50376
1993
71

50376
1993
71

USTL

LIFL Laboratoire d'Informatique Fondamentale de Lille

Thèse de Doctorat

Spécialité:

Informatique

Présentée par:

Christian Even

UNIVERSITE DES Sc. et Techn. BIBLIOTHEQUE UNIVERSITAIRE LILLE

Sujet:

Autour du Lambda-Calcul Partiel

Soutenue le 11 février 1993 devant la commission d'examen

Président :	**Jean-Paul Delahaye**
Rapporteurs :	**Pierre-Louis Curien**
	Furio Honsell
Examinateurs :	**Albert Burroni**
	Gérard Jacob
	Ramón Pino Pérez

Remerciements

Je suis très honoré que Jean-Paul Delahaye aît accepté de présider le jury de ma thèse. Je voudrais qu'il sache que j'ai particulièrement apprécié le séminaire d'Histoire et de Philosophie de la Logique et du Calcul qu'il anime au LIFL, et les échanges et discussions passionnées qu'il a ainsi suscités.

Je remercie vivement Pierre-Louis Curien d'avoir bien voulu être rapporteur de mon travail. A plusieurs reprises, j'ai profité de ses conseils judicieux et je lui suis reconnaissant de l'attention et de la bienveillance qu'il m'a consacrées.

Je tiens à exprimer mes vifs remerciements à Furio Honsell d'avoir également accepté d'être rapporteur de ma thèse. Sa lecture attentive et ses remarques pertinentes m'ont permis d'améliorer sensiblement la cohérence et la lisibilité du texte et de mieux cerner les points qui mériteraient des développements ultérieurs.

Je tiens à faire part à Gérard Jacob, mon directeur de thèse, de ma plus grande reconnaissance pour avoir rendu possible ce travail de recherche en m'ayant chaleureusement accueilli à Lille, et d'avoir veillé à son aboutissement.

Je ne remercierais jamais assez Ramon Pino Pérez, dont le propre travail a été à la source de ma recherche, de son ardente collaboration, de son constant soutien et de sa précieuse amitié. Sa claivoyance et son opiniâtreté m'ont aidé à surmonter bien des difficultés. J'espère bien que notre futur travail commun pourra grandement bénéficier des leçons de celui qui a été accompli.

Je tiens à exprimer ma grande reconnaissance à Albert Burroni pour avoir accepté de participer à ce jury.

Je dois encore remercier de leur excellent et chaleureux accueil tous les chercheurs et collègues du LIFL.

Je remercie spécialement Nour-Eddine Oussous de ses grands secours.

Qu'il me soit permis de saluer ici parmi de nombreux collègues et amis Michel Petitot, Hoang Ngoc Minh, Pierre Desplanques, Pierre Ageron, Christian Lair et Pierre-Vincent Koseleff.

Je voudrais enfin rendre grâce à mes proches, et tout spécialement à Véronique et à mon père, de m'avoir toujours prodigué leur généreux et indispensable soutien.

A Marythé, ma mère,
par delà sa disparition,
et à la baraka paternelle.

Table des matières

0 Introduction 3

1 Préliminaires sur le Lambda-Calcul Partiel 8
- 1.1 le λ_p-calcul . . . 9
- 1.2 Petit panorama des résultats existant sur le λ_p-calcul . . . 11

2 Confluence 13
- 2.1 Introduction . . . 13
- 2.2 La syntaxe du calcul . . . 14
- 2.3 La théorie de la restriction T_{res} . . . 17
- 2.4 La réduction $\rightarrow_{\beta_{pl}}$ . . . 20
- 2.5 Stabilité par $\stackrel{*}{\rightarrow}_S$ de la normalisabilité forte . . . 24
- 2.6 Confluence modulo une relation d'équivalence . . . 27
- 2.7 Plusieurs propriétés du calcul Λ_{pl} . . . 28
- 2.8 La confluence de Λ_{pl} pour les termes fortement normalisables . . . 31

3 Normalisation 36
- 3.1 Syntaxe de Λ_t . . . 37
- 3.2 Les règles de typage . . . 38
- 3.3 Quelques lemmes . . . 38
- 3.4 Préservation du typage par réduction . . . 38
- 3.5 Normalisabilité forte des termes typables . . . 42
- 3.6 Caractérisation d'une classe de termes de Λ_{pl} fortement normalisables . . . 44

4 Typage 45
- 4.1 Le Lambda-calcul par valeur de tête . . . 45
- 4.2 Définitions et Propriétés du λ_{vh}-calcul. . . . 46
- 4.3 Quelques rappels sur le λ_v-calcul . . . 55
- 4.4 Le système $\mathcal{D}*$ . . . 56
- 4.5 Les termes typables sont réductibles à une valeur . . . 57
- 4.6 Les termes réductibles à une valeur sont typables . . . 60
- 4.7 Caractérisation des termes égaux à une valeur dans le λ_p-calcul . . . 63
- 4.8 Caractérisation des termes clos (fortement) normalisables de λ_{pl} . . . 65

5 Algèbres combinatoires avec éléments partiels 72
- 5.1 Lien entre le λ_p-calcul et les algèbres combinatoires . . . 73

6 Approches catégoriques **84**
6.1 Les catégories cartésiennes fermées partielles 85
6.2 Définition 86
6.3 Présentation équationnelle 87
6.4 Propriétés 87
6.5 Interprétation des termes 88
6.6 Une PCCC avec un objet réflexif fournit une λ_p-algèbre 94
6.7 PCCC construite à partir d'une λ_p-algèbre 96
6.8 Vérification de la présentation équationnelle: 99
6.9 D'une λ_p-algèbre à une PCCC et retour 101
6.10 Une possible construction de modèles du λ-calcul partiel 103

7 Annexe: Fin de la démonstration du lemme 61 **106**

8 Conclusion et perspectives de travail **111**
8.1 Conclusion 111
8.2 Perspectives de travail 111

0

Introduction

Le modèle de la calculabilité que constitue le λ-calcul, introduit par Church dans les années 30 [Church] n'a plus aujourd'hui à faire la preuve de sa fécondité dans le domaine de la programmation fonctionnelle.

Ce modèle a eu en effet une nombreuse descendance: il est à la source de la conception de langages de programmation comme LISP, ML, CAML, ses affinités avec la théorie de la démonstration se sont illustrées dans des systèmes tels que AUTOMATH, le calcul des constructions de T.Coquand et G.Huet, et dans la logique linéaire de J.Y.Girard, etc...

Pour conclure ce bref panorama, nous pouvons noter que son étude s'est notamment avérée devoir être reliée avec celle des algèbres combinatoires et avec la théorie des catégories, et certains travaux ont notamment permis de souligner l'intérêt de ces deux aspects (*cf* par exmple[Curien et alt.]).

Parmi les divers problèmes auxquels le λ-calcul permet de donner une formulation, ceux qui vont plus particulièrement nous intéresser sont ceux de la terminaison et de l'équivalence ou (de la transformation) des programmes.

Le λ-calcul comme langage de programmation

Le λ-calcul est un modèle de la calculabilité qui est hautement non-déterministe. Il n'y a en effet qu'une seule règle de calcul (ou réduction), la règle β : $(\lambda x.M)N \rightarrow M[N/x]$, mais un même terme peut contenir diverses occurences où celle-ci peut s'appliquer, d'où il résulte qu'il y a beaucoup de façons d'effectuer le calcul que représente un terme.

Néanmoins la propriété de Church-Rosser signifie notamment que le résultat d'un calcul (c'est-à-dire la réduction itérée d'un terme jusqu'à ce qu'il n'y en aît plus de possible) ne dépend pas de l'ordre dans lequel on a procédé.

Un langage de programmation fonctionnel correspond lui au choix d'une stratégie d'évaluation (par exemple, par l'intermédiaire d'une machine abstraite qui définit la manière dont un terme est évalué), ce que l'on peut ramener au choix d'un ordre dans lequel les réductions seront effectuées.

Il existe dans le λ-calcul une réduction plus naturelle que les autres, la réduction gauche, en ceci qu'elle découvre toujours le résultat d'un calcul, c'est-à-dire un terme *normal* (i.e. dans lequel plus aucune réduction ne puisse être effectuée) si celui-ci peut être trouvé par une stratégie quelconque.

Ceci permet de définir une fonction partielle *Eval* de l'ensemble des termes du λ-calcul dans l'ensemble des termes normaux, qui à un terme associe sa forme normale (soit le résultat qui lui correspond), et est indéfinie si celle-ci n'existe pas (c'est-à-dire que le calcul ne termine jamais).

La notion la plus naturelle d'équivalence de programmes que l'on puisse définir est l'équivalence observationnelle, qui énonce que deux termes définisssent des programmes équivalents, si dans n'importe quel contexte, la substitution de l'un à l'autre ne modifie pas le résultat du calcul. Ce qui s'énonce:

$$M \equiv_{obs} N \iff \text{pour tout contexte } C, Eval(C[M]) = Eval(C[N]).$$

Cependant cette notion se prète mal à la vérification que deux programmes sont effectivement équivalents. Néanmoins toute notion raisonnable d'équivalence de deux programmes $M \equiv N$ doit impliquer leur équivalence observationnelle (i.e. $M \equiv N \Longrightarrow M \equiv_{obs} N$) et sera dite alors *correcte*. A cet égard, la β-équivalence du λ-calcul est correcte au regard de l'évaluation par réduction gauche.

Si maintenant nous modifions la stratégie adoptée, on aimerait également pouvoir disposer d'un cadre commun où représenter la stratégie et l'outil qui nous permet d'etablir l'equivalence des programmes. Nous allons voir que c'est ce qui est rélisé par le λ-calcul par par valeur au regard de la stratégie d'évaluation la plus communément adoptée dans les implantations des langages fonctionnels.

λ_v-calcul et équivalence des programmes évalués par un mécanisme d'appel par valeur

Le mécanisme dont nous parlions est celui qui est dit mécanisme d'appel par valeur. Plotkin, selon ce point de vue que les programmes qui nous intéressent concrétement utilise ce mécanisme, a été amené à introduire le λ_v-calcul, qui a le double avantage de permettre de représenter la stratégie d'évaluation par valeur (qui correspond dans le λ_v-calcul à une stratégie de réduction gauche et paresseuse) et d'offrir en même temps un outil de raisonnement sur l'équivalence des programmes.

Dans le λ_v-calcul l'unique règle permettant d'établir l'équivalence des programmes, à savoir la règle β, est relativisée à ce mode d'évaluation[1].

Pour être plus précis, supposons que l'on a distingué une classe de termes comme étant des valeurs, c'est-à-dire qu'on les considère comme constituant un résultat suffisamment évalué (au moins en tant que résultat intermédiaire d'un calcul), alors la règle β devient la suivante:

$$\beta_v : (\lambda x.M)N = M[N/x], \text{ si } N \text{ est une valeur}$$

Plotkin avait choisi de prendre pour valeurs les abstractions, les constantes et les variables. En effet, la stratégie d'évaluation de la machine abstraite SECD correspond ici à la réduction gauche par valeur, qui est paresseuse, c'est-à-dire qu'on ne réduit pas sous une abstraction. Les termes clos normaux par rapport à cette réduction sont donc les abstractions et les constantes, et de ce point de vue il faut les considérer comme valeurs. Il faut également y adjoindre les variables pour avoir un calcul qui puisse se formuler simplement tout en ayant de bonnes propriétés. L'essentiel étant que les valeurs closes sont les termes clos normaux par rapport à cette réduction.

[1] Le λ_v-calcul demeure un modèle de calculabilité, toutes les fonctions récursives y étant représentables. Notons que ce mode d'évaluation est tout d'abord apparu dans l'implémentation du langage ISWIM par l'intermédiaire de la machine abstraite SECD donnée par Landin, et dont Plotkin a montré l'étroite relation de la sémantique opérationnelle avec le λ_v-calcul et comment son mécanisme s'y traduisait par une stratégie de réduction gauche et paresseuse. Cette approche est générique de la plupart des implémentations des langages fonctionnels et semble pouvoir s'adapter sans trop de mal aux mécanismes définis par d'autres machines abstraites.

Nous allons voir, et c'est un des aspects de ce travail, que ce n'est pas le seul choix raisonnable qui puisse être envisagé et qu'il est également possible et tout aussi raisonnable de supposer que les valeurs soient des formes normales de tête[2].

Cependant ce calcul déçoit l'espoir de fournir une notion satisfaisante d'équivalence de programmes pour les termes dont l'évaluation ne se termine pas, ceux que nous appellerons les *indéfinis* (les autres, par opposition, seront baptisés tout naturellement les *définis*). Moggi a proposé un calcul, le lambda calcul partiel ou λ_p-calcul, issu de considérations sémantiques, qui respecte l'esprit du λ-calcul par valeur, mais dont la plus grande expressivité s'avère permettre d'identifier les programmes qui ne terminent pas, mais ont moralement le même comportement algorithmique dans une stratégie d'évaluation par valeur.

Le λ_p-calcul

Le formalisme du λ_p-calcul est relativement simple, du moins dans la simplification qu'en a proposée Ramon Pino Pérez. Grosso modo, il s'agit d'adjoindre à un terme un certain nombre de mémoires à ses sous-termes. Lors de la réduction d'un sous-terme qui est un rédex, la trace de cette réduction, c'est-à-dire l'argument du rédex, vient s'ajouter à la mémoire correspondant au sous-terme considéré.

La réduction β_p consiste à réduire et en outre à mémoriser l'argument dans cette réduction, à savoir:

$$\beta_p : (\lambda x.M)N = M[N/x]\lceil N$$

On a donc un nouveau constructeur de termes $\lceil$, et $M\lceil N$ est le terme M étiqueté par une mémoire contenant le terme N. De plus on s'autorise à effectuer un certain nombre de manipulations de ces mémoires (c'est l'objet de la *Théorie de la restriction. cf* le chapitre 1). Les égalités prouvables dans λ_v le demeurent dans λ_p[3].

Les résultats présentés dans cette thèse

Le λ_p-calcul reste encore relativement peu étudié. Différentes approches ont étés ici menées pour en clarifier certains aspects et pour obtenir des caractérisations de classes intéressantes de termes. Nous nous sommes également interessés au λ_v-calcul dont l'étude parallèle s'avère indispensable et en avons étudié une variante (λ_{vh}).

La confluence n'a pas encore pu être démontrée, bien que des résultats partiels de confluence aient étés établis par Moggi (*cf* [Moggi 88]). Nous nous sommes tout d'abord à nouveau penchés sur ce problème et obtenu un résultat comparable par d'autres méthodes pour un λ_p-calcul paresseux (λ_{pl}).

Nous avons ensuite chercher à caractériser par des systèmes de types les termes remarquables: normalisables, fortement normalisables, réductibles ou égalisables à une valeur pour différents calculs (λ_v,λ_{vh},λ_p,λ_{pl}..).

Moggi a formulé une conjecture sur l'existence d'un formalisme équivalent au λ_p-calcul qui serait fourni par une notion adéquate d'algèbre combinatoire. Nous répondons ici à cette conjecture, tout en montrant l'équivalence avec des formalismes issus de la théorie des catégories.

---

[2] D'autres variations sur la notion de valeur ont été envisagées dans [Dezani& alt.86, FH-SRDR 87]. Notre approche syntaxique mériterait sans une plus grande investigation pour être fondée plus concrètement, notamment en la corrélant avec la sémantique opérationnelle de machines abstraites, ce qui paraît être tout à fait viable.

