

Remarque : si $T \rightarrow_{\text{op}} T'$ alors $T =_{\beta} T'$. En particulier, si T est normal alors sa classe d'équivalence est réduite à lui seul.

Lemme 2.1.1 *Si $T_1 =_{\text{op}} T_2$ et $U_1 =_{\text{op}} U_2$ alors $T_1[U_1/x] =_{\text{op}} T_2[U_2/x]$.*

Preuve : car $=_{\text{op}}$ passe au contexte. •

Proposition 2.1.2 *Soit T et T' deux termes équivalents opérationnellement ; on a alors*

- *T et T' ont les mêmes variables libres ;*
- *si $\lambda x_1, \ldots, \lambda x_n$ sont les lambda de gauche de T alors ils sont aussi les lambda de gauche de T' ;*

- *si* $V_1, \ldots, V_k$ *sont les arguments de la variable de tête de* T *alors la variable de tête de* T' *a* k *argument* ${V'}_1, \ldots, {V'}_k$ *et pour tout* $i \leq k$, $V_i =_{\text{op}} {V'}_i$ *;*
- *si* $(\lambda y, U)$ *est un redex spinal de* T *alors* T' *a un redex spinal* $(\lambda y, U')$ *et* $U =_{\text{op}} U'$.

Preuve : comme $=_{\text{op}}$ est engendrée par $\rightarrow_{\text{op}}$, il suffit de vérifier la proposition dans les cas $T \rightarrow_{\text{op}} T'$ et $T' \rightarrow_{\text{op}} T$. La réduction opérationnelle procède par commutation d'une application définissant un redex visible avec soit un lambda, soit une application. Donc elle ne peut changer le statut d'un lambda (lambda de gauche ou lambda de redex premier) non plus que celui d'un argument (argument de la variable de tête ou argument d'un redex premier), pas plus qu'elle ne peut échanger deux lambda de gauche ou deux arguments de la variable de tête. •

Proposition 2.1.3 *La réduction opérationnelle termine toujours.*

Preuve : on note $M_T(\lambda x)$ le nombre de fois où λx est en position de fonction dans T i.e., le nombre de parenthèses ouvertes à gauche et fermées à droite de λx dans T. Alors pour chaque réduction opérationnelle, la somme sur tous les lieurs des $M_T(\lambda x)$ diminue strictement. •

Définition 2.1.2 *Les termes ou formes* canoniques *sont définis inductivement par les trois clauses suivantes :*

- *Si* $V_1, \ldots, V_m$ *sont des termes canoniques et* x *une variable alors* $(x)V_1 \ldots V_m$ *est un terme canonique ;*
- *Si* T *est un terme canonique alors* $\lambda x T$ *est un terme canonique ;*
- *si* T *et* U *sont des termes canoniques et* T *ne commence pas par lambda alors* $(\lambda x T)U$ *est un terme canonique.*

La forme générale d'un terme canonique est donc :

$$\lambda x_1 \cdots \lambda x_n (\lambda y_1 \cdots (\lambda y_m (h) U_1 \cdots U_p) V_m \cdots) V_1$$

où les U_i et les V_j sont des termes canoniques et h est soit une variable libre, soit égale à l'un des x_i ou des y_i. Remarquons que les termes canoniques sont les formes normales de la réduction opérationnelle. Dans un terme canonique les lambda de gauche sont exactement les n premiers lambda, les redex premiers sont tous visibles et les arguments de la variable de tête sont immédiatement accessibles. Qui plus est, comme les termes canoniques sont les formes normales de $\rightarrow_{\text{op}}$ et que celle-ci termine toujours, tout terme est opérationnellement équivalent à un terme canonique.

Remarque : Un même terme peut se réécrire en plusieurs termes canoniques différents : si $T \equiv ((\lambda x \lambda y U)V_1)V_2$ alors $T \rightarrow_{\text{op}} (\lambda x(\lambda y U)V_2)V_1$ et $T \rightarrow_{\text{op}} (\lambda y(\lambda x U)V_1)V_2$.

2.2 Permutation de radicaux par équivalence opérationnelle

Proposition 2.2.1 *Soient* T *et* T' *deux termes canoniques. Si* $T =_{\text{op}} T'$ *alors il existe une permutation* σ *telle que les égalités suivantes soient vérifiées :*

$$\begin{aligned} T &\equiv \lambda y_1 \cdots \lambda y_p (\lambda x_1 (\lambda x_2 \cdots (\lambda x_n U_0) U_n \cdots) U_2) U_1 \\ T' &\equiv \lambda y_1 \cdots \lambda y_p (\lambda x_{\sigma(1)} (\lambda x_{\sigma(2)} \cdots (\lambda x_{\sigma(n)} {U'}_0) {U'}_{\sigma(n)} \cdots) {U'}_{\sigma(2)}) {U'}_{\sigma(1)} \end{aligned}$$

où pour tout i, $U_i =_{\text{op}} U'_i$.

Preuve : c'est une conséquence immédiate de la propriété 2.1.2 •

Définition 2.2.1 *On dira qu'une permutation σ de $1, \dots, n$ est* admissible pour un terme T *si* $T \equiv (\lambda x_1(\lambda x_2 \cdots (\lambda x_n U_0)U_n \cdots)U_2)U_1$ *et pour toute inversion (i, j) de σ (i.e., $i < j$ et $\sigma(i) > \sigma(j)$) on a $x_{\sigma(j)} \notin U_{\sigma(i)}$.*

Proposition 2.2.2 *Soit T un terme canonique de la forme :*

$$T \equiv (\lambda x_1(\lambda x_2 \cdots (\lambda x_n U_0)U_n \cdots)U_2)U_1$$

Pour toute permutation σ de $1, \dots, n$, le terme canonique T' :

$$T_\sigma \equiv (\lambda x_{\sigma(1)}(\lambda x_{\sigma(2)} \cdots (\lambda x_{\sigma(n)} U_0)U_{\sigma(n)} \cdots)U_{\sigma(2)})U_{\sigma(1)}$$

est équivalent opérationnellement à T ssi σ est admissible

Preuve : supposons que σ n'est pas admissible pour T i.e., il existe une inversion (i, j) telle que $x_{\sigma(j)} \in U_{\sigma(i)}$. Donc dans T_σ la variable $x_{\sigma(i)}$ de $U_{\sigma(j)}$ est liée par $\lambda x_{\sigma(i)}$ alors qu'elle était libre dans T. Donc $T \neq_\beta T_\sigma$; or l'équivalence opérationnelle conserve la beta-équivalence donc $T \neq_{\text{op}} T_\sigma$.
Réciproquement on montre que si σ est admissible alors $T =_{\text{op}} T_\sigma$. Pour cela on définit la notion de *transposition contigüe* comme étant la transposition de i et $i+1$. On rappelle qu'une permutation ayant m inversions se décompose en un produit de m transpositions contigües. On se sert du lemme suivant :

Lemme 2.2.3 *Soit $\sigma = \tau_m \cdots \tau_1$, où les τ_i sont des transpositions contigües, une permutation ayant m inversions. Alors σ est admissible pour T ssi on a pour tout $i \leq m$:*

1. *$\tau_{i-1} \cdots \tau_1$ est une permutation admissible pour T ;*
2. *τ_i est admissible pour le terme $T_{\tau_{i-1} \cdots \tau_1}$.*

Preuve : on raisonne par induction sur m le nombre d'inversions de σ. Si $m = 0$ c'est que σ est l'identité et le résultat est clair.
Sinon $\sigma = \tau_m \sigma'$ où τ_m est une transposition contigüe et σ' a $m-1$ inversions donc est un produit de $m-1$ transpositions contigües. Donc τ_m échange $\sigma'(i)$ et $\sigma'(j) = \sigma'(i) + 1$ pour un couple (i, j) vérifiant $i < j$; soit σ a exactement les mêmes inversions que σ' plus l'inversion (i, j).
Si σ est admissible pour T, alors σ' ayant moins d'inversions l'est aussi. De plus comme (i, j) est une inversion de σ, on a $x_{\sigma(j)} \notin U_{\sigma(i)}$ donc τ_m est admissible pout $T_{\sigma'}$.
Si σ n'est pas admissible alors on doit voir que soit τ_m, soit σ' ne l'est pas. Supposons que σ' est admissible, alors comme σ a les mêmes inversions que σ' plus (i, j), c'est que cette dernière n'est pas correcte, donc que τ_m n'est pas admissible pour $T_{\sigma'}$. •

Reste pour terminer la démonstration à voir que si $\sigma = \tau_i$ est une transposition contigüe admissible alors T et T_σ sont équivalents opérationnellement. On a donc :

$$T_\sigma \equiv (\lambda x_1 \cdots (\lambda x_{i-1}(\lambda x_{i+1}(\lambda x_i(\lambda x_{i+2} \cdots (\lambda x_n U_0)U_n \cdots)U_{i+2})U_i)U_{i+1})U_{i-1} \cdots)U_1$$

Donc comme τ_i est admissible pour T i.e., $x_i = x_{\tau_i(i+1)} \notin U_{i+1} = U_{\tau_i(i)}$ on a:

$$T =_{\mathrm{op}} T' \equiv (\lambda x_1 \cdots (\lambda x_{i-1}((\lambda x_{i+1} \lambda x_i (\lambda x_{i+2} \cdots (\lambda x_n U_0) U_n \cdots) U_{i+2}) U_{i+1}) U_i) U_{i-1} \cdots) U_1$$

par la clause (2.2) de la définition de l'équivalence opérationnelle, et $T' =_{\mathrm{op}} T_\sigma$ par la clause (2.1). Remarquons que pour la même raison on a :

$$T =_{\mathrm{op}} T'' \equiv (\lambda x_1 \cdots (\lambda x_{i-1}((\lambda x_i \lambda x_{i+1} (\lambda x_{i+2} \cdots (\lambda x_n U_0) U_n \cdots) U_{i+2}) U_i) U_{i+1}) U_{i-1} \cdots) U_1$$

par la clause (2.1) et $T'' =_{\mathrm{op}} T_\sigma$ par la clause (2.2). •

2.3 Equivalence opérationnelle et réduction

On va maintenant s'attacher à montrer que l'équivalence opérationnelle n'introduit aucune différence majeure dans la réduction des termes. Dans ce qui suit on entendra donc par réduction la beta-réduction habituelle, et par redex les redex habituels i.e., les redex visibles, du lambda-calcul.

Proposition 2.3.1 *Si $T_1 \twoheadrightarrow_{\mathrm{op}} T_2$ et $T_1 \rightarrow_\beta T'_1$ alors il existe T'_2 tel que $T_2 \rightarrow_\beta T'_2$ et $T'_1 \twoheadrightarrow_{\mathrm{op}} T'_2$.*

Preuve : on se contente de prouver la proposition quand $T_1 \rightarrow_{\mathrm{op}} T_2$ car le reste de l'argument vient par induction sur la longueur de la réduction opérationnelle menant de T_1 à T_2. Soit $(\lambda x, V_1)$ le redex de T_1 que l'on réduit. Comme on l'a vu c'est aussi un redex de T_2 i.e., caractérisé par le même lambda, dont la réduction donne T'_2. Reste à voir que $T'_1 \twoheadrightarrow_{\mathrm{op}} T'_2$. On a $T_1 \equiv C_1[(\lambda x U_1) V_1]$ et $T'_1 \equiv C_1[U_1[V_1/x]]$. Si $T_2 \equiv C_2[(\lambda x U_1) V_1]$ avec $C_1[] \rightarrow_{\mathrm{op}} C_2[]$ alors on a bien $C_1[U_1[V_1/x]] \rightarrow_{\mathrm{op}} C_2[U_1[V_1/x]]$. Si $T_2 \equiv C_1[(\lambda x U_2) V_1]$ avec $U_1 \rightarrow_{\mathrm{op}} U_2$ alors on a bien $C_1[U_1[V_1/x]] \rightarrow_{\mathrm{op}} C_1[U_2[V_1/x]]$. Si $T_2 \equiv C_1[(\lambda x U_1) V_2]$ avec $V_1 \rightarrow_{\mathrm{op}} V_2$ alors $C_1[U_1[V_1/x]] \overset{n}{\twoheadrightarrow}_{\mathrm{op}} C_2[U_1[V_2/x]]$ où n est le nombre d'occurrences de x dans U_1. Si aucune de ces situations n'a lieu c'est que $T_1 \equiv C[((\lambda x U) V) W]$ et $T_2 \equiv C[(\lambda x (U) W) V]$ ou que $T_1 \equiv C[(\lambda x \lambda y U) V]$ et $T_2 \equiv C[\lambda y (\lambda x U) V]$ mais après réduction du redex on obtient $T'_1 \equiv T'_2$. •

Corollaire 2.3.2 *Si $T_1 \twoheadrightarrow_{\mathrm{op}} T_2$ alors $\|T_1\| \leq \|T_2\|$.*

Preuve : on vient de voir que l'on pouvait calquer toute réduction de T_1, en particulier la plus longue, sur T_2. •

Remarque : l'inégalité inverse est plus délicate : si $T_1 \equiv ((\lambda x \lambda y U) V) W$ et $T_2 \equiv (\lambda x (\lambda y U) W) V$, on a $T_2 \rightarrow_\beta T'_2 \equiv (\lambda x U[W/y]) V$ mais on ne peut pas réduire T_1 en un terme équivalent opérationnellement à T'_2 puisque $(\lambda y, W)$ n'est pas un redex de T_1. Notons qu'avec la réduction des redex premiers, cette difficulté disparait puisque $(\lambda y, W)$ est un redex premier aussi bien de T_1 que de T_2.

2.4 Réduction gauche

Proposition 2.4.1 *Si $T_1 \rightarrow_{\mathrm{op}} T_2$ et $T_1 \rightarrow_{\mathrm{G}} T'_1$ alors il existe T''_1 et T''_2 tels que $T'_1 \overset{n}{\twoheadrightarrow}_{\mathrm{G}} T''_1$, $T_2 \overset{n+1}{\twoheadrightarrow}_{\mathrm{G}} T''_2$ et $T''_1 \twoheadrightarrow_{\mathrm{op}} T''_2$. En particulier on a $\|T_1\|_{\mathrm{G}} = \|T_2\|_{\mathrm{G}}$.*

Preuve : si le redex de gauche de T_2 est caractérisé par le même λy alors comme on l'a vu pour la proposition 2.3.1, $T_2 {\rightarrow_{\mathrm{G}}} T'_2$ et $T'_1 \twoheadrightarrow_{\mathrm{op}} T'2$; on choisit donc $T''_1 \equiv T'_1$ et $T''_2 \equiv T'_2$. La seule possibilité restante est que la réduction opérationnelle ait modifié le redex de gauche : soit $T_1 \equiv \lambda x_1 \cdots \lambda x_n(\lambda x \lambda y U) V_1 V_2 \cdots V_m$ et $T_2 \equiv \lambda x_1 \cdots \lambda x_n(\lambda y(\lambda x U) V_1) V_2 \cdots V_m$. Dans T_1 le redex de gauche est $(\lambda x, V_1)$ alors que dans T_2 c'est $(\lambda y, V_2)$. Cela dit en deux réductions gauches de T_1 et T_2 on aboutit au même terme T''. •

Remarque : la première réduction opérationnelle ne modifie pas le redex de gauche.

Corollaire 2.4.2 *Si $T_1 =_{\mathrm{op}} T_2$ alors :*

1. $\|T_1\|_{\mathrm{G}} = \|T_2\|_{\mathrm{G}}$;
2. $\|T_1\|_{\mathrm{h}} = \|T_2\|_{\mathrm{h}}$;
3. *si T_1^h (resp. T_2^h) est la forme normale de tête obtenue par réduction de tête de T_1 (resp. T_2), alors $T_1^h =_{\mathrm{op}} T_2^h$.*

2.5 Norme Barendregt

Proposition 2.5.1 *Si $T_1 \twoheadrightarrow_{\mathrm{op}} T_2$ alors $\|T_1\|_{\mathrm{B}} = \|T_2\|_{\mathrm{B}}$.*

Preuve : on ne s'intéresse qu'au cas où $T_1 \rightarrow_{\mathrm{op}} T_2$, le reste venant par induction sur la longueur de la réduction opérationnelle.

Lemme 2.5.2 *Si $T_1 \rightarrow_{\mathrm{op}} T_2$ alors l'une des deux propositions suivantes est réalisées :*

1. *il existe T'_1, T'_2 et n tels que : $T_1 \overset{n}{\twoheadrightarrow}_{\mathrm{B}} T'_1$, $T_2 \overset{n}{\twoheadrightarrow}_{\mathrm{B}} T'_2$ et $T'_1 \twoheadrightarrow_{\mathrm{op}} T'_2$;*
2. $\|T_1\|_{\mathrm{B}} = \|T_2\|_{\mathrm{B}} = \infty$.

Notons que les deux cas ne sont pas exclusifs. Prouvons d'abord la proposition : on raisonne par cas sur $inf(\|T_1\|_{\mathrm{B}}, \|T_2\|_{\mathrm{B}})$. Si c'est infini, alors on a bien $\|T_1\|_{\mathrm{B}} = \|T_2\|_{\mathrm{B}}$. Si $\|T_1\|_{\mathrm{B}} < \infty$ et $\|T_2\|_{\mathrm{B}} \leq \infty$ (resp. $\|T_1\|_{\mathrm{B}} \leq \infty$ et $\|T_2\|_{\mathrm{B}} < \infty$), on raisonne par induction sur $\|T_1\|_{\mathrm{B}}$ (resp. $\|T_2\|_{\mathrm{B}}$) : on est forcément dans le cas 1. du lemme et on a une suite $R_0, R_1, \ldots, R_p$ telle que : $R_0 \equiv T'_1$, $R_p \equiv T'_2$, et $R_i \rightarrow_{\mathrm{op}} R_{i+1}$. On a : $\|T'_1\|_{\mathrm{B}} = \|T_1\|_{\mathrm{B}} - n$, $\|T'_2\|_{\mathrm{B}} = \|T_2\|_{\mathrm{B}} - n$ et par induction sur $\|T'_1\|_{\mathrm{B}}$ (resp. $\|T_2\|_{\mathrm{B}}$) : $\|T'_1\|_{\mathrm{B}} = \|R_1\|_{\mathrm{B}} = \|R_2\|_{\mathrm{B}} \cdots = \|T'_2\|_{\mathrm{B}}$. •

Puis le lemme : soient R_1 et R_2 les deux sous-termes de respectivement T_1 et T_2 tels que : (A) $R_1 =_{\mathrm{op}} R_2$, $B(T_1) = B(R_1)$ et $B(T_2) = B(R_2)$. La notation $B(T_1) = B(R_1)$ signifie que les deux termes sont le même sous-terme de T_1 (et pas simplement deux termes α-équivalents). Supposons de plus que R_1 soit minimal pour (A). On raisonne par cas en considérant les premiers symboles de R_1 (parenthèse ouvrante ou λ). On traite successivement les cas : λ, $(\lambda($, $(\lambda\lambda$, $((($, $((\lambda($, $((\lambda\lambda$.

1. $R_1 \equiv \lambda x U_1$ alors $R_2 \equiv \lambda x U_2$ par définition de $\rightarrow_{\mathrm{op}}$. Mais dans ce cas, $B(R_1) \equiv B(U_1)$, $B(R_2) \equiv B(U_2)$ et $U_1 \rightarrow_{\mathrm{op}} U_2$ contredisant la minimalité de R_1.
2. $R_1 \equiv (\lambda x U_1) U_2$, U_1 ne commence pas par λ. Donc $R_2 \equiv (\lambda x V_1) V_2$ avec $U_1 \rightarrow_{\mathrm{op}} V_1$ ou $U_2 \rightarrow_{\mathrm{op}} V_2$. Si $x \in U_1$ alors $x \in V_1$ et $B(R_1) \equiv R_1$, $B(R_2) \equiv R_2$; donc $R_1 {\rightarrow_{\mathrm{B}}} R'_1 \equiv U_1[U_2/x]$ et $R_2 {\rightarrow_{\mathrm{B}}} R'_2 \equiv V_1[V_2/x]$ soit le cas 1. du lemme avec $n = 1$ (car $R'_1 \twoheadrightarrow_{\mathrm{op}} R'_2$ par la proposition 2.1.1).
   Si $x \notin U_1$, alors $B(R_1) \equiv B(U_2)$ et $B(R_2) \equiv B(V_2)$ contredisant la minimalité de R_1.

3. $R_1 \equiv (\lambda x \lambda y U_1)U_2$. On retrouve les deux cas précédents, et un troisième : $R_2 \equiv \lambda y(\lambda x U_1)U_2$. Si $x \in U_1$ alors $B(R_1) \equiv R_1$ et $B(R_2) \equiv (\lambda x U_1)U_2$. Donc par réduction de $B(R_1)$ et $B(R_2)$ dans respectivement T_1 et T_2, on obtient le même T'.
   Si $x \notin U_1$ alors $B(R_1) \equiv B(U_2)$ et $B(R_2) \equiv B(U_2)$ contredisant la minimalité de R_1.

4. $R_1 \equiv ((U_1)U_2)U_3$, U_1 ne commence pas par λ. Alors $R_2 \equiv (V)V_3$ avec soit $(U_1)U_2 \rightarrow_{\mathrm{op}} V$ et $U_3 \equiv V_3$, soit $(U_1)U_2 \equiv V$ et $U_3 \rightarrow_{\mathrm{op}} V_3$. Dans les deux cas $B(R_1) \equiv B\big((U_1)U_2\big)$ (resp. $B(U_3)$ si $(U_1)U_2$ est normal) et $B(R_2) \equiv B(V)$ (resp. $B(V_3)$) contredisant la minimalité de R_1.

5. $R_1 \equiv ((\lambda x U_1)U_2)U_3$, U_1 ne commence pas par λ. Alors on peut considérer le cas : $R_2 \equiv (\lambda x(U_1)U_3)U_2$. Si $x \in U_1$ alors $B(R_1) \equiv (\lambda x U_1)U_2$, $B(R_2) \equiv R_2$ et $R_1, R_2 \rightarrow_{\mathrm{B}} (U_1[U_2/x])U_3$, soit le cas 1. du lemme avec $n = 1$.
   Si $x \notin U_1$ alors $B(R_1) \equiv B(U_2)$ et $B(R_2) \equiv B(U_2)$ contredisant la mininalité de R_1.

6. $R_1 \equiv ((\lambda x \lambda y U_1)U_2)U_3$ ce qui rajoute le cas : $R_2 \equiv (\lambda y(\lambda x U_1)U_2)U_3$.
   Si $x, y \in U_1$ alors $B(R_1) \equiv R_1$ donc $R_1 \rightarrow_{\mathrm{B}} R'_1 \equiv (\lambda y U_1[U_2/x])U_3$, soit $T_1 \rightarrow_{\mathrm{B}} T'_1$. Du fait que $y \in U_1$, on a $B(T'_1) \equiv R'_1$. Donc $R'_1 \rightarrow_{\mathrm{B}} R'' \equiv U_1[U_2/x, U_3/y]$, soit $T'_1 \rightarrow_{\mathrm{B}} T''$. Mais deux réductions de Barendregt mènent au même résultat pour R_2 (compte tenu du fait que par définition de $\rightarrow_{\mathrm{op}}$, $x \notin U_3$ et $y \notin U_2$). Donc $T_1 \stackrel{2}{\twoheadrightarrow}_{\mathrm{B}} T''$ et $T_2 \stackrel{2}{\twoheadrightarrow}_{\mathrm{B}} T''$.
   Si $y \notin U_1$ alors la stratégie B va d'abord mettre U_3 en forme normale. S'il existe n tel que $R_1 \rightarrow_{\mathrm{B}} R'_1 \stackrel{n}{\twoheadrightarrow}_{\mathrm{B}} (\lambda y U_1[U_2/x])U_3^0 \rightarrow_{\mathrm{B}} R'' \equiv U_1[U_2/x]$, soit $T_1 \stackrel{n+2}{\twoheadrightarrow}_{\mathrm{B}} T''$, alors $R_2 \stackrel{n}{\twoheadrightarrow}_{\mathrm{B}} (\lambda y(\lambda x U_1)U_2)U_3^0$ $\rightarrow_{\mathrm{B}} (\lambda x U_1)U_2 \rightarrow_{\mathrm{B}} R''$. Donc $T_2 \stackrel{n+2}{\twoheadrightarrow}_{\mathrm{B}} T''$. Sinon i.e., si la réduction de Barendregt de U_3 est infinie, alors la réduction de Barendregt de R_2 aussi est infinie d'où $\|T_2\|_{\mathrm{B}} = \infty$ soit le cas 2. du lemme.
   Le cas où $x \notin U_1$ et $y \in U_1$ est symétrique du précédent, reste celui où $x, y \notin U_1$. Alors s'il existe n et p tels que $R_1 \stackrel{n}{\twoheadrightarrow}_{\mathrm{B}} ((\lambda x \lambda y U_1)U_2^0)U_3 \rightarrow_{\mathrm{B}} (\lambda y U_1)U_3 \stackrel{p}{\twoheadrightarrow}_{\mathrm{B}} (\lambda y U_1)U_3^0 \rightarrow_{\mathrm{B}} U_1$, donc $T_1 \stackrel{n+p+2}{\twoheadrightarrow}_{\mathrm{B}} T''$, clairement il en est de même de T_2 (en inversant n et p). Si n ou p ou les deux n'existe pas, comme précédemment on en conclut qu'on est dans le cas 2. •

Corollaire 2.5.3 *Si $T_1 =_{\mathrm{op}} T_2$ alors $\|T_1\|_{\mathrm{B}} = \|T_2\|_{\mathrm{B}}$.*

Preuve : $=_{\mathrm{op}}$ est la relation d'équivalence associée à $\rightarrow_{\mathrm{op}}$. •

2.6 Norme et norme Barendregt

La proposition suivante est une généralisation du théorème de Barendregt :

Proposition 2.6.1 *Soient T et T' tels que : $\|T\| < \infty$ et $T \rightarrow_\beta T'$. Alors $\|T\|_{\mathrm{B}} > \|T'\|_{\mathrm{B}}$.*

Remarque : autrement dit, toute réduction fait strictement diminuer la norme Barendregt.
preuve : par induction sur $\|T\|_{\mathrm{B}}$. On utilise le lemme suivant :

Lemme 2.6.2 *Soient T un terme canonique, T' et T_B tels que : $T \rightarrow_\beta T'$ et $T \rightarrow_{\mathrm{B}} T_B$. L'une des deux propositions suivantes est réalisée :*

1. *il existe deux entiers n, p et un terme S tels que $n \geq p$ $T_B \stackrel{n}{\twoheadrightarrow}_\beta S$ et $T' \stackrel{p}{\twoheadrightarrow}_{\mathrm{B}} S$;*
2. *$\|T'\|_{\mathrm{B}} = \infty$ et $\|T_B\| = \infty$*

Commençons par la preuve de la proposition : par induction sur $\|T_B\|$ et sur n on a : pour tout S tel que $T_B \xrightarrow{n}_\beta S$, $\|T_B\|_{\mathrm{B}} \geq \|S\|_{\mathrm{B}} + n$. Supposons que T est en forme canonique de façon à pouvoir appliquer le lemme. Si on est dans le cas 2. on contredit $\|T\| < \infty$. Par conséquent il existe S comme dans le cas 1. Mais alors on a : $\|T\|_{\mathrm{B}} = \|T_B\|_{\mathrm{B}} + 1 \geq \|S\|_{\mathrm{B}} + n + 1 = \|T'\|_{\mathrm{B}} - p + n + 1 \geq \|T'\|_{\mathrm{B}} + 1$ car la réduction de T' à S est de Barendregt.
Si maintenant T n'est pas en forme canonique, soit T_c une forme canonique pour T i.e., $T \twoheadrightarrow_{\mathrm{op}} T_c$. On applique la proposition 2.3.1 à T, T' et T_c obtenant T'_c tel que $T' \twoheadrightarrow_{\mathrm{op}} T'_c$. Par le raisonnement ci-dessus on a $\|T_c\|_{\mathrm{B}} \geq \|T'_c\|_{\mathrm{B}} + 1$ et l'on obtient le résultat pour T et T' par la proposition 2.5.1. $\bullet$
Passons au lemme : on le prouve par induction sur la forme canonique T.

- $T \equiv \lambda x U$: alors on a le résultat par induction sur U.

- $T \equiv (\lambda x U)V$ et $x \in U$: alors $T_B \equiv U[V/x]$.
  Si $T' \equiv T_B$ il n'y a rien à dire. Si $T' \equiv (\lambda x U')V$ et $x \in U'$ alors $T' \to_{\mathrm{B}} U'[V/x]$. Mais $T_B \to_\beta U'[V/x]$ soit $n = p = 1$.
  $T' \equiv (\lambda x U')V$ et $x \notin U'$: s'il existe q tel que $T' \xrightarrow{q}_{\mathrm{B}} (\lambda x U')V_0$ où V_0 est en forme normale alors $T' \xrightarrow{q+1}_{\mathrm{B}} U'$. Notons k le nombre d'occurrences de x dans U, k est donc supérieur ou égal à 1. On a en réduisant les k sous-termes V de T_B en V_0 : $T_B \xrightarrow{kq}_\beta U[V_0/x]$ soit $T_B \xrightarrow{kq+1}_\beta U'$ d'où le résultat avec $n = kq + 1$ et $p = q + 1$. Si q n'existe pas i.e., $\|T'\|_{\mathrm{B}} = \infty$ alors on a une réduction infinie de V donc un réduction infinie de T_B soit le cas 2. du lemme.
  $T' \equiv (\lambda x U)V'$: alors comme $x \in U$, $T' \to_{\mathrm{B}} U[V'/x]$. Mais $T_B \xrightarrow{k}_\beta U[V'/x]$ d'où le résultat avec $n = k$ et $p = 1$.

- $T \equiv (\lambda x U)V$, $x \notin U$ et V n'est pas normal : alors $T_B \equiv (\lambda x U)V_B$. Notons que puisque T est en forme canonique, V l'est aussi.
  Si $T' \equiv (\lambda x U')V$ alors comme $x \notin U'$, $T' \to_{\mathrm{B}} (\lambda x U')V_B$. Mais $T_B \to_\beta (\lambda x U')V_B$ soit $n = p = 1$.
  Si $T' \equiv (\lambda x U)V'$ alors par induction sur V, soit $\|V'\|_{\mathrm{B}} = \infty$ et $\|V_B\| = \infty$ auquel cas il en est de même de T' et T_B, soit il existe n et p, $n \geq p$ et W tels que $V_B \xrightarrow{n}_\beta W$ et $V' \xrightarrow{p}_{\mathrm{B}} W$. Mais alors $T' \xrightarrow{p}_{\mathrm{B}} (\lambda x U)W$ et $T_B \xrightarrow{n}_\beta (\lambda x U)W$.
  Si $T' \equiv U$ alors $T_B \to_{\mathrm{B}} U$ soit $n = 1$ et $p = 0$.

- $T \equiv (\lambda x U)V$, $x \notin U$ et V est normal : alors $T_B \equiv U$. Le cas non trivial est $T' \equiv (\lambda x U')V$: alors $T' \to_{\mathrm{B}} U'$ et $T_B \to_\beta U'$ soit $n = p = 1$.

- $T \equiv (x)V^1 \cdots V^q$: alors $T_B \equiv (x)V^1 \cdots V^j_B \cdots V^q$ où V^j est le premier argument de x non normal. Notons que tous les V^i sont en forme canonique.
  Si $T' \equiv (x)V^1 \cdots V'^j \cdots V^q$ alors par induction sur V^j on obtient le résultat. Sinon $T' \equiv (x)V^1 \cdots V^j \cdots V'_k \cdots V^q$ et $T' \to_{\mathrm{B}} S \equiv (x)V^1 \cdots V^j_B \cdots V'_k \cdots V^q$, et $T_B \to_\beta S$ soit $n = p = 1$.$\bullet$

Remarque : il est intéressant de noter la construction de la preuve. On prouve le lemme par induction sur les formes canoniques ce qui simplifie l'étude par cas. Pour s'en convaincre, considérons le cas où $T \equiv (U)V$, U ne commencant pas par lambda. Si T est une forme canonique, c'est la dernière étape de la preuve, mais si ce n'est pas le cas, l'hypothèse d'induction devient inutilisable : supposons que $T_B \equiv (U_B)V$, et $T' \equiv (U')V$. Par hypothèse d'induction sur U on sait qu'il existe n, p et U'' tels que $U_B \xrightarrow{n}_\beta U''$ et $U' \xrightarrow{p}_{\mathrm{B}} U''$. On en déduit bien que $T_B \xrightarrow{n}_\beta S \equiv (U'')V$ mais pas que $T' \xrightarrow{p}_{\mathrm{B}} S$ car il se peut que : $U' \xrightarrow{p_1}_{\mathrm{B}} \lambda x U_1 \xrightarrow{p_2}_{\mathrm{B}} U'' \equiv \lambda x U_2$ avec $p_1, p_2 \neq 0$, et $p = p_1 + p_2$. Supposons que p_1 soit minimal pour cette propriété, alors $T' \xrightarrow{p_1}_{\mathrm{B}} T_1 \equiv (\lambda x U_1)V$ et ne se réduit donc pas par la stratégie

B en S. On peut résoudre ce problème en renforcant l'hypothèse d'induction mais cela complique considérablement la preuve. Il est beaucoup plus aisé de passer par les formes canoniques.

Corollaire 2.6.3 *Pour tout terme T, $\|T\| = \|T\|_{\text{B}}$.*

Preuve : par définition de $\|T\|$ on a $\|T\| \geq \|T\|_{\text{B}}$, ce qui règle le cas où $\|T\|_{\text{B}} = \infty$. Pour l'autre inégalité, on raisonne par induction sur $\|T\|_{\text{B}}$. Soit T' un réduit à une étape de T. Par la proposition ci-dessus, $\|T\|_{\text{B}} > \|T'\|_{\text{B}}$. Par hypothèse d'induction, $\|T'\| = \|T'\|_{\text{B}}$. Donc : pour tout T' réduit à une étape de T, $\|T\|_{\text{B}} > \|T'\|$. Comme $\|T\| = \|T'\| + 1$ pour au moins un T', on obtient finalement $\|T\|_{\text{B}} \geq \|T\|$. •

Corollaire 2.6.4 *Si $T_1 =_{\text{op}} T_2$ alors $\|T_1\| = \|T_2\|$.*

Preuve : c'est une conséquence des propositions 2.5.1 et 2.6.3. •

2.7 Conclusion

La réduction opérationnelle permet de "visualiser" les redex premiers d'un terme tout en laissant inchangées ses propriétés opérationnelles (norme, norme gauche, norme de tête, nombre de réductions gauches jusqu'à la forme normale, unicité de celle-ci, etc...). Autrement dit on peut, et c'est inoffensif, généraliser la notion de redex à celle de redex premier. Le problème est alors de reconnaitre les redex premiers. Pour ce faire on peut soit les calculer en utilisant leur définition, soit passer par les termes canoniques ; mais aucune de ces opérations n'est atomique. Une troisième solution, qui est présentée ici, consite à trouver une syntaxe pour les lambda-termes dans laquelle les redex premiers sont visibles.

Chapitre 3

Les systèmes D et DΩ

Ce chapitre présente le système **DΩ**, connu en premier lieu sous le nom de *intersection type discipline* de Coppo et Dezani (voir [CoDe86]). Il n'est pas question ici de démontrer les propriétés, nombreuses pourtant du système **DΩ**, mais de faire une petite liste commentée de celles qui nous serons utiles pour la suite. Par conséquent on ne donnera ici que les preuves des propositions qui ne se trouvent pas dans [Kri90], auquel on empruntera les notations.

3.1 Définitions

Le système **DΩ** est un système d'assignation de type aux lambda-termes typant tous les termes. On dérive des *séquents* de la forme :

$$x_1 : X_1, \ldots, x_n : X_n \vdash_{\mathrm{D}\Omega} T : P$$

où les x_i sont des variables du lambda-calcul, les X_i et P sont des *types*. La partie gauche du séquent est appelée le *contexte de variables* du terme ; elle contient les variables libres du terme T. Les contextes de variables seront symbolisés par la lettre Γ. On considérera les contextes comme des ensembles non ordonnés d'occurrences de variables typées.
Les types sont :

- la constante Ω ;
- les variables de type, que l'on choisit dans le même ensemble V que les variables du lambda-calcul ;
- les types *flèche* : si N et P sont des types alors $N \to P$ est un type. On notera $N_1 \to \ldots \to N_n \to P$ le type : $N_1 \to (N_2 \to \ldots \to (N_n \to P) \ldots)$;
- les types *intersection* ou *conjonctifs* : si N_1 et N_2 sont des types alors $N_1 \cap N_2$ est un type.

REMARQUE : le fait que l'on utilise les mêmes noms pour les variables du lambda-calcul et les variables de type n'a aucune conséquence mathématique. Dans la suite on se servira des variables de types pour distinguer différentes occurrences d'une même variable de terme. On verra souvent des contextes de

la forme : $x : x_1, x : x_2$ dans lequel deux occurrences de la variable x sont de type respectivement x_1 et x_2.
On définit le formalisme dans une présentation de type déduction naturelle. Dans ce qui suit, Γ est un contexte de variables, N, P, P_1, P_2 et X des types.

1. **axiome** :

$$\frac{}{\Gamma, x : X \vdash_{\mathrm{D}\Omega} x : X}$$

2. **$\to$-introduction** :

$$\frac{\Gamma, x : X \vdash_{\mathrm{D}\Omega} T : P}{\Gamma \vdash_{\mathrm{D}\Omega} \lambda x T : X \to P}$$

3. **$\to$-élimination**

$$\frac{\Gamma \vdash_{\mathrm{D}\Omega} U : N \to P \quad \Gamma \vdash_{\mathrm{D}\Omega} V : N}{\Gamma \vdash_{\mathrm{D}\Omega} (U)V : P}$$

4. **$\cap$-introduction**

$$\frac{\Gamma \vdash_{\mathrm{D}\Omega} T : P_1 \quad \Gamma \vdash_{\mathrm{D}\Omega} T : P_2}{\Gamma \vdash_{\mathrm{D}\Omega} T : P_1 \cap P_2}$$

5. **$\cap$-élimination**

$$\frac{\Gamma \vdash_{\mathrm{D}\Omega} T : P_1 \cap P_2}{\Gamma \vdash_{\mathrm{D}\Omega} T : P_i}$$

   où i est 1 ou 2.

6. **règle Omega**

$$\frac{}{\Gamma \vdash_{\mathrm{D}\Omega} T : \Omega}$$

Si on supprime la dernière règle on obtient le *système* **D** dans lequel on prouve des séquents de la forme : $\Gamma \vdash_{\mathrm{D}} T : P$. Remarquons que dans **D** le symbole Ω joue exactement le même rôle qu'une variable de type.

Lemme 3.1.1 *Soit Γ un contexte et T un terme typable de type P non conjonctif dans ce contexte :*

- *Si $T \equiv \lambda x U$ alors $P = P_1 \to P_2$ et il existe une dérivation de $\Gamma, x : P_1 \vdash_{\mathrm{D}\Omega} U : P_2$.*
- *Si $T \equiv (U)V$ alors il existe un type N et des dérivations de $\Gamma \vdash_{\mathrm{D}\Omega} U : N \to P$, $\Gamma \vdash_{\mathrm{D}\Omega} V : N$.*

La propriété suivante montre que l'équivalence opérationnelle est indépendante du typage dans **D**Ω.

Proposition 3.1.2 *Soient $T_1 =_{\mathrm{op}} T_2$; pour tout contexte de variables Γ et tout type P, si $\Gamma \vdash_{\mathrm{D}\Omega} T_1 : P$ alors $\Gamma \vdash_{\mathrm{D}\Omega} T_2 : P$.*

Preuve : on commence par supposer $T_1 \twoheadrightarrow_{\mathrm{op}} T_2$. On raisonne par induction sur le terme puis pour un terme donné, sur le type.
Si P est un type conjonctif, soit $P = P_1 \cap P_2$, alors T_1 est de type P_1 par une règle d'élimination du $\cap$. Par hypothèse d'induction T_2 est aussi de type P_1. On fait de même avec P_2 et par une règle d'introduction du $\cap$ on déduit que T_2 est de type $P_1 \cap P_2$. Si $P = \Omega$ alors T_2 est de type P par la règle Ω.
On suppose maintenant que P n'est pas un type conjonctif, ni Ω.
Si T_1 est une variable typée, alors $T_1 \twoheadrightarrow_{\mathrm{op}} T_2$ implique $T_1 \equiv T_2$.
Si $T_1 \equiv \lambda x T'_1$ est de type $X \to P$ dans le contexte Γ avec $T'_1 : P$ dans le contexte $\Gamma, x : X$; comme $T_1 \twoheadrightarrow_{\mathrm{op}} T_2$ on a $T_2 \equiv \lambda x T'_2$ avec $T'_1 \twoheadrightarrow_{\mathrm{op}} T'_2$. Par hypothèse d'induction sur T'_1, $\Gamma, x : X \vdash_{\mathrm{D}\Omega} T'_2 : P$ d'où par application d'une règle d'introduction de flèche : $\Gamma \vdash_{\mathrm{D}\Omega} T_2 : P$.
Si $T_1 \equiv (U_1)V_1$ alors par le lemme on a un type N tel que $\Gamma \vdash_{\mathrm{D}\Omega} U_1 : N \to P$ et $\Gamma \vdash_{\mathrm{D}\Omega} V_1 : N$. Si $T_2 \equiv (U_2)V_2$ avec $U_1 \twoheadrightarrow_{\mathrm{op}} U_2$ et $V_1 \twoheadrightarrow_{\mathrm{op}} V_2$ alors par induction on a $\Gamma \vdash_{\mathrm{D}\Omega} U_2 : N \to P$ et $\Gamma \vdash_{\mathrm{D}\Omega} V_2 : N$ d'où le résultat. Sinon c'est que $U_1 \equiv (\lambda x U'_1)U''_1$ ou $\lambda x \lambda y U'_1$.

- Dans le premier cas, $T_2 \equiv (\lambda x (U'_2)V_2)U''_2$ avec $U'_1 \twoheadrightarrow_{\mathrm{op}} U'_2$, $U''_1 \twoheadrightarrow_{\mathrm{op}} U''_2$. De la dérivation de T_1 on peut extraire des sous-dérivations de $\Gamma, x : N'' \vdash_{\mathrm{D}\Omega} U'_1 : N \to P$, $\Gamma \vdash_{\mathrm{D}\Omega} U''_1 : N''$ et $\Gamma \vdash_{\mathrm{D}\Omega} V_1 : N$ d'où le résultat en utilisant l'hypothèse d'induction et en réarrangeant ces dérivations.
- Sinon $T_2 \equiv \lambda y(\lambda x U_2)V_2$ ce qui se traite similairement.

Dans le cas où $T_2 \twoheadrightarrow_{\mathrm{op}} T_1$ on raisonne exactement de la même manière, d'où l'on déduit le résultat pour $=_{\mathrm{op}}$ par une simple induction. •

3.2 Typages principaux des termes normaux

Théorème 3.2.1 *Si $T =_\beta T'$ alors pour tout contexte de variables Γ et tout type P, $\Gamma \vdash_{\mathrm{D}\Omega} T : P$ ssi $\Gamma \vdash_{\mathrm{D}\Omega} T' : P$.*
Réciproquement, si T et T' ont les mêmes types, alors $T =_\beta T'$.

Autrement dit tous les termes d'une classe de beta-équivalence sont typables des mêmes types dans $\mathbf{D}\Omega$, et réciproquement, l'ensemble des types d'un terme caractérise une classe. Dans le cas où le terme est normalisable, on va voir qu'il existe même un type qui caractérise la beta-classe.

Définition 3.2.1 *On appelle* typage principal *d'un terme normal les dérivations de type dans le système suivant que l'on note* **Dn** *:*
Les types de **Dn** *sont des types de* **D**, *et les règles sont :*

1. **Application simple** *:*

$$\frac{\Gamma_1 \vdash_{\mathrm{Dn}} U_1 : P_1 \ \ldots \ \Gamma_n \vdash_{\mathrm{Dn}} U_n : P_n}{\Gamma_1, \ldots, \Gamma_n, x : P_1 \to P_2 \to \ldots \to x_k \vdash_{\mathrm{Dn}} (x)U_1 \ldots U_n : x_k}$$

   où x_k est une nouvelle variable de type associée à l'occurrence de tête de la variable x dans $(x)U_1 \ldots U_n$.

2. **Abstraction**

$$\frac{\Gamma, x : X_1, \ldots, x : X_n \vdash_{\mathrm{Dn}} T : P}{\Gamma \vdash_{\mathrm{Dn}} \lambda x T : (\bigcap_{1 \leq i \leq n} X_i) \to P}$$

Dans le cas où $n = 0$ on convient que $\bigcap_{1 \leq i \leq 0} X_i = \Omega$ soit la règle :

$$\frac{\Gamma \vdash_{\mathrm{Dn}} T : P}{\Gamma \vdash_{\mathrm{Dn}} \lambda x T : \Omega \to P}$$

REMARQUE : **Dn** type tous les termes normaux, et seulement ceux là. Si on se donne un terme normal alors sa démonstration de typage dans **Dn** se déduit par une simple induction. A chaque occurrence d'une variable x dans le terme T est associée une unique occurrence d'une variable de type x_k.

Définition 3.2.2 *On appellera* type principal *un séquent de la forme : $S = X_1, \ldots, X_n \vdash_{\mathrm{Dn}} P$ tel qu'il existe des variables $x_1, \ldots, x_n$, un terme T et une dérivation de $x_1 : X_1, \ldots, x_n : X_n \vdash_{\mathrm{Dn}} T : P$. On dira alors que S est un type principal de T.*

REMARQUE : tous les types de **D** ne sont pas des types principaux. On a pas de caractérisation des types principaux autre que : "ce sont les conclusions des dérivations de **Dn**". On peut donner des conditions nécessaires pour qu'une séquence de types soit le type principal d'un terme normal ; en fait il y a deux sortes de types dans **Dn** : les types *négatifs* sont les types des occurrences de variables ; ce sont eux que l'on trouve dans les parties gauches des séquents. Ce sont soit des variables de type, soit des types flèche de la forme $P \to X$ où P est un type *positif* et X est un type négatif. Les types positifs sont ceux que l'on trouve à droite des séquents ; ce sont soit des variables de type, soit des types flèche de la forme : $(\bigcap_{1 \leq i \leq n} X_i) \to P$ où les X_i sont des types négatifs et P est un type positif. De plus si $x_1 : X_1, \ldots, x_n : X_n \vdash_{\mathrm{Dn}} T : P$ est un typage principal de T alors les variables de type ont exactement deux occurrences parmi les X_i et P, une en position positive et l'autre en position négative. On verra plus loin une condition suffisante pour qu'une séquence de types soit un type principal.
EXEMPLE : voici une dérivation du typage principal de l'entier de Church 2 :

$$\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{}{y : y \vdash_{\mathrm{Dn}} y : y}}{y : y, g : y \to g_2 \vdash_{\mathrm{Dn}} (g)y : g_2}}{y : y, g : y \to g_2, g : g_2 \to g_1 \vdash_{\mathrm{Dn}} (g)(g)y : g_1}}{g : y \to g_2, g : g_2 \to g_1 \vdash_{\mathrm{Dn}} \lambda y(g)(g)y : y \to g_1}}{\vdash_{\mathrm{Dn}} \lambda g \lambda y(g)(g)y : (y \to g_2 \cap g_2 \to g_1) \to (y \to g_1)}$$

Proposition 3.2.2 *Soit T un terme normal, $\Gamma \vdash_{\mathrm{Dn}} P$ un type principal de T. Alors il existe une dérivation dans* **D** *de : $\Gamma \vdash_{\mathrm{D}} T : P$.*

REMARQUE : donc **Dn** est un sous-système de **D**.
Preuve : la difficulté vient de la gestion différente des contextes dans les deux systèmes. Remarquons que si on a une dérivation de $\Gamma \vdash_{\mathrm{D}} T : P$ alors on a une dérivation de $\Gamma, x_1 : X_1, \ldots, x_n : X_n \vdash_{\mathrm{D}} T : P$. On raisonne par induction sur T :

1. Si $T \equiv (x)U_1 \ldots U_n$, on a une dérivation de $\Gamma_1, \ldots, \Gamma_n, x : P_1 \to \ldots \to P_n \to x \vdash_{\mathrm{Dn}} (x)U_1 \ldots U_n : x$, alors on vient d'utiliser la règle d'application à une variable et on a des dérivations de :

$\Gamma_i \vdash_{\mathrm{Dn}} U_i : P_i$. Par induction on a donc des dérivations dans $\mathbf{D}$ de $\Gamma_i \vdash_{\mathrm{D}} U_i : P_i$ donc de $\Delta \vdash_{\mathrm{D}} U_i : P_i$ où $\Delta = \Gamma_1, \ldots, \Gamma_n, x : P_1 \to \ldots \to P_n \to x$. On fabrique alors la séquence suivante :

$$\dfrac{\dfrac{\dfrac{}{\Delta \vdash_{\mathrm{D}} x : P_1 \to \ldots \to P_n \to x} \quad \begin{matrix}\vdots \\ \Delta \vdash_{\mathrm{D}} U_1 : P_1\end{matrix}}{\Delta \vdash_{\mathrm{D}} (x)U_1 : P_2 \to \ldots \to P_n \to x}}{\begin{matrix}\vdots \\ \Delta \vdash_{\mathrm{D}} (x)U_1 \ldots U_{n-1} : P_n \to x\end{matrix} \qquad \begin{matrix}\vdots \\ \Delta \vdash_{\mathrm{D}} U_n : P_n\end{matrix}}$$

$$\dfrac{\Delta \vdash_{\mathrm{D}} (x)U_1 \ldots U_{n-1} : P_n \to x \qquad \Delta \vdash_{\mathrm{D}} U_n : P_n}{\Delta \vdash_{\mathrm{D}} (x)U_1 \ldots U_n : x}$$

où $\Delta = \Gamma, x : P_1 \to \ldots \to P_n \to x$. On obtient ainsi une dérivation du même séquent dans $\mathbf{D}$.

2. Si $T \equiv \lambda x U$, on a une dérivation de $\Gamma \vdash_{\mathrm{Dn}} \lambda x T : (\bigcap_{1 \leq i \leq n} X_i) \to P$; c'est donc que l'on vient d'utiliser une règle d'abstraction donc que l'on a une dérivation de : $\Gamma, x : X_1, \ldots, x : X_n \vdash_{\mathrm{Dn}} T : P$. Par induction on a une dérivation dans $\mathbf{D}$ de $\Gamma, x : X_1, \ldots, x : X_n \vdash_{\mathrm{D}} T : P$. Remarquons que toutes les règles d'une dérivation de $\mathbf{D}$ se font avec le même contexte de variables, soit dans ce cas particulier le contexte : $\Gamma, x : X_1, \ldots, x : X_n$. Remplaçons maintenant les règles axiome $\Gamma, x : X_1, \ldots, x : X_n \vdash_{\mathrm{D}} x : X_i$ par la séquence suivante :

$$\dfrac{\dfrac{}{\Gamma, x : \bigcap_{1 \leq i \leq n} X_i \vdash_{\mathrm{D}} x : \bigcap_{1 \leq i \leq n} X_i}}{\Gamma, x : \bigcap_{1 \leq i \leq n} X_i \vdash_{\mathrm{D}} x : X_i}$$

On obtient bien une dérivation de $\mathbf{D}$ car les parties gauches des séquents conclusion avant et après la transformation sont les mêmes, et seules ces parties sont les paramètres dont dépendent les règles de $\mathbf{D}$. Donc on a une dérivation de $\Gamma, x : \bigcap_{1 \leq i \leq n} X_i \vdash_{\mathrm{D}} T : P$ et on conclut en appliquant la règle d'abstraction de $\mathbf{D}$ ce qui nous donne une dérivation de : $\Gamma \vdash_{\mathrm{D}} \lambda x T : (\bigcap_{1 \leq i \leq n} X_i) \to P$.
•

Corollaire 3.2.3 *Tout terme normal est typable dans* $\mathbf{D}$*. De plus tout terme normalisable a un type principal dans* $\mathbf{D}\Omega$*, celui de sa forme normale.*

Théorème 3.2.4 *Soit* $\Gamma \vdash_{\mathrm{Dn}} T : P$ *un typage principal du terme normal* T*. Soit* U *tel que* $\Gamma \vdash_{\mathrm{D}\Omega} U : P$ *; alors* U *a une forme normale* U_0 *qui est un êta-réduit de* T*.*

Corollaire 3.2.5 *Si* S *est un type principal, alors il existe un unique terme normal* T *dont* S *est le type principal.*

Donc il est équivalent de trouver un type principal de la forme normale d'un terme ou de trouver sa forme normale, à êta-réduction près. Le problème du calcul des formes normales peut être transposé en celui du calcul des types principaux des termes. Une solution évidente à ce problème consiste à réduire le terme, puis trouver un typage principal de sa forme normale. On va voir que l'on peut procéder autrement, en définissant un calcul sur les types.

Deuxième partie

Réseaux

Les réseaux de preuve furent introduits par J.Y. Girard dans [Gir87] comme une syntaxe graphique des démonstrations en logique linéaire. En plus d'une théorie comprenant en particulier un critère caractérisant les réseaux de preuve parmi toutes les structures graphiques (on peut voir aussi [DaRe87] et [FlRe88] à ce sujet), on trouvait dans le même papier une traduction de la logique intuitionniste dans la logique linéaire (voir aussi [Ret87]). Danos a ensuite montré dans sa thèse ([Dan90]) comment définir une syntaxe dérivée des réseaux de preuve permettant de coder les lambda-termes purs, les *réseaux.* La différence majeure entre les réseaux de Danos et ceux qui suivent réside dans le traitement du lien **pourquoi pas**. A l'instar des réseaux de preuve, les réseaux de Danos possèdent trois instances de ce lien : l'affaiblissement d'arité 0, la dérèliction d'arité 1 et la contraction d'arité 2. Ici on introduit un lien **pourquoi pas** d'arité quelconque qui unifie les trois derniers.
Les réseaux sont, à la différence des réseaux de preuve, des objets non typés. C'est pourquoi la logique linéaire n'apparait que par allusion dans ce qui suit, à travers la terminologie et la définition du critère de correction qui est presque identique à celle de Girard.
On fait une présentation des réseaux basée sur une idée d'Eric Duquesne qui a montré que les réseaux normaux sont une écriture graphique des types principaux du système **D**, donc que la réduction des réseaux est équivalente à la recherche de type principaux telle que décrite par les règles de résolution d'équations de **De** ([Duq90]). Cela permet d'établir précisément le lien entre la beta-réduction et la réduction des réseaux.
On commence par proposer un système associant à un terme des équations entre types de **D** dont la résolution délivre un typage principal de la forme normale du terme. Ce système, que l'on appellera **De**, ainsi que le calcul associé est similaire à l'algorithme de recherche de P.T.S. décrit dans [Ron90]. On donnera ensuite une définition des réseaux et on montrera que ceux-ci sont isomorphes à des typages dans **De** et que leurs réductions sont isomorphes au calcul de **De**. En utilisant les outils connus sur les réseaux on verra alors que l'on peut caractériser les types principaux et trouver le terme normal dont S est le type principal.

Chapitre 1

Le système De

1.1 Définition

On se donne un ensemble dénombrable d'étiquettes $\mathcal{B} = \{b_1, \ldots, b_n, \ldots\}$. On dénotera par les lettres α, β ou γ les suites finies d'étiquettes deux à deux distinctes.
Les formules sont divisées en deux groupes, les formules *positives* et les formules *négatives*. On les définit inductivement :

1. si $P_1, \ldots, P_n$ sont des formules positives, x une variable de type et $b_1, \ldots, b_n$ des étiquettes alors

$$P_1^{b_1} \to \ldots \to P_n^{b_n} \to x$$

   est une formule négative. Dans la suite on les appellera *formules de genre* X ;

2. si $X_1, \ldots, X_n$ sont des formules négatives de genre X et $\beta_1, \ldots, \beta_n$ des suites d'étiquettes alors

$$\bigcap_{1 \leq i \leq n} X\beta_i$$

   est une formule négative *de genre* N. En particulier quand $n = 0$, Ω est une formule négative ;

3. si $N_1, \ldots, N_n$ sont des formules négatives de genre N et x est une variable de type alors

$$N_1 \to \ldots \to N_n \to x$$

   est une formule positive. On les appellera *formules de genre* P.

Remarque : il y a deux sortes de formules dont le connecteur principal est une flèche : les formules positives de genre $N \to P$ et les négatives de genre $P \to X$. Une variable de type est indifféremment une formule positive de genre P ou négative de genre X. Donc les genres X, et P sont d'intersection non vide.
Le système **De** dérive des séquents de la forme :

$$x_1 : X_1^{\beta_1}, \ldots, x_n : X_n^{\beta_n} \vdash_{\mathrm{De}} T : P, (N_1 = P_1^{b_1})^{\gamma_1}, \ldots, (N_p = P_p^{b_p})^{\gamma_p}$$

Remarque : comme on le verra, tous les séquents de cette forme ne sont pas dérivables dans **De**, exactement pour la même raison que toutes les formules de **D** ne sont pas des types principaux.

On considérera toujours les listes d'équations comme des ensembles non ordonnés d'équations. On appellera *étiquetage de* S ou de F l'ensemble des étiquettes figurant dans un séquent S ou dans une formule F.
On se donne les notations suivantes : si $N = X^{\beta}$ est une formule portant les étiquettes de β alors N^b est la formule $X^{b\beta}$ portant les étiquettes b puis celles de β. On dira alors que X *porte* b et les étiquettes de β, et que b *gouverne* toutes les sous-formules de X, ainsi que les éléments de l'étiquetage de X^{β}. De même, si $\mathcal{E} = \left((N_1 = P_1^{b_1})^{\gamma_1}, \ldots, (N_p = P_p^{b_p})^{\gamma_p}\right)$ est une suite d'équations alors $\mathcal{E}^b$ est la suite d'équations : $(N_1 = P_1^{b_1})^{b\gamma_1}, \ldots, (N_p = P_p^{b_p})^{b\gamma_p}$ et si $\Gamma = x_1 : X_1^{\beta_1}, \ldots, x_n : X_n^{\beta_n}$ est un contexte de variables alors Γ^b est le contexte : $x_1 : X_1^{b\beta_1}, \ldots, x_n : X_n^{b\beta_n}$. Comme ci-dessus, on dira que b gouverne les N_i, les P_i, les b_i et les étiquettes des γ_i, et dans le cas des contextes, les X_i et les β_i.
Le système **De** est défini par les règles :

1. **Application simple**

$$\frac{\Gamma_1 \vdash_{\mathrm{De}} U_1 : P_1, \mathcal{E}_1 \ \ldots\ \Gamma_n \vdash_{\mathrm{De}} U_n : P_n, \mathcal{E}_n}{\Gamma_1^{b_1}, \ldots \Gamma_n^{b_n}, x : P_1^{b_1} \to \ldots P_n^{b_n} \to x_k \vdash_{\mathrm{De}} (x)U_1 \ldots U_n : x_k, \mathcal{E}_1^{b_1}, \ldots, \mathcal{E}_n^{b_n}}$$

   où les étiquetages des n séquents prémisses sont disjoints, les b_i sont de nouvelles étiquettes deux à deux distinctes et x_k est une nouvelle variable de type. Cette règle correspond à n applications de la règle de **flèche gauche** du calcul des séquents.

2. **Abstraction**

$$\frac{\Gamma, x : X_1^{\beta_1}, \ldots, x : X_n^{\beta_n} \vdash_{\mathrm{De}} T : P, \mathcal{E}}{\Gamma \vdash_{\mathrm{De}} \lambda x T : \bigcap_{1 \le i \le n} X_i^{\beta_i} \to P, \mathcal{E}}$$

   où x n'occure pas dans Γ. Cette règle correspond à n applications de la règle de **contraction gauche** suivies d'une règle de **flèche droite** en calcul des séquents.

3. **Application radicale**

$$\frac{\Gamma, x : X_1^{\beta_1}, \ldots, x : X_n^{\beta_n} \vdash_{\mathrm{De}} T : y, \mathcal{E} \quad \Gamma' \vdash_{\mathrm{De}} U : P, \mathcal{E}'}{\Gamma, \Gamma'^b \vdash_{\mathrm{De}} (\lambda x T)U : y, \mathcal{E}, \mathcal{E}'^b, \bigcap_{1 \le i \le n} X_i^{\beta_i} = P^b}$$

   où les étiquetages des deux séquents prémisses sont disjoints, b est une nouvelle étiquette, y une variable de type et la variable x n'occure pas dans Γ. Remarquons que la condition imposant que T soit de type y se paraphrase en "T n'a pas de lambda de gauche". Cette règle correspond à n **contractions gauches** suivies d'une règle de **coupure**. Ceci explique l'introduction de l'équation ; en effet pour que la coupure soit valide il faut qu'il y ait cette égalité entre les types.

Les systèmes **De** et **Dn** sont définis "à la calcul des séquents". Il serait en effet beaucoup plus compliqué de les présenter comme des formalismes de type déduction naturelle à cause du fait que l'on veut introduire une nouvelle équation pour chaque redex premier du terme ce qui est une notion difficile à manipuler. Dans un formalisme comme le calcul des séquents, il est par contre aisé de se restreindre, comme on le fait ici, aux termes canoniques.

Remarque : un terme a un unique type au nom des variables de type près. En effet, le terme détermine la suite de règles de dérivations ; donc le seul choix reste dans le nom des variables de type introduites.