[3] et les égalités entre termes purs qui se prouvent dans λ_p se prouvent dans λ, comme tout terme de λ_p est égal à un terme pur, la situation se résume ainsi: $\lambda_v \not\subset \lambda_p \not\subset \lambda$.

L'élaboration d'une sémantique dénotationnelle du λ_p-calcul n'est pas encore entièrement satisfaisante (*cf* [Pino 92]). En effet la construction de graph-modèles ou modèles fonctionnels, suivant une construction analogue à celle de Krivine (*cf* [Krivine 90]) fonctionne à condition d'identifier tous les indéfinis, mais remédier à cette imperfection est resté une voie inaboutie de ce travail. Nous ouvrons enfin diverses perspectives de recherche.

Nous prouvons au chapitre 2, que pour un λ_p-calcul paresseux et par suite plus simple (λ_{pl}), dans lequel il n'y a en quelque sorte qu'une restriction par un ensemble de termes, la réduction jouit de cette propriété de confluence pour les termes fortement normalisables, ce qui laisse présager que cela puisse être vrai de tout le λ_p-calcul.

En effet, dans ce cas auquel nous nous sommes intéressés, on peut faire de la théorie de la restriction un système de réécriture nœthérien et confluent, ce qui n'apparaît pas possible dans le cas du calcul tout entier (*cf* [Pino 89]), or on ne connaît pour ce problème, qui est celui de la *confluence modulo une relation d'équivalence*, de solutions que sous des hypothèses restrictives et notre étude d'ailleurs fournit une technique nouvelle lorsque la relation d'équivalence est présentée par un système de réécriture nœthérien et confluent.

Nous montrons également au chapitre 3 que, par un système de types, on peut caractériser certains termes fortement normalisables du λ_p-calcul.

Le chapitre 4 a but de caractériser les termes qui sont égalisables à une valeur du λ_p-calcul. Une valeur y est un terme égal par la théorie de la restriction à une abstraction , une variable ou une constante.

Nous présentons une variante du λ_v-calcul dans lequel les valeurs sont des termes en forme normale de tête et nous prouvons la confluence de ce calcul et un théorème de standardisation, ce qui montre que ce calcul est tout aussi viable - du moins quant à être pourvu de bonnes propriétés syntaxiques - que le λ_v-calcul originellement introduit par Plotkin. Nous proposons un système de types qui permet de caractériser pour ces deux calculs les termes qui peuvent se réduire à une valeur et montrons comment ce résultat caractérise les termes égalisables à une valeur dans le λ_p-calcul et les termes clos du λ_{pl}calcul qui le sont également. Nous prouvons du même coup que les termes clos du λ_{pl}égalisables à une valeur sont aussi fortement normalisables (en une valeur).

Dans l'introduction de sa thèse, dont les thèmes sont étroitement liés à ceux du présent travail, R.Pino Pérez nous explique pourquoi la notion d'équivalence qu'offre le λ_p-calcul est plus raisonnable que les équivalences opérationnelle ou observationnelle: celles-ci identifient des programmes qui divergent alors que leur comportement algorithmique sont sensiblement différents. Le λ_p-calcul n'a pas ce défaut et une de nos préoccupations a été de montrer qu'on peut trouver des modèles du λ_p-calcul qui ont cette même propriété de ne pas identifier tous les termes indéfinis, et nous proposons une construction de tels modèles.

Nous montrons au chapitre 5 en effet qu'il existe une équivalence entre le λ_p-calcul (trés légèrement modifié) et certaines algèbres combinatoires qui sont des modèles de λ_p - nous précisons cette notion d'algèbre combinatoire - et que nous avons baptisées λ_p-algèbres, de même qu'il existe une équivalence entre le λ-calcul pur et les algèbres combinatoires introduites par Curry, résolvant ainsi la conjecture de Moggi. Nous montrons que ces λ_p-algèbres sont les algèbres combinatoires qui vérifient un ensemble fini d'équations, et qui correspondent aux équations de Curry pour le λ-calcul pur.

Nous montrons dans le chapitre 6 comment une catégorie cartésienne fermée partielle muni d'un objet réflexif fournit une λ_p-algèbre. Nous montrons enfin que ce résultat admet en partie une réciproque: la sous-λ_p-algèbre engendrée par S, K et I d'une λ_p-algèbre permet de construire une catégorie cartésienne fermée partielle muni d'un objet réflexif.

Nous terminons en prouvant un théorème analogue à celui de Koymans pour le λ-calcul

pur, à savoir que la λ_p-algèbre que fournit cette dernière catégorie est isomorphe à celle qui a servi à la construire.

Enfin, nous montrons comment, à modifier de façon négligeable les axiomes du λ_p-calcul, et corrélativement celle de λ_p-algèbre et d'interprétation catégorique, cette réciproque se généralise à toute λ_p-algèbre.

Nous conluons et tentons de dégager des développements ultérieurs de ce travail.

1

Préliminaires sur le Lambda-Calcul Partiel

1.1 le λ_p-calcul

Le λ_p-calcul représente un cadre théorique pour étudier l'équivalence des programmes dont l'éxécution se fait par un mécanisme d'appel par valeur. Plotkin a introduit dans son article *Call by name, call by value and the λ-calculus* [Plotkin 75] un formalisme, le λ_v-calcul ou λ-calcul par valeur, qui est opérationnellement équivalent à ce mécanisme.

Comme nous l'avons dit, le λ_p-calcul a été introduit par E.Moggi pour répondre à certaines insuffisances du λ_v-calcul quant à offrir un formalisme satisfaisant pour étudier l'équivalence des programmes. De fait, Moggi a découvert le λ_p-calcul par des considérations modèle-théoriques, comme étant le seul calcul cohérent et complet au regard d'une certaine classe d'interprétations dans les catégories cartésiennes partielles fermées. En fait le formalisme qu'il a adopté était différent que celui que nous allons utiliser. Il ne considérait en effet que certains termes du λ_p-calcul que nous considérons, ceux qu'il a baptisés les $p\lambda$-termes. R.Pino Pérez a montré que ce formalisme est équivalent à celui que nous présentons. Nous allons ultérieurement montrer comment ce formalisme se lie aux interprétations catégoriques et à certaines algèbres combinatoires, le tout donnant lieu à une harmonisation de ces différentes approches.

Pour ce que nous avons dit concernant l'équivalence des programmes, considérons par exemple les termes

$$P_1 = (\lambda xy.c)(\Omega\Omega)\Omega \text{ et } P_2 = (\lambda xy.c)\Omega(\Omega\Omega).$$

La confluence du λ_v-calcul prouve qu'il ne sont pas équivalents. Or leur comportement algorithmique est le même, au sens où les calculs infinis que ces termes engendrent sont les mêmes. Nous souhaiterions donc les identifier. C'est ce que nous va nous permettre le λ_p-calcul.

La syntaxe du λ_p-calcul est la suivante:

terme	= variable		x
	= constante		c
	= abstraction		$\lambda x.M$
	= application		(MN)
	= restriction d'un terme	par un terme	$M\lceil N$
		par un ensemble de termes	$M\lceil D$

Nous noterons Λ_P (resp. $\Lambda_P{}^0$) l'ensemble des termes (resp. termes clos) du λ_p-calcul.

$M\lceil N$ représente le terme M sous réserve que N soit une valeur (autrement dit cette condition est notée explicitement), sinon cette "réserve" ne s'élimine pas, mais peut se transformer par réduction, ce qui va s'expliquer de la suite. $M\lceil N$ se lit "M restreint par N".

Les termes sont considérés modulo une théorie $T\lceil$ qui décrit les manipulations qu'on s'autorise à faire sur les mémoires, dite *théorie de la restriction.* Celle-ci est constituée des axiomes suivants:

$M\lceil\emptyset = M$: si la mémoire est vide, on peut l'ignorer.

$M\lceil x = M, M\lceil c = M, M\lceil(\lambda x.P) = M$: si on a une valeur en mémoire, on peut la décharger

$M\lceil\{M\} = M$: le calcul de M comprend la vérification que M est une valeur.

$(M\lceil D)N = (MN)\lceil D, M(N\lceil D) = (MN)\lceil D$: la vérification des conditions mémorisées peut être retardée lors d'une application.

$(M\lceil D)\lceil E = M\lceil(D \cup E), M\lceil\{N\lceil D\} = M\lceil(\{N\} \cup D)$: on peut compacter les mémoires qui étiquètent un même terme.

$(\lambda x.M\lceil N)\lceil N = (\lambda x.M)\lceil N$: si une condition est à vérifier hors d'une d'une abstraction, il est inutile de la vérifier à l'intérieur. Ce que nous exprimerons en disant qu'une information globale peut se déduire à un niveau local (elle y est déjà présente implicitement).

Les règles d'inférence sont les suivantes:

$$(\mu)\ \frac{M\ \mathcal{R}\ N}{MP\ \mathcal{R}\ NP} \qquad (\rho)\ \frac{}{M\ \mathcal{R}\ M}$$

$$(\nu)\ \frac{M\ \mathcal{R}\ N}{PM\ \mathcal{R}\ PN} \qquad (\sigma)\ \frac{M\ \mathcal{R}\ N}{N\ \mathcal{R}\ M}$$

$$(\gamma)\ \frac{M\ \mathcal{R}\ N}{M\lceil D\ \mathcal{R}\ N\lceil D} \qquad (\tau)\ \frac{M\ \mathcal{R}\ N \quad N\ \mathcal{R}\ P}{M\ \mathcal{R}\ P}$$

$$(\delta)\ \frac{N\ \mathcal{R}\ P}{M\lceil D \sqcup \{N\}\ \mathcal{R}\ M\lceil D \sqcup \{P\}} \qquad (\xi)\ \frac{M\ \mathcal{R}\ N}{\lambda x.M\ \mathcal{R}\ \lambda x.P}$$

En fait il est possible de se limiter à ne restreindre les termes que par des termes et non par des ensembles de termes et dans certains cas cela simplifie agréablement le formalisme.

Dans ce cas la théorie de la restriction $T\lceil$ s'écrit:

$M\lceil x = M$

$M\lceil c = M$

$M\lceil(\lambda x.P) = M$

$M\lceil M = M$

$P(Q\lceil N) = (PQ)\lceil N$

$(P\lceil N)Q = (PQ)\lceil N$

$P\lceil(Q\lceil N) = P\lceil(N\lceil Q)$

$P\lceil(Q\lceil N) = (P\lceil N)\lceil Q.$

$(\lambda x.M\lceil N)\lceil N = (\lambda x.M)\lceil N$

les règles d'inférence devenant alors:

$$(\mu)\ \frac{M\ \mathcal{R}\ N}{MP\ \mathcal{R}\ NP} \qquad (\rho)\ \frac{}{M\ \mathcal{R}\ M}$$

$$(\nu)\ \frac{M\ \mathcal{R}\ N}{PM\ \mathcal{R}\ PN} \qquad (\sigma)\ \frac{M\ \mathcal{R}\ N}{N\ \mathcal{R}\ M}$$

$$(\gamma)\ \frac{M\ \mathcal{R}\ N}{M\lceil P\ \mathcal{R}\ N\lceil P} \qquad (\tau)\ \frac{M\ \mathcal{R}\ N \quad N\ \mathcal{R}\ P}{M\ \mathcal{R}\ P}$$

$$(\delta)\ \frac{N\ \mathcal{R}\ P}{M\lceil N\ \mathcal{R}\ M\lceil P} \qquad (\xi)\ \frac{M\ \mathcal{R}\ N}{\lambda x.M\ \mathcal{R}\ \lambda x.P}$$

Il est facile de passer d'un formalisme à l'autre, $M\lceil\{N_1,\cdots,N_n\}$ peut se traduire dans la deuxième présentation par $(\cdots(M\lceil N_1)\cdots N_n)$ et il est aisé de montrer ainsi l'équivalence de ces deux formalismes.

La réduction β_p s'écrit:

$$\beta_p : (\lambda x.M)N = M[N/x]\lceil N$$

Si les termes M et N sont équivalents par la relation d'équivalence engendrée par les axiomes et les règles de la théorie de la restriction, on le notera $T\lceil\vdash M = N$

Si les termes M et N sont équivalents par la relation d'équivalence engendrée par les axiomes et les règles de la théorie de la restriction et l'axiome β_p, on le notera $\lambda_p \vdash M = N$

Les *valeurs* (que nous nous dirons encore termes *définis* ou *totaux*) du λ_p-calcul sont les termes M tels qu'il existe un terme N qui est une abstraction, une variable ou une constante tel que $T\lceil\vdash M = N$.

Les termes sans restriction seront appelés *termes purs.*

Maintenant nous sommes en mesure d'identifier les termes

P_1 et P_2, en effet $P_1\beta_p c\lceil\Omega\Omega, \Omega$ et $P_2\beta_p c\lceil\Omega, \Omega\Omega$, donc $\lambda_p \vdash P_1 = P_2$. Rien ne nous dit que nous ayons ainsi identifié tous les programmes qu'il serait souhaitable d'identifier[1], mais nous avons du moins progressé dans cette voie et obtenu un langage plus expressif.

1.2 Petit panorama des résultats existant sur le λ_p-calcul

Le formalisme plus délicat de Moggi lui avait néanmoins permis de définir une réduction parallèle $\rightarrow_1$ qui est une sorte de réduction par niveau en ceci qu'elle ne vérifie pas la règle xi, dont il a prouvé la confluence pour les $p\lambda$-termes fortement normalisables. Nous verrons que le formalisme que nous avons adopté et qui consiste à considérer un calcul paresseux conduit à un résultat qui peut être mis en parallèle à celui de Moggi, mais par des techniques toutes différentes.

[1] un souhait plus exigeant resterait à formuler. Remarquons à cet endroit que la structure des termes du λ-calcul dont la réduction de tête ne termine pas, peuvent être distingués selon leurs propriétés: termes récurrents, h-cycliques, faciles, etc...(*cf* par exemple C.Szylberach [Szylberach].) Il semble probable qu'on puisse identifier davantage de programmes, y compris dans un mécanisme d'appel par valeur, mais la syntaxe de ces identifications reste à élaborer.

Si divers résultats de caractérisation des termes remarquables de λ_v existaient déjà (et nous les réévoquerons plus tard), rien n'avait à ma connaissance élaboré de tel pour le λ_p-calcul, ni bien sûr pour le λ_{vh}-calcul que nous introduisons et qui nous a servi de contrepoint: peut-on lui appliquer les mêmes constructions qu'au λ_v-calcul, est-il associé à un λ_p-calcul qui aît des liens non moins naturels avec d'autres formalismes et modèles: question que nous n'avons à vrai dire fait qu'effleurer.

Moggi a développé une étude de diverses logiques portant sur les fonctions partielles et les modèles correspondants, notamment dans le cadre du λ-calcul. R.Pino Pérez à sa suite a construit divers modèles et développé certaines notions d'algèbres combinatoires, mais un certain nombre de difficultés ont perduré concernant cette notion d'algèbre combinatoire et des logiques partielles où elles trouvent leur cadre. Nous montrons qu'il est tout à fait possible de se placer dans une logique classique pour dégager une notion d'algèbre combinatoire qui soit sans doute plus maniable que les précédentes.