Si on oublie les étiquettes et la dernière règle, on a exactement le système **Dn**. Comme pour **Dn**, à chaque occurrence d'une variable x dans le terme T sont associées deux occurrences d'une variable de type x_k. Toutes les formules de type P apparaissant dans le contexte ou dans les équations i.e., toutes sauf celle qui type le terme, portent une unique étiquette et chaque étiquette apparaissant est portée par une unique formule de genre P (et éventuellement par plusieurs formules de genre X). Les équations ont toujours la forme : $(N = P^b)^\beta$. Donc l'ensemble des formules et équations portant l'étiquette b est le type dans **De**d'un sous-terme.
Exemple : voici une dérivation de $\mathcal{Q} = (\lambda x(x)x)\lambda g \lambda y(g)(g)y$:

$$\dfrac{\dfrac{\dfrac{}{x : x_2 \vdash_{\mathrm{De}} x : x_2}}{x : x_2^{b_1}, x : x_2^{b_1} \to x_1 \vdash_{\mathrm{De}} (x)x : x_1} \quad \dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{y : y \vdash_{\mathrm{De}} y : y}}{g : y^{b_2} \to g_2, y : y^{b_2} \vdash_{\mathrm{De}} (g)y : g_2}}{g : g_2^{b_3} \to g_1, g : (y^{b_2} \to g_2)^{b_3}, y^{b_3 b_2} \vdash_{\mathrm{De}} (g)(g)y : g_1}}{g : g_2^{b_3} \to g_1, g : (y^{b_2} \to g_2)^{b_3} \vdash_{\mathrm{De}} \lambda y(g)(g)y : y^{b_3 b_2} \to g_1}}{\vdash_{\mathrm{De}} \lambda g \lambda y(g)(g)y : (g_2^{b_3} \to g_1 \cap (y^{b_2} \to g_2)^{b_3}) \to (y^{b_3 b_2} \to g_1)}}{\vdash_{\mathrm{De}} (\lambda x(x)x)\lambda g \lambda y(g)(g)y : x_1, x_2^{b_1} \cap x_2^{b_1} \to x_1 = ((g_2^{b_3} \to g_1 \cap (y^{b_2} \to g_2)^{b_3}) \to (y^{b_3 b_2} \to g_1))^{b_4}}$$

1.2 De et l'équivalence opérationnelle

Définition 1.2.1 *On appellera* type principal *un séquent de la forme :* $S = X_1^{\beta_1}, \ldots, X_n^{\beta_n} \vdash_{\mathrm{De}} P, N_1 = P_1^{b_1}, \ldots, N_p = P_p^{b_p}$ *tel qu'il existe des variables* x_i*, un terme* T *et une dérivation dans* **De** *de :* $x_1 : X_1^{\beta_1}, \ldots, x_n : X_n^{\beta_n} \vdash_{\mathrm{De}} T : P, N_1 = P_1^{b_1}, \ldots, N_p = P_p^{b_p}$*. On dira que* P *est le type de* T*, que les* X_i *sont les types des variables libres de* T *et que les équations sont les types des redex de* T*.*

Proposition 1.2.1 *Soit* $S = X_1^{\beta_1}, \ldots, X_n^{\beta_n} \vdash_{\mathrm{De}} P, N_1 = P_1^{b_1}, \ldots, N_p = P_p^{b_p}$ *un type principal d'un terme* T *; alors pour tout terme canonique* T'*,* T' *est typable de type* S *dans* **De** *ssi* $T' =_{\mathrm{op}} T$ *i.e., si* $T' = T_\sigma$ *pour une permutation* σ *admissible pour* T*.*

Preuve : on raisonne par induction sur T :

1. Si $T \equiv (y)U_1 \ldots U_k$ alors T est obtenu par une règle d'application simple de la forme :

$$\dfrac{\Gamma_1 \vdash_{\mathrm{De}} U_1 : P_1, \mathcal{E}_1 \ \ldots \ \Gamma_k \vdash_{\mathrm{De}} U_k : P_k, \mathcal{E}_k}{\Gamma_1{}_1^b, \ldots, \Gamma_k{}^{b_k}, y : P_1{}^{b_1} \ldots \to P_k{}^{b_k} \to y \vdash_{\mathrm{De}} (y)U_1 \ldots U_k : y, \mathcal{E}_1{}^{b_1}, \ldots, \mathcal{E}_k{}^{b_k}}$$

Donc S est un type principal dont aucune équation n'est libre d'étiquette ; tout terme T' de type S ne peut donc être obtenu que par la même règle et est donc de la forme $(y)U'_1 \ldots U'_k$ où les U'_i ont

les mêmes types que les U_i. Par induction les U_i et les U'_i sont donc équivalents opérationnellement, soit $T =_{\mathrm{op}} T'$ car l'équivalence opérationnelle passe au contexte.
Réciproquement, si $T' =_{\mathrm{op}} T$ alors $T' \equiv (y)U'_1 \ldots U'_k$ où les U'_i sont équivalents opérationnellement aux U_i, donc par induction sont de même type. Par une règle d'application simple on obtient donc que T' est de type S.

2. Si $T \equiv \lambda y U$ alors T est obtenu par une règle d'abstraction de la forme :

$$\frac{\Gamma, y : N_1, \ldots, y : N_k \vdash_{\mathrm{De}} U : P', \mathcal{E}}{\Gamma \vdash_{\mathrm{De}} \lambda y U : \bigcap_{1 \leq i \leq k} N_i \to P', \mathcal{E}}$$

Mais alors P est un type flèche et comme ce genre de type ne peut être obtenu que par des règles d'abstraction, si T' a le type S alors il est dérivé par la même règle, soit :

$$\frac{\Gamma, y : N_1, \ldots, y : N_k \vdash_{\mathrm{De}} U' : P', \mathcal{E}}{\Gamma \vdash_{\mathrm{De}} \lambda y U' : \bigcap_{1 \leq i \leq k} N_i \to P', \mathcal{E}}$$

Par hypothèse d'induction on a $U =_{\mathrm{op}} U'$ et donc on obtient $T =_{\mathrm{op}} T'$.
Réciproquement si $T' =_{\mathrm{op}} T$ alors T' est de la forme $\lambda x U'$ avec $U' =_{\mathrm{op}} U$. Par induction U' a le même type principal que U et donc par une régle d'abstraction on obtient que S est un type principal pour T'.

3. Si $T \equiv (\lambda x_1 \ldots (\lambda x_k U) U_k \ldots) U_1$ où U est en forme normale de tête alors T est obtenu par exactement k applications successives de règles d'application radicale. Donc P est une variable et S compte exactement k équations ne portant pas d'étiquettes. Mais le seul moyen d'obtenir un tel type est par k applications de la règle d'application radicale portant sur les mêmes prémisses dans un ordre différent, donc $T' \equiv (\lambda x_{\sigma(1)} \ldots (\lambda x_{\sigma(n)} U') U'_{\sigma(n)} \ldots) U'_{\sigma(1)}$ où les U et les U' ont été dérivés de même type principal, donc par hypothèse d'induction sont équivalents opérationnellement. Donc $T' =_{\mathrm{op}} T$ ssi σ est admissible. Supposons qu'il y a une inversion (i, j) de σ avec $x_{\sigma(j)} \in U_{\sigma(i)}$. Comme $U'_{\sigma(i)} =_{\mathrm{op}} U_{\sigma(i)}$ on a aussi $x_{\sigma(j)} \in U'_{\sigma(i)}$ donc la variable $x_{\sigma(j)}$ occure dans le contexte du type de $U'_{\sigma(i)}$. Mais dans la dérivation de T' la règle introduisant le redex $(\lambda x_{\sigma(j)}, U'_{\sigma(j)})$ est avant celle introduisant le redex $(\lambda_{\sigma(i)}, U'_{\sigma(i)})$ donc comme la variable $x_{\sigma(j)}$ est abstraite du contexte uniquement lors de la première règle, la seconde réintroduit dans le contexte des occurrence de $x_{\sigma(j)}$ qui ne disparaitront plus. Donc le type de T' ne peut être le même que celui de T puisqu'il a un contexte plus grand.
Réciproquement, si $T' =_{\mathrm{op}} T$ alors il faut voir que S est un type principal pour T'. D'après le lemme 2.2.3 on peut se restreindre au cas où T' est obtenu par une transposition contigüe admissible pour T. Supposons donc que : $T \equiv C[(\lambda x(\lambda y U) W) V]$ et $T' \equiv C[(\lambda y(\lambda x U) V) W]$ avec $x \notin W$. La dérivation de typage de T est donc de la forme :

$$\dfrac{\dfrac{\begin{array}{c}\vdots\\ \Gamma, x : X_1^{\beta_1}, \ldots, x : X_p^{\beta_p}, y : Y_1^{\gamma_1}, \ldots, y : Y_q^{\gamma_q} \vdash_{\mathrm{De}} U : u, \mathcal{E}\end{array} \quad \begin{array}{c}\vdots\\ \Gamma'' \vdash_{\mathrm{De}} W : P''\end{array}}{\Gamma, {\Gamma''}^{b''}, x : X_1^{\beta_1}, \ldots, x : X_p^{\beta_p} \vdash_{\mathrm{De}} (\lambda y U) W : u, \mathcal{E}, \bigcap_{1 \leq i \leq q} Y_i^{\gamma_i} = {P''}^{b''}, {\mathcal{E}''}^{b''}} \quad \begin{array}{c}\vdots\\ \Gamma' \vdash_{\mathrm{De}} V : P', \mathcal{E}'\end{array}}{\begin{array}{c}\Gamma, {\Gamma'}^{b'}, {\Gamma''}^{b''} \vdash_{\mathrm{De}} (\lambda x(\lambda y U) W) V : u, \mathcal{E}, \bigcap_{1 \leq i \leq p} X_i^{\beta_i} = {P'}^{b'}, {\mathcal{E}'}^{b'}, \bigcap_{1 \leq i \leq q} Y_i^{\gamma_i} = {P''}^{b''}, {\mathcal{E}''}^{b''}\\ \vdots\\ \vdash_{\mathrm{De}} T : S\end{array}}$$

La condition $x \notin W$ est équivalente à “x n'occure pas dans Γ''”, donc les deux règles d'application radicale peuvent être commutées ce qui donne :

$$\dfrac{\dfrac{\dfrac{\vdots}{\Gamma, x : X_1^{\beta_1}, \ldots, x : X_p^{\beta_p}, y : Y_1^{\gamma_1}, \ldots, y : Y_q^{\gamma_q} \vdash_{\text{De}} U : u, \mathcal{E}} \quad \dfrac{\vdots}{\Gamma' \vdash_{\text{De}} V : P'}}{\Gamma, {\Gamma''}^{b''}, y : Y_1^{\gamma_1}, \ldots, y : Y_q^{\gamma_q} \vdash_{\text{De}} (\lambda x U)V : u, \mathcal{E}, \bigcap_{1 \leq i \leq p} X_i^{\beta_i} = {P'}^{b'}, {\mathcal{E}'}^{b'}} \quad \dfrac{\vdots}{\Gamma'' \vdash_{\text{De}} W : P'', \mathcal{E}'}}{\dfrac{\Gamma, {\Gamma'}^{b'}, {\Gamma''}^{b''} \vdash_{\text{De}} (\lambda y(\lambda x U)V)W : u, \mathcal{E}, \bigcap_{1 \leq i \leq p} X_i^{\beta_i} = {P'}^{b'}, {\mathcal{E}'}^{b'}, \bigcap_{1 \leq i \leq q} Y_i^{\gamma_i} = {P''}^{b''}, {\mathcal{E}''}^{b''}}{\dfrac{\vdots}{\vdash_{\text{De}} T' : S}}}$$

et T' a donc bien le type S •

Donc un type principal $S = X_1, \ldots, X_n \vdash_{\text{De}} P, \mathcal{E}$ caractérise une unique classe d'équivalence opérationnelle, de même que pour **Dn** il caractérisait un unique terme normal.

1.3 Résolution des équations

On ne considérera plus, dans ce qui suit, de typage, mais des types principaux. Aussi le symbole Γ représentera non plus un contexte de variables typées mais un contexte de formules de genre X^β.
Soit S un type principal, B l'étiquetage de S et V l'ensemble des variables de types apparaissant dans S. On se donne une fonction de renommage qui est une injection de $(B \cup V) \times \mathbb{N}$ dans $\mathcal{B} \cup \mathbf{V} \setminus (B \cup V)$ qui associe à chaque variable de type x et chaque entier j une nouvelle variable de type $x^{(j)}$ et à chaque étiquette B et chaque entier j une nouvelle étiquette $b^{(j)}$. On étend par induction la fonction de renommage à toutes les formules apparaissant dans S. Donc $F^{(j)}$ est une formule identique au nom des variables de type et des étiquettes près à F, et $F^{(i)}$ et $F^{(j)}$ sont identiques ssi $i = j$.
Soit b une étiquette : on note b_N l'ensemble des formules de genre N apparaissant dans S qui portent l'étiquette b. Donc une formule de b_N est de la forme $X^{\beta_1 b \beta_2}$. On note b_E l'ensemble des équations de S qui portent b soit $E \in b_E$ ssi E est de la forme : $(X^\beta = P^c)^{\beta_1 b \beta_2}$. On définit $S^{b \leftarrow \gamma_1 \cdots \gamma_n}$ comme étant S dans lequel :

1. chaque formule $X^{\beta_1 b \beta_2}$ de b_N est remplacée par la formule :

$$\bigcap_{1 \leq i \leq n} (X^{(i)})^{\beta_1 \gamma_i \beta_2^{(i)}}$$

2. chaque équation $(X^\beta = P^c)^{\beta_1 b \beta_2}$ de b_E est remplacée par la suite d'équations :

$$\left((X^\beta)^{(i)} = (P^c)^{(i)}\right)^{\beta_1 \gamma_i \beta_2^{(i)}}$$

Remarque : $S^{b \leftarrow \gamma_1 \cdots \gamma_n}$ est bien définie car par définition des règles de **De**, si N est dans b_N (resp. E est dans b_E) alors aucune sous-formule propre de N (resp. de E) n'est dans b_N (resp. n'est dans b_E).

On est maintenant en mesure de donner les règles de résolution d'équations ; elles ont la forme :

$$P, \mathcal{E}, E \rightarrow_{\text{De}} P', \mathcal{E}'$$

où E est l'équation que l'on réécrit.

- **Variable**

$$\begin{array}{rcl} \Gamma \vdash_{\text{De}} P, \mathcal{E}, (x = P')^{\beta} & \rightarrow_{\text{De}} & (P, \mathcal{E})[P'/x] \\ \Gamma \vdash_{\text{De}} P, \mathcal{E}, (X = x)^{\beta} & \rightarrow_{\text{De}} & (X, \mathcal{E})[X/x] \end{array}$$

  où la notation $\mathcal{X}[F/x]$ signifie : "$\mathcal{X}$ dans lequel on a substitué la variable de type x par la formule F" (il n'y a qu'une occurrence de x car les variables de type n'apparaissent que deux fois chacune). Remarquons que l'on impose dans le premier cas à la formule P de ne pas porter d'étiquette, dans le second que la formule soit de genre X. De telles équations ne font donc pas partie d'un type principal de terme, mais comme on va le voir tout de suite, elles apparaissent pendant la résolution.

- **Flèche**

$$\Gamma \vdash_{\text{De}} P, \mathcal{E}, (P_1 \rightarrow N_1 = N_2 \rightarrow P_2)^{\beta} \quad \rightarrow_{\text{De}} \quad \Gamma \vdash_{\text{De}} P, \mathcal{E}, (N_1 = P_2)^{\beta}, (N_2 = P_1)^{\beta}$$

  Ici aussi on impose des conditions sur les formules égalées. Encore une fois, l'équation E n'est pas une équation provenant d'un type principal, mais de la réduction d'un type principal.

- **Duplication**

$$\Gamma \vdash_{\text{De}} P, \mathcal{E}, \Big(\bigcap_{1 \leq i \leq n} X_i^{\beta_i} = P'^{b} \Big)^{\beta} \quad \rightarrow_{\text{De}} \quad (\Gamma \vdash_{\text{De}} P, \mathcal{E})^{b \leftarrow \beta_1, \ldots, \beta_n}, \Big(\big(X_i = P'^{(i)}\big)^{\beta\beta_i} \Big)_{1 \leq i \leq n}$$

  Ici par contre, l'équation E est bien une équation de type principal. C'est la règle de duplication qui fait apparaitre des équations non standards, à savoir les n copies de E.

On trouvera dans l'annexe A un exemple de résolution d'équation. On donne dans le prochain chapitre la preuve que les réductions des types principaux par les règles ci-dessus donnent encore des types principaux et que l'éventuelle solution est bien le type principal de la forme normale. On verra aussi que cette solution n'existe que si le terme est normalisable.

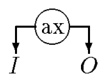
Chapitre 2

Structure de réseaux

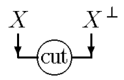
2.1 Définition

Les *formules* sont : I, O, $?I$ et $!O$. Les formules I et $?I$ seront dites *négatives* et les formules O et $!O$ *positives*. On note $(X^{\perp})$ la formule *duale* de X i.e., I et O sont duales de même que $?I$ et $!O$. Remarquons que pour toute formule on a $(X^{\perp})^{\perp} = X$. Les formules servent à étiqueter les arêtes ; une arête α étiquetée par une formule X (ou *portant la formule* X) sera dite *de genre* X.
Les *liens* sont des sommets dont les arêtes incidentes sont appelées *prémisses* et les arêtes émergentes sont appelées *conclusions*. Si α est conclusion (resp. prémisse) d'un lien l, on dira que l est le lien *prémisse* (resp. *conclusion*) de α. Il y a six types de liens, chaque type étant caractérisé par le nombre et le genre des prémisses et des conclusions du lien :

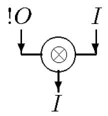
1. **lien axiome** : ce lien n'a pas de prémisses mais deux conclusions portant respectivement les formules I et O ; on le représente comme suit :



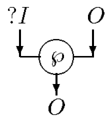
2. **lien coupure**, noté **cut** : deux prémisses X et $X^{\perp}$ ($X = I$ ou $?I$), pas de conclusion ; on le représente ainsi :



3. **lien tenseur**, noté $\otimes$: deux prémisses : la prémisse *gauche* de genre $!O$ et la prémisse *droite* de genre I ; une conclusion I ;



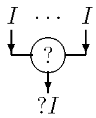
4. **lien par**, noté $\wp$: deux prémisses : la gauche de genre $?I$ et la droite de genre O ; une conclusion O :



Ces quatre liens sont appelés les liens *multiplicatifs*. Voici maintenant les liens *exponentiels* :

5. **lien bien sur**, noté ! : une prémisse O et une conclusion $!O$:

6. **lien pourquoi pas**, noté ? : n prémisses I et une conclusion $?I$. Dans le cas où $n = 0$ on l'appellera lien **affaiblissement** :



On représentera toujours les liens comme ci-dessus ; en particulier la prémisse I (resp. O) d'un lien **tenseur** (resp. **par**) sera toujours à droite et la prémisse $!O$ (resp. $?I$) d'un lien **tenseur** (resp. **par**) sera toujours à gauche. Cette convention nous permet d'éviter d'annoter lourdement les dessins puisqu'elle lève les éventuelles ambiguités sur le genre des arêtes.
Une *structure de réseau* S est définie par :

1. un graphe orienté G, sans cycle orienté, dont les arêtes sont étiquetées par des formules et les sommets sont soit des liens, soit des *conclusions* de S i.e., des sommets à une prémisse et pas de conclusion.

2. la donnée pour chaque lien **bien sur** l de G d'un sous-graphe $b(l)$ de G appelé *boîte* contenant l i.e., $b(l)$ est un ensemble de liens de G contenant l. Les boîtes doivent vérifier les conditions suivantes :

   (a) si b est une boîte de S alors b n'a pas d'arête incidente i.e., il n'existe pas d'arête α de G qui soit prémisse d'un sommet dans b et conclusion d'un sommet non dans b. On appellera *portes* de b les arêtes émergentes de b ;

   (b) si b_1 et b_2 sont deux boîtes de S alors on a soit $b_1 \cap b_2 = \emptyset$, soit $b_1 \subset b_2$;

   (c) si $b(l)$ est la boîte associée au lien **bien sur** l alors $b(l)$ a une unique porte de genre $!O$ et c'est la conclusion de l ; on l'appelle la *porte principale de* $b(l)$. Les autres portes sont toutes de genre I et ce sont les *portes auxiliaires de* $b(l)$.

   (d) si α est porte auxiliaire d'une boîte alors α est soit prémisse d'un lien **pourquoi pas**, soit prémisse d'un sommet conclusion de S.

Remarque : une arête α de genre $!O$ est la porte principale de la boîte $b(l)$ où l est le lien **bien sur** prémisse de α. Par contre une arête α de genre I peut être porte auxiliaire de plusieurs boîtes $b_1, \ldots, b_n$ qui sont alors totalement ordonnées par inclusion.

On dira que X *ou* α *est conclusion de* S si S a un sommet conclusion dont l'arête prémisse α porte la formule X. Si α est une arête conclusion ou une prémisse d'un lien **cut** alors on dira que α est une arête *terminale*. Dans ce cas si l est le lien prémisse de α i.e., si α est conclusion du lien l, alors l sera dit terminal aussi.
On dira qu'on emprunte une arête *en descendant* quand on la suit le long de son orientation, et *en montant* si on la prend à l'envers. On dira que l'arête ou le lien x_1 est *au-dessus* de l'arête ou le lien x_2 ou que x_2 est *conclusion héréditaire* de x_1 ou que x_1 est *prémisse héréditaire* de x_2 s'il existe un chemin orienté dans G menant de x_1 à x_2.
On dira qu'une boîte ou un lien ou une arête est *externe* si elle n'est dans aucune autre. On appellera *intérieur* d'une boîte b, l'ensemble des liens de b et des arêtes dont les liens prémisse sont dans b. L'intérieur d'une boîte est donc à son tour une structure de réseau. On appellera *extérieur* de la boîte l'ensemble des liens qui ne sont pas dans b ainsi que les arêtes dont le lien conclusion n'est pas dans b. Remarquons que l'intersection entre l'intérieur et l'extérieur d'une boîte est exactement l'ensemble de ses portes.

Remarque : l'intérieur d'une boîte b est caractérisée par la donnée de ses portes, plus celle des liens **coupure** qu'elle contient. En effet, les portes et les prémisses des liens **coupure** sont les seules arêtes de b qui n'ont aucune conclusion héréditaires dans b.

Soit une arête α portant la formule I dans une structure R. Par définition elle ne peut être conclusion que d'un lien **axiome**, ou d'un lien **tenseur**. Dans ce dernier cas, la même remarque s'applique à la prémisse I du **tenseur**, etc..., jusqu'à arriver en un lien **axiome** a. On dira que α *est conclusion d'un arbre* **tenseur** *dont la prémisse de tête est* a. Chaque lien **tenseur** de l'arbre a aussi une prémisse $!O$ de boîte associée b_i. On dira que les b_i sont les *arguments de l'***axiome** a. On définit similairement pour une arête α portant la formule O un *arbre de* **par** *dont* α *est la conclusion*, les prémisses $?I$ des liens **par** étant les *lambda de tête de l'***axiome** *de tête* a. Chaque lien **axiome** est donc axiome de tête d'un arbre de **tenseur** du coté de sa conclusion I et d'un arbre de **par** du coté de sa conclusion O. Réciproquement, chaque arête de genre I ou O a un unique **axiome** de tête, à savoir celui de l'arbre **tenseur** ou **par** dont elle fait partie.
Soit S une structure de réseau ayant des conclusions $X_1, \ldots, X_n$. On dit que l'on *lie* les conclusions X_i par un lien l quand on ajoute à S un lien l dont les prémisses sont les n arêtes portant les X_i.
Il y a trois types de lien **coupure** suivant la nature des prémisses :

1. la coupure *axiome* est une coupure dont les prémisses portent les formules I et O et telles que l'une d'elle est conclusion d'un lien **axiome** (l'autre étant conclusion d'un lien multiplicatif) ;

2. la coupure *binaire* dont les prémisses portent les formules I et O et sont respectivement conclusion d'un lien **tenseur** et d'un lien **par** ;

3. la coupure *exponentielle* dont les prémisses portent les formules $?I$ et $!O$ et sont donc respectivement conclusion d'un lien **pourquoi pas** et d'un lien **bien sur**. Dans le cas où le lien **pourquoi pas** est un **affaiblissement** on parlera d'une coupure *affaiblissement*. Les coupures non-**affaiblissement** sont les *coupures strictes*.

Les coupures **axiome** et binaires sont appelées les coupures *multiplicatives*.

2.2 Types principaux et structures de réseaux

Ici l'on montre comment écrire un type principal de **De** sous forme graphique i.e., comment associer à un type principal $S = X_1^{\beta_1}, \ldots, X_n^{\beta_n} \vdash_{\mathrm{De}} P, (N_1 = P_1^{b_1})^{\gamma_1}, \ldots, (N_p = P_p^{b_p})^{\gamma_p}$ une structure de réseau $(S)^\circ$ ayant n conclusions I respectivement porte auxiliaire des boîtes des β_i, une conclusion O et p liens **coupure** exponentielle respectivement dans les boîtes des γ_i.
On associe à chaque sous-formule F de S un arbre ayant des arêtes *hypothèses* i.e., sans lien prémisse, associées aux occurrences des variables de type de S :

- Si $F = N_1 \to \cdots \to N_n \to x$ est de genre P alors soient $(N_1)^\circ, \ldots, (N_p)^\circ$ les arbres associés aux N_i. Comme on va le voir ci-dessous, les $(N_i)^\circ$ ont pour conclusion une arête $?I$. On construit alors un arbre de **par**, ayant comme prémisse O initiale une arête hypothèse que l'on associe à la variable de type x, et dont les prémisses $?I$ sont respectivement les conclusions des $(N_i)^\circ$ pour i variant de p, à 1 (c'est à dire que $(N_1)^\circ$ est en haut de l'arbre, et $(N_p)^\circ$ en bas).

  Si F est l'une des formules P_i qui apparaissent dans les équations de S, donc si F porte une étiquette b_i, on ajoute à la conclusion O de $(F)^\circ$ un lien **bien sur** associé à une boîte que l'on appellera b_i.

- Si $F = P_1^{b_1} \to \cdots \to P_p^{b_p} \to x$ est une formule de genre X alors par induction on a des arbres $(P_1)^\circ, \ldots, (P_p)^\circ$: on note $(P_i^{b_i})^\circ$ l'arbre $!(P_i)^\circ$ i.e., l'arbre $(P_i)^\circ$ à la conclusion O duquel on a ajouté un lien **bien sur**, puis on construit un arbre de **tenseur** dont la prémisse de tête est une arête hypothèse portant la formule I que l'on associe à la variable de type x, et dont les prémisses $!O$ sont les conclusions des $(P_i^{b_i})^\circ$ pour i variant de p à 1. On définit les boîtes b_i associées aux nouveaux liens **bien sur** comme contenant exactement les traductions des formules portant l'étiquette b_i.

- Si $F = X_1^{\beta_1} \cap \cdots \cap X_p^{\beta_p}$ est une formule de genre N, alors par induction on a des arbres $(X_i)^\circ$ de conclusion I, que l'on lie au moyen d'un lien **pourquoi pas**, obtenant ainsi un arbre de conclusion $?I$. De plus, chaque arbre $(X_i)^\circ$ est contenu dans les boîtes de la liste β_i, donc chaque conclusion I des $(P_i)^\circ$ est porte auxiliaire des β_i.

On obtient ainsi une forêt comptant :

- n arbres **tenseur** associés aux n formules du contexte de S ; on déclare chacune des conclusion porte auxiliaire respectivement des boîtes des β_i ;

- un arbre **par** correspondant à la formule P ;

- p couples d'arbres de conclusions $(?I, !O)$ correspondant aux équations. On ajoute alors p liens **cut** de prémisse chacune des conclusions des composantes de ces couples, chacun des liens étant déclaré appartenir aux boîtes respectivement des γ_i.

En vertu du fait que S est un type principal, à chaque variable de type x correspond exactement une arête hypothèse O et une arête hypothèse I associées à x que l'on lie alors au moyen d'un lien **axiome** de conclusions ces deux arêtes Si chaque arête appartient à une boîte b alors on déclare le nouveau lien **axiome** dans b aussi. Remarquons que l'on a pas l'assurance que les deux arêtes appartiennent exactement aux même boîtes. On va voir dans ce qui suit que c'est toujours le cas.

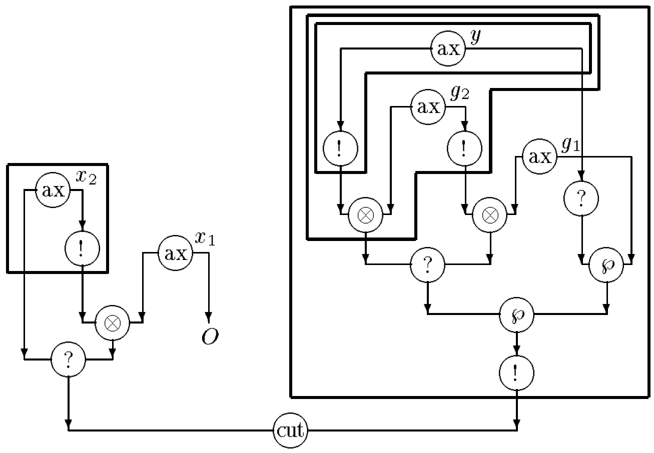
Exemple : on rappelle que :

$$\mathcal{Q} = (\lambda x(x)x)\lambda g\lambda y(g)(g)y$$

et que le type principal de $\mathcal{Q}$ est :

$$x_1, x_2^{b_1} \cap x_2^{b_1} \rightarrow x_1 = ((g_2^{b_3} \rightarrow g_1 \cap (y^{b_2} \rightarrow g_2)^{b_3}) \rightarrow (y^{b_3 b_2} \rightarrow g_1))^{b_4}$$

Voici le réseau $(\mathcal{Q})^\circ$ correspondant à $\mathcal{Q}$:



On a indiqué à côté de chaque lien **axiome**, l'occurrence de variable de Q auquel il correspond (en numérotant les occurrences de gauche à droite dans le terme). Le tableau sur la page suivante représente les correspondances entre le réseau et le terme.

le réseau n'a qu'une conclusion O	le terme n'a pas de variable libre
l'axiome x_1 n'est dans aucune boîte et sa conclusion O est conclusion du réseau	x_1 est l'occurrence de tête de Q et à ce titre ne peut être argument de personne
la conclusion O de l'axiome x_2 est (héréditairement) porte principale d'une boîte b_{g_2} contenant l'axiome x_2	x_2 est l'occurrence de tête du terme x_2 qui est argument d'une application
la conclusion I de x_2 est porte auxiliaire de b_{x_2}	x_2 est une occurrence libre dans le terme x_2
la porte principale de b_{x_2} est prémisse du même lien **tenseur** que la conclusion I de x_1	le terme x_2 est l'argument de x_1
les conclusions I de x_1 et x_2 sont (héréditairement) prémisse du même lien **pourquoi pas**	x_1 et x_2 sont les deux occurrences de la variable x
la conclusion de ce lien **pourquoi pas** est prémisse d'une coupure	λx définit un redex
la conclusion O de y est porte principale d'une boîte b_y contenant l'axiome y	y est l'occurrence de tête du terme y qui est argument d'une application
la conclusion I de y est porte auxiliaire de b_y	y est une occurrence libre du terme y
la porte principale de b_y est prémisse du même lien **tenseur** que la conclusion I de g_2	le terme y est l'argument de g_2
la conclusion O de g_2 est prémisse d'une boîte b_{g_2} contenant la boîte b_y, l'axiome g_2 et le lien **tenseur** précédent	g_2 est l'occurrence de tête du terme $(g_2)y$ qui est argument d'une application
b_{g_2} a pour portes auxiliaires les conclusions I (héréditaires) des axiomes y et g_2	y et g_2 sont des occurrences libres dans $(g_2)y$
la porte principale de b_{g_2} est prémisse du même **tenseur** que la conclusion I de g_1	le terme $(g_2)y$ est l'argument de g_1
la conclusion I de y est l'unique prémisse d'un lien **pourquoi pas**	y est l'unique occurrence de la variable y
ce lien **pourquoi pas** est prémisse du même **par** que la conclusion O de g_1	λy est immédiatement à gauche de g_1
les conclusions I de g_1 et g_2 sont héréditairement prémisse d'un même lien **pourquoi pas**	g_1 et g_2 sont les deux occurrences de la variable g
ce lien **pourquoi pas** est prémisse d'un second lien **par** dont la prémisse O est conclusion du précédent (et donc héréditairement de la conclusion O de g_1)	λg est le second lambda à gauche de g_1
la conclusion O de g_1 est prémisse héréditaire de la porte principale d'une boîte b_{g_1} contenant la boîte b_{g_2}, le **tenseur** entre b_{g_2} et la conclusion I de g_1 et les deux liens **par**	g_1 est l'occurrence de tête de $\lambda g \lambda y(g_1)(g_2)y$ qui est argument d'une application
la porte principale de b_{g_1} est prémisse de la coupure	le terme $\lambda g \lambda y(g_1)(g_2)y$ est l'argument du redex défini par λx

Chapitre 3

Réseaux

3.1 Le critère de correction

Un *interrupteur* d'une structure de réseau S est un sous-graphe non orienté de S obtenu par les opérations :

- réduire certaines boîtes, choisies arbitrairement, à de simples sommets, oubliant ainsi l'intérieur et connectant toutes les portes.