Enfin notons que R.Pino Pérez a défini une notion de C-monoïde partiel qui est le contrepoint du cas classique et qui est à rapprocher de l'interprétation catégorique que nous définissons (les deux approches reposent sur la présentation équationnelle des PCCC). Celle-ci reforme le lien entre les interprétations dans catégories cartésiennes fermées partielles sous un jour nouveau (rappelons que c'est de là qu'était parti Moggi). Je n'ai eu qu'une connaissance indirecte de la thèse de Luke Ong dont certains résultats sur le $\lambda_{L,C}$-calcul semblent à rapprocher de ceux qui sont ici présentés (*cf* [Ong]).

2

Confluence

2.1 Introduction

Jusqu'à présent les résultats de confluence dans le lambda calcul partiel sont restés lacunaires et nous avons déjà évoqué a cet égard le résultat de Moggi au chapitre précédent (cf [Moggi 88]). Or cette propriété est évidemment souhaitable si l'on veut non seulement avoir un outil efficace et raisonnable pour étudier l'équivalence de progammes - ce qui est un des premiers intérêts présenté par le lambda calcul partiel - mais aussi avoir un mode de calcul qui puisse s'implémenter. Ici nous définissons un λ-calcul partiel paresseux. La notion de réduction considérée permet d'avoir une stratégie moins restrictive que celle du calcul par valeur tout en gardant l'heuristique (ce fait sera davantage explicité au chapitre 4). En effet, dans le calcul que nous considérons- et que appelerons λ_{pl}- les restrictions n'apparaissent pas sous une abstraction, et peuvent être regroupées en une seule restriction en quelque sorte accolée à un terme pur. Ainsi un terme de la forme $(M\lceil(N\lceil R))(P\lceil Q)$ s'égalera au terme $MP\lceil\{P,N,R\}$.

Ce calcul correspond à une stratégie d'évaluation paresseuse: aucune réduction n'est effectuée sous une abstraction, et la réduction est définie de telle sorte que les restrictions qui sont produites par les réductions n'apparaissent pas sous une abstraction (*cf* 2.4). Ainsi nous construisons une sorte de calcul faible intermédiaire entre le λ_v-calcul et le λ_p-calcul, et qui est au λ_v-calcul paresseux ce que le λ_p-calcul est au λ_v-calcul[1].

Nous sommes repartis de la version simplifiée proposée par Pino Pérez dans [Pino 87, Pino 89], mais en éliminant une des règles qu'il proposait (la règle $(\lambda x.M\lceil N)\lceil N = (\lambda x.M)\lceil N$, qui vaut si $x \notin VL(N)$, règle qui s'élimine ici naturellement, puisqu'elle n'a plus lieu de s'appliquer) et en substituant la règle $M\lceil M = M$ par deux autres qui lui sont équivalentes mais qui éliminent des paires critiques.

D'autre part nous nous sommes inspirés d'un article de G.Huet [Huet 80] pour trouver une nouvelle technique qui nous permette d'établir la confluence de notre calcul *modulo* l'équivalence des termes. Nous avons étés amenés à considérer ce calcul restreint, eu égard à la difficulté d'établir le même résultat pour le calcul tout entier, ce qui est lié à l'équation:

$$(\lambda x.M\lceil N)\lceil N = (\lambda x.M)\lceil N$$

qui interdit d'orienter les autres équations: elle suscite des paires critiques qui semblent inéliminables. Ceci est à rapprocher de la difficulté que présente la décidabilité de la théorie équationnelle de la restriction (*cf* [Pino 89, Pino 92]).

[1] En fait, il est possible d'imaginer des calculs qui se rapprochent davantage du λ_p-calcul tout en restant paresseux, ici encore voir le chapitre 4. Mais ceci n'auriat fait que renforcer la difficulté technique sans pour autant fournir un résultat substantiellement plus intéressant. Nous considérons les résultats de ce chapitre plutôt comme une approche, loin d'être pleinement satisfaisante, des techniques à mettre en œuvre pour prouver la confluence de λ_p.

D'autre part, nous avons dû légèrement modifier la syntaxe, les restrictions se faisant non plus par des ensembles, mais par des multi-ensembles de termes. Ceci est nécessaire pour avoir un contrôle de la la taille de réduction d'un terme. Nous expliquerons dans la section 2.5 le pourquoi de cette modification.

Nous montrons tout d'abord que les règles de la théorie de la restriction peuvent s'orienter en donnant un système de réécriture confluent et nœthérien et ceci d'autant plus facilement que la forme normale d'un terme relativement à ce système de réécriture peut se définir directement (mais moyennant un certain nombre de notions préalables) par induction sur ce terme.

Nous définissons ensuite plus précisément la réduction $\rightarrow_{\beta_{pl}}$ de notre calcul et étudions diverses propriétés qui prépare la preuve de la proposition 1 qui énonce que le système de réécriture issu de la théorie de la restriction préserve la normalisabilité forte.

A cette étape nous sommes dans des hypothèses où savons prouver la confluence de la réduction modulo cette théorie de la restriction pour les termes fortement normalisables. Nous montrons en effet qu'une sorte de locale confluence (qui se dédouble en deux propriétés α et γ^+) est vérifiée. La confluence en résulte pour les termes fortement normalisables par induction nœthérienne.

2.2 La syntaxe du calcul

Definition 1 *Les termes*

Le lambda calcul partiel contient un constructeur supplémentaire par rapport au lambda calcul pur, le constructeur $\lceil$, *appelé restriction. Ici nous considérons une variante* Λ_{pl} *du lambda calcul partiel dans laquelle les restrictions ne peuvent se trouver sous une abstraction. Plus formellement, on se donne deux ensembles prédéfinis* V *et* C, *l'un de variables, l'autre de constantes. Alors:*

terme partiel = *valeur pure*[2] *construit à partir de* V *et de* C,
MN, *où* M *et* N *sont des termes partiels,*
$M\lceil D$, *où* M *est un terme partiel et* D *un multi-ensemble fini de termes partiels.*

L'ensemble des termes sera noté Λ_{pl} . *Il est à remarquer que tous les termes purs sont également des termes partiels:* $\Lambda \subset \Lambda_{pl}$.

Rappelons qu'un multi-ensemble est un ensemble où les éléments sont comptés avec multiplicité. Cette autre divergence vis-à-vis du lambda calcul partiel usuellement considéré est nécessitée pour avoir un bon comportement par rapport à la taille de réduction des termes fortement normalisables (cf infra 14). Le multi-ensemble D *augmenté d'un élément* N *sera dénoté par* $D \sqcup N$.

On notera $\mathcal{P}_{mf}(\Lambda_{pl})$, *l'ensemble des multi-ensembles finis dont les élément appartiennent à* Λ_{pl}. *Si* $\mathcal{R}$ *est une relation sur* Λ_{pl}, *alors celle-ci sétend à* $\mathcal{P}_{mf}(\Lambda_{pl})$ *en posant* $D\mathcal{R}D'$, D *et* D' *étant deux multi-ensembles finis, si* $D = E \sqcup \{M\}$ *et* $D = E \sqcup \{M'\}$, *avec* $M\mathcal{R}M'$.

On identifiera M *et* $M\lceil\emptyset$.

Remarque 1 *Il n'y a pas de restriction sous les* λ. *En particulier, si* $\lambda x.M \in \Lambda_{pl}$ *alors* $\lambda x.M \in \Lambda$.

Definition 2 *Contexte factoriel*

Intuitivement, un contexte est un terme construit sur V *augmenté d'une variable particulière* [ ], *qui est -si l'on veut- un trou. Un contexte sera dit* factoriel *s'il n'y a aucune*

occurence de ce trou sous une abstraction. Un contexte factoriel sera dit propre *s'il y a au moins une occurence de* [ ], simple *s'il y en a une et une seule,* multiple *sinon.*

Plus formellement:

Contexte factoriel = [ ],
M, *où* M *est un terme partiel,*
CC', *où* C *et* C' *sont des contextes factoriels,*
$C\lceil D$, *où* C *est un contexte factoriel et* D *un multi-ensemble fini de contextes factoriels.*

$C[M/[\,]]$ *sera noté plus simplement* $C[M]$.

Remarque 2 *Cette notion de contexte factoriel est appropriée à notre calcul puisque on a la propriété suivante, qui se montre très simplement par induction sur* C*:*

Si C *est un contexte factoriel et* $M \in \Lambda_{pl}$ *alors* $C[M] \in \Lambda_{pl}$.

Definition 3 *Relation passant au contexte factoriel*

On dira qu'une relation $\mathcal{R}$ *sur* Λ_{pl} *passe au contexte factoriel si pour tout contexte factoriel simple* $C[\,]$, *on a:*

$$x\ \mathcal{R}\ y \Rightarrow C[x]\ \mathcal{R}\ C[y]$$

Remarque 3 *Si* $\mathcal{R}$ *est transitive, on peut omettre la mention* simple *dans la précédente définition. Pour le vérifier, il suffit de remarquer qu'un contexte factoriel multiple dans lequel on substitue une des occurences de* [ ] *par n'importe quel terme reste un contexte factoriel; il suffit dès lors d'appliquer récursivement la propriété précédente pour s'assurer du bien-fondé de cette remarque.* □

Proposition 1 *Une relation* $\mathcal{R}$ *passe au contexte factoriel si et seulement si, pour tout terme* M *et pour tout multi-ensemble* D *de termes:*

$$\begin{aligned} x\ \mathcal{R}\ y &\Rightarrow Mx\ \mathcal{R}\ My \\ &\Rightarrow xM\ \mathcal{R}\ yM \\ &\Rightarrow x\lceil D\ \mathcal{R}\ y\lceil D \\ &\Rightarrow M\lceil D \sqcup \{x\}\ \mathcal{R}\ M\lceil D \sqcup \{y\} \end{aligned}$$

Preuve: Par induction sur le contexte factoriel C. □

Remarque 4 *La plus petite relation (resp. relation d'équivalence)* $\mathcal{R}$ *passant au contexte factoriel et vérifiant une liste d'axiomes peut être vue comme ce qui est dérivable dans le système formé par ces axiomes et les règles suivantes:*

$$(\mu)\ \frac{M\ \mathcal{R}\ N}{MP\ \mathcal{R}\ NP} \qquad (\nu)\ \frac{M\ \mathcal{R}\ N}{PM\ \mathcal{R}\ PN} \qquad (\gamma)\ \frac{M\ \mathcal{R}\ N}{M\lceil D\ \mathcal{R}\ N\lceil D} \qquad (\delta)\ \frac{N\ \mathcal{R}\ P}{M\lceil D \sqcup \{N\}\ \mathcal{R}\ M\lceil D \sqcup \{P\}}$$

si $\mathcal{R}$ est une relation d'équivalence.

$$(\rho)\ \frac{}{M\ \mathcal{R}\ M} \qquad (\sigma)\ \frac{M\ \mathcal{R}\ N}{N\ \mathcal{R}\ M} \qquad (\tau)\ \frac{M\ \mathcal{R}\ N \quad N\ \mathcal{R}\ P}{M\ \mathcal{R}\ P}$$

Definition 4 *Nous définissons deux fonctions qui vont nous être très utiles:* $\mathsf{h} : \Lambda_{pl} \to \Lambda_{pl}$ *et* $\mathsf{r} : \Lambda_{pl} \to \mathcal{P}_{mf}(\Lambda_{pl})$ *comme suit:*

$$\begin{array}{ll} \mathsf{h}(x) = x & \mathsf{r}(x) = \emptyset \\ \mathsf{h}(c) = c & \mathsf{r}(c) = \emptyset \\ \mathsf{h}(\lambda x.M) = \lambda x.M & \mathsf{r}(\lambda x.M) = \emptyset \\ \mathsf{h}(MN) = \mathsf{h}(M)\mathsf{h}(N) & \mathsf{r}(MN) = \mathsf{r}(M) \sqcup \mathsf{r}(N) \\ \mathsf{h}(M\lceil D) = \mathsf{h}(M) & \mathsf{r}(M\lceil D) = \mathsf{r}(M) \sqcup D \end{array}$$

Exemple: *Soit* $M = (y(x\lceil\{xx\lceil(\lambda y.yy)xz\}))\lceil\{xy, (\lambda y.y)x\}$. *Alors*

$$\begin{aligned} \mathsf{h}(M) &= yx \\ \mathsf{r}(M) &= \{xy, (\lambda y.y)x, xx\lceil(\lambda y.yy)xz\} \end{aligned}$$

Remarque 5 *Si* M *est un terme pur alors* $\mathsf{h}(M) = M$ *et* $\mathsf{r}(M) = \emptyset$.

Definition 5 *On définit inductivement comme suit l'ensemble des facteurs d'un terme* M, *noté* $Fact(M)$:

$$\begin{aligned} Fact(x) &= \{x\}, \\ Fact(c) &= \{c\}, \\ Fact(\lambda x.M) &= \{\lambda x.M\}, \\ Fact(MN) &= Fact(M) \cup Fact(N) \cup \{\mathsf{h}(M)\mathsf{h}(N)\}, \\ Fact(M\lceil D) &= Fact(M) \cup \bigcup_{N \in D} Fact(N), \end{aligned}$$

Il est facile de vérifier que $Fact(M)$ *est un ensemble fini de termes purs.*

L'ensemble des facteurs propres de M, $Factp(M)$, *est défini comme* $Fact(M) - \{M\}$. *Pour un ensemble* D, *on posera:*

$Fact(D) = \bigcup_{N \in D} Fact(N)$ *et* $Factp(D) = \bigcup_{N \in D} Factp(N)$.

(On peut remarquer qu'on n'a pas $Factp(D) = Fact(D) - D$, *puisqu'il se peut que* $d, d' \in D$ *soient tels que* $d \in Factp(d')$ *).*

Lemme 1 *Soit* A *un ensemble fini.*

Si $B \subset Factp(A)$ *alors* $Factp(A - B) = Factp(A)$.

Preuve: La seule difficulté est de prouver $Factp(A) \subset Factp(A - B)$. Munissons $Fact(A)$ de la relation $\mathcal{R}$ définie comme suit:

$$M \mathcal{R} N \Leftrightarrow (M \in Factp(N) \text{ ou } M = N).$$

$\mathcal{R}$ est une relation d'ordre. Pour le vérifier, il suffit de prouver la transitivité en remarquant que $M \in Factp(N)$ implique $Fact(M) \subset Fact(N)$, ce qui se prouve par induction sur N. L'antisymétrie résulte de ce que - si $M \mathcal{R} N$ et $N \mathcal{R} M$ avec $M \neq N$ - alors $M \in Factp(N)$ et $N \in Factp(M)$, d'où $M \in Factp(M)$ par transitivité, ce qui est par définition exclu.

Soit $M \in Factp(A)$. Considérons l'ensemble des majorants de M, celui-ci étant fini, admet nécéssairement un élément maximal P, qui est donc tel que $P \notin Factp(A)$ (et en particulier $P \neq M$), d'où $P \notin B$ et enfin, en remarquant que les éléments de $Fact(A)$ qui ne sont pas dans $Factp(A)$ sont dans A on en déduit $P \in A - B$. Par conséquent $M \in Factp(P) \subset Factp(A - B)$. □

Lemme 2 *Si* M *est un terme pur alors:* $N \in Fact(M)$ *si et seulement si il existe un contexte factoriel* C, *tel que* $M = C[N]$.