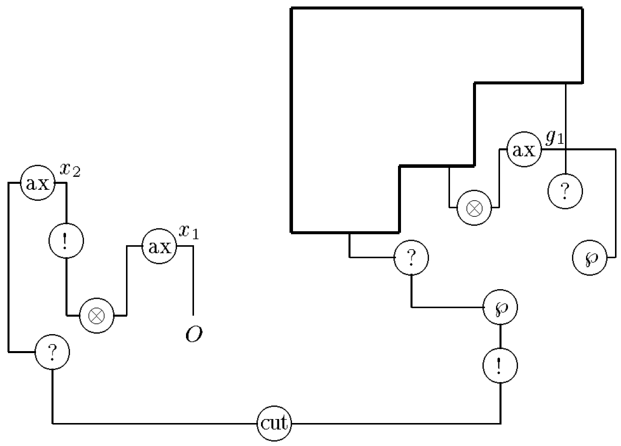
- pour chaque lien **par** et **pourquoi pas** restant, effacer toutes ses arêtes prémisses sauf une ;

Définition 3.1.1 *Un* réseau *est une structure R vérifiant la condition suivante appellée* critère de correction *(en abrégé* **CC***) :*

POUR TOUT INTERRUPTEUR G DE R, G EST ACYCLIQUE ET COMPTE UNE COMPOSANTE CONNEXE DE PLUS QUE DE LIENS **affaiblissement**.

REMARQUE : ceci impose que le critère soit vérifié à l'intérieur de chaque boîte. En effet si on prend un interrupteur G qui réduit une boîte b à un simple sommet, on compte n liens **affaiblissement** et $n+1$ composantes connexes. Soit G' un interrupteur de l'intérieur de b et $G+G'$ celui de la structure complète i.e., celui où l'on remplace dans G le sommet auquel b était réduit par G' : comme on a ajouté à G les p liens **affaiblissement** de la boîte b, $G+G'$ doit maintenant compter $n+p+1$ composantes connexes. Donc G' compte $p+1$ composantes connexes. De plus $G+G'$ acyclique implique que G' est acyclique. Donc si R est un réseau alors l'intérieur de toute boîte de R est à son tour un réseau.

Exemple : voici un interrupteur de $(\mathcal{Q})^{\circ}$ dans lequel la boîte b_{g_2} est réduite à un sommet :



La démonstration du lemme suivant, qui nous sera utile plus loin est un exemple d'utilisation de **CC** :

Lemme 3.1.1 *Un réseau R a toujours une conclusion positive*

Preuve : soit G l'interrupteur de R défini en réduisant toutes les boîtes à des sommets, et en enlevant pour chaque lien **par** sa prémisse $?I$, pour chaque lien **pourquoi pas**, toutes ses prémisses sauf une. Considérons une arête terminale dans G. Si elle est positive et conclusion de G alors elle est conclusion de R. Sinon elle peut être positive et prémisse d'une coupure c auquel cas on passe à la prémisse négative de c que l'on emprunte en montant. Dans tous les cas on a soit trouvé du premier coup une conclusion positive, soit une arête terminale négative. On monte alors dans celle-ci. Si c'est une arête $?I$, on passe par un lien **pourquoi pas** l, puis par la prémisse I de l qui figure dans G. Si l est un **affaiblissement**, le critère **CC** nous garantit alors l'existence d'une autre composante connexe dans G sur laquelle on recommence. On arrive donc dans une arête I en montant. En continuant à monter, on parcourt un arbre **tenseur** jusqu'à son **axiome** de tête a. On passe alors à la conclusion O de a et on descend à travers un arbre **par** — dont justement on a gardé les arêtes prémisse O — dont la conclusion est soit conclusion du réseau, ce que l'on cherchait, soit prémisse d'une coupure que l'on traverse alors pour remonter le long de sa prémisse négative, soit prémisse d'un lien **bien sur**. Dans ce dernier cas on le traverse en descendant aboutissant soit à une conclusion, soit à une prémisse de coupure dont on repart alors en montant par la prémisse négative, soit en un lien **tenseur** d'où l'on repart alors en montant par sa prémisse négative.
On va voyager ainsi dans G en montant dans les arête portant des formules négatives et en descendant dans les positives, en sautant à une autre composante connexe — qui existe toujours d'après **CC**— quand on arrive dans un lien **affaiblissement** et sans jamais repasser deux fois par le même lien puisque sinon G serait cyclique. Comme G est de taille finie, la procédure termine forcément et son seul cas de terminaison est dans une conclusion positive de G donc de R. •

3.2 Le système De et les réseaux

Proposition 3.2.1 *Soit S un type principal de* **De**. *Alors $(S)^{\circ}$ est un réseau.*

Preuve : comme on l'a vu, on associe à un type principal $S = \Gamma \vdash_{\mathrm{De}} P, \mathcal{E}$ une structure ayant autant de boîtes qu'il y a d'étiquettes dans S, de conclusions I qu'il y a de formules de genre X dans Γ, chacune étant porte auxiliaire des boîtes décrites par les étiquettes de la formule X correspondante et autant de liens **coupure** qu'il y a d'équations dans $\mathcal{E}$, chacun étant dans les boîtes correspondant aux étiquettes portées par l'équation dans $\mathcal{E}$. On va montrer par induction sur la dérivation de S que $(S)^\circ$ vérifie **CC**.

- Si $S = \Gamma_1^{b_1}, \dots \Gamma_n^{b_n}, P_1^{b_1} \cdots P_n^{b_n} \to x_k \vdash_{\mathrm{De}} x_k, \mathcal{E}_1^{b_1}, \dots, \mathcal{E}_n^{b_n}$ a été obtenu par une règle d'application simple alors par induction on a n réseaux $R_1 = (S_1)^\circ, \dots, R_n = (S_n)^\circ$ correspondant aux n types principaux $S_i = \Gamma_i \vdash_{\mathrm{De}} P_i, \mathcal{E}_i$. On procède exactement comme précédemment à ceci près que l'arbre **tenseur** que l'on fabrique a maintenant comme prémisses $!O$ non les conclusions des arbres traduction de formules, mais les conclusions $!O$ des réseaux $(R_i^{b_i})^\circ$. Clairement on obtient bien ainsi une structure $R = (S)^\circ$.

  Vérifions que c'est un réseau : soient $G_1, \dots G_n$ des interrupteurs de $R_1, \dots, R_n$ et soit $G = G_1 \otimes \cdots \otimes G_n \otimes a$ l'interrupteur correspondant de R. Alors, comme on a connecté les G_i par un seul lien, G est acyclique car les G_i le sont, et a comme nombre de composantes connexes la somme de celles des G_i moins n, soit exactement le nombre de liens **affaiblissement** de G plus un.

- Si $S = \Gamma \vdash_{\mathrm{De}} \bigcap_{1 \leq i \leq n} X_i^{\beta_i} \to P, \mathcal{E}$ est obtenu par une règle d'abstraction alors par induction on a un réseau $R' = (S')^\circ$ correspondant au type principal $S' = \Gamma, X_1^{\beta_1}, \dots, X_n^{\beta_n} \vdash_{\mathrm{De}} P, \mathcal{E}$ ayant en particulier n conclusions I correspondant aux X_i. On lie celles-ci par un lien **pourquoi pas** l puis la nouvelle conclusion $?I$ avec la conclusion O de R' par un lien **par** obtenant ainsi R qui est bien égale à $(S)^\circ$.

  Soit G' un interrupteur de R'. On obtient — et tout interrupteur de R s'obtient ainsi — un interrupteur ajoutant à G les liens **pourquoi pas** et **par** de R, et en effaçant toutes, sauf une, leurs prémisses. Si l n'est pas un **affaiblissement**, par induction G' est acyclique et a k composantes connexes. Comme G a deux arêtes et deux sommets de plus, il est aussi acyclique et a aussi k composantes connexes. Si l est un **affaiblissement** alors G a seulement une arête de plus que G', donc une composante connexe de plus. Mais justement il a un **affaiblissement** de plus aussi.

- Si $S = \Gamma, \Gamma'^b \vdash_{\mathrm{De}} y, \mathcal{E}, \mathcal{E}'^b, \bigcap_{1 \leq i \leq n} X_i^{\beta_i} = P^b$ est obtenu par une règle d'application radicale alors par induction on a deux réseaux $R_1 = (S_1)^\circ$ et $R_2 = (S_2)^\circ$ correspondant aux types principaux $S_1 = \Gamma, X_1^{\beta_1}, \dots, X_n^{\beta_n} \vdash_{\mathrm{De}} y, \mathcal{E}$ et $S_2 = \Gamma' \vdash_{\mathrm{De}} P, \mathcal{E}'$. On lie les n conclusions I correspondant aux X_i de R_1 par un lien **pourquoi pas**, on ajoute un lien **bien sur** à la conclusion O de R_2 créant ainsi une boîte b et on lie les deux nouvelles conclusions $?I$ et $!O$ de respectivement R_1 et R_2 par un lien **coupure** obtenant ainsi $R = (S)^\circ$. Encore une fois la vérification que c'est un réseau est immédiate. •

Si S et S' sont deux types principaux d'un terme T alors $(S)^\circ = (S')^\circ$ car S et S' ne diffèrent que par les nom de leurs variables de type, et ceux-ci n'apparaissent pas dans les réseaux. Si T et T' sont deux termes canoniques équivalent opérationnellement alors ils ont les mêmes types principaux puisqu'ils ne différent que par l'ordre dans lequel sont écrits leur redex. Ces deux remarques légitiment la définition suivante :

Définition 3.2.1 *Soit T un terme du lambda-calcul, T_c une forme canonique de T et S un type principal de T_c. On appelle* traduit *de T et on note $(T)^\circ$ le réseau $(S)^\circ$.*

NOTATIONS : pour traduire un type principal, on a vu qu'il fallait associer à chaque lien **axiome** une variable de type. Par ailleurs, chaque variable de type correspond à une occurrence de variable dans le terme. Autrement dit, chaque lien **axiome** d'un réseau traduit correspond à une occurrence de variable dans le terme. Par conséquent, si $R = (T)^\circ$ est le réseau traduit du terme T et si a est un lien **axiome** de R, on notera $a = (x)^\circ$ où x est l'occurrence de variable correspondant à a.
De même à chaque redex C du terme correspond une équation dans son type principal et donc une coupure dans le réseau traduit que l'on notera $(C)^\circ$.
A chaque lambda λx du terme liant les occurrences $x_1, \ldots, x_n$ de la variable x correspond un lien **pourquoi pas** que l'on notera $(\lambda x)^\circ$ dont les n prémisses I ont pour **axiome** de tête les $(x_j)^\circ$. A chaque sous-terme V argument soit d'un redex, soit d'une application simple correspond une boîte dont l'intérieur est $(V)^\circ$.

3.3 Séquentialisation

On montre ici la réciproque de la dernière proposition : sous certaines hypothèses, un réseau est la traduction d'un lambda-terme.

Définition 3.3.1 *On appellera* réseau exponentiel *un réseau n'ayant que des coupures exponentielles et pas de conclusion $?I$.*
On appellera chaîne *une suite de boîtes $(b_i)_{1\leq i\leq n}$ d'un réseau telle que pour tout i, la boîte b_i a une porte auxiliaire α_i qui est prémisse d'un lien* **pourquoi pas**, *lui-même prémisse d'une coupure avec la porte principale de la boîte b_{i+1}.*
On appelle coupure spéciale *d'un réseau une coupure exponentielle externe telle que la boîte associée à sa prémisse $!O$ a toute ses portes auxiliaires conclusion du réseau. Autrement dit, une coupure spéciale décompose un réseau R en $R = !R_1 \mathbf{cut} R_2$*

Lemme 3.3.1 *Soit une chaîne $(b_i)_{1\leq i\leq n}$ d'une structure R. On note l_i le lien* **pourquoi pas** *par lequel b_i est coupée avec b_{i+1}. Si R est un réseau on a : pour tout $i, j \leq n$, si $i \neq j$ alors $b_i \neq b_j$ et $l_i \neq l_j$.*

Preuve : par induction sur n : si $n = 1$ il n'y a rien à dire. Sinon soit $(b_i)_{1\leq i\leq n+1}$ une chaîne. On prend le graphe G défini en réduisant toutes, et uniquement celles-là, les boîtes b_i à des sommets, et pour $i < n$ en effaçant les prémisses du lien l_i sauf la porte auxiliaire de b_i. Par hypothèse sur n il y a un chemin dans G de b_1 à b_n passant par tous les sommets b_i. Si l_n est l'un des l_i, soit l_{i_0}, alors une porte auxiliaire de b_n et une porte auxiliaire de b_{i_0} sont prémisse de $l_n = l_{i_0}$. Comme il y a un chemin dans G de b_{i_0} à b_n, donc de l_{i_0} à b_n, on déduit que tout interrupteur qui efface les prémisses des l_i sauf la porte auxiliaire de b_i pour $i > i_0$ est cyclique. Mais cela contredit l'assertion que R est un réseau. Donc l_n est distinct de tous les l_i et en particulier, les prémisses de l_n ne sont pas des portes auxiliaires des b_i donc n'ont pas été effacées dans G. On construit G' en les effaçant toutes sauf la porte auxiliaire de b_n. Si b_{n+1} est l'une des b_i soit b_{i_0} alors G' a un cycle $b_{i_0} b_{i_0+1} \cdots b_{n+1}$. Mais on en déduit immédiatement un interrupteur cyclique de R. •

Théorème 3.3.2 (Séquentialisation) *Soit R un réseau exponentiel. Il existe un type principal S de* **De** *tel que $R = (S)^\circ$.*

Preuve : on raisonne par induction sur R.

- Si R est un **axiome** alors il est la traduction du type principal $S = x \vdash_{\mathrm{De}} x$ obtenu par une règle d'application simple sans prémisse.

- Si R se termine par un lien **par**, soit $R = \wp R'$, alors R' a une conclusion $?I$, prémisse du **par**, donc un lien **pourquoi pas** terminal de prémisses n arêtes I, soit $R' = ?R''$. R'' est donc un réseau exponentiel donc par induction il existe S'' tel que $R'' = (S'')^\circ$. En particulier, S'' a la forme: $S'' = \Gamma, X_1^{\beta_1}, \ldots, X_n^{\beta_n} \vdash_{\mathrm{De}} P, \mathcal{E}$ où les X_i correspondent aux n conclusions I de R''. Donc par une application de la règle d'abstraction sur S'' on obtient un type principal $S = \Gamma \vdash_{\mathrm{De}} \bigcap_{1 \leq i \leq n} X_i^{\beta_i} \to P, \mathcal{E}$ dont R est la traduction.

- Si R n'a pas de liens **par** terminal, mais des coupures externes alors comme il est exponentiel il n'a pas non plus de lien **pourquoi pas** terminal. Soit c une coupure externe de R dont la prémisse $!O$ a une boîte associée b. Soit α une porte auxiliaire de b. Elle ne peut être prémisse que d'un lien **pourquoi pas** l ou conclusion du réseau car elle est porte auxiliaire de boîte. Dans le premier cas, soit α_1 la conclusion $?I$ de l. Elle ne peut qu'être la prémisse d'une coupure avec un $!O$ de boîte associée b_1. Sinon α_1 est prémisse d'un lien **par** dont la conclusion O est prémisse d'un **par** etc...jusqu'à arriver à une conclusion O du réseau. Il ne peut y avoir de lien **bien sur** car sinon sa boîte associée contiendrait b, or celle-ci est externe. Mais alors R a des liens **par** terminaux, ce qui contredit l'hypothèse. Donc α est soit conclusion du réseau, soit prémisse d'un lien **pourquoi pas** dont la conclusion est liée par une coupure à la porte principale d'une boîte externe b_2. Dans ce dernier cas on peut recommencer le raisonnement et exhiber ainsi une chaîne (b_i). Comme le réseau est de taille finie et qu'une chaîne ne peut être cyclique, la suite se termine sur une boîte b_p dont toutes les portes auxiliaires sont conclusions du réseau R. La porte principale de b_p est donc précisément la prémisse d'une coupure spéciale de R et on a $R = !R_1 \mathbf{cut} ?R_2$ où R_1 est l'intérieur de b_p et R_2 est le réseau R auquel on enlevé le lien **pourquoi pas**, remplaçant ainsi une conclusion $?I$ par p conclusions I. Donc par induction sur R_1 et R_2 on a deux types principaux $S_1 = \Gamma_1 \vdash_{\mathrm{De}} P_1, \mathcal{E}_1$ et $S_2 = \Gamma_2, X_1^{\beta_1}, \ldots, X_p^{\beta_n} \vdash_{\mathrm{De}} P, \mathcal{E}_2$ qui par une règle d'application radicale donnent $S = \Gamma_1^b, \Gamma_2 \vdash_{\mathrm{De}} P, \mathcal{E}_1^b, \mathcal{E}_2, \bigcap_{1 \leq i \leq p} X_i^{\beta_i} = P_1^b$ dont la traduction est R.

- enfin si R n'a que des liens **tenseur** terminaux alors l'un d'eux est externe. En effet notons a l'**axiome** dont la conclusion O est la conclusion de R. Considérons la conclusion de l'arbre **tenseur** dont a est l'**axiome** de tête. C'est une arête externe car a l'est. Elle ne peut être que conclusion du réseau ou prémisse d'un lien **pourquoi pas** l. Dans le dernier cas la conclusion de l est l'un des lambda de tête d'un arbre **par** dont la conclusion, toujours externe ne peut donc être prémisse d'un **bien sur**. Comme il n'y a pas de coupure multiplicative c'est donc qu'elle est conclusion du réseau, ce qui contredit l'hypothèse qu'il n'y a pas de **par** terminal. Donc la conclusion de l'arbre **tenseur** est conclusion, externe du réseau.

  Notons que si cet arbre est réduit à une simple arête i.e., il ne possède aucun sommet, alors le réseau est composé du seul lien **axiome** a. Prenons un interrupteur G qui réduit toutes les boîtes à des sommets; alors G n'a aucun lien **affaiblissement**, sinon c'est que R possède un lien **affaiblissement** externe, dont la conclusion par le même raisonnement que ci-dessus ne peut être que conclusion de R. Mais alors R n'est pas exponentiel puisqu'un réseau exponentiel n'a que des conclusions I. Si G n'a pas de lien **affaiblissement**, alors d'après **CC**, il doit avoir exactement une composante connexe, et celle-ci ne peut être que l'**axiome** a.

Revenons à notre arbre **tenseur** puisque l'on est maintenant assuré qu'il possède au moins un lien, donc un lien terminal. Soit b la boîte associée à sa prémisse $!O$; par le même raisonnement que dans le cas précédent on voit que toutes les portes auxiliaires de b sont des conclusions de R puisqu'il n'y a pas de coupures externes. Donc $R =!R_1 \otimes R'_1$ où R_1 est l'intérieur de b et R'_1 est le réseau R auquel on a enlevé le lien **tenseur** terminal externe ainsi que la boîte b. On réitère le procédé avec l'avant-dernier lien **tenseur**, etc..., obtenant $R =!R_1 \otimes \cdots \otimes R_p \otimes a$. Par induction sur les R_i on a des types principaux S_i correspondant que l'on peut tous joindre au moyen d'une application simple obtenant ainsi un type principal S correspondant au réseau R. •

Remarque : le système **De** n'a pas de règle de dérivation dont la traduction produirait un réseau avec des coupures multiplicatives. C'est la raison pour laquelle on a restreint le théorème de séquentialisation aux réseaux exponentiels.

Remarque : la preuve du théorème de séquentialisation se généralise au cas typé. On obtient alors que tout réseau typé dans un fragment adéquat de la logique linéaire (le fragment intuitionniste) est la traduction d'un lambda-terme typé. L'argument est développé dans l'appendice B.

Corollaire 3.3.3 *Un réseau exponentiel a une unique conclusion positive ; s'il n'a pas de lien* **par** *terminal, alors il a une unique conclusion I qui n'est porte auxiliaire d'aucune boîte. Elles sont toutes les deux conclusions héréditaire d'un même lien* **axiome**.

Preuve : on le vérifie par induction sur la séquentialisation du réseau ; la conclusion O correspond à la formule P du type principal, et la conclusion I dans le cas où le réseau n'a pas de lien **par** terminal, i.e., où P est une variable de type, correspond à l'occurrence négative de cette variable •

Cette propriété va nous être prétexte à introduire un peu de terminologie sur les réseaux. On appellera *axiome de tête* le lien **axiome** a dont la conclusion O a pour conclusion héréditaire la conclusion positive du réseau i.e., le lien **axiome** de tête de l'arbre **par** dont la conclusion est aussi celle du réseau. On appellera *liens* **tenseur** *de tête* et *liens* **par** *de tête* les liens **tenseur** et **par** appartennant aux arbres respectivement **tenseur** et **par** dont a est l'axiome de tête.
Remarquons que la conclusion d'un **tenseur** de tête ne peut être prémisse héréditaire que d'autres **tenseur** de tête, puis d'un lien **pourquoi pas**, et enfin de **par** de tête. La conclusion d'un **par** de tête ne peut être prémisse héréditaire que d'autre **par** de tête et éventuellement d'un lien **bien sur**. L'**axiome** de tête d'un réseau est exactement celui qui correspond à l'occurrence de la variable de tête du terme dont il provient.

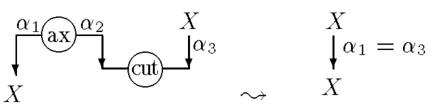
Chapitre 4

Réduction des réseaux

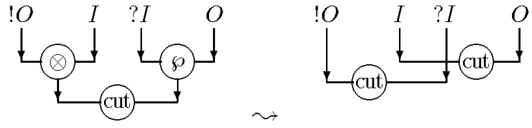
4.1 Définition

Eliminer une coupure c, ou *réduire un réseau R selon la coupure c* dont les prémisses sont conclusion de deux liens l_1 et l_2 dans un réseau R, c'est la remplacer par des coupures $c_1, \ldots, c_n$ dont les prémisses sont les prémisses de l_1 et l_2 en dupliquant ou en effaçant certains sous-réseaux de R. On obtient ainsi une structure R' appelée *réduit à une étape de R* dans laquelle l_1, l_2 et c ont disparu. Les coupures $c_1, \ldots, c_n$ qui remplacent c sont les coupures *créées* par la réduction de c. On dira de toutes les autres coupures de R' qu'elles étaient *présentes* dans R ou que *R les possédait*.
On montre les différents cas de réduction suivant la nature de la coupure c i.e., on définit pour chaque type de coupure la structure R' :

1. **coupure axiome** : l_1 est un lien **axiome** dont une arête conclusion α_1 porte une formule X ($X = I$ ou O), la seconde α_2 porte la formule $X^\perp$ et c est une coupure de prémisse α_2 et α_3 portant la formule X (α_3 est donc conclusion de l_2). La coupure s'élimine en effaçant c, l_1, α_2 et α_3, et en identifiant les arêtes α_1 et α_3 :

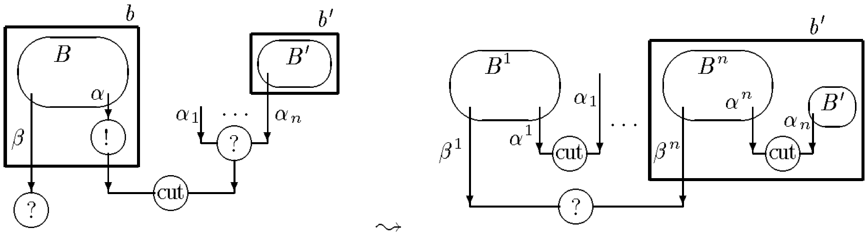


2. **coupure binaire**, notée $\otimes/\wp$: l_1 est un lien **tenseur** de prémisses α_1 (portant $!O$) et α_2 (portant I) et l_2 est un lien **par** de prémisse β_1 (portant $?I$) et β_2 (portant O). On efface l_1, l_2, leur conclusion et c ; on remplace cette dernière par une coupure exponentielle c_1 de prémisses α_1 et β_1 et une coupure multiplicative de prémisses α_2 et β_2.



Remarquons que les deux règles de réductions multiplicatives font strictement décroître la taille du réseau.

3. **coupure exponentielle**, notée $!/?$: l_1 est un lien **bien sur** de boîte associée b et de prémisse une arête α portant la formule O ; l_2 est un lien **pourquoi pas** ayant n prémisses $\alpha_1, \ldots, \alpha_n$ portant la formule I. On efface l_1 et l_2 ainsi que leur conclusion, si bien que la boîte b n'existe plus (elle n'a plus de lien **bien sur** associée). On copie n fois (on efface donc si $n = 0$) l'intérieur B de la boîte b obtenant n réseaux (puisque l'intérieur d'une boîte est un réseau) $B^1, \ldots, B^n$. Si α^i est la $i$ème copie de α alors on lie α^i à la $i$ème prémisse de l_1 par une coupure multiplicative c_i. Si β est une porte auxiliaire de b et que β est prémisse d'un lien **pourquoi pas** l alors on remplace β par ses n copies β^i si bien que l a $n-1$ prémisses de plus dans R'. Enfin si α_i est porte auxiliaire d'une boîte b' d'intérieur B' alors, dans R', la boîte b' contient B^i en entier ainsi que α_i et la coupure c_i, et a pour nouvelles portes auxiliaires les conclusions I de B^i. Cette réduction augmente en général la taille du réseau.



Pour ne pas alourdir le dessin, on s'est contenté d'une seule porte auxiliaire à b et on a supposé que seule α_n est porte auxiliaire d'une unique boîte b'.

Remarque : si β est une porte auxiliaire de la boîte dupliquée mais qu'elle n'est pas prémisse d'un lien **pourquoi pas**, alors c'est qu'elle est conclusion de R. Dans ce cas R' comptera $n-1$ conclusions de plus que R.

On dira qu'une arête α' de R' *provient* d'une arête α de R quand, soit α n'a pas été changée par la réduction de c et $\alpha' = \alpha$, soit α était dans la boîte b et $\alpha' = \alpha^i$ pour un i.
On dira que R' est *un réduit en n étapes de R* s'il existe une suite de réseaux $R_0 = R, \ldots, R_n = R'$ tels que pour tout i, R_{i+1} est le produit de l'élimination d'une coupure c_i de R_i. Si pour tout i la coupure c_i est stricte, on dira que la réduction est *stricte* et que R' est un réduit strict de R.

Remarque : dans une structure de réseaux, il peut apparaître des coupures que l'on ne saurait réduire ; il y en a deux :

1. une coupure **axiome** dont les deux prémisses sont les conclusions d'un même lien **axiome** ;
2. une coupure exponentielle c dont la prémisse $?I$ est conclusion héréditaire d'une porte auxiliaire de la boîte associée à la prémisse $!O$ de c.

Mais de telles coupures ne peuvent jamais apparaître dans un réseau, sinon on en déduit un interrupteur cyclique. Remarquons que l'on ne se sert que de la condition d'acyclicité.

Proposition 4.1.1 *Tout réduit d'un réseau est un réseau.*

Preuve : il suffit de le voir pour un réduit à une étape de R: on suppose que R se réduit en R' par une élimination de la coupure c. On montre que si R est un réseau, alors R' vérifie **CC**.

1. La vérification que le réduit **axiome** R' de R est encore un réseau est immédiate puisque R' est obtenu en effaçant deux arêtes et deux sommets, et en recollant les bouts i.e., en racourcissant un chemin, donc sans rien changer ni à la cyclicité, ni à la connexité de tout interrupteur de R.

2. Notons α_1 et α_2 les deux prémisses du **tenseur** et α'_1, α'_2 celles du **par** et soit G un interrupteur de R dont on déduit G' un interrupteur de R' en faisant les mêmes effacements d'arêtes prémisse de **par** et de **pourquoi pas** que pour G. Notons $G(\alpha)$ le sous-graphe de G connecté au lien prémisse de l'arête α quand on efface celle-ci. Comme G est acyclique on a donc que $G(\alpha_1)$ et $G(\alpha_2)$ sont disjoints, de même que le sont $G(\alpha_i)$ et $G(\alpha'_j)$. Donc G' est acyclique. Supposons que $G(\alpha'_1)$ et $G(\alpha'_2)$ soient disjoints et notons $G(c)$ la composante connexe de c dans G. Alors $G(c)$ contient $G(\alpha_1)$, $G(\alpha_2)$ et $G(\alpha'_1)$ (en supposant que G est obtenu en effaçant α'_2) et est distinct de $G(\alpha'_2)$. Mais alors $G'(c_1)$ est distinct de $G'(c_2)$, et donc G' a le même nombre de composantes connexes que G puisque tout est égal par ailleurs. Si $G(\alpha'_1) = G(\alpha'_2)$ alors $G(c)$ contient tout le monde, de même que $G'(c_1)$ ou $G'(c_2)$ donc G et G' ont encore le même nombre de composantes connexes. Finalement on a bien que R' est un réseau ssi R l'est.