Preuve: Par induction sur M. □

2.3 La théorie de la restriction T_{res}

Les termes du calcul sont à considérer modulo une théorie que nous précisons maintenant.

Definition 6 *T_{res} est la théorie constituée des axiomes suivants:*

1. $M\lceil\{x\} \sqcup D \equiv M\lceil D$,
2. $M\lceil\{c\} \sqcup D \equiv M\lceil D$,
3. $M\lceil\{\lambda x.N\} \sqcup D \equiv M\lceil D$,
4. $M(N\lceil D) \equiv MN\lceil D$,
5. $(M\lceil D)N \equiv MN\lceil D$,
6. $(M\lceil D)\lceil E \equiv M\lceil D \sqcup E$,
7. $M\lceil\{N, N\} \sqcup D \equiv M\lceil\{N\} \sqcup D$,
8. $M\lceil\{N\lceil D\} \sqcup E \equiv M\lceil\{N\} \sqcup D \sqcup E$,
9. $M\lceil\{C[N], N\} \sqcup D \equiv M\lceil\{C[N]\} \sqcup D$, *si N est un terme pur et C un contexte factoriel,*
10. $C[N]\lceil\{N\} \sqcup D \equiv C[N]\lceil D$, *si N est un terme pur et C un contexte factoriel,*

Au cas, où plusieurs règles peuvent s'appliquer, on peut supposer avoir appliquer la premiére d'entre elles, dans l'ordre où elles sont écrites, et ceci sans ambiguïté puisqu'elles fournissent alors le même résultat.

$\overset{T_{res}}{=}$ est la plus petite relation d'équivalence $\equiv$ passant au contexte factoriel et vérifiant les axiomes de T_{res}.

La plus petite relation passant au contexte factoriel et vérifiant ces mêmes axiomes sera notée $\rightarrow_S$, celle-ci peut être vue comme une relation de réécriture.

Lemme 3 *Si $M \rightarrow_S M'$ (ou mieux si $M \overset{T_{res}}{=} M'$) alors $\mathsf{h}(M) = \mathsf{h}(M')$.*

Preuve: Par induction sur la dérivation de $M \rightarrow_S M'$ (ou de $M \overset{T_{res}}{=} M'$) (*cf* la remarque 4). □

Lemme 4 $M \overset{T_{res}}{=} \mathsf{h}(M)\lceil\mathsf{r}(M)$.

Preuve: Une simple induction sur M (en appliquant les axiomes 4,5 et 6). □

Definition 7 *On définit le simplifié d'un terme M comme le terme $S(M) = S_1(M)\lceil S_2(M)$, où $S_2(M)$ est un "vrai" ensemble, qu'on identifiera à un multi-ensemble dont les éléments*

sont tous de multiplicité 1, défini inductivement comme suit:

$$S_1(x) = x \qquad S_2(x) = \emptyset$$
$$S_1(c) = c \qquad S_2(c) = \emptyset$$
$$S_1(\lambda x.N) = \lambda x.N \qquad S_2(\lambda x.N) = \emptyset$$
$$S_1(MN) = S_1(M)S_1(N) \qquad S_2(MN) = \begin{pmatrix} S_2(M) \\ \cup \\ S_2(N)) \end{pmatrix} - \begin{pmatrix} Fact(S_1(M)S_1(N)) \\ \cup \\ Factp(S_2(M) \cup S_2(N)) \end{pmatrix}$$

$$S_1(M\lceil D) = S_1(M) \qquad S_2(M\lceil D) = \begin{pmatrix} S_2(M) \\ \cup \\ S_1(D)^* \\ \cup \\ S_2(D) \end{pmatrix} - \begin{pmatrix} Fact(S_1(M)) \\ \cup \\ Factp \begin{pmatrix} S_2(M) \\ \cup \\ S_2(D) \\ \cup \\ S_1(D)^* \end{pmatrix} \end{pmatrix}$$

où D^* est l'ensemble D privé de tous les termes qui sont des variables, des constantes ou des abstractions.

Il est aisé de vérifier par induction que $S_1(M)$ n'est autre que $\mathsf{h}(M)$ et que $S_2(M)$ est un ensemble de termes purs tel que $S_2(M)^* = S_2(M)$.

Lemme 5 *$S(M)$ est normal relativement à la relation $\rightarrow_S$.*

Preuve: Le fait que $S_1(M)$ est un terme pur et que $S_2(M)$ est un ensemble de termes purs tel que $S_2(M)^* = S_2(M)$ permet de s'assurer que les règles de 1 à 8 ne peuvent plus s'appliquer. Le lemme 2 et une induction sur M permettent de vérifier que $S_2(M) = S_2(M)^* - (Fact(S_1(M) \cup Factp(S_2(M))$, ce qui exclut que les deux dernières règles puissent s'appliquer. Ceci signifie que toutes les suppressions possibles ont déjà été effectuées. □

Remarque 6 *Si M est un terme pur et D un ensemble de termes purs alors:*
$M\lceil D \xrightarrow{*}_S M\lceil(D - (Fact(M) \cup Factp(D))$

Preuve: le lemme 2 permet de voir qu'il suffit d'appliquer les règles 9 et 10. □

Lemme 6 *Pour tout terme M , $M \xrightarrow{*}_S S(M)$.*

Preuve: Ceci se prouve par induction sur M.

Si $M = x, c$ ou $\lambda x.N$, c'est immédiat.

Si $M = NP$ alors par induction:

$$NP \underset{\text{induction}}{\xrightarrow{*}_S} (S_1(N)\lceil S_2(N))(S_1(P)\lceil S_2(P))$$

d'où: $NP \underset{\text{règles 4,5,6,7}}{\xrightarrow{*}_S} S_1(N)S_1(P)\lceil S_2(N) \cup S_2(P)$

Enfin la remarque 6 permet d'obtenir:

$$NP \xrightarrow{*}_S S_1(N)S_1(P)\lceil \begin{pmatrix} S_2(N) \\ \cup \\ S_2(P)) \end{pmatrix} - \begin{pmatrix} Fact(S_1(N)S_1(P)) \\ \cup \\ Factp(S_2(N) \cup S_2(P)) \end{pmatrix}$$

Si $M = N\lceil D$, alors:

$$N\lceil D \xrightarrow[\text{induction et règle 7}]{}{}^{*}_{S} (S_1(N)\lceil S_2(N))\lceil\{S_1(Q)\lceil S_2(Q), Q \in D\} \xrightarrow[\text{règles 1 à 8}]{}{}^{*}_{S} S_1(N)\lceil(S_2(N) \cup S_1(D)^* \cup S_2(D))$$

La remarque 6 permet de voir que:

$$N\lceil D \xrightarrow{*}_S S_1(N)\lceil \begin{pmatrix} S_2(N) \\ \cup \\ S_1(D)^* \\ \cup \\ S_2(D) \end{pmatrix} - \begin{pmatrix} Fact(S_1(N)) \\ \cup \\ Factp \begin{pmatrix} S_2(N) \\ \cup \\ S_1(D)^* \\ \cup \\ S_2(D)) \end{pmatrix} \end{pmatrix}$$

D'où le résultat. □

Lemme 7 *Si* $M \overset{T_{res}}{=} M'$ *alors* $S(M) = S(M')$.

Preuve: La démonstration se fait par induction sur la dérivation de $M \overset{T_{res}}{=} M'$.

•Pour les axiomes:

Pour les axiomes 1, 2 et 3 c'est élémentaire.

Pour l'axiome 4: $M(N\lceil D) \overset{T_{res}}{=} MN\lceil D$, par définition de S:

$$S(M\lceil(N\lceil D)) =$$

$$S_1(M)S_1(N)\lceil \begin{pmatrix} S_2(M) \\ \cup \\ (A - B) \end{pmatrix} - \begin{pmatrix} Factp \begin{pmatrix} S_2(M) \\ \cup \\ S_2(N\lceil D) \end{pmatrix} \\ \cup \\ Fact(S_1(M)S_1(N)) \end{pmatrix}$$

avec

$$A = S_2(N) \cup S_1(D)^* \cup S_2(D)$$

$$B = Fact(S_1(N)) \cup Factp(S_2(N) \cup S_2(D) \cup S_1(D)^*)$$

comme $Fact(S_1(N)) \subset Fact(S_1(M)S_1(N))$, on peut conserver l'égalité en supprimant $Fact(S_1(N))$ et d'autre part le lemme 1 permet de constater que:

$$Factp \begin{pmatrix} S_2(N) \\ \cup \\ S_2(D) \\ \cup \\ S_1(D)^*) \end{pmatrix} \subset Factp(S_2(N\lceil D)) \cup Fact(S_1(M)S_1(N))$$

et

$$S(M\lceil(N\lceil D)) =$$
$$S_1(M)S_1(N)\lceil\begin{pmatrix} S_2(M) \\ \cup \\ S_2(N) \\ \cup \\ S_1(D)^* \\ \cup \\ S_2(D) \end{pmatrix} - \begin{pmatrix} Fact(S_1(M)S_1(N)) \\ \cup \\ Factp\begin{pmatrix} S_2(M) \\ \cup \\ S_2(N) \\ \cup \\ S_1(D)^* \\ \cup \\ S_2(D) \end{pmatrix} \end{pmatrix}$$

et un calcul analogue prouve qu'on trouve le même résultat pour $S(MN\lceil P)$.

Pour les axiomes de 5 à 8: ces cas sont analogues au précédent et leur résolution tiennent au lemme 1.

pour l'axiome 9: $M\lceil\{C[N], N\} \sqcup D \overset{T_{res}}{=} M\lceil\{C[N]\} \sqcup D$, si N est un terme pur et C un contexte factoriel, alors on peut supposer $N \neq C[N]$ (sinon on peut appliquer la règle 7) et le lemme 2 nous dit que $N \in Factp(C[N])$, ce dont il est facile de déduire l'égalité des simplifiés.

pour l'axiome 10: $C[N]\lceil\{N\} \sqcup D \overset{T_{res}}{=} C[N]\lceil D$, si N est un terme pur et C un contexte factoriel, ici le lemme 2 nous dit que $N \in Fact(C[N])$, ce dont il est facile de déduire l'égalité des simplifiés.

•Pour les règles: L'hypothèse d'induction donne immédiatement le résultat. □

Théorème 1 *La relation $\rightarrow_S$ est noethérienne et confluente, de plus la forme normale d'un terme M est $S(M)$. Par conséquent T_{res} est décidable ($M \overset{T_{res}}{=} N \Leftrightarrow S(M) = S(N)$).*

Preuve:

• $\rightarrow_S$ est noethérienne, une mesure qui le prouve est le couple formé par la taille du terme (nombre de caractères de son écriture) et la somme des profondeurs des restrictions, considéré dans l'ordre lexicographique. Il est facile de voir que les règles qui suppriment quelque chose diminuent la taille, les autres laissent celle-ci inchangée mais font remonter les restrictions.

• $\rightarrow_S$ est confluente et n'admet qu'une unique forme normale. En effet si $M \overset{*}{\rightarrow}_S M_1$ et $M \overset{*}{\rightarrow}_S M_2$, alors on a $M_1 \overset{T_{res}}{=} M_2$, d'òu par le lemme 7 $S(M_1) = S(M_2)$ et par le lemme $6 M_i \overset{*}{\rightarrow}_S S(M_i)$ pour $i = 1, 2$.

•Enfin $\overset{T_{res}}{=}$ n'est autre que la clôture réflexive,transitive et symétrique de $\rightarrow_S$. □

2.4 La réduction $\rightarrow_{\beta_{pl}}$

Comme nous l'avons indiqué dans l'introduction de ce chapitre, la notion de réduction de notre calcul tient sa définition un peu particulière à la double exigence de garder l'heuristique du lambda calcul par valeur, mais toujours dans l'esprit du λ_p-calcul - soit en gardant en mémoire les termes dont on aura à vérifier qu'ils sont des valeurs - et d'être cohérent avec l'idée qu'aucune restriction ne peut se trouver sous une abstraction. Dans la réduction de notre calcul restreint Λ_{pl}, les restrictions qui figurent dans l'argument d'un rédex ne peuvent passer sous la portée d'une abstraction, on doit donc les garder comme restrictions du contractum

et c'est ce que permet la fonction r. Ceci n'est pas nécessaire lorsque le corps de la fonction est un contexte factoriel de la variable à substituer.

Definition 8 *La relation $\rightarrow_{\beta_{pl}}$ est la plus petite relation passant au contexte factoriel vérifiant:*

Si $\mathsf{h}(M) = \lambda x.P$ *alors:*

$MN \rightarrow_{\beta_{pl}} P[\mathsf{h}(N)]\lceil(\mathsf{r}(M) \sqcup \{N\})$ *si N est une valeur, c'est-à-dire* $\mathsf{h}(N)$ *est une abstraction, une variable ou une constante ou bien si P est un contexte factoriel en x (i.e. $P[[\ ]/x]$ est un contexte factoriel).*

On vérifie aisément que les conditions imposées sur cette réduction en font bien une relation définie sur Λ_{pl}.

Exemple:

1. $\mathsf{h}((\lambda z.z)\lceil u) = \lambda z.z$ est une valeur, d'où:

$$((\lambda xy.y(\lambda v.x))\lceil ab)((\lambda z.z)\lceil u) \rightarrow_{\beta_{pl}} (\lambda y.y(\lambda v.(\lambda z.z)))\lceil\{ab, (\lambda z.z)\lceil u)\}$$

2. yx est un contexte factoriel en x, d'où:

$$((\lambda x.yx)\lceil a)(v(w\lceil c)) \rightarrow_{\beta_{pl}} y(vw)\lceil\{a, c\}$$

Remarquons en particulier que les termes P_1 et P_2 de notre introduction sont bien égalés par λ_{pl}.

Remarque 7 *Si nous notons* $\lambda_{pl} \vdash \mathrm{M} = \mathrm{N}$ *la relation d'équivalence engendrée par* $\overset{T_{res}}{=}$ *et $\rightarrow_{\beta_{pl}}$, alors il est facile de montrer que si $\Lambda_{pl} \vdash M = N$ alors $\lambda_p \vdash M = N$. En effet les termes obtenus par l'une ou l'autre des réductions d'un même rédex sont équivalents par $T\lceil$. En effet, si $M \rightarrow_{\beta_{pl}} P$ alors il existe $P' \in \Lambda_P$ tel que $T\lceil \vdash P = P'$.*

Soient D et D' deux multi-ensembles, on notera $D \rightarrow_{\beta_{pl}} D'$ si $D = \{M\} \sqcup E$ et $D' = \{M'\} \sqcup E$, avec $M \rightarrow_{\beta_{pl}} M'$.

Lemme 8 *Si $M \rightarrow_{\beta_{pl}} M'$ et* $\mathsf{h}(M)$ *est normal alors* $\mathsf{r}(M) \rightarrow_{\beta_{pl}} \mathsf{r}(M')$ *et* $\mathsf{h}(M) = \mathsf{h}(M')$.

Preuve: Par induction sur la dérivation de $M \rightarrow_{\beta_{pl}} M'$. □

Definition 9 *Nous appellerons taille de réduction d'un terme $M \in \Lambda_{pl}$ et nous noterons $\nu(M)$ la longueur maximale (et éventuellement infinie) d'une réduction partant du terme M, et $\nu(D)$ la somme des $\nu(M)$ pour M parcourant le multi-ensemble fini D*

Il est à noter que $\nu(D \sqcup E) = \nu(D) + \nu(E)$, puisque $\nu(M)$ est compté autant de fois que M figure dans le multi-ensemble considéré.

Voici une remarque aussi simple qu'utile:

Remarque 8 $\nu(M\lceil D) = \nu(M) + \nu(D)$

Preuve: Il suffit de remarquer que si $M\lceil D \rightarrow_{\beta_{pl}} P$ alors soit $P = M'\lceil D$ avec $M \rightarrow_{\beta_{pl}} M'$, soit $P = M\lceil D'$ avec $D \rightarrow_{\beta_{pl}} D'$. □

Lemme 9 *(M est normal pour $\rightarrow_{\beta_{pl}}$) $\Leftrightarrow$ ($\mathsf{h}(M)\lceil\mathsf{r}(M)$ est normal pour $\rightarrow_{\beta_{pl}}$)*

Preuve: Montrons ce résultat par induction sur M :

- Si M est une valeur, alors c'est évident, puisque $M = \mathsf{h}(M)\lceil\mathsf{r}(M)$.
- Si $M = PQ$, alors s'il y a un rédex dans M, soit il se trouve dans P ou dans Q, et alors par hypothèse d'induction il y en a un dans $\mathsf{h}(M)\lceil\mathsf{r}(M) = \mathsf{h}(P)\mathsf{h}(Q)\lceil\mathsf{r}(P) \sqcup \mathsf{r}(Q)$, soit PQ est un rédex en vertu du fait que $\mathsf{h}(P)$ est une abstraction, mais alors $\mathsf{h}(P)\mathsf{h}(Q)$ est aussi un rédex.

  Réciproquement, si il existe un rédex dans $\mathsf{h}(M)\lceil\mathsf{r}(M)$ il est soit dans $\mathsf{h}(P)$ ou $\mathsf{r}(P)$, auquel cas il en existe un par hypothèse d'induction dans P et donc dans M, soit dans $\mathsf{h}(Q)$ ou $\mathsf{r}(Q)$ auquel cas il en existe un dans Q, soit $\mathsf{h}(P)$ est une abstraction et alors M est lui-même un rédex.
- Si $M = P\lceil D$, alors s'il y a un rédex dans M, soit il se trouve dans P et alors par hypothèse d'induction , il en existe un dans $\mathsf{h}(P)\lceil\mathsf{r}(P)$, et par conséquent dans $\mathsf{h}(M)\lceil\mathsf{r}(M) = \mathsf{h}(P)\lceil\mathsf{r}(P) \sqcup D$, soit il se trouve dans D, et c'est évident.

  Réciproquement, si il existe un rédex dans $\mathsf{h}(M)\lceil\mathsf{r}(M)$ il est soit dans $\mathsf{h}(P)$, soit dans $\mathsf{r}(P)$, soit dans D, et dans les deux premiers on conclut par hypothèse d'induction et le troisième est un évidence. □

Remarque 9

$$\nu(\mathsf{h}(P)\lceil\mathsf{r}(P)) \leq \nu(\mathsf{h}(PQ)\lceil\mathsf{r}(PQ))$$
$$\nu(\mathsf{h}(Q)\lceil\mathsf{r}(Q)) \leq \nu(\mathsf{h}(PQ)\lceil\mathsf{r}(PQ))$$
$$\nu(\mathsf{h}(N)\lceil\mathsf{r}(N)) \leq \nu(\mathsf{h}(N\lceil D)\lceil\mathsf{r}(N\lceil D))$$

Preuve:

Pour les deux premières assertions: $\mathsf{h}(PQ)\lceil\mathsf{r}(PQ) = \mathsf{h}(P)\mathsf{h}(Q)\lceil\mathsf{r}(P) \sqcup \mathsf{r}(Q)$, et par la remarque 8 $\nu(\mathsf{h}(PQ)\lceil\mathsf{r}(PQ)) = \nu(\mathsf{h}(P)\mathsf{h}(Q)) + \nu(\mathsf{r}(P)) + \nu(\mathsf{r}(Q))$, $\nu(\mathsf{h}(P)\lceil\mathsf{r}(P)) = \nu(\mathsf{h}(P)) + \nu(\mathsf{r}(P))$ et idem pour Q. Il suffit pour conclure de remarquer que $\nu(\mathsf{h}(P))$ et $\nu(\mathsf{h}(Q))$ sont tous deux plus petits que $\nu(\mathsf{h}(P)\mathsf{h}(Q))$.

Pour la dernière : la remarque 8 montre que $\nu(\mathsf{h}(N\lceil D)\lceil\mathsf{r}(N\lceil D)) = \nu(\mathsf{h}(N)\lceil\mathsf{r}(N)) + \nu(D)$.

□

Lemme 10 *Soit un terme M tel que M ou $\mathsf{h}(M)\lceil\mathsf{r}(M)$ soient fortement normalisable, alors:*

$$\nu(M) = \nu(\mathsf{h}(M)\lceil\mathsf{r}(M))$$

Preuve: Nous allons en fait montrer par induction lexicographique sur le couple (n, M) la propriété $\mathcal{P}_{n,M}$:

Si $inf\{\nu(M), \nu(\mathsf{h}(M)\lceil\mathsf{r}(M))\} \leq n$, alors:
$\nu(M) = \nu(\mathsf{h}(M)\lceil\mathsf{r}(M))$ et de plus

1. si $M \to_{\beta_{pl}} U$, alors il existe V tel que $\mathsf{h}(M)\lceil\mathsf{r}(M) \to_{\beta_{pl}} V$ avec :
   $\mathsf{h}(V) = \mathsf{h}(U)$, $\nu(\mathsf{r}(V)) = \nu(\mathsf{r}(U))$ et $\nu(U) = \nu(V)$.
2. si $\mathsf{h}(M)\lceil\mathsf{r}(M) \to_{\beta_{pl}} Q$, alors il existe U tel que $M \to_{\beta_{pl}} U$ avec :
   $\mathsf{h}(U) = \mathsf{h}(V)$, $\nu(\mathsf{r}(U)) = \nu(\mathsf{r}(V))$ et $\nu(U) = \nu(V)$.

Le lemme est manifestement vérifié si la propriété $\mathcal{P}_{n,M}$ est vraie pour tout couple $(n, M) \in \mathbb{N} \times \Lambda_{pl}$, la première partie de cette propriété donnant le résultat souhaité.

•Le cas $n = 0$ a été en fait l'objet du lemme 9.

•Pour $n + 1$

Il suffit alors de montrer les points *1* et *2*. En effet, $\nu(M) = \nu(\mathsf{h}(M)\lceil\mathsf{r}(M))$ apparaît comme une conséquence de ces deux points associés à l'hypothèse d'induction : les points *1* et *2* permettent de constater qu'au plus long chemin de réduction de M, quelqu'en soit la première étape, répond un chemin de réduction de même longueur de $\mathsf{h}(M)\lceil\mathsf{r}(M)$, et réciproquement.

On utilisera ici le fait que si $N = P$ ou $N = Q$, et $M = PQ$, ou encore si $M = N\lceil D$, alors le couple $(\nu(N), N)$ est strictement plus petit que le couple $(\nu(M), M)$, et de même le couple $(\nu(\mathsf{h}(N)\lceil\mathsf{r}(N)), N)$ est strictement plus petit que le couple $(\nu(\mathsf{h}(M)\lceil\mathsf{r}(M)), M)$ (*cf* la remarque 9). Il en ressort que l'un des couples $(\nu(N), N)$ et $(\nu(\mathsf{h}(N)\lceil\mathsf{r}(N)), N)$ est strictement plus petit que le couple n, M. Mais alors l'hypothèse d'induction nous dit qu'ils sont égaux. Ce couple $(\nu(N), N) = (\nu(\mathsf{h}(N)\lceil\mathsf{r}(N)), N)$ est donc plus petit que les deux couples $(\nu(M), M)$ et $(\nu(\mathsf{h}(M)\lceil\mathsf{r}(M)), M)$, et donc plus petit que leur inf, c'est-à-dire que n. Ce qui fait qu'on peut appliquer l'hypothèse d'induction sur N, sans se soucier de savoir si on se place sous l'hypothèse que $\nu(M) \leq n$ ou sous l'hypothèse que $\nu(\mathsf{h}(M)\lceil\mathsf{r}(M)) \leq n$.

Montrons *1* et *2* par induction sur M:

- Dans tous les cas, si M est une valeur, alors $M = \mathsf{h}(M)\lceil\mathsf{r}(M)$ et c'est immédiat.

- Si $M = PQ$:

  Pour *1*:

  Si on réduit P en P' dans M $(U = P'Q)$, alors par hypothèse d'induction sur M, il existe P_1 et D_1 tels que $\mathsf{h}(P_1) = \mathsf{h}(P')$, $\mathsf{r}(P_1\lceil D_1) = \mathsf{r}(P_1) \sqcup D_1 = \mathsf{r}(P')$ et vérifiant en outre $\mathsf{h}(P')\lceil\mathsf{r}(P') \rightarrow_{\beta_{pl}} P_1\lceil D_1$.

  Posons $V = P_1\mathsf{h}(Q)\lceil D_1 \sqcup \mathsf{r}(Q)$

  Alors $\mathsf{h}(PQ)\lceil\mathsf{r}(PQ) = \mathsf{h}(P)\mathsf{h}(Q)\lceil\mathsf{r}(P) \sqcup \mathsf{r}(Q) \rightarrow_{\beta_{pl}} V$,

  $\mathsf{h}(V) = \mathsf{h}(P_1)\mathsf{h}(Q) = \mathsf{h}(P')\mathsf{h}(Q) = \mathsf{h}(P'Q)$,

  $\mathsf{r}(V) = \mathsf{r}(P_1) \sqcup D_1 \sqcup \mathsf{r}(Q) = \mathsf{r}(P') \sqcup \mathsf{r}(Q) = \mathsf{r}(U)$, d'où $\nu(U) = \nu(V)$.

  Si on réduit Q en Q' dans M $(U = PQ')$, alors le raisonnement est similaire à celui que nous venons de voir, l'hypothèse d'induction s'appliquant cette fois à Q.

  Si PQ est un rédex que l'on réduit, soit $\mathsf{h}(P) = \lambda x.R$ et $U = R[\mathsf{h}(Q)/x]\lceil\mathsf{r}(P) \sqcup \{Q\}$. Alors $\mathsf{h}(P)\mathsf{h}(Q)$ est aussi un rédex, dont la réduction conduit à $V = (R[\mathsf{h}(Q)/x]\lceil\{\mathsf{h}(Q)\})$ $\lceil\mathsf{r}(P) \sqcup \mathsf{r}(Q)$. $\mathsf{h}(U)$ et $\mathsf{h}(V)$ sont manifestement identiques et $\nu(\mathsf{r}(U)) = \nu(\mathsf{r}(P)) + \nu(Q)$. Par hypothèse d'induction sur Q on a $\nu(Q) = \nu(\mathsf{h}(Q)\lceil\mathsf{r}(Q))$, d'où $\nu(\mathsf{r}(U)) = \nu(\mathsf{r}(V))$.

  Pour *2*:

  Si on réduit $\mathsf{h}(P)\mathsf{h}(Q)\lceil\mathsf{r}(P) \sqcup \mathsf{r}(Q)$ en réduisant l'un des sous-termes parmi $\mathsf{h}(P)$, $\mathsf{h}(Q)$, $\mathsf{r}(P)$, $\mathsf{r}(Q)$, pour le terme V. Alors il suffit d'appliquer l'hypothèse d'induction sur P ou sur Q. Traitons par exemple le cas où l'on a réduit $\mathsf{r}(Q)$ en D. Alors $V = \mathsf{h}(P)\mathsf{h}(Q)\lceil\mathsf{r}(P) \sqcup D$ et $\mathsf{h}(Q)\lceil\mathsf{r}(Q) \rightarrow_{\beta_{pl}} \mathsf{h}(Q)\lceil D$, d'où par hypothèse d'induction sur Q, il existe W tel que $Q \rightarrow_{\beta_{pl}} W$, $\mathsf{h}(W) = \mathsf{h}(Q)$ et $\mathsf{r}(W) = D$. Alors $PQ \rightarrow_{\beta_{pl}} PW = U$. $\mathsf{h}(V) = \mathsf{h}(U)$ va de soi et $\mathsf{r}(U) = \mathsf{r}(PW) = \mathsf{r}(P) \sqcup D = \mathsf{r}(V)$

  Si on réduit le rédex $\mathsf{h}(P)\mathsf{h}(Q)$, alors de l'autre côté on réduit le rédex PQ, et on déjà vu dans le cas de *1* que les termes U et V vérifient les relations souhaitées.

- Si $M = P\lceil D$

  Pour *1*:

  Si on réduit P en P' ($U = P'\lceil D$), alors on sait par hypothèse d'induction sur P que $\mathsf{h}(P)\lceil\mathsf{r}(P) \to_{\beta_{pl}} P_1\lceil D_1$ avec $\mathsf{h}(P_1) = \mathsf{h}(P')$ et $\mathsf{r}(P_1) \sqcup \mathsf{r}(D_1) = \mathsf{r}(P')$. Vérifions que $V = P_1\lceil D_1 \sqcup D$ convient:

$$\mathsf{h}(V) = \mathsf{h}(U) = \mathsf{h}(P')$$

$$\mathsf{r}(V) = \mathsf{r}(P_1) \sqcup \mathsf{r}(D_1) \sqcup \mathsf{r}(D) = \mathsf{r}(P') \sqcup \mathsf{r}(D) = \mathsf{r}(U)$$

  Si on réduit D en D' ($U = P\lceil D'$), alors $D = E \sqcup \{N\}$ et $D' = E \sqcup \{N'\}$. Alors $\mathsf{h}(M\lceil E \sqcup \{N\})\lceil\mathsf{r}(M\lceil E \sqcup \{N\}) = \mathsf{h}(M)\lceil\mathsf{r}(M) \sqcup E \sqcup \{N\} \to_{\beta_{pl}} \mathsf{h}(M)\lceil\mathsf{r}(M) \sqcup E \sqcup \{N'\} = V$. On vérifie alors aisément que $\mathsf{h}(U)\lceil\mathsf{r}(U) = V = \mathsf{h}(V)\lceil\mathsf{r}(V)$, à partir de quoi on conclut aisément.