3. Soit G un interrupteur de R qui réduit la boîte b à un sommet. G se décompose en un interrupteur G_1 de R_1, et un interrupteur G_2 de R_2. Considérons deux arêtes frontières : si elles sont connectées dans G_2 alors c'est que ce sont deux arêtes prémisses de l_2. En effet supposons par exemple que l'une soit α_1 et l'autre ι_1. Alors, comme b est réduit à un seul sommet, si on a choisit un interrupteur G qui garde les arête ι_1 et α_1, on en déduit un cycle. Le seul cas où ce raisonnement ne s'applique pas est celui où les deux arêtes sont prémisses de l_2 car aucun interrupteur ne peut garder les deux à la fois. Mais deux arêtes prémisses de l_2 ne peuvent être connectées puisque chaque interrupteur efface au moins l'une d'elle. Donc toute paire d'arêtes de la frontière est déconnectées dans G_2.

   Soit G' un interrupteur de R' qui se décompose donc en G'_1 et G_2 car R_2 est laissée invariant par la réduction. G'_1 ne peut être cyclique car sinon c'est que l'interrupteur de l'un des B^i l'est donc qu'il existe un interrupteur cyclique de R car les B^i sont des copies de l'intérieur de b. Si G' est cyclique, c'est donc qu'il y a deux arêtes de la frontière entre R'_1 et R_2 qui sont connectées d'un coté par G'_1, de l'autre par G_2. Mais on vient de voir que deux arêtes de la frontière ne peuvent être connectées dans G_2. Donc G' est acyclique.

   Soient $G_1, \ldots, G_n$ des interrupteurs de respectivement $B^1, \ldots, B^n$. Comme les B^i sont des réseaux, le nombre de composantes connexes de chaque G_i est égal à $a_i + 1$ où a_i est le nombre de liens **affaiblissement** contenus dans G_i. Donc l'interrupteur de R_1 que forment la réunion des G_i a $a_1 + \cdots + a_n + n$ composantes connexes. Soit G' l'interrupteur de R' dont la restriction aux B^i donne les G_i. Tout se passe comme si G' était obtenu à partir de G en cassant le sommet auquel on avait réduit b en $a_1 + \cdots + a_n + n$ composantes connexes, et en ajoutant n arêtes, celles des prémisses du lien l_2 qui ne sont plus effacées car ce lien n'est plus dans R', le tout sans créer de cycle. Donc la composante connexe de G qui contenait b est devenue $a_1 + \cdots + a_n + 1$ composantes connexes. Comme G' a par ailleurs a composantes connexes de plus, où a est le nombre de liens **affaiblissement** de G, on obtient finalement le résultat souhaité que G' a $a_1 + a_n + a + 1$ composantes connexes. •

On vérifie sans difficulté que la réduction des réseaux est isomorphe à la résolution d'équations des types principaux. Plus précisément on a :

Théorème 4.1.2 *Si $R = (S)^\circ$ où S est un type principal de* **De** *alors :*

1. *R se réduit par une coupure* **axiome** *en R' ssi S se réécrit en S' par une règle de variable. La variable de type qui disparait dans S correspond au lien* **axiome** *qui disparait dans R.*

2. *R se réduit en R' par une coupure $\otimes/\wp$ ssi S se réécrit en S' par une règle de flèche.*

3. *R se réduit en R' par une coupure exponentielle ssi S se réécrit en S' par une règle de duplication. Cette dernière duplique les arêtes qui portent l'étiquette b, qui correspond à la boîte dupliquée dans R. Dans R' la boîte n'existe plus, de même que dans S' l'étiquette b a disparu.*

Finalement avec ce dernier résultat, on a achevé de montrer que la résolution d'équations, isomorphe à l'élimination des coupures des réseaux, est bien définie. Remarquons que de même qu'un réseau exponentiel ne se réduit pas en général en un réseau exponentiel à cause des coupures multiplicatives qui apparaissent, de même un type principal de **De** laisse apparaître par réécriture des équations $X = P$.

4.2 beta-réduction et élimination des coupures

On appelle *réduction beta* d'un réseau exponentiel R la suite de réductions suivantes :

1. choisir une coupure exponentielle c et la réduire ;

2. réduire toutes les coupures multiplicatives créées jusqu'à retrouver un réseau exponentiel R'.

Cette définition fait bien du sens puisque les réductions multiplicatives diminuent la taille du réseau. Détaillons les étapes d'une réduction beta pour un réseau traduit d'un terme T. Si T a un redex R de lambda λx liant des occurrences de variables $x_1, \ldots, x_n$, d'argument $\lambda y_1 \ldots \lambda y_p V$ alors $(T)^\circ$ a une coupure $(R)^\circ$ dont la prémisse $!O$ a une boîte associée d'intérieur $\wp^p(V)^\circ$ et dont la prémisse $?I$ est issue d'un lien **pourquoi pas** $(\lambda x)^\circ$ à n prémisses I. Plus précisément, si x_j est la $j$ème occurrence de x et a m arguments $U_1, \ldots, U_m$ alors la $j$ème prémisse de $(\lambda x)^\circ$ est conclusion d'un arbre de m liens **tenseur**, chacun d'eux ayant pour prémisse $!O$ la conclusion de la boîte $!(U_i)^\circ$, et dont l'axiome de tête est $(x_j)^\circ$. Chacun des p liens **par** de $(V)^\circ$ a pour prémisse la conclusion d'un lien **pourquoi pas** $(\lambda y_j)^\circ$.
Si on réduit la coupure c on trouve n coupures multiplicatives, dont la $j$ème porte sur la conclusion de l'arbre **tenseur**, et la conclusion de l'arbre **par** de $(V)^\circ$. Après réduction des $\mathsf{inf}(p, m)$ coupures $\otimes/\wp$ on trouve $\mathsf{inf}(p, m)$ coupures exponentielles de prémisses respectivement $!(U_i)^\circ$ et $(\lambda y_i)^\circ$ et une coupure **axiome** a. Si par exemple p est plus grand que m alors a a pour prémisse la conclusion I de $(x_j)^\circ$ et la fin de l'arbre **par** soit $(\lambda y_{m+1} \ldots \lambda y_p V)^\circ$. La réduction de la coupure **axiome** va donc ajouter l'arbre **par** à celui qui existait déja sous la conclusion O de a. Autrement dit on a rajouté au lien **axiome** de tête de $(V)^\circ$ les liens **par** de l'axiome $(x_j)^\circ$ ou, en termes de lambda-calcul, on a mis les lambda de tête de l'occurrence x_j en tête de V. Si p est plus petit que m, on va rajouter au lien **axiome** de tête de $(V)^\circ$ les arguments en trop de $(x_j)^\circ$. On voit donc que par réduction beta, $(T)^\circ$ se réécrit en $(T')^\circ$ où T' est la forme canonique du beta-réduit de T. D'où la proposition suivante :

Proposition 4.2.1 *Soit T un terme. T se réduit en T' par le redex premier R ssi $(T)^\circ$ se réduit par une réduction beta commencant par la coupure exponentielle $(R)^\circ$.*

Donc $(T)^{\circ}$ se réduit en un réseau sans coupure R_0 ssi R_0 est la traduction de la forme normale de T.

Corollaire 4.2.2 *Si S est un type principal du terme T et S se réécrit en S_0 sans équations, alors S_0 est le type principal de la forme normale de T.*

4.3 Digression sur la réduction stricte

Dans la suite on se restreindra souvent à la réduction stricte des réseaux. On montre ici que ce faisant, du point de vue opérationnel qui est celui qui nous intéresse, on ne perd pas de généralité.

Définition 4.3.1 *Soient R un réseau. On appelle* approximation stricte *de R un réseau R_S qui se contracte en R uniquement par des coupures* **affaiblissement** *contenues dans R_S.*
Une forme normale stricte de R *est un réduit strict R_0 de R tel que R_0 est une approximation stricte de la forme normale de R.*

Remarque : donc si R_0 est une forme normale stricte de R et si c est une coupure stricte de R_0 alors c est contenue dans une boîte dont la porte principale est prémisse d'une coupure **affaiblissement**. En particulier le nombre d'éliminations de coupures menant de R_0 à la forme normale de R est au minimum égal au nombre de coupures **affaiblissement** de R''. Notons qu'il n'y a pas unicité de la forme normale stricte d'un réseau.

Lemme 4.3.1 (Commutation des coupures affaiblissement) *Soient R et R' deux réseaux tels que R se réduit en R' par une coupure* **affaiblissement** *a, puis une coupure stricte c. Alors R se réduit en R' par la coupure (stricte) c puis par des coupures* **affaiblissement** *$a^1, \ldots, a^n$.*

Preuve : remarquons d'abord que puisque on peut réduire c après a, c'est que c n'est pas dans la boîte effacée par a. Donc c est une coupure stricte de R que l'on peut réduire :

1. si c est une coupure multiplicative, alors après élimination de c la coupure a est toujours présente : on peut donc la réduire obtenant ainsi le réseau R' ;
2. si c est une coupure exponentielle dont la boîte ne contient pas a alors on réduit c d'abord, puis a qui est restée inchangée obtenant encore le réseau R' ;
3. si c est une coupure exponentielle dont la boîte prémisse contient a, alors la réduction de c duplique a en n ($n \neq 0$ car c est stricte) coupures **affaiblissement** $a^1, \ldots, a^n$ dont les réductions permettent encore d'obtenir le réseau R'. •

Proposition 4.3.2 *Soit R un réseau se réduisant en R' par une suite d'éliminations de coupures S. Soit S' la suite déduite de S en lui enlevant les coupures* **affaiblissement**. *Alors S' est une suite de réductions de R menant à une approximation stricte R'' de R'.*

Remarque : S' est donc une réduction stricte contenant exactement les mêmes coupures strictes, dans le même ordre, que S.

Preuve : par induction sur la longueur de S en appliquant le lemme de commutation des coupures **affaiblissement**. La difficulté est de voir que S' est bien une suite de réductions de R. Mai on a vu que si on réduit une coupure **affaiblissement** a puis une coupure stricte c alors on pouvait réduire c avant, donc le fait de supprimer la réduction de a n'empêche pas de réduire c. •

En clair on vient de montrer que l'on peut toujours simuler une réduction quelconque par une réduction stricte ; il suffit de remplacer les termes "formes normales" par "forme normale stricte", "se réduit en" par "se réduit strictement en", pour passer de l'une à l'autre et ce sans autre effet sur les réductions que de laisser des coupures **affaiblissement**.

Corollaire 4.3.3 *R est fortement normalisable ssi R est fortement normalisable pour la réduction stricte. R est normalisable ssi R a une forme normale stricte.*

REMARQUE : cette propriété est l'analogue pour les réseaux de l'équivalence en lambda-calcul entre forte normalisation et terminaison de la stratégie de Barendregt. En effet celle-ci peut être vue comme une manière de définir une beta-réduction stricte des lambda-termes.

Preuve : si R est fortemement normalisable alors par définition il est strictement fortement normalisable. Réciproquement supposons qu'il y ait une réduction infinie de R; alors celle-ci possède une infinité de réduction strictes car les réductions **affaiblissement** font diminuer la taille de R. Mais par la propriété précédente on en déduit des réductions strictes de R de tailles arbitraires. Donc R n'est pas strictement fortement normalisable.
Si R a une forme normale stricte alors il a une forme normale par définition. Réciproquement s'il existe une réduction de R menant à une forme normale alors en appliquant le lemme pour faire commuter les réductions **affaiblissement**, on obtient une forme normale stricte de R. •

4.4 Conclusion

L'ensemble des réseaux est isomorphe à l'ensemble des lambda-termes quotienté par l'équivalence opérationnelle. Le fait que l'on ait maintenant deux types de redex — les coupures multiplicatives et exponentielles — est le prix a payer pour cela. Les coupures multiplicatives codent la réduction opérationnelle, tandis que les coupures exponentielles interprètent la beta-réduction. La correspondance étroite entre la réduction des réseaux et la beta-réduction justifie l'emploi que l'on fait des premiers pour l'étude de la seconde.

Troisième partie

Exécution des réseaux

Dans cette partie on va définir une notion d'exécution des réseaux basée sur des déplacement dans les graphes. L'intuition est que l'on cherche à définir la notion de "déplacement" d'une coupure dans un réseau. Pour cela on se donne la notion de *traces dérivables* qui sont les chemins dans le réseau non-modifiés par réduction.

Chapitre 1

Traces dans un réseau

1.1 Définition

A chaque arête α de prémisse l_1 et de conclusion l_2 d'un réseau R on associe son arête *inverse* ou *duale* α^* de prémisse l_2 et de conclusion l_1. On a donc $(\alpha^*)^* = \alpha$. On appelle *trace de* R une suite γ d'arêtes et d'arêtes inverses de R vérifiant : pour tout couple (α, β) d'arêtes ou d'arêtes inverses de R consécutives dans γ on a :

1. si α et β sont des arêtes alors α est prémisse d'un lien l et β est conclusion de l ;
2. si α et β sont deux arêtes inverses alors α est conclusion d'un lien l et β est prémisse de l.
3. si α est une arête et β une arête inverse alors α et β sont les deux prémisses d'un lien **cut** l ;
4. si α est une arête inverse et β une arête alors α et β sont les deux conclusions d'un lien **axiome** l.

Remarque : une trace de R est un chemin dans le graphe R mais un chemin dans R n'est pas en général une trace *de* R puisque celles-ci ne changent de sens que dans les coupures et les axiomes.

Dans tous les cas on dira que γ *passe* ou *emprunte* le lien l. On dira que γ commence (resp. se termine) en *descendant* (resp. en *montant*) si le premier (resp. le dernier) élément de γ est une arête (resp. une arête inverse). Le premier (resp. le dernier) élément de γ s'appelle l'arête *initiale* (resp. *finale*) de γ. Une trace maximale de R est donc une trace dont les arêtes initiale et finale sont des conclusions de R
On se donne une arithmétique formelle sur les traces :

- On appellera *traces discontinues* les suites d'arêtes qui ne sont pas des traces. Si $\gamma_1\gamma_2$ est une trace on dira que γ_1 et γ_2 sont *composables* ;

  On applique une relation d'équivalence sur l'ensemble des suites d'arêtes qui égalise toutes les traces discontinues : une classe d'équivalence est soit un singleton contenant une trace, soit l'ensemble des traces discontinues qui sera par définition la trace *nulle*.

- le produit de deux traces est leur concaténation en tant que suites ; le produit des traces γ_1 et γ_2 est non nul ssi la suite $\gamma_1\gamma_2$ est une trace. Donc 0 est l'élément absorbant du produit. Une trace est un produit d'arêtes i.e., de traces de longueurs 1 ;
- la trace unité noté 1 est la trace vide i.e., une suite vide d'arêtes. La trace 1 est l'élément neutre du produit.
- une somme de traces est le multi-ensemble de ces traces. La somme est commutative, associative, d'élément neutre 0 et distributive sur le produit ;
- la *duale* d'une trace $\gamma = (\alpha_i)_{0\leq i\leq n}$ notée γ^* est la trace $({\alpha_i}^*)_{n\leq i\leq 0}$. L'opération de dualité se distribue sur les sommes de traces i.e., $(\gamma_1+\gamma_2)^* = \gamma_1^* + \gamma_2^*$.

Si γ est une trace on dira que δ est une *sous-trace* de γ si δ est contenue dans γ i.e., γ s'écrit : $\gamma = \delta_1\delta\delta_2$. Si γ est une somme de traces alors une sous-trace de γ est une sous-trace de l'un des termes de γ. Si γ_1 est une somme de traces dans un réseau R_1 et si γ_2 est une somme de traces dans un réseau R_2 on a $\gamma_1 = \gamma_2$ ssi R_1 et R_2 sont le même réseau et γ_1 et γ_2 les mêmes sommes de traces.

1.2 Dérivation

1.2.1 Définitions

Soient R un réseau, c une coupure de R et R' le réseau obtenu en réduisant c dans R. Soit γ une trace de R ; on appelle *dérivée de γ dans R'* ou *dérivée de γ par la coupure c* la somme de traces (éventuellement nulle) γ' de R' définie par :

- si γ est la trace nulle alors sa dérivée est la trace nulle ;
- si c est une coupure multiplicative (i.e., une coupure **axiome** ou $\otimes/\wp$) dont les prémisses sont α_1 et α_2 alors γ' est la trace γ dans laquelle on a supprimé toutes les occurrences de α_1 et α_2 ;
- si c est une coupure exponentielle de prémisses α_1 conclusion d'un lien **pourquoi pas** à n prémisses et α_2 conclusion d'un lien **bien sur** de boîte b associée, dupliquant l'intérieur de la boîte b en n réseaux $B_1, \ldots, B_n$, alors γ' est γ dans laquelle on a :
  1. supprimé les arêtes α_1 et α_2 ;
  2. remplacé chaque occurrence d'une arête α appartenant à b ou porte auxiliaire de b par la somme $\alpha^1 + \ldots + \alpha^n$ de ses copies dans R'.

  Donc, par distributivité de la somme sur le produit, la dérivée d'une trace par une coupure exponentielle est en général une somme de plusieurs traces. On définit la dérivée d'une somme de traces comme la somme des dérivés de ses termes.

Comme on a supprimé toutes les occurrences des prémisses de c dans γ, et remplacé toutes les arêtes dupliquées par leur copies, on a bien que γ' est une somme de traces (éventuellement nulle) de R'.
Soient R un réseau et γ une somme de traces de R. On dira que l'on *réduit R relativement à γ* ou que l'on *réduit γ* si à chaque étape de la réduction on choisit une coupure empruntée par γ. Les dérivées de γ sont les *réduits de γ*.

REMARQUE : si γ est une trace ne passant par aucune coupure de R, en particulier si γ est la trace nulle, alors aucune réduction de R n'est relative à γ.

1.2.2 Quatre lemmes

Définition 1.2.1 *On dira qu'une trace* γ échange des prémisses en passant par une coupure c *dans les deux cas suivants :*

- *c est une coupure $\otimes/\wp$ entre deux liens l_1 et l_2 de prémisses respectivement α_1^1, α_1^2 et α_2^1, α_2^2 et de conclusions respectivement β_1 et β_2 ; γ contient une sous-trace de la forme $\alpha_1^i\beta_1{\beta_2}^*{\alpha_2^j}^*$ avec $i \neq j$;*
- *c est une coupure exponentielle entre un lien* **pourquoi pas** *ayant (au moins) deux prémisses α_1 et α_2, et de conclusion α et un lien* **bien sur** *de conclusion β et de boîte associée b ; γ a une sous-trace de la forme $\alpha_1\alpha\beta^*\delta\beta\alpha^*\alpha_2^*$ où δ est entièrement contenue dans b.*

REMARQUE : si γ échange les prémisses d'une coupure multiplicatives alors les prémisses échangées sont respectivement de genre I et $?I$ ou O et $!O$.

Lemme 1.2.1 (Echange de prémisses) *Soient c une coupure stricte d'un réseau R et γ une trace non nulle de R. Alors la dérivée γ' de γ par c est nulle ssi γ échange des prémisses en passant par c. Si c est une coupure* **affaiblissement** *dont la boîte prémisse est b alors γ' est nulle ssi γ a une sous-trace $\alpha^*\delta\beta$ où α et β sont des portes auxiliaires de b et δ est une trace entièrement contenue dans b.*

Preuve : soit c une coupure stricte : il y a deux cas suivant que c est multiplicative ou exponentielle ; on utilise les notations de la définition :

- c est multiplicative : comme la dérivation procède par effacement des arêtes β_1 et β_2 de γ, γ' est obtenue en remplaçant les sous-traces de la forme : $\alpha_1^i\beta_1{\beta_2}^*{\alpha_2^j}^*$ par $\alpha_1^i{\alpha_2^j}^*$. Donc γ' est nulle — i.e., discontinue — ssi $i \neq j$ — i.e., γ échange des prémisses en passant par c.
- c est exponentielle alors, en convenant que δ représente une trace entièrement incluse dans la boîte b, les sous-traces de γ qui sont modifiées par la dérivation sont de quatre types :
  1. *sous-traces de type* **pp** : elles ont la forme $\sigma = \alpha_i\alpha\beta^*\delta\beta\alpha^*\alpha_j^*$. Elles sont remplacées par $\sigma' = \alpha_i(\sum\delta^k)\alpha_j^*$. Or si $i \neq k$ alors $\alpha_i\delta^k$ est la trace nulle puisque δ^k est dans la $k$ième copie de b et que α_i est coupée avec la $i$ème copie de b. Donc $\sigma' = \alpha_i\delta^i\alpha_j$ et σ' est continue ssi $i = j$, car δ^i est la $i$ème copie de δ et donc se termine en la copie c_i de la coupure c dont l'autre prémisse est α_i. Mais ceci équivaut à dire que σ n'échange pas les prémisses de c.
  2. *sous-traces de type* **pa** : elles ont la forme $\sigma = \alpha_i\alpha\beta^*\delta\beta'$ où β' est une porte auxiliaire de b. Elles sont remplacées par $\sigma' = \alpha_i(\sum\delta^j\beta'^j) = \alpha_i\delta^i\beta^i$ ce qui n'introduit aucune discontinuité.
  3. *sous-traces de type* **ap** : ce sont les duales des précédentes, le raisonnement est identique.
  4. *sous-traces de type* **aa** : elles ont la forme ${\beta_1}^*\delta\beta_2$ où β_1 et β_2 sont des portes auxiliaires de b. Elles sont remplacées par : $\sum{\beta_1^i}^*\delta^i\beta_2^i$. Elles n'introduisent de discontinuité que si la somme est nulle i.e., c est une coupure **affaiblissement**. Remarquons que c'est le seul cas où la dérivée d'une trace est une somme de plus d'un terme.

Si c est une coupure **affaiblissement** alors la sous-trace $\alpha^*\delta\beta$ de γ se dérive en $\alpha^*\beta$ puisque la réduction de c efface la boîte b. Mais comme α et β sont des conclusions de lien **affaiblissement** dans R' c'est une trace discontinue donc γ' est nulle. Réciproquement si γ' est nulle c'est que γ a une sous-trace $\alpha^*\delta\beta$ se dérivant en $\alpha^*\beta$ discontinue. Mais la seule possibilité pour ce faire est que α et β soient des portes auxiliaires de b et δ une trace contenue dans b. •

Définition 1.2.2 *Soient γ une trace dans un réseau R et c une coupure exponentielle, de boîte associée b, traversée par γ. On dit que c est* spéciale *pour γ si γ est de la forme : $\gamma = \gamma_1\delta_1\delta\delta_2\gamma_2$ où γ_1 et γ_2, si elles sont non vides, sont en dehors de la boîte b et ne passent par aucune coupure ; δ_1 et δ_2, si elles sont non vides, sont respectivement des traces de type* **ap** *et* **pa** *dans b, et δ n'a que des sous-traces de type* **pp** *dans b. On dit alors que γ est* initialement (resp. finalement) de type **ap** (resp. **pa**) dans *b si δ_1 (resp. δ_2) est non vide.*
On appelle réduction spéciale *de γ une réduction qui choisit à chaque étape ; soit une coupure multiplicative s'il y en a, soit une coupure exponentielle spéciale pour la dérivée de γ.*

Lemme 1.2.2 (Coupures spéciales) *Si γ est une trace dans un réseau R ne traversant que des coupures exponentielles alors l'une d'elles est spéciale pour γ.*
La dérivée de γ par une coupure spéciale est soit la trace nulle, soit une trace dans le réduit de R.

Preuve : par induction sur la taille de R.

- Si R a des liens **par** ou **pourquoi pas** terminaux alors soit R' le réseau obtenu en les supprimant et γ' la suite d'arêtes déduite en effaçant de γ les arêtes qui ont disparu dans R'. Comme on n'a supprimé que des liens terminaux on a $\gamma = \gamma_1\gamma'\gamma_2$ et γ' est une trace de R' ne passant que par des coupures exponentielles donc par hypothèse d'induction possédant une coupure spéciale pour γ'. Mais alors celle-ci est aussi une coupure spéciale pour γ dans R.

- Si R a des coupures multiplicatives alors γ ne passe par aucune d'entre elles puisque γ ne passe que par des coupures exponentielles. Soit m l'une d'elles et R' le réduit de R par m. Alors R' est plus petit que R et γ est une trace non nulle de R', donc par hypothèse d'induction traverse une coupure exponentielle s spéciale pour γ dans R' donc spéciale pour γ dans R.

- Si R n'a plus ni liens **par** ou **pourquoi pas** terminaux, ni coupures multiplicatives mais des coupures externes alors on a vu que l'une d'elle, soit s, est spéciale, donc que $R = R_1 \mathbf{cut} !R_2$.

  Si γ ne passe pas par c alors γ est entièrement contenue soit dans R_1 soit dans R_2. Par induction elle passe par une coupure spéciale pour elle même s. Mais alors s est spéciale pour γ dans R.

  Si γ passe par c alors aucune sous-trace de γ ne peut être de type **aa**. En effet c est spéciale donc les portes auxiliaires de $!R_2$ sont des conclusions du réseau. Si γ commence dans b, soit $\gamma = \delta_1\gamma'$ où δ_1 la est sous-trace maximale de γ contenue dans b, alors γ' ne contient aucune sous-trace de type **ap** dans b, sinon une telle sous-trace commencerait en une porte auxiliaire de b, mais celles-ci sont des conclusions de R. Donc γ est initialement de type **ap**. Le même raisonnement s'applique si γ se termine dans b.

- si R n'a que des liens **tenseur** externes alors on a vu que $R = !R_1 \otimes R_2$. Donc γ est entièrement incluse soit dans R_1 soit dans R_2, mais une coupure spéciale pour γ dans R_i est spéciale pour γ dans R.

Le fait que la dérivée de γ par une coupure spéciale pour γ est une somme d'au plus un terme est une conséquence de la preuve précédente. En effet on a vu que seules les traces de type **aa** se dérivent en des sommes de plus d'un terme ; comme s est spéciale pour γ, celle-ci ne possède pas de telles sous-traces i.e., toutes ses sous-traces se dérivent en une somme d'au plus un terme, donc il en est de même pour γ. $\bullet$

Lemme 1.2.3 (Intégration des traces) *Soient R un réseau, R' un réduit strict de R et γ' une trace de R'. Il existe une unique trace γ de R tel que γ' soit l'un des termes de la somme dérivée de γ dans R'.*

Remarque : γ est l'intégrale de γ' le long de la réduction de R à R'

Preuve : on peut se restreindre au cas où R' est obtenu par la réduction d'une coupure c de R, le cas général s'en déduit par induction sur la longueur de la réduction. Notons $c_1, \ldots, c_n$ les coupures créées par la réduction de c, α_i et β_i les prémisses de c_i dans R', α et β les prémisses de c dans R et, dans le cas où c est une coupure exponentielle, ι^j les arêtes de R' qui sont des copies d'arêtes ι de R. On construit γ en remplaçant les sous-traces de γ' de la forme : $\alpha_i {\beta_i}^*$ par : $\alpha_i \alpha \beta^* {\beta_i}^*$ puis en remplaçant toutes les arêtes ι^j par ι. On détermine ainsi une trace γ dans R qui par construction est celle recherchée. $\bullet$

Lemme 1.2.4 (Commutation des réductions affaiblissement) *Soient R un réseau possèdant une coupure* **affaiblissement** *a et une coupure stricte c non contenue dans la boîte prémisse de a, et γ une somme de traces de R. Soit R' obtenu en réduisant d'abord a puis c et γ' la dérivée de γ dans R'. Alors γ' est aussi la dérivée de γ si on réduit d'abord c puis les coupures* **affaiblissement** *provenant de a.*

Remarque : ce dernier lemme étend aux réductions de traces les résultats de la partie 4.3.

Preuve : soit b la boîte prémisse de a. On obtient γ' en effaçant les prémisses de a et les arêtes contenues dans b puis en effaçant les prémisses de c. Dans l'autre cas on dérive γ en effaçant d'abord les prémisses de c, puis celles des éventuelles copies de a ainsi que celles contenues dans (les copies de) b obtenant γ'' dans R'. Si γ' n'est pas nulle alors c'est que γ n'échange pas de prémisses de c et qu'il n'a pas de sous-trace de type **aa** dans b. Mais alors γ'' n'est pas nulle non plus et est clairement égale à γ'. Si γ échange les prémisses de c alors γ' et γ'' sont nulles ; de même si γ a une sous-trace de type **aa** dans b. $\bullet$

Chapitre 2

Confluence des réductions

2.1 Développements finis

Définition 2.1.1 *On dit qu'une trace est* multiplicative *si elle ne passe que par des liens multiplicatifs. Soit γ une somme de traces dans un réseau R. On note γ° la somme des sous-traces de γ multiplicatives et passant par une coupure. On appelle* développement fini de γ *une réduction de γ°.*

Remarque : un développement fini de γ peut réduire des coupures exponentielles. Si γ passe par une coupure exponentielle c alors la sous-trace de γ ne passant que par c est multiplicative, donc la réduction de c est un développement fini de γ.

Remarque : un développement fini d'une somme de traces γ n'est pas en général une réduction de γ. Supposons par exemple que γ soit une trace qui échange les prémisses d'une coupure c et qui passe par une autre coupure c'. Alors la réduction de c puis de c' est un développement fini de γ car γ° contient les traces ne passant que par c et c', mais pas une réduction de γ puisque la dérivée de γ par c est nulle.

Remarque : à chaque coupure c de R on peut associer la trace $\overline{c}$ qui ne passe que par le lien c. Soit γ la somme des $\overline{c}$ sur toutes les coupures de R. Alors toute réduction de γ est un dévellopement fini de γ puisque chaque $\overline{c}$ est une trace multiplicative. Mais réduire γ correspond à réduire les coupures de R sans réduire les coupures créées soit exactement à opérer à un développement fini de R. Autrement dit, les développements finis de R sont des cas particuliers de développements finis de traces.

2.1.1 Confluence locale

Proposition 2.1.1 (Confluence locale des développements finis) *Soient R un réseau possédant deux coupures strictes c_1 et c_2, et γ une somme de traces de R passant par c_1 et c_2. Soient R_1 et R_2 les réduits de R par respectivement c_1 et c_2, γ_1 et γ_2 les réduites de γ dans R_1 et R_2. Alors R_1 et R_2 se réduisent en un même réseau R_3, différent de R_1 et R_2 et vérifiant :*

1. *γ_1 et γ_2 se dérivent en une même somme de trace γ_3 dans R_3 ;*

2. *les réductions de R à R_3 en passant par R_1 ou R_2 sont des développements finis de γ ;*

Preuve : Considérons les différents types possibles de c_1 et c_2 :

1. c_1 et c_2 sont des coupures multiplicatives: alors γ_1 est γ dans laquelle on a supprimé les arêtes prémisses de c_1, soit α_1 et β_1, et γ_2 est γ dans laquelle on a supprimé les prémisses de c_2, soit α_2 et β_2. On obtient R_3 en éliminant c_2 dans R_1, ou c_1 dans R_2. Si ni γ_1 ni γ_2 ne sont nulles alors la dérivée de γ_1 est obtenue en effaçant α_2 et β_2 et celle de γ_2 en effaçant α_1 et β_1 ce qui dans les deux cas produit bien la même somme de traces γ_3.