  Pour *2*: Remarquons que $\mathsf{h}(P\lceil D)\lceil\mathsf{r}(P\lceil D) = \mathsf{h}(P)\lceil\mathsf{r}(P) \sqcup D$, d'où :

  Si $\mathsf{h}(P)\lceil\mathsf{r}(P) \sqcup D \to_{\beta_{pl}} P_1\lceil D_1 \sqcup D = V$, alors par hypothèse d'induction sur P, comme $\mathsf{h}(P)\lceil\mathsf{r}(P) \to_{\beta_{pl}} P_1\lceil D_1$, $P \to_{\beta_{pl}} U_1$ tel que $\mathsf{h}(U_1) = \mathsf{h}(P_1)$ et $\mathsf{r}(U_1) = \mathsf{r}(P_1) \sqcup D_1$, d'où $P\lceil D \to_{\beta_{pl}} U_1\lceil D = U$ et on a:

$$\mathsf{h}(U) = \mathsf{h}(U_1) = \mathsf{h}(P_1) = \mathsf{h}(V)$$

$$\mathsf{r}(U) = \mathsf{r}(U_1) \sqcup D = \mathsf{r}(P_1) \sqcup D_1 \sqcup D = \mathsf{r}(V)$$

  Si on réduit D, alors on réduit un certain terme $N \in D$ en N'. Il suffit de faire la même chose dans $P\lceil D$ et nous avons déjà vu dans le cas de *1* que les termes U et V vérifient les relations souhaitées. □

2.5 Stabilité par $\xrightarrow{*}_S$ de la normalisabilité forte

Nous allons maintenant prouver la proprosition suivante, qui exprime que la classe des termes fortement normalisables est stable par simplification (par $\xrightarrow{*}_S$):

Proposition 1 *Si $M \xrightarrow{*}_S M'$ et si M est fortement normalisable, alors M' est fortement normalisable.*

Cette proprosition va apparaître comme un corollaire immédiat du lemme suivant :

Lemme 11
Soit U un terme fortement normalisable et U' tel que $U \xrightarrow{*}_S U'$, alors $\nu(U) \geq \nu(U')$ et $\nu(\mathsf{r}(U)) \geq \nu(\mathsf{r}(U'))$.

Preuve: Nous allons démontrer ce résultat par induction lexicographique sur le couple formé par $n = \nu(U)$ et par la preuve de $U \xrightarrow{*}_S U'$ (*cf* la remarque 4).

Si $n = 0$, il s'agit de vérifier que si U est normal et que $U \xrightarrow{*}_S U'$, alors U' est également normal. Il suffit donc de considérer les différents axiomes et règles d'inférences pour constater que si il existe un rédex dans U, alors il en existe également un dans U'. Nous laissons au lecteur le soin d'effectuer ce rapide examen (il faut également utiliser la remarque 3, par laquelle on sait que $\mathsf{h}(U) = \mathsf{h}(U')$).

Si $n > 0$, Considérons la dernière règle appliquée dans la dérivation de $U \xrightarrow{*}_S U'$:

- Pour tous les axiomes autres que *4* et *5*, la remarque 8 permet de constater que $\nu(U)$ s'exprime comme la somme des tailles de réduction de certains de ses sous-termes, expression dans laquelle il suffit de supprimer certains termes pour obtenir l'expression de $\nu(U')$. Il devient alors évident que $\nu(U) \geq \nu(U')$.

  Illustrons ceci par le cas de l'axiome *9*:

$$\begin{aligned} \nu(U) &= \nu(M\lceil\{C[N]\} \sqcup \{N\} \sqcup D) = \nu(M) + \nu(C[N]) + \nu(N) + \nu(D) \\ \nu(U') &= \nu(M\lceil\{C[N]\} \sqcup D) = \nu(M) + \nu(C[N]) + \nu(D) \end{aligned}$$

- Pour tous les axiomes, et pour toutes les règles d'inférence (en utilisant l'hypothèse d'induction sur la preuve de $U \stackrel{*}{\to}_S U'$), la remarque 8 et la définition de $\mathsf{r}(M)$ montrent qu'on passe de même de $\nu(\mathsf{r}(U))$ à $\nu(\mathsf{r}(U'))$. Il devient alors évident que $\nu(\mathsf{r}(U)) \geq \nu(\mathsf{r}(U'))$.

  Illustrons ceci par les cas de l'axiome *8* et de la règle (δ):

  axiome *8*: $U = M\lceil\{N, N\} \sqcup D \equiv M\lceil\{N\} \sqcup D = U'$,

$$\begin{aligned} \nu(\mathsf{r}(U)) &= \nu(\mathsf{r}(M\lceil\{N, N\} \sqcup D)) \\ &= \nu(\mathsf{r}(M)) + \nu(\{N, N\} \sqcup D) \\ &= \nu(\mathsf{r}(M)) + \nu(N) + \nu(N) + \nu(D) \\ \nu(\mathsf{r}(U')) &= \nu(\mathsf{r}(M\lceil\{N\} \sqcup D)) \\ &= \nu(\mathsf{r}(M)) + \nu(\{N\} \sqcup D) \\ &= \nu(\mathsf{r}(M)) + \nu(N) + \nu(D) \end{aligned}$$

$$(\delta)\ \frac{N \stackrel{*}{\to}_S P}{U = M\lceil D \sqcup \{N\} \stackrel{*}{\to}_S M\lceil D \sqcup \{P\} = U'}$$

  Alors, par hypothèse d'induction $\nu(P) \leq \nu(N)$ et on a:

$$\begin{aligned} \nu(\mathsf{r}(M\lceil D \sqcup \{N\})) &= \nu(\mathsf{r}(M)) + \nu(D) + \nu(N) \\ \nu(\mathsf{r}(M\lceil D \sqcup \{P\})) &= \nu(\mathsf{r}(M)) + \nu(D) + \nu(P) \end{aligned}$$

- Pour les axiomes *4* et *5*, il reste à vérifier que $\nu(U) \geq \nu(U')$ et nous allons alors nous servir de l'hypothèse d'induction sur n:

  ○Pour les axiomes *4* et *5*: On vérifie aisément que $\mathsf{h}(U)\lceil\mathsf{r}(U) = \mathsf{h}(U')\lceil\mathsf{r}(U')$. Par la remarque 10, il s'ensuit que U' est fortement normalisable et que $\nu(U') = \nu(U)$.

Il nous reste à vérifier que $\nu(U) \geq \nu(U')$ pour les règles d'inférence:

- $(\mu)\ \dfrac{M \stackrel{*}{\to}_S N}{U = MP \stackrel{*}{\to}_S NP = U'}$

  Par hypothèse d'induction et par la remarque 10 $\nu(M) = \nu(\mathsf{h}(M)) + \nu(\mathsf{r}(M)) \geq \nu(N) = \nu(\mathsf{h}(N) + \nu(\mathsf{r}(N))$, d'où par la remarque 3: $\nu(\mathsf{r}(M)) \geq \nu(\mathsf{r}(N))$.

  Or $\mathsf{h}(NP)\lceil\mathsf{r}(NP) = \mathsf{h}(NP)\lceil\mathsf{r}(N) \sqcup \mathsf{r}(P)$. La remarque 3 montre que $\mathsf{h}(NP) = \mathsf{h}(MP)$. Ainsi

$$\begin{aligned} \nu(\mathsf{h}(U)\lceil\mathsf{r}(U)) &= \nu(\mathsf{h}(NP)\lceil\mathsf{r}(N) \sqcup \mathsf{r}(P)) \\ &= \nu(\mathsf{h}(NP)) + \nu(\mathsf{r}(N)) + \nu(\mathsf{r}(P)) \\ &\leq \nu(\mathsf{h}(MP)) + \nu(\mathsf{r}(M)) + \nu(\mathsf{r}(P)) \\ &= \nu(U) \end{aligned}$$

$\mathsf{h}(U)\lceil\mathsf{r}(U)$ est ainsi fortement normalisable pour une taille $\leq \nu(U)$, il en va donc de même de U en vertu de la remarque 10.

- $(\nu)\ \dfrac{M \overset{*}{\to}_S N}{U = PM \overset{*}{\to}_S PN = U'}$

  Se résout par un raisonnement tout à fait similaire au cas précédent.

- $(\gamma)\ \dfrac{M \overset{*}{\to}_S N}{U = M\lceil D \overset{*}{\to}_S N\lceil D = U'}$

  Idem

- $(\delta)\ \dfrac{N \overset{*}{\to}_S P}{U = M\lceil D \sqcup \{N\} \overset{*}{\to}_S M\lceil D \sqcup \{P\} = U'}$

  Conséquence immédiate de la remarque 8 et de l'hypothèse d'induction .

- $(\tau)\ \dfrac{M \overset{*}{\to}_S N \quad N \overset{*}{\to}_S P}{U = M \overset{*}{\to}_S P = U'}$

  Par transitivité appliquée à l'hypothèse d'induction . □

Il est à remarquer qu'il est nécessaire de restreindre les termes par des multi-ensembles et non par des ensembles pour espérer avoir le résultat du lemme précédent et donc la proposition 1.

En effet, supposons que nous ayons opté pour la restriction par des ensembles. Nous ne changeons rien à la théorie de la restriction, ni à la définition de la réduction, si ce n'est que les réunions disjointes ($\sqcup$) deviennent des réunions ensemblistes ($\cup$).

Voici deux contrexemples qui interdisent d'espérer dans ce cadre un résultat analogue à celui de la proposition 1:

$$\begin{array}{ccc} U = ((\lambda x.R)\lceil\{\lambda y.P\lceil Q\})(\lambda y.P\lceil Q) & \to_S & ((\lambda x.R)\lceil\{\lambda y.P, Q\})(\lambda y.P\lceil Q) = U' \\ \downarrow & & \downarrow \\ V = R[\lambda y.P/x]\lceil\{\lambda y.P\lceil Q)\} & & R[\lambda y.P/x]\lceil\{\lambda y.P, Q, \lambda y.P\lceil Q\} = V' \end{array}$$

On constate ici que la technique que nous avons employée et qui s'impose naturellement, à savoir de réduire le rédex de U qui correspond à celui qui a été réduit dans U' pour obtenir V', conduit à un terme V, mais alors il suffit de choisir un terme Q qui ne soit pas normal pour que l'on aît $\nu(V) < \nu(V')$. On pourrait se dire qu'il est possible de modifier la définition de la réduction, de sorte de faire remonter toutes les restrictions à un seul et même niveau, mais un deuxième contrexemple[3] vient y objecter:

[3] En fait, ce contrexemple montre quelque chose de plus profond: le fait que nous avons orienté les axiomes $M(N\lceil D) \to_S (MN)\lceil D$ et $(M\lceil D)N \to_S (MN)\lceil D$ à "rebrousse poil" de la réduction. C'est en effet dans le terme $(M\lceil D)N$ que l'information que constitue la restriction par D permet au mieux d'anticiper sur les simplifications et, si on avait pris l'orientation $(MN)\lceil D \to_S (M\lceil D)N$, on aurait tenu compte du caractère

$$\begin{array}{ccc} U = (((\lambda x.R)\lceil\lambda y.P)\lceil Q)(\lambda y.P\lceil Q) & \rightarrow_S & (((\lambda x.R)\lceil\{\lambda y.P, Q\})(\lambda y.P\lceil Q))\lceil Q = U' \\ \downarrow & & \downarrow \\ V = R[\lambda y.P/x]\lceil\{\lambda y.P\lceil Q), Q\} & & (R[\lambda y.P/x]\lceil\{\lambda y.P, Q\})\lceil Q = V' \end{array}$$

2.6 Confluence modulo une relation d'équivalence

Nous empruntons les notions qui suivent à la présentation qu'en donne Huet ([Huet 80]).

Definition 10 *Soit un ensemble $\mathcal{E}$ muni d'une relation de réduction $\rightarrow$ et d'une relation d'équivalence $\sim$, alors nous dirons que la relation $\rightarrow$ est* confluente modulo $\sim$ *si et seulement si la propriété suivante est satisfaite:*

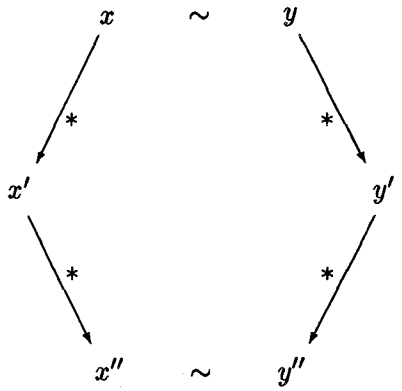
$$\forall x, y, x', y'\ x \sim y \text{ et } x \xrightarrow{*} x' \text{ et } y \xrightarrow{*} y'$$

$$\rightarrow$$

$$\exists x'', y''\ x' \xrightarrow{*} x'' \text{ et } y' \xrightarrow{*} y'' \text{ et } x'' \sim y''.$$

On notera la propriété $\exists x'', y''\ x' \xrightarrow{*} x''$ et $y' \xrightarrow{*} y''$ et $x'' \sim y''$ par $x' \overset{\sim}{\downarrow} y'$.

Ceci est illustré par la figure suivante:



local du calcul. Le défaut de cette option est que la syntaxe des manipulations s'en serait sensiblement alourdie, et c'est la raison pour laquelle nous ne l'avons pas adoptée ici, et ce, bien qu'on puisse alors en obtenir des propriétés plus fortes, notamment la commutation de la simplification $\xrightarrow{*}_S$ et de la réduction. Cette remarque ne devrait pas être négligeable, s'agissant du même problème pour λ_p que celui que nous résolvons dans ce chapitre pour Λ_{pl}. Les multi-ensembles ne semblent finalement nécessités que du fait d'avoir adopté un point de vue qui va contre la nature locale du calcul.

2.7 Plusieurs propriétés du calcul Λ_{pl}

Lemme 12 *Soit R un contexte factoriel en x et deux termes P et P' tels que $P \to_{\beta_{pl}} P'$, alors il existe un terme T tel que $T \overset{T_{res}}{=} R[\mathsf{h}(P')/x]\lceil\{P'\}$ et $R[\mathsf{h}(P)/x]\lceil\{P\} \overset{*}{\to}_{\beta_{pl}} T$.*

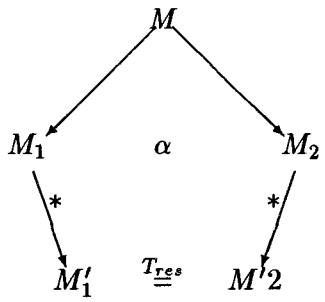
Preuve: C'est très simple si $\mathsf{h}(P) = \mathsf{h}(P')$. Supposons donc $\mathsf{h}(P) \neq \mathsf{h}(P')$. Alors le redex que l'on réduit est dans $\mathsf{h}(P)$. Si on réduit le rédex correspondant dans $\mathsf{h}(P)$ alors on trouve un terme U tel que $\mathsf{h}(U) = \mathsf{h}(P')$. D'autre part $\mathsf{r}(U) = \{N\}$ où N est l'argument dans le rédex de $\mathsf{h}(P)$, autrement dit $\{N\}$ est la seule restriction qu'il y a dans U. Comme R est un contexte factoriel, on peut transporter cette restriction à l'extérieur de R par les axiomes *4,5* et *6*, on a ainsi:

$$\begin{aligned} R[\mathsf{h}(P)/x]\lceil\{P\} &\to_{\beta_{pl}} R[U/x]\lceil\{P\} \\ &\to_{\beta_{pl}} R[U/x]\lceil\{P'\} \\ &\overset{T_{res}}{=} R[\mathsf{h}(P')/x]\lceil\{P', N\} \end{aligned}$$

Enfin il est aisé de vérifier que $N \in Fact(P')$ et peut donc être éliminé par l'axiome *10*. □

Lemme 13 *$\to_{\beta_{pl}}$ vérifie la propriété α*
Celle-ci s'énonce comme suit:
Si $M \to_{\beta_{pl}} M_1$ et $M \to_{\beta_{pl}} M_2$ alors il existe M_1' et M_2' tels que $M_i \overset{*}{\to}_{\beta_{pl}} M_i'$ pour $i = 1,2$
.

La figure suivante illustre cette propriété:



Preuve: Par induction sur M:

$M = x, c$ ou $\lambda x.N$: Ce cas est évident puisqu'alors M est normal.