   Si γ_1 est nulle alors tous les termes de γ échangent les prémisses de c_1. Mais alors les termes de γ_2 aussi échangent les prémisses de c_1 et donc sa dérivée γ_3 est nulle.

2. c_2 est une coupure exponentielle dont la boîte prémisse b ne contient pas c_1, alors le raisonnement est le même en remplaçant les arêtes α contenues dans b par la somme $\alpha^1 + \ldots + \alpha^n$ de leurs copies.

3. c_2 est une coupure exponentielle dont la boîte prémisse b contient c_1, alors γ_2 est γ dans laquelle on a supprimé les prémisses α_2 et β_2 de c_2 et copié les arêtes α de b, et en particulier α_1 et β_1, en des sommes $\alpha^1 + \ldots + \alpha^n$, ($n \neq 0$ puisque c_2 est stricte). R_3 est obtenu de R_1 en réduisant c_2 et de R_2 en réduisant les n copies de c_1.

   Si ni γ_1 ni γ_2 ne sont nulles, la dérivée de γ_1 est obtenue en supprimant α_2 et β_2, et en remplaçant chaque arête α de b par $\alpha^1 + \ldots + \alpha^n$; la dérivée de γ_2 est obtenue en supprimant les copies α_1^i et β_1^i de γ_2. Dans les deux cas on retombe sur la même somme de traces γ_3.

   Si γ_2 est nulle c'est que tous les termes de γ échange les prémisses de c_2, mais alors ceux de γ_1 échangent les prémisses de c_2 aussi et donc sa dérivée γ_3 est nulle.

   Si γ_1 est nulle alors γ échange les prémisses de c_1. Donc γ_2 échange les copies des prémisses des copies de c_1. Donc chaque terme de la somme γ_2 va se dériver en 0 lorsque l'on réduit la copie correspondante de c_1. Au total la dérivée de γ_2 dans R_3 est bien nulle puisqu'on obtient R_3 en réduisant toutes les copies de c_1 dans R_2. Remarquons que ce raisonnement ne tient plus s'il n'y a pas de copie de c_1 i.e., si c_2 est une coupure **affaiblissement**: R_3 est alors égal à R_2 et γ_2 n'est pas nulle alors que γ_1 l'est et donc sa dérivée dans $R_3 = R_2$ aussi.

4. c_1 et c_2 sont des coupures exponentielles ; alors tout se passe comme dans le cas où c_1 est multiplicative, à ceci près que l'on duplique le contenu de sa boîte.

Dans tous les cas la confluence s'obtient en ne réduisant que (les copies de) c_1 et c_2 dans R_2 et R_1. Ces réductions sont donc bien des développements finis de γ. •

2.1.2 Terminaison des développements finis

Définition 2.1.2 *On dira qu'une trace multiplicative δ* connecte *une arête α de type $?I$ à une arête β de type $!O$ si :*

1. *l'arête initiale de δ est α ; l'arête finale de δ est β^* ;*

2. *δ passe exactement par le même nombre de liens* **par** *et* **tenseur** *;*

3. *toute sous-trace de δ commencant en α passe par strictement plus de liens* **par** *que de liens* **tenseur**.

REMARQUE : la dernière condition est équivalente à : "toute sous-trace de δ d'arête finale β^* passe par strictement plus de liens **tenseur** que de liens **par**".

REMARQUE : une trace multiplicative connectant deux arêtes α et β ne traverse aucune autre arête exponentielle que α et β.

Lemme 2.1.2 *Soit R un réseau possèdant une coupure stricte c, R' le réduit de R par c, α et β deux arêtes de R qui ne sont pas toutes deux prémisses de c. Si α et β sont connectées par une trace multiplicative δ dans R alors chaque copie de α et β est connectée dans R' par un terme de la dérivée δ' de δ par c.*

Preuve : Si δ ne passe pas par c alors la réduction de c peut éventuellement dupliquer δ mais les copies de la dérivée continuent à connecter les copies de α et β dans R'.
Si δ passe par c alors c est une coupure multiplicative. En effet δ est multiplicative donc si c est une coupure exponentielle i.e., ses deux prémisses sont des arêtes exponentielles, δ est la trace ne passant que par c. Mais dans ce cas $\delta = (\alpha\beta^*)$ et donc α et β sont les prémisses de c.
Si c est une coupure **axiome**, alors la réduction de c ne change pas la structure de δ qui traverse les mêmes liens **par** et **tenseur** dans le même ordre ; donc δ' connecte toujours α et β.
Si c est une coupure $\otimes/\wp$ alors δ ne peut échanger les prémisses de c. Sinon l'une de ces prémisses est exponentielle donc égale à α ou β, soit par exemple α ; donc δ commence par croiser un lien **par**, puis la coupure c, puis un lien **tenseur** en remontant par la prémisse I. Mais ceci est une sous-trace de δ commencant en α et possèdant un nombre égal de **par** et de **tenseur**.
Si δ n'échange pas les prémisses de c, δ' traverse exactement un lien **par** et un lien **tenseur** consécutifs de moins que δ donc connecte α et β dans R'. •

REMARQUE : deux arêtes connectées par γ sont au même niveau i.e., appartiennent aux mêmes boîtes.

Proposition 2.1.3 (Terminaison des développements finis) *Soit γ une somme de traces dans un réseau R ; les développements finis de R relativement à γ sont de longueurs bornées.*

Preuve : La démonstration est une adaptation de celle de terminaison des développements finis des lambda-termes ([Bar89]).
On dira qu'une arête est *duplicante* si elle est prémisse d'un lien **pourquoi pas** ou si c'est une conclusion portant la formule I du réseau. Soient R' un réduit de R obtenu par développement fini de γ et $\gamma^{\circ\prime}$ la dérivée de γ° dans R'. On appellera *coupures actives* les coupures par lesquelles passe un terme de $\gamma^{\circ\prime}$. On dira qu'un lien **pourquoi pas** l de R' est *connecté par* γ à une boîte b si il existe une sous-trace de $\gamma^{\circ\prime}$ qui connecte la conclusion de l à la porte principale de b. Dans ce cas on dira aussi des prémisses de l qu'elles sont connectées par γ à la boîte b. Remarquons qu'une arête duplicante est connectée par γ à au plus une boîte.
Soit R' un réduit de R par développement fini de γ. A chaque arête duplicante α de R' on va associer une mesure entière $m_{R'}(\alpha)$. La mesure $m_{R'}(b)$ d'une boîte b de R' sera la somme sur toutes les arêtes duplicantes contenues dans b et les portes auxiliaires duplicantes de b de leur mesure, et la mesure $m_{R'}(R')$ de R' sera la somme sur toutes les arêtes duplicantes de R' de leur mesure. La preuve est en deux parties :

1. On suppose que $m_{R'}$ vérifie l'hypothèse $H(R')$ suivante : *pour toute arête duplicante α dans R', si α est connectée par γ à une boîte b alors : $m_{R'}(\alpha) > m_{R'}(b)$.*

Soit c une coupure active de R' et soit R'' le réduit de R par c. On définit la mesure $m_{R''}$ par : toute arête duplicante de R'' a la même mesure que l'arête dont elle provient dans R'. Remarquons que si c est exponentielle entre un lien **pourquoi pas** l et une boîte b, alors les prémisses de l, qui sont des arêtes de R'', ne sont plus duplicantes donc n'ont plus de mesure dans R''. On montre alors que :

- $m_{R''}$ vérifie l'hypothèse $H(R'')$;
- si c est exponentielle alors $m_{R''}(R'') < m_{R'}(R')$; si c est multiplicative alors $m_{R''}(R'') = m_{R'}(R')$.

2. On montre ensuite qu'il existe une mesure m_R des arêtes de R vérifiant $H(R)$.

Muni de 1. et 2. on raisonne par induction sur $m_R(R)$ puis sur la taille de R et on déduit immédiatement la terminaison puisque l'élimination des coupures multiplicatives diminue la taille du réseau.

1. Si c est multiplicative alors la réduction de c n'élimine aucune arête duplicante. De plus si l est un lien **pourquoi pas** connecté par γ avec une boîte b avant réduction alors, d'après le lemme ci-dessus, c'est encore le cas après. La mesure des prémisses de l n'a pas changé, non plus que celle de b. Donc on a bien $m_{R''}(R'') = m_{R'}(R')$ et $m_{R''}$ vérifie l'hypothèse $H(R'')$.

   Si c est une coupure exponentielle active entre un lien **pourquoi pas** l et une boîte b de portes auxiliaires $\beta_1, \ldots, \beta_n$ alors notons $\alpha_1, \ldots, \alpha_n$ les prémisses de l qui sont donc des arêtes duplicantes de R' connectées à b par γ donc vérifiant $m_{R'}(\alpha_i) > m_{R'}(b)$. Après réduction de c la situation dans R'' est la suivante :

   - les α_i ne sont plus duplicantes. Par contre les n copies de b ont toutes la même mesure que b. Rien n'ayant changé par ailleurs la mesure de R'' est donc : $m_{R''}(R'') = m_{R'}(R') - m_{R'}(b) - \sum_{1 \leq i \leq n} m_{R'}(\alpha_i) + n m_{R'}(b)$. Mais comme $m_{R'}$ vérifie l'hypothèse $H(R')$ on a $m_{R''}(R'') < m_{R'}(R')$.
   - Si α_i est porte auxiliaire d'une boîte b_i dans R' alors dans R'' la mesure de b_i est : $m_{R'}(b_i) - m_{R'}(\alpha_i) + m_{R'}(b)$ puisque on a copié les arêtes duplicantes de b dans b_i sans changer leur mesure et que α_i n'es plus duplicante dans R''. Donc la mesure de b_i a strictement diminué par l'hypothèse $H(R')$. Par conséquent si la porte principale de b_i est connectée par γ avec un lien **pourquoi pas** dans R'', donc aussi dans R', alors en notant α une arête prémisse du **pourquoi pas** on a bien $m_{R''}(\alpha) > m_{R'}(b_i) > m_{R''}(b_i)$. Remarquons que l'on a implicitement supposé que α n'est pas une porte auxiliaire de b, non plus que de b_i ce qui est correct puisque le réseau l'est.
   - Si β_j^i est connectée à b' dans R'' alors β_j est connectée à b' dans R' donc $m_{R''}(\beta_j^i) = m_{R'}(\beta_j) > m_{R'}(b') = m_{R''}(b')$
   - Si α est une arête duplicante de R' autre que celles considérées ci-dessus alors dans R'' les éventuelles copies de α ainsi que celles de b ont la même mesure et sont deux à deux connectées. Donc $m_{R''}(\alpha^i) > m_{R''}(b''^i)$. Ici on se sert du fait que si α'' est connectée par γ avec b'' alors α et b'' sont au même niveau.

2. On appelle arêtes *actives* de R les arêtes duplicantes qui sont connectées par γ à une boîte. Les arêtes duplicantes non actives seront les arêtes *libres*. On montre par induction sur R que pour

tout choix de mesure des arêtes libres de R, on peut trouver une mesure des arêtes actives telle que le tout vérifie l'hypothèse $H(R)$.

On appelle *γ-chaîne* une suite (b_i) de boîtes de R vérifiant : b_i a une porte auxiliaire β_i connectée à la boîte b_{i+1} par γ.

Une γ-chaîne C ne peut être cyclique i.e., contenir deux fois une même boîte. Soit R' un réduit multiplicatif de R dans lequel il n'y a plus de coupures multiplicatives (un tel réseau existe puisque les coupures multiplicatives font décroître la taille du réseau) ; alors d'après le lemme 2.1.2, C est encore une γ-chaîne de R'. Mais comme R' n'a plus de coupures multiplicatives on en déduit que toute boîte b_{i+1} de C a sa porte principale coupée par une coupure exponentielle sur le lien l_i. Donc C est une chaîne de R', au sens défini pour le théorème de séquentialisation, et C ne peut être cyclique par le lemme 3.3.1.

On se donne une mesure arbitraire m des arêtes libres de R. On dira qu'une boîte b est *mesurée* si :

(a) les arêtes duplicantes et les portes auxiliaires de b ont une mesure m_b vérifiant $H(b)$;
(b) si α est une arête duplicante de b i.e., α est dans b ou α est une porte auxiliaire de b, et si α est libre dans R alors $m_b(\alpha) = m(\alpha)$;
(c) si β est une porte auxiliaire de b connectée par γ à une boîte b' alors b' est mesurée et $m_b(\beta) > m_{b'}(b)$.

Si toutes les boîtes externes de R sont mesurées alors on définit m_R par :

- si α est une arête duplicante d'une boîte externe b alors $m_R(\alpha) = m_b(\alpha)$;
- si α est une arête externe libre alors $m_R(\alpha) = m(\alpha)$;
- si α est une arête externe active connectée par γ à une boîte b mais n'est porte auxiliaire d'aucune boîte alors $m_R(\alpha) = m_b(b) + 1$

Clairement m_R vérifie $H(R)$ et est donc la mesure cherchée.

Reste à mesurer les boîtes externes. Soit b une boîte externe non mesurée. On dira qu'une γ-chaîne C est *issue* de b si b est le premier élément de C. On raisonne par induction sur la somme $S(b)$ des longueurs des γ-chaînes issue de b. Cette somme est finie car les γ-chaînes ne sont pas cycliques donc de longueurs bornées. Si $S(b)$ est égale à 1 alors les portes auxiliaires de b sont libres dans R et on obtient la mesure m_b par induction sur l'intérieur de b. Sinon certaines portes auxiliaires de b sont connectées à des boîtes b'. Mais pour chaque telle boîte, $S(b') < S(b)$ puisque les γ-chaînes issues de b' sont contenues dans celles issues de b. Donc par induction sur $S(b')$ on sait construire la mesure $m_{b'}$. On donne alors à chaque porte auxiliaire β de b connectée à une boîte b' (rappellons qu'au plus une boîte est connectée à β) la mesure $m_{b'}(b') + 1$. Comme β est une arête libre de l'intérieur de b, on sait alors construire la mesure m_b par induction sur b. •

Corollaire 2.1.4 (Confluence des développements finis) *Soient R un réseau, γ une somme de traces dans R, R_1 et R_2 deux développements finis stricts relativement à γ de R, et γ_1, γ_2 les dérivées de γ dans R_1 et R_2.*
Il existe un réseau R_3 tel que :

- *R_1 et R_2 se réduisent en R_3. De plus les réductions de R à R_3 passant par R_1 ou R_2 sont des développements finis stricts de R relativement à γ :*

– *γ_1 et γ_2 se dérivent en la même somme de traces γ_3 dans R_3 ;*

Preuve : c'est une conséquence immédiate de la confluence locale et de la terminaison des développements finis. •

2.2 Confluence des réductions de sommes de traces

Définition 2.2.1 *Soit γ une somme de traces dans un réseau R. On appelle* saturée *de γ la somme $\overline{\gamma}$ des sous-traces de γ. On dira que γ est* saturée *si toute sous-trace de γ est un terme de γ.*

Lemme 2.2.1 *Si γ est une somme saturée de traces alors tout développement fini de γ est une réduction de γ.*
Soient γ' dans R' et γ'' dans R'' telles que : γ se réduit en γ', γ' en γ'' et γ' et γ'' sont obtenues par des développements finis de γ. Alors γ'' est un développement fini de γ'.

Preuve : en effet si γ est saturée alors tous les termes de γ° sont des termes de γ. Comme un développement fini de γ est une réduction de γ°, c'est bien aussi une réduction de γ.
Il faut voir que les termes de la dérivée $\gamma^{\circ\prime}$ de γ° dans R' sont des termes de γ'°. Mais cela est clair puisque γ est saturée et donc γ' contient tous les termes de $\gamma^{\circ\prime}$. •

Remarque : si γ est une trace et que γ' est nulle, alors γ'', nulle elle aussi, n'est pas un développement fini de γ'.

Lemme 2.2.2 *Soient γ une somme saturée de traces et γ' une réduite de γ ; alors γ' est une somme saturée de traces.*

Preuve : soit δ' une sous-trace de γ'. Alors un terme de γ' s'écrit : $\gamma_i = \delta'_1\delta\delta'_2$. Mais comme γ' est la dérivée de γ, γ'_i est la dérivée de l'un des termes $\gamma_i = \delta_1\delta\delta_2$ de γ et δ_i se dérive en δ'_i, δ se dérive en δ'. Donc δ est une sous-trace de γ et comme celle-ci est saturée, δ est un terme de γ. Donc δ' est un terme de γ'. •

Théorème 2.2.3 (Confluence des réductions de traces) *Soient R un réseau, γ une somme de traces de R, R_1 et R_2 deux réduits stricts de R relativement à γ, γ_1 et γ_2 les réduites de γ dans R_1 et R_2. Alors R_1 et R_2 se réduisent strictement en un même réseau R_3 tel que :*

– *γ_1 et γ_2 se dérivent en une même somme de traces γ_3 de R_3 ;*

– *les réductions de R à R_3 en passant par R_1 ou R_2 sont des réductions de $\overline{\gamma}$;*

Preuve : notons $\overline{\gamma}_1$ (resp. $\overline{\gamma}_2$) la réduite de $\overline{\gamma}$ dans R_1 (resp. R_2). Clairement si il existe des réductions de $\overline{\gamma}_1$ et $\overline{\gamma}_2$ en une même somme de trace $\overline{\gamma}_3$ dans un réseau R_3, alors les dérivées dans R_3 de γ_1 et γ_2 sont égales à une même somme de traces γ_3. Donc il suffit de montrer le théorème pour des sommes saturées γ_1 et γ_2.
Dans un premier temps on se place dans le cas où γ_2 est obtenue par développement fini de γ. On note $(\gamma_1^i)_{0\leq i\leq n}$ les réduites successives de γ jusqu'à γ_1, (soit $\gamma = \gamma_1^0$ et $\gamma_1 = \gamma_1^n$). On raisonne par induction sur n.
Si $n = 0$ i.e., $\gamma_1 = \gamma$, alors $R_3 = R_2$, $\gamma_3 = \gamma_2$, est la solution cherchée.

Si $n \geq 1$ alors on applique le théorème de confluence des développements finis à γ_1^1 et γ_2 ce qui donne une somme de traces γ_3^1 dans un réseau R_3^1. Comme γ_3^1 est obtenue par développement fini de γ qui est saturée, d'après le lemme 2.2.1 les deux réductions menant à R_3^1 sont des réductions de γ. D'après le lemme 2.2.1, γ_3^1 est un développement fini de γ_1^1 qui, d'après le lemme 2.2.2, est saturée ; donc on peut appliquer l'hypothèse d'induction à γ_1^1 (puisque la réduction de γ_1^1 à $\gamma_1^n = \gamma_1$ est de longueur $n-1$) ce qui donne une réduite γ_3^n de γ_1^n et γ_3^1. Mais γ_3^n est donc une réduite de γ_1 et de γ_2.
On montre ensuite le résultat géneral i.e., quand γ_2 est obtenue par une réduction quelconque de γ de la même manière i.e., par induction sur la longueur de la réduction de γ à γ_2. •

REMARQUE : la preuve de la confluence de la réduction des traces est essentiellement la même que celle de confluence de la beta-réduction. La différence provient de la possibilité d'annuler une trace par réduction, alors qu'il n'existe pas de notion de "lambda-terme nul". C'est pour cela que l'on est obligé d'introduire les sommes saturées.

Définition 2.2.2 *Soit γ une trace dans un réseau R. On appelle* forme normale *de γ un réduite stricte de γ ne passant par aucune coupure stricte.*

REMARQUE : toute réduction d'un réseau R ne mène pas à une forme normale d'une trace γ de R. En effet une forme normale de γ doit être obtenue par réduction de γ.

Corollaire 2.2.4 (Unicité de la forme normale) *Soient γ une somme de traces, γ_0 et ${\gamma'}_0$ deux formes normales de γ. Alors γ_0 est nulle ssi ${\gamma'}_0$ l'est. Si de plus γ est saturée alors $\gamma_0 = {\gamma'}_0$.*

Preuve : si on réduit une coupure stricte c alors la dérivée $\gamma_0{}'$ de γ_0 par c est normale. De plus, comme γ_0 ne passe pas par c, donc ne peut échanger les prémisses de c, $\gamma_0{}'$ est nulle ssi γ_0 l'est. Donc, par induction sur la longueur des réductions, toute dérivée de γ_0 est une somme normale de traces, nulle ssi γ_0 l'est. Comme γ_0 et ${\gamma'}_0$ sont des réduites de γ, par le théorème de confluence on peut les dériver en ${\gamma''}_0$ qui est donc nulle ssi les deux le sont.
Si γ est saturée alors γ_0 et ${\gamma'}_0$ se *réduisent* en ${\gamma''}_0$. Mais comme elles sont normales, donc n'admettent aucune réduction, elles sont égales. •

Chapitre 3

Réduction de traces

3.1 Forte normalisation

Théorème 3.1.1 (Terminaison des dérivations) *Toute réduction d'une somme de traces γ dans un réseau R termine.*

REMARQUE : autrement dit les traces offrent une "mesure " des réseaux qui diminue par réduction. Un exemple d'une autre telle mesure est donné par la norme des termes fortement normalisables.

Preuve : en vertu du lemme de commutation des réductions **affaiblissement**, on peut se limiter à des réductions strictes. De plus on peut se restreindre à montrer la terminaisons pour les sommes saturées de traces puisque si γ a une réduction infinie alors $\overline{\gamma}$ a une réduction infinie.
On appelle *forme pré-normale d'une somme de traces* γ une réduite non nulle de γ n'ayant aucune réduite stricte non nulle. Donc si γ' est une forme pré-normale de γ alors pour toute coupure stricte c par laquelle passe γ', tous les termes de γ' échangent les prémisses de c. On montre que :

1. il existe une réduction stricte de γ menant à une forme pré-normale de γ ;

2. si une somme de traces γ a une réduction stricte infinie alors toute réduite stricte non nulle à une étape de γ a une réduction stricte infinie.

Une forme pré-normale ne peut se réduire au plus qu'une fois donc n'a pas de réduction infinie. Donc par 2., aucune réduite stricte de γ ne peut avoir de réduction infinie.

1. On raisonne par induction sur la longueur du plus long terme de γ puis sur la taille du réseau. On se sert du fait qu'une réduction d'un réseau n'augmente jamais la longueur d'une trace, et la diminue même dans les bons cas.

   Si R a des liens **par** ou **pourquoi pas** terminaux, alors soit R' le réseau obtenu en les enlevant et γ' la somme de traces déduite de γ dans R'. Comme le plus long terme de γ' n'est pas plus long que celui de γ, par induction sur R' on a une réduction de γ' menant à une forme pré-normale de γ'. Mais cette réduction en est aussi une de γ et mène à une forme pré-normale de γ.

   Si R a des coupures multiplicatives, on les réduit ce qui mène à un réseau R' plus petit. Les termes de la dérivée γ' de γ dans R' ne sont pas plus longs que ceux de γ donc on peut appliquer l'hypothèse d'induction sur R'. Donc γ' a une forme pré-normale. Mais comme γ' est une dérivée de γ uniquement par des coupures multiplicatives, on en déduit immédiatement la même chose pour γ.

Si R n'a plus ni liens **par** et **pourquoi pas** terminaux, ni coupures multiplicatives alors on sait que R s'écrit $R_1 \otimes !R_2$ ou $R_1 \mathbf{cut} !R_2$. Dans le premier cas γ se décompose en deux sommes γ_1 dans R_1 et γ_2 dans R_2. Par induction sur R_1 et R_2 on obtient des réductions de γ_1 et γ_2 menant à des formes prénormales γ'_1 et γ'_2 dans des réseaux R'_1 et R'_2. Mais ces deux réductions forment une réduction de γ menant à une forme pré-normale $\gamma' = \gamma'_1 + \gamma'_2$ dans le réseau $R'_1 \otimes R'_2$.

Enfin si $R = R_1 \mathbf{cut} R_2$ pour une coupure c, alors γ se décompose en trois sommes γ_1 dans R_1, γ_2 dans R_2 et γ_3 qui est la somme des termes de γ passant par c. Par induction sur R_1 et R_2 on a des réductions de γ_1 et γ_2 menant à des formes prénormales γ'_1 dans R'_1 de γ_1 et γ'_2 dans R'_2 de γ_2. Donc, si γ'_3 est la dérivée de γ_3 par ces deux réductions, la réduite de γ est $\gamma'_1 + \gamma'_2 + \gamma'_3$ dans le réseau $R'_1 \mathbf{cut} ! R'_2$. Or γ'_3 est une somme de traces passant par c, puisque c n'a pas été réduite. Donc si on réduit c, la dérivée γ''_3 de γ'_3 est un somme de traces toutes plus courtes que le plus long terme de γ'_3. On obtient alors le résultat par induction sur le plus long terme de γ''_3 qui est strictement plus court que celui de γ'_3 lui-même plus court que celui de γ.

2. Soit $\gamma_1, \ldots, \gamma_k, \ldots$ la suite infinie des réduites de γ et γ' une réduite stricte à une étape de γ. On va fabriquer une réduction infinie $\gamma'_0 = \gamma', \gamma'_1, \ldots, \gamma'_k, \ldots$ de γ' telle que γ'_k soit un développement fini de γ_k. Comme γ est saturée, d'après le lemme 2.2.2, toute réduite de γ est saturée. On produit γ'_{k+1} par une application du théorème de confluence des développements à γ_k : donc γ'_{k+1} est la dérivée commune à γ_{k+1} et γ'_k. Mais comme les sommes sont saturées, γ'_{k+1} est une réduite donc un développement fini de γ'_k et de γ_{k+1}. •

3.2 Traces dérivables

Définition 3.2.1 *Soient R un réseau et γ une trace dans R. On dira que γ est* dérivable *si pour toute réduction stricte de γ en une somme de traces γ', γ' est non nulle.*

Remarque : une trace est dérivable ssi l'une de ses réduites strictes l'est. Sinon on contredit l'unicité de la forme normale.

Théorème 3.2.1 (Fidélité) *Les propositions suivantes sont équivalentes :*

1. *R est un réseau fortement normalisable ;*
2. *R n'a qu'un nombre fini de traces dérivables ;*
3. *il existe N tel que toute trace dérivable de R soit de longueur inférieure à N.*

Preuve : l'équivalence entre les deux dernières est immédiate puisque les traces sont des listes d'arêtes occurant dans R, donc des listes d'objets pris dans un ensemble fini.
Comme on a vu que la forte normalisation et la forte normalisation pour la réduction stricte sont équivalentes pour les réseaux, on se restreint dans la suite à des réductions strictes.
Le fait que 2 entraîne 1 est une conséquence immédiate du théorème de terminaisons des réductions de sommes de traces. Soit γ la somme de toutes les traces dérivables de R. Si γ est en forme normale

alors R est en forme normale ; sinon R a une coupure et il existe une trace dérivable, celle qui ne passe que par cette coupure, non normale. Si γ n'est pas normale, on raisonne par induction sur sa norme i.e., la longueur de sa plus longue réduction. Toute réduction stricte de R est une réduction stricte de γ donc par induction sur la norme de γ, le réduit est fortement normalisable.
On montre que 1 entraine 3 par induction sur la norme de R. Si R a une coupure stricte c de prémisses α_1 et α_2, alors soient R' le réduit de R par c et γ une trace dérivable de R. Si γ ne passe pas par c alors on a déja vu que la dérivée γ' de γ par c est une somme de traces toutes de même longueur que γ donc celle-ci est bornée par hypothèse d'induction sur la norme de R'.
Si γ passe n fois par c, soit $l(\gamma)$ la longueur de γ ; alors γ' est une somme de traces toutes de longueur $l(\gamma) - 2n$ puisqu'elles sont des copies de γ dans lesquels on a effacé α_1 et α_2. Mais α_1 et α_2 sont conclusion de deux liens non-**affaiblissement** l_1 et l_2 puisque c est stricte. Donc chaque passage de γ dans c s'écrit : $\alpha\alpha_1{\alpha_2}^*\beta^*$ où α et β sont des prémisses de respectivement l_1 et l_2, ou si par exemple l_1 est un lien **axiome** : $\alpha^*\alpha_1{\alpha_2}^*\beta^*$, voire si l_1 et l_2 sont des liens **axiome** : $\alpha^*\alpha_1{\alpha_2}^*\beta$. En aucun cas α et β ne peuvent être égales entre elles, ou égales à α_1 ou α_2 sinon on trouve un interrupteur cyclique du réseau. Donc $l(\gamma)$ est au moins égale à $4n$. Donc $l(\gamma') = l(\gamma) - 2n$ est au moins égale à $2n$. Mais comme par hypothèse d'induction $l(\gamma')$ est bornée par un entier N', on en déduit que n est borné par $N'/2$ et donc que $l(\gamma)$ est bornée par $2N'$. •

3.3 Géomètrie des traces dérivables

Théorème 3.3.1 (Apériodicité des traces dérivables) *Une trace de la forme $\gamma = \delta\delta$ n'est pas dérivable.*

Preuve : remarquons que γ commence et se termine en un même lien. En effet $\gamma = \delta\delta$ donc, si α est l'arête initiale de δ et β son arête finale, α et β doivent être consécutives i.e., respectivement prémisse et conclusion ou respectivement conclusion et prémisse d'un même lien. Sinon $\delta\delta$ est la trace nulle.
On montre qu'une réduction par une coupure spéciale pour γ produit soit une trace nulle, soit une trace de la forme $\delta'\delta'$. Donc la forme normale de γ est soit nulle, soit de la forme $\delta_0\delta_0$. Mais le deuxième cas est impossible car la trace $\delta_0\delta_o$ passe forcément par une coupure donc n'est pas normale.
Si c est une coupure multiplicative n'échangeant pas les prémisses de γ alors en notant α_1 et α_2 les prémisses de c, la dérivée γ' de γ par c est γ dans laquelle on a supprimé α_1 et α_2. Donc si δ' est la dérivée de δ on a bien $\gamma' = \delta'\delta'$.
Si c est une coupure exponentielle spéciale pour γ alors notons α la prémisse $?I$ de c, β la prémisse $!O$ de c, b la boîte associée à β et $\alpha_1, \ldots, \alpha_n$ les prémisses du lien **pourquoi pas**. Remarquons que $n \neq 0$ i.e., c n'est pas une coupure **affaiblissement**. En effet puisque $\gamma = \delta\delta$, γ passe au moins deux fois par c ce qui est impossible si c est une coupure **affaiblissement**.
Supposons que γ ne commence pas dans b, et donc ne se termine pas dans b. Du fait que c est spéciale pour γ on déduit que γ n'a que des sous-traces de type **pp** dans b. Si γ n'échange pas les prémisses de c alors δ est de la forme :

$$\delta_1\alpha_{i_1}\alpha\beta^*\sigma_1\beta\alpha^*\alpha_{i_1}^* \ldots \delta_p\alpha_{i_p}\alpha\beta^*\sigma_p\beta\alpha^*\alpha_{i_p}^*\delta_{p+1}$$

où les σ_i sont contenues dans la boîte b et les δ_i en dehors. Donc δ' est de la forme :

$$\delta_1\alpha_{i_1}\sigma_1^{i_1}\alpha_{i_1}^* \ldots \delta_p\alpha_{i_p}\sigma_p^{i_p}\alpha_{i_p}^*\delta_{p+1}$$

où σ_i^j est la $j$ème copie de σ_i. Par conséquent, on a encore $\gamma' = \delta'\delta'$.
Si γ commence dans b alors δ est de la forme :

$$\delta = \sigma_1\beta\alpha^*\alpha_{i_1}^*\delta_1\alpha_{i_2}\alpha\beta^*\sigma_2\cdots\delta_p\alpha_{i_{p+1}}\alpha\beta^*\sigma_{p+1}$$

Soit δ_c la trace obtenue par permutation circulaire de δ de la manière suivante :

$$\delta_c = \delta_1\alpha_{i_2}\alpha\beta^*\sigma_2\cdots\delta_p\alpha_{i_{p+1}}\alpha\beta^*\sigma_{p+1}\sigma_1\beta\alpha^*\alpha_{i_1}^*$$

δ_c est bien une trace puisque δ commence et se termine en un même lien donc $\sigma_{p+1}\sigma_1$ est une trace. δ_c commence et se termine en un même lien en dehors de b donc $\gamma_c = \delta_c\delta_c$ est une trace qui commence et se termine en dehors de b. Si $\alpha_{i_{p+1}} \neq \alpha_1$ alors la dérivée de $\gamma = \delta\delta$ par c est nulle, de même que celle de γ_c. Sinon si γ n'échange pas les prémisses de c alors il en de même de γ_c qui se dérive donc par le raisonnement ci-dessus en ${\gamma'}_c = {\delta'}_c{\delta'}_c$ où ${\delta'}_c$ est la dérivée de δ_c par c. Mais en appliquant la permutation circulaire inverse à ${\gamma'}_c$ on trouve alors que $\gamma' = \delta'\delta'$. •

Chapitre 4

Réductions étiquetées

4.1 Définitions

Soit $\mathbf{V}^* = \{x^*, x \in \mathbf{V}\}$ où $\mathbf{V}$ est comme d'habitude l'ensemble des variables du lambda-calcul. On note $\mathcal{E}$ la structure libre engendrée par $\mathbf{V} \cup \mathbf{V}^*$ et deux fonctions binaires : $x, y \mapsto x.y$ et $x, y \mapsto x * y$. On appellera *noms* les éléments de $\mathcal{E}$, *noms atomiques* les éléments de $\mathbf{V} \cup \mathbf{V}^*$ et *nom principal* d'un nom x le nom le plus à gauche de x soit : si x est atomique alors son nom principal est x ; si $x = x_1 * x_2$ ou $x_1.x_2$ alors x_1 est son nom principal. On appelle *dual* d'un nom x le nom x^* défini par :

1. si $x \in \mathbf{V}$ alors son dual est $x^* \in \mathbf{V}^*$;
2. si $x = y^* \in \mathbf{V}^*$ alors son dual est $y \in \mathbf{V}$;
3. si $x = x_1.x_2$ alors son dual est $x_2^*.x_1^*$;
4. si $x = x_1 * x_2$ alors son dual est $x_1 * x_2^*$.

Un *nommage* d'un réseau R est une fonction injective de l'ensemble des arêtes duplicantes de R dans $\mathcal{E}$. Si on se donne un nommage de R, on dit que R est *étiqueté*. Soit R' un réduit de R par une coupure c. On définit les noms des arêtes de R' à partir de ceux des arêtes de R par :

1. si c est multiplicative alors toute arête α de R' porte le même nom que celle dont elle provient dans R ;
2. si c est exponentielle : notons $\alpha_1, \ldots, \alpha_n$ les prémisses du lien **pourquoi pas** et $x_1, \ldots, x_n$ leurs noms. Soit α une arête de R portant le nom x :
   - Si α n'est pas dans b et n'est pas prémisse de c alors α porte le même nom dans R' ;
   - si α est une porte auxiliaire de b alors la $i$ème copie de α dans R' porte le nom $x_i.x$.
   - Si α est dans b, alors la $i$ème copie de α dans R' porte le nom $x_i * x$;

Si deux arêtes de R' portent le même nom, c'est que l'une provient d'une arête non dans b de R et porte le nom $x_i * x$ ou $x_i.x$, et que l'autre provient d'une arête dans b ou d'une porte auxiliaire de b et qu'elle a le nom x dans R. Si on suppose que le nommage de R est *clair* i.e., pour toute arête

duplicante de R de nom x, x n'est le nom principal d'aucune autre arête de R, alors non seulement on a bien défini un nommage de R', mais en plus celui-ci est aussi clair puisque les α_i ne sont plus des arêtes duplicantes de R'. Dans la suite on supposera toujours que le nommage de R est clair.
On appellera *nommage réduit* le nommage de R' obtenu. On appelle *réduction étiquetée* d'un réseau R la donnée d'un nommage clair de R, d'une réduction de R en un réseau R' et du nommage réduit correspondant dans R'.
On dira qu'un réseau est *librement étiqueté* si chaque arête a pour nom un élément distinct du noms des autres arêtes de $\mathbf{V}$. Remarquons qu'un nommage libre est clair.
On note $\mathcal{E}^g$ le groupe libre engendré par $\mathbf{V}$. On note x^* l'inverse de x dans $\mathcal{E}^g$. On va associer à chaque nom x de $\mathcal{E}$ un élément x^g de $\mathcal{E}^g$ par :

1. si $x \in \mathbf{V}$ alors $x^g = x$;
2. si $x = y^* \in \mathbf{V}^*$ alors $x^g = y^*$;
3. si $x = x_1.x_2$ alors $x^g = x_1^g x_2^g$;
4. si $x = x_1 * x_2$ alors $x^g = x_1^g x_2^g x_1^{g*}$.

4.2 Traces maximales

On rappelle qu'une trace est maximale si elle commence en montant et se termine en descendant en des conclusions du réseau. Remarquons que la dérivée d'une trace maximale, si elle est non nulle, est une somme de traces maximales.

Lemme 4.2.1 *Soient α_1 et α_2 deux arêtes (ou arêtes inverses) d'un réseau R, γ_1 et γ_2 deux traces de même arête initiale α_1 et de même arête finale α_2. Si γ_1 et γ_2 ne traverse aucune autre arête duplicante que (éventuellement) α_1 et α_2 alors $\gamma_1 = \gamma_2$ et γ_1 est un chemin dans un interrupteur de R.*

Preuve : du fait que γ_1 et γ_2 ne traversent pas d'autre arête duplicante que α_1 et α_2 (si celles-ci sont duplicantes), on déduit que γ_1 et γ_2 ne traversent au plus que deux liens **pourquoi pas**, ceux de prémisses respectivement α_1 et α_2. Mais alors γ_1 et γ_2 sont des chemins qui joignent les conclusions des ces deux liens **pourquoi pas**. On peut donc supposer sans perdre de généralité que γ_1 et γ_2 ne traverse aucune formule duplicante ni aucun lien **pourquoi pas**. Dans ce cas γ_1 et γ_2 ne possèdent aucune autre arête de genre $?I$ que éventuellement α_1 et α_2. Soit un interrupteur G de R défini en effaçant pour chaque lien **par**, sa prémisse $?I$ si elle est différente de α_1 et α_2, sa prémisse O sinon. Alors γ_1 et γ_2 sont deux chemins dans le graphe G, joignant α_1 et α_2 qui sont bien des arêtes de G. Mais comme G est acyclique on a forcément $\gamma_1 = \gamma_2$. •

Lemme 4.2.2 *Soient γ_1 et γ_2 deux traces maximales dans un réseau R. Soient γ_1^d (resp. γ_2^d) la suite d'arêtes obtenue en effaçant de γ_1 (resp. γ_2) toutes les arêtes non duplicantes. Si $\gamma_1^d = \gamma_2^d$ alors $\gamma_1 = \gamma_2$.*

Preuve : soit α_1 (resp. α_2) l'arête initiale de γ_1 (resp. γ_2). Comme γ_1 est maximale, α_1 est une arête inverse conclusion du réseau. Si α_1 porte la formule I alors par définition c'est une formule duplicante et comme $\gamma_2^g = \gamma_1^g$, on a en particulier $\alpha_2 = \alpha_1$. Sinon si elle porte la formule $?I$ alors elle est

conclusion d'un lien **pourquoi pas** dont γ_1 emprunte une prémisse en montant. Donc γ_2 emprunte la même prémisse en montant et comme γ_2 est maximale, γ_2 commence donc en la conclusion du lien **pourquoi pas**, soit à nouveau $\alpha_1 = \alpha_2$. Si enfin α_1 est la conclusion O ou $!O$ du réseau, alors le même raisonnement appliqué à α_2 montre que α_2 ne peut porter la formule I ou $?I$ donc encore une fois $\alpha_1 = \alpha_2$. On obtient similairement que γ_1 et γ_2 se terminent en la même conclusion de R.
On a donc :

$$\begin{aligned}\gamma_1 &= \alpha^1\gamma_1^1\alpha^2\gamma_1^2\cdots\alpha^n\gamma_1^n\alpha^{n+1}\\ \gamma_2 &= \alpha^1\gamma_2^1\alpha^2\gamma_2^2\cdots\alpha^n\gamma_2^n\alpha^{n+1}\end{aligned}$$

où les γ_i^j ne contiennent aucune arête duplicante. Par le lemme ci-dessus on déduit immédiatement que pour tout j, $\gamma_1^j = \gamma_2^j$ soit $\gamma_1 = \gamma_2$ •

Définition 4.2.1 *Soit γ une trace maximale dans un réseau R. On appelle* étiquette *ou* nom *de γ la suite des noms des arêtes duplicantes qu'elle traverse, en convenant que si α a le nom x alors α^* a le nom x^*.*
Si $(x_1,\ldots,x_n)$ est le nom de γ alors on note γ^g le produit $x_1^g\cdots x_n^g$ dans $\mathcal{E}^g$.

Remarque : en vertu de la proposition ci-dessus, le nom de γ caractérise complètement γ. Dans la suite on identifiera les traces maximales avec leur nom.

Lemme 4.2.3 *Si γ et γ' sont des traces maximales dans un réseau librement étiqueté alors $\gamma^g = \gamma'^g$ entraine $\gamma = \gamma'$.*

Preuve : on a $\gamma = (x_1,\ldots,x_n)$ et $\gamma' = (y_1,\ldots,y_p)$. Pour tout i, on ne peut avoir $x_i = x_{i+1}^*$. Notons α l'arête de nom x_i. Si $x_i = x_{i+1}^*$ alors il existe une trace ne traversant aucune arête duplicante joignant α^* à α. Mais par le même argument que ci-dessus i.e., qu'un tel chemin est aussi un chemin dans un interrupteur de R, on déduit une contradiction. Du fait que R est librement étiqueté on déduit que les x_i est les y_j sont des éléments de $\mathbf{V}\cup\mathbf{V}^*$ et de ce qui précède on obtient que γ et γ' sont des mots réduits du groupe $\mathcal{E}^g$. Or deux mots réduits ne sont égaux dans un groupe libre que s'ils sont exactement la même suite de générateurs et d'inverses de générateurs, soit $\gamma = \gamma'$. •

4.3 Intégration des traces maximales

Théorème 4.3.1 *Soient γ une trace maximale dérivable dans un réseau R librement étiqueté, R' un réduit étiquetté strict de R et γ' la somme de traces dérivée de γ dans R'. Une trace δ de R' est l'un des termes de γ' ssi $\delta^g = \gamma^g$.*

Preuve : supposons que pour toute trace γ se dérivant en γ' non nulle dans R', et pour tout terme γ'_i de γ' on ait $\gamma^g = {\gamma'_i}^g$. Soit γ une trace de R et δ' une trace de R' tel que $\delta'^g = \gamma^g$. Par le lemme d'intégration (1.2.3) on sait qu'il existe une unique trace δ dans R tel que δ' soit un terme de la dérivée de δ dans R'. Donc $\delta^g = \delta'^g = \gamma^g$. Mais le nommage de R est libre donc par le lemme ce-dessus $\delta = \gamma$ et δ' est donc un terme de la dérivée de γ dans R'.
Pour montrer la réciproque i.e., ${\gamma'_i}^g = \gamma^g$ pour tout terme γ'_i de la somme γ', on se restreint au cas où R' est obtenu par réduction d'une coupure c, le cas général s'en déduit par induction sur la

longueur de la réduction. Il s'agit donc de voir que si γ est une somme de traces maximales telle que tous ses termes ont même nom dans $\mathcal{E}^g$, alors il en est de même de sa dérivée γ' par une coupure stricte c. Passons tout de suite le cas où c est multiplicative puisque la réduction de c ne change pas les noms des termes de γ.
Si c est une coupure exponentielle entre un lien **pourquoi pas** l dont les prémisses $\alpha_1, \ldots, \alpha_n$ portent les noms $x_1, \ldots, x_n$, et une boîte b alors comme chaque terme γ_i de γ est maximal, γ_i ne commence ni ne se termine dans b. Donc les sous-traces de γ_i modifiées par la réduction sont de types **pp**, **ap**, **pa**, ou **aa**. De plus, comme γ est dérivable, il y a au moins un terme de γ qui n'échange pas les prémisses de l. Comme ceux qui le font se dérivent en 0 donc n'ont pas de nom dans le réduit, on ne considérera que les premiers. Pour ne pas alourdir les notations, si x est le nom d'une arête, on écrira x au lieu de x^g pour le nom de cette arête interprété dans $\mathcal{E}^g$.

1. sous-traces de type **pp** ; elles ont la forme :
$$\sigma = \alpha_i \alpha \beta^* \delta \beta \alpha^* \alpha_i^*$$
où α est la conclusion de l, β la porte principale de b et δ est une trace contenue dans b. Donc si $u_1, \ldots, u_p$ est l'étiquette de δ, celle de σ est : $x_i, u_1, \ldots, u_p, x_i^*$. La dérivée de σ est :
$$\sigma' = \alpha_i \delta^i \alpha_i^*$$
où δ^i est la $i$ème copie de δ donc où les arêtes de δ^i portent le nom $x_i * u_j$. L'étiquette de σ' est alors : $x_i * u_1, \ldots, x_i * u_p$. Donc $\sigma'^g = x_i u_1 x_i{}^* x_i u_2 x_i{}^* \cdots x_i u_p x_i{}^* = x_i u_1 \cdots u_p x_i{}^* = \sigma^g$.
2. sous-traces de type **pa** : elles ont la forme :
$$\sigma = \alpha_i \alpha \beta^* \delta \iota$$
où ι est une porte auxiliaire de b dont on notera y le nom. σ se dérive en :
$$\sigma' = \alpha_i \delta^i \iota$$
Le nom de σ est $x_i, u_1, \ldots, u_p, y$ et celui de σ' dans le nommage réduit est $x_i * u_1, \ldots, x_i * u_p, x_i.y$ (car ι a le nom $x_i.y$ dans le nommage réduit). Donc $\sigma'^g = x_i u_1 x_i{}^* \cdots x_i u_p x_i{}^* x_i y = x_i u_1 \cdots u_p y = \sigma^g$.
3. le cas **pa** est symétrique du précédent.
4. sous-traces de types **aa** ; elles ont la forme :
$$\sigma = \iota_1^* \delta \iota_2$$
où ι_1 et ι_2 sont des portes auxiliaires de b portant les étiquettes y_1 et y_2. Donc la dérivée de σ est :
$$\sigma' = \sum_{i=1}^{p} \iota_1^{i\,*} \delta^i \iota_2^i$$
Le nom de σ est $y_1{}^*, u_1, \ldots, u_p, y_2$ et celui du $i$ème terme σ^i de σ' est $(x_i.y_1)^*, x_i * u_1, \ldots, x_i * u_p, x_i.y_2$ donc : $\sigma^{i\,g} = y_1{}^* x_i{}^* x_i u_1 x_i{}^* \cdots x_i u_p x_i{}^* x_i y_2 = y_1{}^* u_1 \cdots u_p y_2 = \sigma^g$. Donc tous les termes de σ' porte le même nom que σ modulo $\mathcal{E}^g$.

Comme toutes les sous-traces de γ_i qui sont modifiées par la réduction n'ont pas, modulo $\mathcal{E}^g$, changé de nom, il en est de même de γ_i. •
Donc par réduction étiquettée de R on peut déterminer les traces maximales dérivables de R.

4.4 conclusion

On peut voir la réduction d'une coupure c comme une consommation de ses prémisses, puis des prémisses de ses prémisses, etc... La question est alors de déterminer les paires d'arêtes qui deviendront prémisses de c au bout de n étapes de réduction. Cette question est précisément celle à laquelle répond l'ensemble des traces dérivables passant par c. La donnée de toutes les traces dérivables maximales passant par c détermine donc toutes les paires de prémisses potentielles de c, *sans réduire le réseau*. Reste maintenant pour produire un calcul des réseaux sans réduction à trouver les traces maximales dérivables. C'est le sujet de la partie qui suit.

Quatrième partie

L'algèbre dynamique

L'algèbre dynamique $\mathbf{\Lambda}^*$a été introduite par Girard ([Gir89]) sous la forme d'une algèbre d'opérateurs sur un espace de Hilbert. Les réseaux de la logique linéaire étaient interprétés par des couples (σ, π) de termes de $\mathbf{\Lambda}^*$, et la réduction correspondait à la formule d'exécution :

$$EX(\sigma, \pi) = (1 - \sigma^2)\pi \sum (\sigma\pi)^n (1 - \sigma^2)$$

Une autre manière de voir cette formule, comme on le montrera dans cette partie, est d'associer à chaque trace d'un réseau un poids dans $\mathbf{\Lambda}^*$. Les traces dérivables sont alors celles qui ont un poids non nul.
Dans le premier chapitre on définit l'algèbre comme une théorie axiomatisée par des équations dont on donne un modèle. Dans la suite on ne travaillera plus que dans le modèle des termes de $\mathbf{\Lambda}^*$. On montre alors que les équations une fois convenablement orientées donne un système de réécriture simple qui décide si un terme est prouvablement nul. Puis on définit les poids des traces et on montre qu'une trace est dérivable ssi elle est de poids non (prouvablement) nul.

Chapitre 1

L'algèbre $\mathbf{\Lambda}^*$

1.1 La théorie

On définit l'algèbre $\mathbf{\Lambda}^*$ par deux ensembles S et C d'axiomes équationnels. L'égalité morphologique entre deux termes de $\mathbf{\Lambda}^*$ (i.e. deux termes sont exactement la même suite de symboles) sera notée $\equiv$. Les axiomes de S définissent l'égalité *syntaxique* notée $\approx$ i.e., on notera $u \approx v$ quand l'égalité entre u et v est uniquement conséquence de S. On garde le symbole $=$ pour les conséquences de $\mathbf{\Lambda}^* = \mathcal{S} + \mathcal{C}$. Dans ce qui suit on dira qu'un terme est non nul si il est non prouvablement nul dans $\mathbf{\Lambda}^*$.

1.1.1 Le langage et l'égalité syntaxique ...

Le langage est constitué de :

1. deux constantes *multiplicatives* p et q ; *exponentielles.*

On notera $\mathbf{C}$ l'ensemble $\mathbf{V} \cup \{p, q\}$.

3. Deux constantes 0 et 1 ;

4. deux symboles de fonctions unaire $*$ et $!$. On notera u^* le terme $*(u)$;

5. un symbole de fonction binaire . que l'on notera comme d'habitude $.(u, v) = uv$.

Les axiomes constituant S sont :

1. la multiplication est associative, d'élément neutre 1 ;

2. 0 est absorbant pour la multiplication ;

3. $*$ est une involution et vérifie : $\forall u \forall v, (uv)^* \approx v^* u^*$, $1^* \approx 1$ et $0^* \approx 0$;

4. ! est un endomorphisme pour tout ce qui précède, soit : $\forall u \forall v, !(u)!(v) \approx !(uv)$; $(!(u))^* \approx !(u^*)$; $!(1) \approx 1$; $!(0) \approx 0$. On notera $!^n$ la composition n fois de la fonction !.

On dira qu'un terme u est *syntaxiquement égal* à un terme v si on peut montrer $u \approx v$.
On appelle *monômes* les termes de $\mathbf{\Lambda}^*$. Un monôme non nul ne comportant pas le symbole $*$ sera dit *positif*. Les monômes positifs se définissent par induction : si x est une constante de $\mathbf{C}$ alors x est un monôme positif et si u est un monôme positif alors $!(u)$ et xu sont des monômes positifs. Un monôme non nul ne comportant pas le symbole ! sera dit *plat*. De même on les définit par induction : si x est une constante de $\mathbf{C}$ alors x et x^* sont des monômes plats et si u est un monôme plat alors xu et x^*u sont des monômes plats. Les monômes $!(u)$ seront appelés des *!-formes*.
On appelle *monôme syntaxiquement normal* le monôme 1 ou ceux de la forme :

$$u \equiv f_1!(u_1)\dots f_n!(u_n)f_{n+1}$$

où pour tout i, $f_i \not\approx 0, 1$, $u_i \not\approx 0, 1$, les f_i sont des monômes plats et les u_i sont des monômes normaux.

Lemme 1.1.1 *Soit u un monôme non nul ; il existe un unique monôme normal u_0 tel que $u \approx u_0$.*

Preuve : on peut orient les équations de S de façon à ce que le terme de droite comporte strictement moins de symboles que celui de gauche. On obtient ainsi un système de réécriture des monômes. Clairement celui-ci n'admet pas de réduction infini. On vérifie aisément qu'il a la propriété de confluence locale, d'où la conclusion. •

1.1.2 ...et les équations pour calculer

On définit maintenant l'égalité $=$ sur $\mathbf{\Lambda}^*$. A chaque constante exponentielle x on associe un entier $c(x)$ appelé la *charge* de x. On suppose que pour tout n l'ensemble des constantes exponentielles de charge n est dénombrable.
La première clause est : $u \approx v \Rightarrow u = v$. Il y a deux autres types d'équations :

1. Equations *d'anihilation* :
$$x^*y = \delta_{xy}$$
où x et y sont soit des constantes multiplicatives, soit des constantes exponentielles et δ_{xy} est la fonction caractéristique du prédicat $x = y$. Il n'y a pas d'équation si x est une constante multiplicative (resp. exponentielle) et y est une constante exponentielle (resp. multiplicative).

2. Equations de commutation :
$$\begin{aligned} !(u)x &= x!^{c(x)}(u) \\ x^*!(u) &= !^{c(x)}(u)x^* \end{aligned}$$
où x est une constante exponentielle et u un monôme quelconque.

Remarque : la seconde équation de commutation se déduit de la première par application des équations de $*$.

1.2 Un modèle de $\mathbf{\Lambda}^*$

On connait plusieurs modèles de $\mathbf{\Lambda}^*$. Le premier d'entre eux, qui fut à l'origine de la théorie, est défini dans [Gir89] comme un ensemble d'opérateurs sur ℓ^2. On trouve dans le même article un modèle dont l'ensemble de base est constitué d'opérateurs sur les arbres binaires. Tous deux sont isomorphes à celui que l'on présente ici.
On va interpréter les monômes de $\mathbf{\Lambda}^*$ comme des bijections partielles sur $\mathbb{N}$. Une *bijection partielle* u de $\mathbb{N}$ dans $\mathbb{N}$ est définie par la donnée de son *domaine* $\mathbf{Dom}(u) \subset \mathbb{N}$, de son *codomaine* $\mathbf{Codom}(u) \subset \mathbb{N}$ et d'une bijection de $\mathbf{Dom}(u)$ sur $\mathbf{Codom}(u)$. On note Ł l'ensembles des bijections partielles de $\mathbb{N}$ dans $\mathbb{N}$, 0 la bijection de domaine vide, 1 l'identité sur $\mathbb{N}$ et 1_X l'identité sur $X \subset \mathbb{N}$.
On définit le produit de u et de v de la manière standard par : $x \in \mathbf{Dom}(uv)$ ssi $x \in \mathbf{Dom}(v)$ et $v(x) \in \mathbf{Dom}(u)$ et, si $x \in \mathbf{Dom}(uv)$ alors $uv(x) = u \circ v(x)$. On a clairement pour toute bijection partielle u, $0u = u0 = 0$, $1u = u1 = u$.
Exemple : soit u définie sur les entiers pairs par $u(2k) = k$ et v définie sur $\mathbb{N}$ par $v(x) = 2x+1$. Alors $uv = 0$. D'une manière générale le produit de deux bijections partielles u et v est nul ssi le codomaine de v et le domaine de u sont disjoints. Si u est une bijection partielle de X dans Y, on note u^* la bijection *réciproque* de Y dans X définie par $u^*(u(x)) = x$. Donc $u^*u = 1_X$ et $uu^* = 1_Y$. Notons que $*$ est une involution sur Ł et que $0^* = 0$, $1^* = 1$ et $(uv)^* = v^*u^*$ pour tout u, v dans Ł.
On se donne une bijection : $(x_1, x_2) \mapsto \langle x_1, x_2\rangle$ de $\mathbb{N}^2$ sur $\mathbb{N}$. On note π_1 et π_2 les "projections" soit $\pi_i(\langle x_1, x_2\rangle) = x_i$. On définit un morphisme ! de Ł dans Ł par : $x \in \mathbf{Dom}(!(u))$ ssi $\pi_2(x) \in \mathbf{Dom}(u)$ et $!(u)(x) = \langle \pi_1(x), u(\pi_2(x))\rangle$. La vérification que ! est un morphisme injectif sur Ł est immédiate et résulte du fait que $\langle x, y\rangle$ est bijective. De plus il est clair que $(!(u))^* = !(u^*)$ pour tout u dans Ł.
On se donne deux bijections partielles p et q de domaines pleins (i.e., égal à $\mathbb{N}$) et de codomaines disjoints. Par exemple on peut prendre $p(x) = 2x$ et $q(x) = 2x + 1$. On a alors satisfait les équations $p^*p = q^*q = 1$ (car les domaines de p est q sont pleins) et $p^*q = q^*p = 0$ (car les codomaines sont disjoints).
On note $\langle x_1, \ldots x_n\rangle$ l'entier $\langle x_1, \langle x_2, \ldots, \langle x_{n-1}, x_n\rangle \cdots\rangle\rangle$. Remarquons que pour tout entier x et pour tout n il existe un unique n-uplet $(x_1, \ldots, x_n)$ d'entiers tels que $x = \langle x_1, \ldots, x_n\rangle$. On a alors $!^n(u)(x) = \langle x_1, \ldots, x_{n-1}, u(x_n)\rangle$. Pour chaque couple d'entiers (k, c) on définit une bijection partielle u_k^c de domaine plein par :

$$u_k^c(\langle x_1, \ldots, x_c\rangle) = \langle\langle\langle k, c\rangle, x_1, \ldots, x_{c-1}\rangle, x_c\rangle$$

Les u_k^c sont de codomaines deux à deux disjoints : $u_k^c(x) = u_l^d(y)$ ssi $\langle\langle\langle k, c\rangle, x_1, \ldots, x_{c-1}\rangle, x_c\rangle = \langle\langle\langle l, d\rangle, y_1, \ldots, y_{d-1}\rangle, y_d\rangle$ ce qui entraine que $\langle k, c\rangle = \langle l, d\rangle$ soit $k = l$ et $c = d$. Donc $(u_k^c)^* u_l^d = \delta_{(k,c)=(l,d)}$.
Considérons le produit $!(v)u_k^c$ pour une bijection partielle v quelconque. On a :

$$\begin{aligned} !(v)u_k^c(\langle x_1, \ldots, x_c\rangle) &= !(v)\langle\langle\langle k, c\rangle, x_1, \ldots, x_{c-1}\rangle, x_c\rangle \\ &= \langle\langle\langle k, c\rangle, x_1, \ldots, x_{c-1}\rangle, v(x_c)\rangle \\ &= u_k^c(\langle x_1, \ldots, x_{c-1}, v(x_c)\rangle) \\ &= u_k^c !^c(v)(\langle x_1, \ldots, x_c\rangle) \end{aligned}$$

Donc u_k^c satisfait toutes les équations d'une constante exponentielle de charge c. Comme l'ensemble de celles-ci est au plus dénombrable, on peut les énumérer et interpréter la première par u_k^1, la seconde par u_k^2, etc...

On a donc construit dans L une interprétation de chacune des constantes de $\mathbf{\Lambda}^*$, ainsi que des fonctions ! et $*$ satisfaisant toutes les équations de $\mathbf{\Lambda}^*$.
Ceci permet de montrer des propriétés élémentaires sur les monômes telles que :

Proposition 1.2.1 *Si a est un monôme positif alors son interprétation est de domaine plein.*
Si a et b sont des monômes positifs et u est un monôme non nul alors aub^* est non nul.

Preuve : si a est positif alors son interprétation dans L est de domaine plein : si $a = x$ où x est une constante alors par définition de l'interprétation x est de domaine plein ; sinon $a = xa'$ où a' est un monôme positif donc par récurrence sur la longueur de a', de domaine plein et comme x l'est aussi, il en est de même pour a.
Comme b est positif on a donc que l'interprétation de b^* est de codomaine plein, donc comme u est non nul, ub^* non plus et comme a est de domaine plein il en est de même de aub^*. $\bullet$

Chapitre 2

Traces

2.1 Poids d'une trace

Une formule dans un réseau R contenue dans k boîtes sera dite de *profondeur* k. Si α est une formule duplicante ou une conclusion I de R, on appelle *charge* de α et on note $c(\alpha)$ le nombre de boîtes dont α est porte auxiliaire. A chaque formule α de profondeur k d'un réseau R on associe son *poids* $w(\alpha)$:

1. si α est la prémisse I d'un lien **tenseur** ou la prémisse O d'un lien **par** alors $w(\alpha) = !^k(q)$;
2. si α est la prémisse $!O$ d'un **tenseur** ou la prémisse $?I$ d'un **par** alors $w(\alpha) = !^k(p)$;
3. si α est prémisse d'un lien **pourquoi pas** ou conclusion I de R alors $w(\alpha) = !^k(x)$ où x est une nouvelle constante exponentielle (i.e. n'est le poids d'aucune autre formule de R) telle que $c(x) = c(\alpha)$;
4. toutes les autres formules ont pour poids 1.

Le poids $w(\gamma)$ d'une trace γ est défini par induction :

1. la trace nulle est de poids nul ;
2. si γ est la trace vide, alors $w(\gamma) = 1$;
3. si $\gamma = \alpha\gamma'$ alors $w(\gamma) = w(\gamma')w(\alpha)$;
4. si $\gamma = \alpha^*\gamma'$ alors $w(\gamma) = w(\gamma')(w(\alpha))^*$.

Remarque : le poids d'une trace est un monôme non syntaxiquement nul.

2.2 Traces dérivables

Définition 2.2.1 *Un monôme u a une* forme stable *si $u = a!(v)b^*$ où a et b sont des monômes positifs plats et v a une forme stable.*

Théorème 2.2.1 *Si γ est une trace alors:*

1. *$w(\gamma)$ est nul ou a une forme stable w_0 ;*
2. *$w(\gamma)$ est non nul ssi γ est dérivable.*

Preuve : on montre que :

1. $w(\gamma)$ est nul ssi le réduit γ' de γ obtenue par élimination d'une coupure c spéciale pour γ est de poids nul ; $w(\gamma)$ a une forme stable ssi $w(\gamma')$ en a une.
2. Une trace normale est de poids non nul ssi elle est non nulle. Dans ce cas son poids est une forme stable.