$M = P\lceil D$, trois cas se présentent:

- $M_i = P_i\lceil D$ pour i = 1,2 avec $P \to_{\beta_{pl}} P_2$ alors par hypothèse d'induction il existe P' tels que $P_i \overset{*}{\to}_{\beta_{pl}} P'$ pour i = 1,2 . Il suffit de prendre $M_1' = M_2' = P'\lceil D$ pour i = 1,2.
- $M_i = P\lceil E \sqcup \{R_i\}$, avec $M_i = P\lceil E \sqcup \{R\}$ et $R \to_{\beta_{pl}} R_i$, pour $i = 1,2$. Immédiat par induction comme dans le cas précédent .
- $M_1 = P_1\lceil D$ et $M_2 = P\lceil E \sqcup \{P_2\}$, avec $P \to_{\beta_{pl}} P_1$, $D = E \sqcup \{R\}$ et $R \to_{\beta_{pl}} P_2$ alors il suffit de prendre $M_1' = M_2' = P_1\lceil E \sqcup \{P_2\}$.

$M = NP$ Plusieurs cas se présentent:

$M_i = NP_i$ avec $P \to_{\beta_{pl}} P_i$ pour $i = 1,2$ ou $M_i = N_iP$ avec $N \to_{\beta_{pl}} N_i$ pour $i = 1,2$ .Alors c'est immédiat par induction.

$M_1 = N_1P$ et $M_2 = NP_2$, avec $N \to_{\beta_{pl}} N_1$ et $P \to_{\beta_{pl}} P_2$, alors il suffit de prendre $M_1' = M_2' = N_1P_2$.

Sinon on a un "vrai" chevauchement: $\mathsf{h}(N) = \lambda x.R$ et on peut supposer $M_1 = R[\mathsf{h}(P)/x]\lceil(\mathsf{r}(N) \sqcup \{P\})$. Alors ici deux cas se présentent:

- $M_2 = N'P$ avec $N \to_{\beta_{pl}} N'$:
  Alors par le lemme 8, puisque $\mathsf{h}(N) = \lambda x.R$ est normal, on a $\mathsf{r}(N) \to_{\beta_{pl}} \mathsf{r}(N')$ et $\mathsf{h}(N) = \mathsf{h}(N') = \lambda x.R$. D'où: $M_2 = N'P \to_{\beta_{pl}} R[\mathsf{h}(P)/x]\lceil(\mathsf{r}(N') \sqcup \{P\}) = Q$ et d'autre part $M_1 \to_{\beta_{pl}} R[\mathsf{h}(P)/x]\lceil(\mathsf{r}(N') \sqcup \{P\}) = Q$.Il suffit de prendre $M_1' = M_2' = Q$.
- $M_2 = NP'$ avec $P \to_{\beta_{pl}} P'$:
  - Si P est une valeur alors le lemme 8 nous dit que $\mathsf{h}(P) = \mathsf{h}(P')$ d'où $M_2 = NP' \to_{\beta_{pl}} R[\mathsf{h}(P)/x]\lceil(\mathsf{r}(N) \sqcup \{P'\}) = Q$, comme d'autre part $M_1 \to_{\beta_{pl}} R[\mathsf{h}(P)/x]\lceil(\mathsf{r}(N) \sqcup \{P\}) \to_{\beta_{pl}} Q$, il suffit de prendre $M_1' = M_2' = Q$.
  - Si P n'est pas une valeur. Alors soit $\mathsf{h}(P) = \mathsf{h}(P')$ et le raisonnement est exactement le même que ci-dessus, soit $\mathsf{h}(P) \neq \mathsf{h}(P')$ et R est un contexte factoriel en x, d'où par le lemme 12 il existe un terme M_1' tel que $M_1' \overset{T_{res}}{=} R[\mathsf{h}(P')/x]\lceil(\mathsf{r}(M) \sqcup \{P'\}) = M_2'$ et $R[\mathsf{h}(P)/x]\lceil(\mathsf{r}(M) \sqcup \{P\}) \to^*_{\beta_{pl}} M_1'$. Remarquons que c'est le seul cas pour lequel on a pris $M_1' \neq M_2'$. □

Remarque 10

Soit C un contexte factoriel simple et un terme N. Alors si $C[N] \to_{\beta_{pl}} P$, P est de l'une des formes suivantes:

$C[N']$ avec $N \to_{\beta_{pl}} N'$,

$C'[N]$ où C' est un contexte factoriel et $C \to_{\beta_{pl}} C'$,

$C[N] = D[NM]$, où D est un contexte factoriel et NM est un rédex $\mathsf{h}(N) = \lambda x.R$) que l'on réduit d'où $P = D[R[\mathsf{h}(M)/x]\lceil(\mathsf{r}(N) \sqcup \{M\})]$.

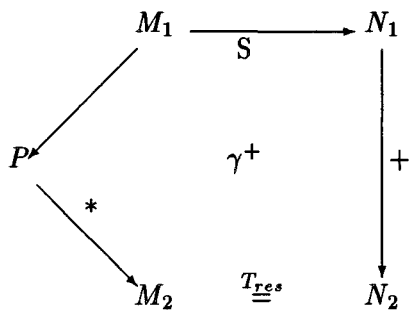
$C[N] = D[MN]$, où D est un contexte factoriel et MN est un rédex avec $\mathsf{h}(M) = \lambda x.Q$ et $P = D[Q[\mathsf{h}(N)/x]\lceil\mathsf{r}(M) \sqcup \{N\}]$.

Preuve: Par induction sur C. □

Lemme 14 *propriété γ^+ Celle-ci s'énonce comme suit:*

Soient trois termes M_1, N_1 et P tels que $M_1 \to_S N_1$ et $M_1 \to_{\beta_{pl}} P$ alors il existe deux termes M_2 et N_2 tels que $M_2 \overset{T_{res}}{=} N_2$, $P \to^*_{\beta_{pl}} M_2$ et $N_1 \to^+_{\beta_{pl}} N_2$.

La figure suivante illustre cette propriété:



Cette propriété ne serait pas vérifiée si l'on avait choisi de restreindre les termes par des ensembles plutôt que par des multi-ensembles.

En effet la règle *3* de T_{res}, donnerait par exemple $M\lceil\lambda x.P \to_S M$. Supposons que M soit normal et que $P \to_{\beta_{pl}} P'$, alors $M\lceil\lambda x.P \to_{\beta_{pl}} M\lceil\lambda x.P'$, et comme on ne peut plus réduire M, on voit que la propriété γ^+ ne peut pas être vérifiée.

Preuve: Par induction sur la dérivation de $M \to_S N$:

Pour les axiomes:

axiomes 1 à 3: $M_1 = M\lceil\{x\} \sqcup D, M\lceil\{c\} \sqcup D, M\lceil\{\lambda x.N\} \sqcup D \to_S M \sqcup D = N_1$, alors nécessairement on réduit M en M'ou D en D' , et le même axiome s'applique toujours après avoir réduit M ou D des deux cotés.

axiome 4: $M_1 = M(N\lceil D) \to_S MN\lceil D = N_1$, si on réduit M, N ou D alors c'est évident. Sinon c'est que $M(N\lceil D)$ est un rédex avec $\mathsf{h}(M) = \lambda x.R$ que l'on réduit donc en $M_2 = R[\mathsf{h}(N)/x]\lceil(\mathsf{r}(M) \sqcup N\lceil D)$, d'autre part $MN\lceil D$ se réduit donc en $N_2 = (R[\mathsf{h}(N)/x]\lceil(\mathsf{r}(M) \sqcup \{N\}))\lceil D$. Les axiomes 4,6 et 8 permettent de vérifier $M_2 \overset{T_{res}}{=} N_2$.

axiome 5: $M_1 = (M\lceil D)N \to_S MN\lceil D = N_1$, si on réduit M, N ou D alors c'est évident. Sinon c'est que $(M\lceil D)N$ est un rédex avec $\mathsf{h}(M) = \lambda x.R$ que l'on réduit donc en $M_2 = R[\mathsf{h}(N)/x]\lceil(\mathsf{r}(M\lceil D) \sqcup \{N\})$, or par définition $\mathsf{r}(M\lceil D) = \mathsf{r}(M) \sqcup D$ et d'autre part $MN\lceil D$ se réduit donc en $N_2 = (R[\mathsf{h}(N)/x]\lceil(\mathsf{r}(M) \sqcup N))\lceil D$. Les axiomes 4, 6 et 8 permettent ici encore de vérifier $M_2 \overset{T_{res}}{=} N_2$.

axiome 6: $M_1 = (M\lceil D)\lceil E \to_S M\lceil D \sqcup E = N_1$, On ne peut réduire que M, D ou E et c'est alors évident.

axiome 7: $M_1 = M\lceil\{N, N\} \sqcup D \to_S M\lceil\{N\} \sqcup D = N_1$, C'est évident si on réduit M ou D. Si on réduit N en N' , alors $P_1 = M\lceil\{N, N'\} \to_{\beta_{pl}} M\lceil\{N', N'\} = M_2$ et $N_2 = M\lceil\{N'\}$, il suffit de réappliquer le même axiome.

axiome 8: $M_1 = M\lceil\{N\lceil D\} \sqcup E \to_S M\lceil\{N\} \sqcup D \sqcup E = N_1$, On ne peut réduire que M, N, D ou E et c'est alors évident.

axiome 9: $M_1 = M\lceil\{C[N], N\} \sqcup D \to_S M\lceil\{C[N]\} \sqcup D = N_1$, si N est un terme pur et C un contexte factoriel,

•Si on réduit M ou D, c'est évident.

•Si on réduit N en N', alors puisque C est un contexte factoriel $C[N] \to_{\beta_{pl}} C[N']$ et par conséquent $P_1 = M\lceil\{C[N], N'\} \sqcup D \to_{\beta_{pl}} M\lceil\{C[N'], N'\} \sqcup D = M_2$ et on a $N_1 \to_{\beta_{pl}} M\lceil\{C[N']\} \sqcup D = N_2$. Il suffit alors de réappliquer le même axiome.

•Si on réduit $C[N]$ alors la remarque 10 nous dit que celui-ci se réduit

o Soit en $C[N']$ (et alors il suffit de prendre $M_2 = M_2 M\lceil\{C[N'], N'\} \sqcup D$ et $N_2 = M\lceil\{C[N']\} \sqcup D$ comme dans le cas précédent),

o Soit en $C'[N]$ ou C' est encore un contexte factoriel et par conséquent $P_1 = M\lceil\{C'[N], N\} \sqcup D = M_2$ et il suffit de prendre $N_2 = M\lceil\{C'[N]\} \sqcup D$ et de réappliquer le même axiome.

o Soit en $S = D[R[\mathsf{h}(Q)/x]\lceil(\mathsf{r}(N) \sqcup \{Q\})]$ avec $C[N] = D[NM]$, où D est un contexte factoriel et NM est un redex $\mathsf{h}(N) = \lambda x.R$) que l'on a réduit. Alors $N_1 \rightarrow_{\beta_{pl}} M\lceil\{S\} \sqcup D = N_2$. D'autre part posons $M_2 = M\lceil\{S, N\} \sqcup D$ et remarquons que S est un contexte factoriel en $\mathsf{r}(N)$, alors le lemme 4 et l'axiome 8 nous montrent que $M_2 \overset{T_{res}}{=} M\lceil\{S, \mathsf{h}(N)\} \sqcup \mathsf{r}(N) \sqcup D$. $\mathsf{h}(N)$ est une abstraction donc peut être supprimé par l'axiome 1 dans ce dernier terme et comme S est un contexte factoriel en $\mathsf{r}(N)$, celui-ci peut aussi être supprimé par l'axiome 9. D'où enfin $M_2 \overset{T_{res}}{=} N_2$.

o Soit en $S = D[Q[\mathsf{h}(N)/x]\lceil\mathsf{r}(M) \sqcup \{N\}]$, où MN est un rédex avec $\mathsf{h}(M) = \lambda x.Q$ et $C[N] = D[MN]$. Or D est un contexte factoriel, d'où il apparaît que $S = E[N]$, avec E contexte factoriel, et par conséquent $P = M\lceil\{E[N], N\} \sqcup D$. Il suffit de prendre $M_2 = P$ et $N_2 = M\lceil\{E[N]\} \sqcup D = M\lceil\{S\} \sqcup D$.

axiome 10: ce cas se déduit mutatis mutandis du précédent.

Pour les règles : c'est immédiat par induction. □

2.8 La confluence de Λ_{pl} pour les termes fortement normalisables

Nous allons maintenant montrer que le calcul que nous avons considéré est confluent pour les termes fortement normalisables, que nous savons constituer une classe stable par $\rightarrow_S$. Le bénéfice de ne considérer que des termes fortement normalisables est de permettre l'utilisation du principe d'induction nœthérienne (cf [Huet 80]).

Definition 11 *preuve standard*

Nous avons vu que la relation $\rightarrow_S$ est noethérienne et confluente, il en résulte que si deux termes M et N sont équivalents par la théorie T_{res} alors il existe un terme P,et deux entiers n et m tel que $M \overset{n}{\rightarrow}_S P$ et $N \overset{m}{\rightarrow}_S P$. Si $n+m$ est de plus minimal alors on dira que P fournit une preuve standard de $M \overset{T_{res}}{=} N$ de taille $n+m$.

Lemme 15 *Si $M \overset{T_{res}}{=} M'$ et M est normal pour $\rightarrow_{\beta_{pl}}$ alors M' est aussi normal pour $\rightarrow_{\beta_{pl}}$.*

Preuve: Si $M \overset{T_{res}}{=} M'$, soit P en donnant une preuve standard de taille $n \geq 1$ (pour n = 0, le résultat est évident). Alors le lemme 14 montre que si M' n'est pas normal, P ne l'est pas non plus. Or le lemme 11 permet d'affirmer que $\nu(M) \geq \nu(P)$, d'où contradiction. □

Théorème 2 *La réduction $\rightarrow_{\beta_{pl}}$ est confluente modulo la théorie T_{res}.*

Preuve: Il s'agit de prouver la propriété suivante:

$$\mathcal{P}(x,y) : x \overset{T_{res}}{=} y \Longrightarrow \forall x', y' (x \overset{*}{\rightarrow}_{\beta_{pl}} x' \text{ et } y \overset{*}{\rightarrow}_{\beta_{pl}} y') \Longrightarrow (x' \overset{T_{res}}{\downarrow} y')$$

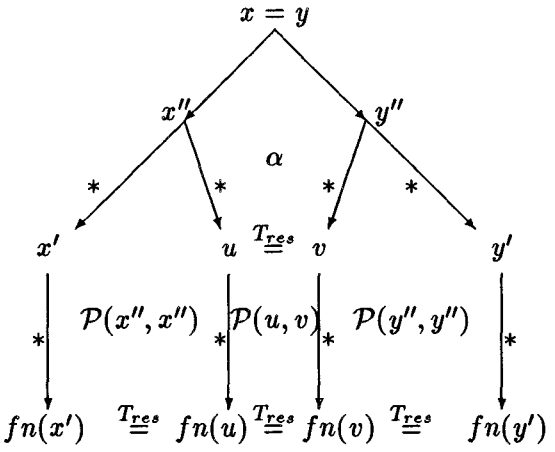
Elle est évidemment vérifiée si on n'a pas $x \overset{T_{res}}{=} y$. Il suffit donc de la prouver pour les couples de termes équivalents par la théorie T_{res}.