On sait que la réduction spéciale de γ termine, comme toutes les réductions. Si γ_0 est la forme normale obtenue on a donc $w(\gamma) = 0$ ssi $w(\gamma_0) = 0$ et $w(\gamma)$ a une forme stable ssi $w(\gamma_0)$ en a une. Reste donc à voir (compte tenu de 2.) que si γ_0 est non nulle alors γ est dérivable ce qui est exactement le corollaire de la confluence des traces de la partie précédente.

1. Soit donc c une coupure spéciale pour γ que l'on réduit en γ' par c :

   (a) si c est une coupure **axiome** alors γ' est obtenue en effaçant les deux prémisses de c dans γ. Comme celles-ci sont toutes deux de poids 1 on a $w(\gamma) = w(\gamma')$.

   (b) Si c est une coupure $\otimes/\wp$; notons α_1 et α_2 les prémisses du lien **tenseur** et α sa conclusion, et β_1 et β_2 celles du lien **par** et β sa conclusion. Si γ échange les prémisses de c alors c'est que γ a une sous-trace δ de la forme $\alpha_i\alpha\beta^*\beta_j{}^*$. Le poids de δ est alors $w(\delta) = q^*p = 0$ si $i = 1$ et $j = 2$ ou $p^*q = 0$ dans le cas contraire. Donc $w(\gamma) = 0$ par une application d'équation d'anihilation. Mais puisque γ échange les prémisses de c, γ' est la trace nulle donc de poids nul. Si γ n'échange pas les prémisses de c alors γ' est non nulle. De plus les sous-traces de γ passant par c (qui sont les seules à être modifiées par la réduction) ont toutes la forme $\delta = \alpha_i\alpha\beta^*\beta_i{}^*$. On a $w(\delta) = p^*p$ si $i = 1$ ou q^*q si $i = 2$ soit $w(\delta) = 1$. Comme δ se dérive par c en $\alpha_i\beta_i^*$ de poids 1 on obtient $w(\gamma) = w(\gamma')$.

   (c) si c est une coupure exponentielle spéciale pour γ : notons α_i les prémisses du lien **pourquoi pas**, α sa conclusion, β la conclusion du lien **bien sur**, et β_i les portes auxiliaires de la boîte. Les sous-traces de γ qui sont modifiées par la réduction ont trois formes :

      i. sous-traces de type **pp** :
      $$\delta = \alpha_i\alpha\beta^*\delta_0\beta\alpha^*\alpha_j^*$$
      où δ_0 est une trace dans la boîte b prémisse de c. Ils se réduisent en :
      $$\delta' = \alpha_i\delta_0^i\alpha_j^*$$
      où δ_0^i est la $i$ème copie de δ et est maintenant une trace dans les boîtes dont α_i est porte auxiliaire. Donc si δ_0 est de profondeur k dans R, alors δ_0^i est de profondeur $k-1+c(\alpha_i)$ dans R'. Notons w_0 le poids de δ_0 dans b, x_i et x_j les constantes exponentielles associées à α_i et α_j. On a
      $$w(\delta) = x_j^*!(w_0)x_i^* = x_j^*x_i!^{c(x_i)}(w_0)$$
      Si $i \neq j$ i.e., si γ échange les prémisses de c, alors $w(\delta) = 0$. Comme δ' est dans ce cas la trace nulle on a $w(\delta) = w(\delta')$. Sinon $w(\delta) = !^{c(x_i)}(w_0)$ soit à nouveau $w(\delta) = w(\delta')$.

ii. sous-trace de type **pa** :

$$\delta = \alpha_i \alpha \beta^* \delta_0 \beta_j$$

et son dérivé δ' est :

$$\delta' = \alpha_i \delta_0^i \beta_j^i$$

où δ_0^i est la $i$ème copie de δ. Notons y_j la constante exponentielle associée à β_j et y_j^i celle associée à β_j^i. On a $w(\delta) = y_j!(w_0)x_i = y_j x_i!^{c(x_i)}(w_0)$ et $w(\delta') = y_j^i!^{c(x_i)}(w_0)$. Par le lemme des coupures spéciales, on sait que δ est la seule sous-trace de type **pa** de γ et que γ s'écrit $\gamma_1\delta\gamma_2$ où γ_2 ne passe par aucune coupure et est en dehors de b. Donc γ_2 est une trace ne contenant pas d'arête inverse et $w(\gamma_2)$ est un monôme positif. Donc $w(\gamma) = w(\gamma_2)w(\delta)w(\gamma_1)$ est nul ssi $w(\delta)w(\gamma_1)$ l'est. Soit $w(\gamma)$ est nul ssi $y_j x_i!^{c(x_i)}(w_0)w(\gamma_1) = 0$ ssi (car $y_j x_i$ est un monôme positif) $!^{c(x_i)}(w_0)w(\gamma_1) = 0$. Comme γ_2 est en dehors de b, γ' s'écrit $\gamma'_1\delta'\gamma_2$ et donc $w(\gamma') = w(\gamma_2)y_j^i!^{c(x_i)}(w_0)w(\gamma_1)$ est nul ssi $w(\gamma)$ l'est (car $w(\gamma_2)y_j^i$ est un monôme positif). De plus si $w(\gamma')$ a une forme stable alors celle-ci s'écrit $w' = w(\gamma_2)y_j^i w''$ où w'' a une forme stable. Mais alors $w(\gamma) = w(\gamma_2)y_j x_i w''$ a aussi une forme stable. On utilise ici le fait que si xw a une forme stable où x est une constante de $\mathbf{C}$, alors celle-ci est de la forme xw_0.

iii. sous-trace de type **ap** : c'est le cas symétrique du précédent. Il se traîte similairement à ceci près que l'on utilise l'équation duale $x_i^*!(w_0) = !^{c(x_i)}(w_0)$.

2. Une trace γ normale et non nulle est de la forme $\gamma_1\gamma_2$ où γ_1 ne contient que des arêtes inverses, et γ_2 ne contient que des arêtes. Donc $w(\gamma) = w(\gamma_2)w(\gamma_1)$ et $w(\gamma_1)$ est un monôme négatif, $w(\gamma_2)$ est un monôme positif. Donc $w(\gamma)$ est non nul par la proposition 1.2.1 et en forme stable.
   •

2.3 Calcul des formes stables

Dans la preuve du théorème précédent on utilise les équations de gauche vers la droite. Par exemple on a jamais besoin d'écrire $1 = p^*p$. Ceci suggère que le calcul de la forme stable d'un poids de trace peut s'effectuer par réécriture en orientant les équations. Et en effet :

Théorème 2.3.1 *Soit w un monôme de $\mathbf{\Lambda}^*$. Alors toute réécriture de w par les équations orientées terminent. Si w a une forme stable w_0, alors w a une unique forme normale pour la réécriture et c'est w_0.*

Preuve : la terminaison de la réécriture provient du fait que toutes les équations orientées sauf les équations de commutation, font strictement diminuer le nombre de symboles. Soit w un monôme contenant une constante exponentielle x. Si on note $g_w(x)$ le nombre de ! à gauche de x dans w alors $g_w(x)$ diminue pour toute équation orientée appliquée à w, et diminue strictement si c'est une équation de commutation entre x et une !-forme (puisque justement dans le membre droit de l'équation, x est à *gauche* de la !-forme). La même remarque s'applique si on considère la duale d'une constante exponentielle x^* et le nombre $d_w(x^*)$ de !-formes à *droite* de x^* dans w. Donc le couple dont le premier terme est la somme des $g_w(x)$ et des $d_w(x^*)$ sur toutes les constantes exponentielles occurrant dans w, et dont le second terme est le nombre de symboles de w, diminue strictement pour l'ordre lexicographique à chaque application d'équation orientée. D'où la terminaison de la réécriture.

Reste à montrer la confluence locale. On suppose donc que w se réécrit en w_1 (resp. w_2) par application d'une équation orientée e_1 (resp. e_2). Aucun cas ne pose de problème sauf celui où e_1 et e_2 sont des équations de commutations partageant la même !-forme i.e., $w = u_1 x^* !(u) y u_2$ où u_1, u_2 et u sont des monômes et x et y sont des constantes exponentielles : $w_1 = u_1 x^* y !^{c(y)}(u) u_2$ et $w_2 = u_1 !^{c(x)}(u) x^* y u_2$. Comme w a une forme stable et que celle-ci ne peut être nulle, w n'est pas nul. Donc $x = y$ et par conséquent w_1 et w_2 se réécrivent par une équation d'anihilation et une équation d'élément neutre en $w' = u_1 !^{c(x)}(u) u_2$ •

Annexe A

Une résolution d'équation dans De

On prend l'exemple du chapitre 1.1. Rappelons que le lambda-terme était $\mathcal{Q} = (\lambda x(x)x)(\lambda g \lambda y(g)(g)y$ et que son type principal était $S = x_1, x_2^{b_1} \cap x_2^{b_1} \to x_1 = ((g_2^{b_3} \to g_1 \cap (y^{b_2} \to g_2)^{b_3}) \to (y^{b_3 b_2} \to g_1))^{b_4}$. On s'attend donc à trouver comme solution le type principal de l'entier quatre de Church soit, au nom des variables de type et des étiquettes près :

$$\big((y^{b_4} \to g_4)^{b_1 b_2 b_3} \cap (g_4^{b_3} \to g_3)^{b_1 b_2} \cap (g_3^{b_2} \to g_2)^{b_1} \cap g_2^{b_1} \to g_1\big) \to (y^{b_1 b_2 b_3 b_4} \to g_1)$$

On va montrer une suite de type principaux, chacun dérivé du précédent par application d'une règle : on terminera par le symbole $\bullet$ l'équation à laquelle on applique la règle.

$$x_1, x_2^{b_1} \cap x_2^{b_1} \to x_1 = ((g_2^{b_3} \to g_1 \cap (y^{b_2} \to g_2)^{b_3}) \to (y^{b_3 b_2} \to g_1))^{b_4} \qquad \bullet$$

Règle de duplication : on duplique b_4, on renomme b_i pour $i = 1, 2, 3$ en respectivement b_i^1 et b_i^2 et les variables de types x en respectivement x^1 et x^2. On renommera toujours en portant des numéros en exposant.

$$\begin{cases} x_1 \\ x_2^{b_1} \to x_1 = ({g_2^1}^{b_3^1} \to g_1^1 \cap ({y^1}^{b_2^1} \to g_2^1)^{b_3^1}) \to ({y^1}^{b_3^1 b_2^1} \to g_1^1) \\ \big(x_2 = ({g_2^2}^{b_3^2} \to g_1^2 \cap ({y^2}^{b_2^2} \to g_2^2)^{b_3^2}) \to ({y^2}^{b_3^2 b_2^2} \to g_1^2)\big)^{b_1} \quad \bullet \end{cases}$$

Règle de variable : on substitue x_2.

$$\begin{cases} x_1 \\ \big(({g_2^2}^{b_3^2} \to g_1^2 \cap ({y^2}^{b_2^2} \to g_2^2)^{b_3^2}) \to ({y^2}^{b_3^2 b_2^2} \to g_1^2)\big)^{b_1} \to x_1 = \\ ({g_2^1}^{b_3^1} \to g_1^1 \cap ({y^1}^{b_2^1} \to g_2^1)^{b_3^1}) \to ({y^1}^{b_3^1 b_2^1} \to g_1^1) \quad \bullet \end{cases}$$

Règle de flèche :

$$\left\{\begin{array}{ll}
x_1 & \\
x_1 = {y^1}^{b_3^1 b_2^1} \rightarrow g_1^1 & \bullet \\
{g_2^1}^{b_3^1} \rightarrow g_1^1 \cap ({y_1}^{b_2^1} \rightarrow g_2^1)^{b_3^1} = \big(({g_2^2}^{b_3^2} \rightarrow g_1^2 \cap ({y^2}^{b_2^2} \rightarrow g_2^2)^{b_3^2}) \rightarrow ({y^2}^{b_3^2 b_2^2} \rightarrow g_1^2)\big)^{b_1} &
\end{array}\right.$$

Règle de variable : on substitue x_1 :

$$\left\{\begin{array}{ll}
{y^1}^{b_3^1 b_2^1} \rightarrow g_1^1 & \\
{g_2^1}^{b_3^1} \rightarrow g_1^1 \cap ({y_1}^{b_2^1} \rightarrow g_2^1)^{b_3^1} = \big(({g_2^2}^{b_3^2} \rightarrow g_1^2 \cap ({y^2}^{b_2^2} \rightarrow g_2^2)^{b_3^2}) \rightarrow ({y^2}^{b_3^2 b_2^2} \rightarrow g_1^2)\big)^{b_1} & \bullet
\end{array}\right.$$

Règle de duplication : on introduit des exposants 3 et 4 :

$$\left\{\begin{array}{ll}
{y^1}^{b_3^1 b_2^1} \rightarrow g_1^1 & \\
\big({y_1}^{b_2^1} \rightarrow g_2^1 = ({g_2^3}^{b_3^3} \rightarrow g_1^3 \cap ({y^3}^{b_2^3} \rightarrow g_2^3)^{b_3^3}) \rightarrow ({y^3}^{b_3^3 b_2^3} \rightarrow g_1^3)\big)^{b_3^1} & \bullet \\
{g_2^1}^{b_3^1} \rightarrow g_1^1 = ({g_2^4}^{b_3^4} \rightarrow g_1^4 \cap ({y^4}^{b_2^4} \rightarrow g_2^4)^{b_3^4}) \rightarrow ({y^4}^{b_3^4 b_2^4} \rightarrow g_1^4) & \bullet
\end{array}\right.$$

On fait d'un coup les deux règles de flèches :

$$\left\{\begin{array}{ll}
{y^1}^{b_3^1 b_2^1} \rightarrow g_1^1 & \\
\big(g_2^1 = {y^3}^{b_3^3 b_2^3} \rightarrow g_1^3\big)^{b_3^1} & \bullet \\
\big({g_2^3}^{b_3^3} \rightarrow g_1^3 \cap ({y^3}^{b_2^3} \rightarrow g_2^3)^{b_3^3} = {y^1}^{b_2^1}\big)^{b_3^1} & \\
g_1^1 = {y^4}^{b_3^4 b_2^4} \rightarrow g_1^4 & \bullet \\
{g_2^4}^{b_3^4} \rightarrow g_1^4 \cap ({y^4}^{b_2^4} \rightarrow g_2^4)^{b_3^4} = {g_2^1}^{b_3^1} &
\end{array}\right.$$

On fait les deux règles de variable qui sont marquées ; notons que les deux autres équations ne sont pas de type variable mais de type duplication :

$$\left\{\begin{array}{ll}
{y^1}^{b_3^1 b_2^1} \rightarrow ({y^4}^{b_3^4 b_2^4} \rightarrow g_1^4) & \\
\big({g_2^3}^{b_3^3} \rightarrow g_1^3 \cap ({y^3}^{b_2^3} \rightarrow g_2^3)^{b_3^3} = {y^1}^{b_2^1}\big)^{b_3^1} & \bullet \\
{g_2^4}^{b_3^4} \rightarrow g_1^4 \cap ({y^4}^{b_2^4} \rightarrow g_2^4)^{b_3^4} = ({y^3}^{b_3^3 b_2^3} \rightarrow g_1^3)^{b_3^1} &
\end{array}\right.$$

Pour la première fois, on va dupliquer une formule non close i.e., telle que certaines de ses variables de type ont une occurrence ailleurs, ou équivalement, ses étiquettes apparaissent ailleurs : on renomme donc la variable y^1 en y^5 et y^6 et on remplace l'étiquette b_2^1 par respectivement rien et b_3^3 :

$$\begin{cases} (y^{5\,b_3^1} \cap y^{6\,b_3^1 b_3^3}) \rightarrow (y^{4\,b_3^4 b_2^4} \rightarrow g_1^4) & \\ g_2^{3\,b_3^3} \rightarrow g_1^3 = y^5 & \bullet \\ \left(y^{3\,b_2^3} \rightarrow g_2^3 = y^6\right)^{b_3^3} & \bullet \\ g_2^{4\,b_3^4} \rightarrow g_1^4 \cap (y^{4\,b_2^4} \rightarrow g_2^4)^{b_3^4} = (y^{3\,b_3^3 b_2^3} \rightarrow g_1^3)^{b_3^1} & \end{cases}$$

On fait les deux substitutions :

$$\begin{cases} \left((g_2^{3\,b_3^3} \rightarrow g_1^3)^{b_3^1} \cap (y^{3\,b_2^3} \rightarrow g_2^3)^{b_3^1 b_3^3}\right) \rightarrow (y^{4\,b_3^4 b_2^4} \rightarrow g_1^4) & \\ g_2^{4\,b_3^4} \rightarrow g_1^4 \cap (y^{4\,b_2^4} \rightarrow g_2^4)^{b_3^4} = (y^{3\,b_3^3 b_2^3} \rightarrow g_1^3)^{b_3^1} & \bullet \end{cases}$$

On duplique les exposant 3 en respectivement 7 et 8 et on remplace b_3^1 par respectivement rien et b_3^4 :

$$\begin{cases} \left(g_2^{7\,b_3^7} \rightarrow g_1^7 \cap (g_2^{8\,b_3^8} \rightarrow g_1^8)^{b_3^4} \cap (y^{7\,b_2^7} \rightarrow g_2^7)^{b_3^7} \cap (y^{8\,b_2^8} \rightarrow g_2^8)^{b_3^4 b_3^8}\right) \rightarrow (y^{4\,b_3^4 b_2^4} \rightarrow g_1^4) & \\ g_2^{4\,b_3^4} \rightarrow g_1^4 = y^{7\,b_3^7 b_2^7} \rightarrow g_1^7 & \bullet \\ \left(y^{4\,b_2^4} \rightarrow g_2^4 = y^{8\,b_3^8 b_2^8} \rightarrow g_1^8\right)^{b_3^4} & \bullet \end{cases}$$

On fait les deux règles de flèche :

$$\begin{cases} \left(g_2^{7\,b_3^7} \rightarrow g_1^7 \cap (g_2^{8\,b_3^8} \rightarrow g_1^8)^{b_3^4} \cap (y^{7\,b_2^7} \rightarrow g_2^7)^{b_3^7} \cap (y^{8\,b_2^8} \rightarrow g_2^8)^{b_3^4 b_3^8}\right) \rightarrow (y^{4\,b_3^4 b_2^4} \rightarrow g_1^4) & \\ g_1^4 = g_1^7 & \bullet \\ y^{7\,b_3^7 b_2^7} = g_2^{4\,b_3^4} & \\ \left(g_2^4 = g_1^8\right)^{b_3^4} & \bullet \\ \left(y^{8\,b_3^8 b_2^8} = y^{4\,b_2^4}\right)^{b_3^4} & \end{cases}$$

Puis les deux règles de variables. On substitue g_1^7 par g_1^4 et g_1^8 par g_2^4 :

$$\begin{cases} \left(g_2^{7\,b_3^7} \rightarrow g_1^4 \cap (g_2^{8\,b_3^8} \rightarrow g_2^4)^{b_3^4} \cap (y^{7\,b_2^7} \rightarrow g_2^7)^{b_3^7} \cap (y^{8\,b_2^8} \rightarrow g_2^8)^{b_3^4 b_3^8}\right) \rightarrow (y^{4\,b_3^4 b_2^4} \rightarrow g_1^4) & \\ y^{7\,b_3^7 b_2^7} = g_2^{4\,b_3^4} & \bullet \\ \left(y^{8\,b_3^8 b_2^8} = y^{4\,b_2^4}\right)^{b_3^4} & \bullet \end{cases}$$

Les deux équations de duplication ... ne duplique pas ce qui nous évitera de renommer variables de type et étiquettes. Leur seul effet est de remplacer partout b_3^4 par $b_3^7 b_2^7$ et b_2^4 par $b_3^8 b_2^8$:

$$\left\{\begin{array}{ll} \big(g_2^{7^{b_3^7}} \rightarrow g_1^4 \cap (g_2^{8^{b_3^8}} \rightarrow g_2^4)^{b_3^7 b_2^7} \cap (y^{7^{b_2^7}} \rightarrow g_2^7)^{b_3^7} \cap (y^{8^{b_2^8}} \rightarrow g_2^8)^{b_3^7 b_2^7 b_3^8}\big) \rightarrow (y^{4^{b_3^7 b_2^7 b_3^8 b_2^8}} \rightarrow g_1^4) & \\ \big(y^7 = g_2^4\big)^{b_3^7 b_2^7} & \bullet \\ \big(y^8 = y^4\big)^{b_3^7 b_2^7 b_3^8 b_2^8} & \bullet \end{array}\right.$$

Enfin on termine en substituant les deux variables y^7 et y^8 par g_2^4 et y^4 obtenant comme solution :

$$\left\{ \big(g_2^{7^{b_3^7}} \rightarrow g_1^4 \cap (g_2^{8^{b_3^8}} \rightarrow g_2^4)^{b_3^7 b_2^7} \cap (g_2^{4^{b_2^7}} \rightarrow g_2^7)^{b_3^7} \cap (y^{4^{b_2^8}} \rightarrow g_2^8)^{b_3^7 b_2^7 b_3^8}\big) \rightarrow (y^{4^{b_3^7 b_2^7 b_3^8 b_2^8}} \rightarrow g_1^4) \right.$$

Le lecteur est invité à vérifier que c'est bien le type principal de l'entier de Church 4.

Annexe B

Séquentialisation des réseaux typés

Il existe une caractérisation intrinsèque des réseaux multiplicatifs : le *critère de correction*. Mais celui-ci est énoncé pour les réseaux sans éléments neutres. Autrement dit on ne sait pas caractériser les réseaux multiplicatifs avec eléments neutres (1 et $\perp$) autrement que par “ce sont les traductions des preuves du calcul des séquents”. Par contre, dans le cas où l'on restreint les règles de **tenseur** et de **par** comme on le fait pour les réseaux, une telle caractérisation existe et c'est celle que l'on donne dans le chapitre 3. Cet appendice vise à généraliser le théorème de séquentialisation au cas des réseaux typés par la logique linéaire. Il s'agit d'un résultat sur les réseaux de preuves de Girard qui n'est donc pas relié directement au sujet de cette thèse (voir [Gir87]).
Dans ce qui suit on se limitera au fragment multiplicatif de la logique linéaire. C'est là en effet que se concentre tous les problèmes liés à la séquentialisation.

B.1 Calcul des séquents et réseaux multiplicatifs

Les formules sont les formules atomiques et leurs négations, les constantes 1 et $\perp$, les **tenseur** et les **par**, soit : si A et B sont des formules alors $A \otimes B$ et $A \wp B$ sont des formules. On quotiente les formules par les lois de de Morgan de façon à simplifier les règles du calcul des séquents :

1. $A^{\perp\perp} = A$, $1^{\perp} = \perp$;
2. $(A \otimes B)^{\perp} = A^{\perp} \wp B^{\perp}$, $(A \wp B)^{\perp} = A^{\perp} \otimes B^{\perp}$.

On appelle $\perp$-*formules* (resp. 1-*formules*) les formules définies par :

1. $\perp$ (resp. 1) est une $\perp$-formule (resp. une 1-formule) ;
2. si A et B sont des $\perp$-formules (resp. des 1-formules) alors $A \wp B$ est une $\perp$-formule (resp. $A \otimes B$ est une 1-formule)

Les règles du calcul des séquents sont : :

1. axiome et coupure :

$$\frac{}{\vdash A, A^{\perp}} \qquad\qquad \frac{\vdash A, \Gamma \quad \vdash A^{\perp}, \Gamma'}{\vdash \Gamma, \Gamma'}$$

2. $\bot$ et 1 :

$$\frac{\vdash \Gamma}{\vdash \Gamma, \bot} \qquad\qquad \frac{}{\vdash 1}$$

3. les règles de **par** et **tenseur** :

$$\frac{\vdash \Gamma, A, B}{\vdash \Gamma, A \wp B} \qquad\qquad \frac{\vdash \Gamma, A \quad \vdash \Gamma', B}{\vdash \Gamma, \Gamma', A \otimes B}$$

   où l'on impose que, pour la règle **par** (resp. pour la règle **tenseur**), A et B ne soient pas des 1-formules (resp. des $\bot$-formules).

où Γ et Γ' sont des multi-ensembles de formules.
Les structure de réseaux sont définies comme dans la partie II : ce sont des graphes dont les arêtes sont étiquetées par des formules, et dont les sommets ou liens sont définis par :

1. les liens **axiome** ont pour conclusions A et $A^{\bot}$ et n'ont pas de prémisse ; les liens coupure ont pour prémisses A et $A^{\bot}$ et pas de conclusion ;
2. le lien 1 n'a pas de prémisse et 1 pour conclusion ; le lien $\bot$ n'a pas de prémisse et $\bot$ pour conclusion ;
3. le lien **tenseur** a deux prémisses A et B qui ne sont pas des $\bot$-formules et une conclusion $A \otimes B$; le lien **par** a deux prémisses A et B qui ne sont pas des 1-formules et une conclusion $A \wp B$.

Si R est une structure alors un *interrupteur* G de R est un sous-graphe de R obtenu en effaçant une de ses prémisses à chaque lien **par**. Un réseau est une structure R vérifiant la condition **CC** :

TOUT INTERRUPTEUR G DE R EST ACYCLIQUE ET COMPTE UNE COMPOSANTE CONNEXE DE PLUS QUE DE LIENS $\bot$.

On peut traduire les preuves du calcul des séquents en des réseaux suivant le principe suivant : si π est une preuve du séquent $\vdash A_1, \ldots, A_n$ alors $(\pi)^\circ$ est un réseau de conclusions $A_1, \ldots, A_n$. La vérification que $(\pi)^\circ$ est une réseau est immédiate.

REMARQUE : les contraintes sur les $\bot$ et les 1 formules sont respectées par la logique intuitionniste i.e., par les traductions de lambda-termes typés en logique linéaire.

B.2 Séquentialisation

Théorème B.2.1 *Soit R un réseau. Il existe une preuve π en calcul des séquents telle que $(\pi)^\circ = R$.*

Preuve : par induction sur le nombre de liens $\bot$ de R. Si R a un lien $\bot$ conclusion l alors soit R' la structure R sans l. Tout interrupteur de R' a une composante connexe et un lien $\bot$ de moins qu'un interrupteur de R et est clairement acyclique. Donc R' est un réseau provenant d'une preuve π'. On obtient π en ajoutant une règle $\bot$ à π'.
Si R a une $\bot$-coupure c i.e., c a pour prémisses une $\bot$-formule A et une 1-formule $A^{\bot}$ alors A et $A^{\bot}$ sont déconnectées du reste puisque leurs seules prémisses héréditaires sont des liens $\bot$ ou 1.

Appelons C le sous-graphe de R constitué de c et de toutes ses prémisses et R' le reste de R. On a donc $C = C_1\mathbf{cut}C_2$ où C_1 a pour conclusion A, ne contient que des liens **par** et $\perp$ et C_2 a pour conclusion $A^\perp$ et ne contient que des liens **tenseur** et 1. Donc R est constitué des deux graphes C et R'. On vérifie aisément que tout interrupteur de R décompose C en n composantes connexes où n est le nombre de liens $\perp$ apparaissant dans C. Donc tout interrupteur de R décompose R' en $p+1$ composantes connexes où p est le nombre de liens $\perp$ apparaissant dans R'. Donc R' est un réseau. Comme tout interrupteur de R décompose C_1 en n composantes connexes, $R' + C_1$ est encore un réseau. De plus C_2 est lui même un réseau (C_2 n'a qu'un interrupteur et celui-ci est égal a C_2). Donc on a deux preuves π' et ρ de conclusions respectivement celles de R' plus A, et $A^\perp$. On peut donc fabriquer π en ajoutant une règle de coupure entre π' et ρ.
Supposons donc que R n'a plus ni liens $\perp$ conclusion, ni $\perp$-coupures. Alors dans R toute $\perp$-formule est prémisse d'un lien **par**. Considérons b, un lien $\perp$ prémisse d'un lien **par** l. Soit l' l'autre lien prémisse de l. On construit une structure R' en enlevant à R les liens b et l et en connectant l'arête conclusion A de l' au lien conclusion de l. R' compte donc un lien $\perp$ de moins que R. Clairement tout interrupteur de R' compte une composante connexe de moins qu'un interrupteur de R et est acyclique. Donc R' est un réseau. Soit π' la preuve en calcul des séquents associée à R'. Cette preuve contient une règle produisant la formule A. Si on met après cette règle, une règle $\perp$ et une règle **par** entre la nouvelle formule $\perp$ et A, et si on substitue dans la suite la formule A par $A \wp \perp$ on obtient une preuve π qui par construction se traduit en R.
Le dernier cas est celui d'un réseau sans lien $\perp$, donc pour lequel tout interrupteur est simplement connexe. On obtient le résultat par application du théorème de séquentialisation de Girard. •

Bibliographie

[Bar89] H. Barendregt, *The lambda calculus*, North Holland, 1989.

[CoDe86] M. Copo, M. Dezzani, *A new type assignment system for λ-terms*, Archives for Mathematical Logic 19, 1978.

[Dan90] V. Danos, *Logique Linéaire, Etude Statique et Dynamique*, thèse de doctorat de l'université Paris 7, Juin 1989.

[DaRe87] V. Danos et Laurent Regnier, *The structure of multiplicatives*, Archive for Mathematical logic 28, 1989.

[Duq90] E. Duquesne, *Elimination des coupures et types principaux dans le système $D\omega$*, communication au séminaire de lambda-calcul typé à l'université Paris 7, Novembre 90.

[FlRe88] A. Fleury. et C. Rétoré. *Logique Linéaire : Réseaux du fragment multiplicatif avec éléments neutres.* Prépublication de l'équipe de logique de Paris VII. Avril 1988.

[Gir87] J.Y. Girard, *Linear Logic*, Theoretical Computer Science 50, 1987.

[Gir89] J.Y. Girard, *Geometry of Interaction, an Interpretation of System F*, Proceedings of A.S.L. meeting, Padoue, Août 1988.

[Kri90] J.L. Krivine, *Lambda-calcul: types et modèles*, ed. Masson, col. Etudes et Recherches en Informatique, 1990.

[Lev80] J.J. Lévy, *Optimal reduction in the lambda-calculus*, To H.B. Curry : Essays on combinatory logic, λ-calculus and formalism, ed. J. Hindley & J. Seldin, Academic Press, 1980.

[Ret87] C. Rétoré *Mémoire de D.E.A.*, Université Paris 7, Septembre 1987.

[Ron90] S. Ronchi della Roca, *Principal Type Scheme and unification for intersection type discipline*, communication à International Joint Conference on Theory and Practice of Software development, Pisa 87, Theoritical Computer Science 59, no 1–2, 1988.

[Vid89] D. Vidal *Nouvelles notions de réductions en lambda-calcul.* Thèse de doctorat d'informatique. Université de Nancy I. Février 89.

Table des matières