Si $x \overset{T_{res}}{=} y$ alors prouvons la propriété $\mathcal{P}(x,y)$ par induction lexicographique sur le couple $(sup(\nu(x),\nu(y)), d(x,y))$, où $d(x,y)$ est la taille d'une preuve standard de $x \overset{T_{res}}{=} y$.

•c'est trivial si $sup(\nu(x),\nu(y)) = 0$.

•Si $sup(\nu(x),\nu(y)) > 0$:

Si la taille d'une preuve standard de $x \overset{T_{res}}{=} y$ est égale à 0, c'est que $x = y$, alors si $x = x'$ ou si $y = y'$, le résultat cherché est évident, sinon c'est qu'il existe x'' et y'' tels que $x \to_{\beta_{pl}} x'' \overset{*}{\to}_{\beta_{pl}} x'$ et $y \to_{\beta_{pl}} y'' \overset{*}{\to}_{\beta_{pl}} y'$ et d'après le lemme 13 il existe donc u et v tels que $x'' \overset{*}{\to}_{\beta_{pl}} u$ et $y'' \overset{*}{\to}_{\beta_{pl}} v$ avec $u \overset{T_{res}}{=} v$. L'hypothèse d'induction nous montre, puisque le *sup* des tailles de réduction est plus petit, qu'on a les propriétés suivantes: $\mathcal{P}(x'',x'')$ et $\mathcal{P}(y'',y'')$, d'où il s'ensuit si on réduit x', u, v et y' jusqu'à des formes normales $fn(x'), fn(u), fn(v)$ et $fn(y')$ on a: $fn(x') \overset{T_{res}}{=} fn(u) \overset{T_{res}}{=} fn(y')$, d'où enfin $x' \overset{T_{res}}{\downarrow} y'$, ce qui montre $\mathcal{P}(x,y)$. Tout ceci est illustré par la figure suivante:

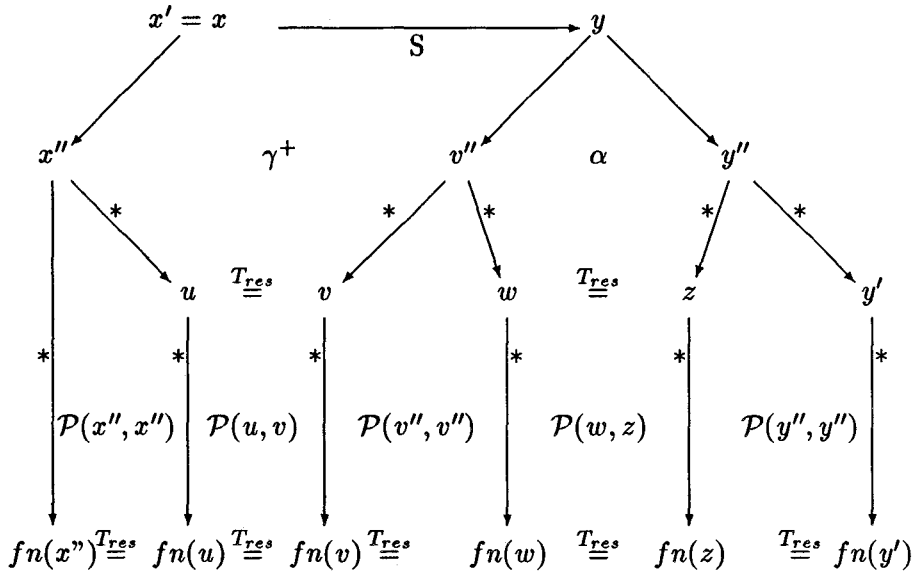


Si la taille d'une preuve standard de $x \overset{T_{res}}{=} y$ est égale à 1, alors $x \to_S y$.

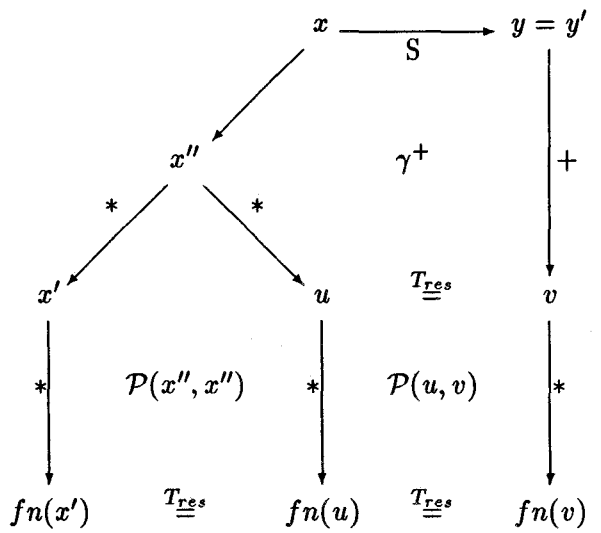
Si $x' = x$, on peut supposer $y' \neq y$, et soit $fn(x)$, $fn(y')$ des formes normales de x et y'. Alors le lemme 15 prouve que $x \neq fn(x)$. Soit par conséquent x'' tel que $x \to_{\beta_{pl}} x'' \overset{*}{\to}_{\beta_{pl}} fn(x)$. le lemme 14 permet d'affirmer qu'il existe u et v tels que $x'' \overset{*}{\to}_{\beta_{pl}} u$, $y \overset{+}{\to}_{\beta_{pl}} v$ avec $u \overset{T_{res}}{=} v$. Soient v'' et y'' tels que $y \to_{\beta_{pl}} v'' \overset{*}{\to}_{\beta_{pl}} v$ et $y \to_{\beta_{pl}} y'' \overset{*}{\to}_{\beta_{pl}} y$, alors le lemme 13 assure l'existence de w et de z tels que $v'' \overset{*}{\to}_{\beta_{pl}} w$ et $y' \overset{*}{\to}_{\beta_{pl}} z$, avec $w \overset{T_{res}}{=} z$. L'hypothèse d'induction montre - puisqu'alors le *sup* des tailles de réduction est plus petit - qu' en ayant réduit u, v, w, z et y' jusqu'à une forme normale, on a:

$$fn(x) \overset{T_{res}}{=} fn(u) \overset{T_{res}}{=} fn(v) \overset{T_{res}}{=} fn(w) \overset{T_{res}}{=} fn(z) \overset{T_{res}}{=} fn(y')$$

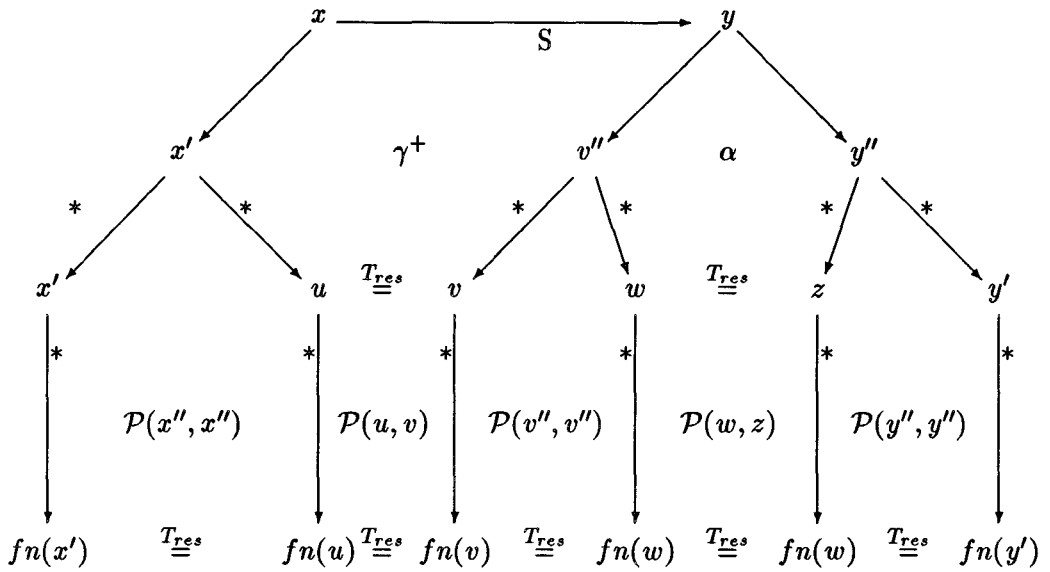
d'où $x' \overset{T_{res}}{\downarrow} y'$.



Si $x' \neq x$ et $y' = y$. La figure suivante indique le raisonnement à suivre:



Si $x' \neq x$ et $y' \neq y$. La figure suivante indique le raisonnement à suivre:

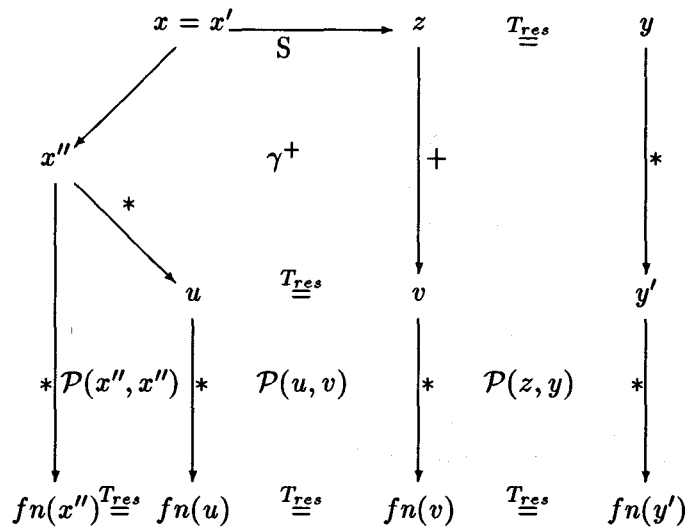


Si $d(x,y) \geq 2$, alors considèrons une preuve standard de $x \overset{T_{res}}{=} y$ qui est de la forme suivante (quitte à échanger x et y qui jouent un rôle symétrique):

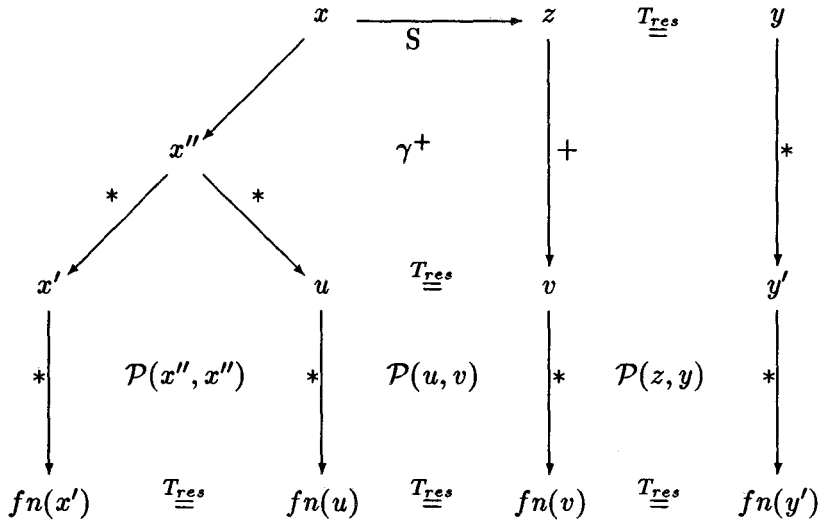
$$x \to_S z \overset{p}{\to}_S w \;{}_S\!\overset{q}{\leftarrow} y$$

Si $x' = x$, on peut supposer $y' \neq y$, et soit $fn(x)$, $fn(y')$ des formes normales de x et y'. Alors le lemme 15 prouve que $x \neq fn(x)$. Soit par conséquent x'' tel que $x \to_{\beta_{pl}} x'' \overset{*}{\to}_{\beta_{pl}} fn(x)$. le lemme 14 permet d'affirmer qu'il existe u et v tels que $x'' \overset{*}{\to}_{\beta_{pl}} u$, $z \overset{+}{\to}_{\beta_{pl}} v$ avec $u \overset{T_{res}}{=} v$. $\mathcal{P}(z,y)$ est vrai par hypothèse d'induction puisque le lemme 11 montre que $sup(\nu(z), \nu(y)) \leq sup(\nu(x), \nu(y))$

$d(z,y) \leq d(x,y)$, la taille de la preuve standard a diminuée et la figure suivante montre comment se conclut le raisonnement:



Si $x \neq x'$, c'est qu'il existe x'' tel que $x \rightarrow_{\beta_{pl}} x'' \stackrel{*}{\rightarrow}_{\beta_{pl}} x'$. Appliquons alors le lemme 14: il existe u et v tels que $u \stackrel{T_{res}}{=} v$, $x'' \stackrel{*}{\rightarrow}_{\beta_{pl}} u$ et $z \stackrel{+}{\rightarrow}_{\beta_{pl}} v$. Le lemme 11 prouve que $sup(\nu(u), \nu(v)) \leq sup(\nu(x), \nu(y))$. La figure suivante montre comment se conclut le raisonnement, analogue au précédent:



□

3

Normalisation

Nous allons maintenant définir un calcul typé Λ_t, qui n'est autre que le λ_p-calcul simplement typé. Nous allons ensuite montrer que tout terme de ce calcul est fortement normalisable en y adaptant la technique de réductibilité due à Tait, et revisitée par Girard, qui est un renforcement de l'hypothèse d'induction nécessaire à démontrer la normalisabilité forte des termes.

La terminaison des calculs n'est pas aussi immédiate qu'il peut paraître, puisque dans ce calcul, et à la différence du cas classique, si on a une réduction d'un terme typable de la forme $(\lambda x.(P\lceil Q))R \to P[R/x]\lceil Q[R/x]\lceil R$ il faut que le typage assure notamment la terminaison du calcul de $Q[R/x]$ et c'est ce qui va présenter une difficulté nouvelle de cette démonstration par ailleurs standard (que résolvent les lemmes 20 et 21).

Remarquons que la notion de terme fortement normalisable diffère de celle considérée au chapitre suivant de terme réductible à une valeur. Par exemple le terme xy, pour n'être pas une valeur, n'en est pas moins sous forme normale.

On établit ici que si les restrictions qu'on introduit sont représentées par des termes typables (et donc fortement normalisables), alors les termes typés obtenus demeurent fortement normalisables.

Nous présentons les regs de typages et quelques lemmes de cohérences. Nous allons ensuite prouver la préservation du typage pour deux notions de réductions, l'une qui correspond au λ_p-calcul et l'autre qui englobera la réduction $\to_{\beta_{pl}}$ du précédent chapitre ce qui nous permettra de considérer ultérieurement les termes de Λ_{pl} qui sont typables, ce qui permettra d'appliquer les résultats du précédent chapitre à cette classe de termes.

Nous suivons pour le reste la technique classique si ce n'est que le lemme 21 est nécessaire à l'adapter aux termes partiels.

3.1 Syntaxe de Λ_t

Les termes de Λ_t sont les termes de λ_p qui sont typables dans le système que nous allons présenter.

Les types sont les formules construites avec des constantes de type α et le connecteur $\to$, chaque constante c de λ_p étant associée à un type constant α_c.

Une déclaration de variable est, par définition, un couple (x, A), où x est une variable du λ_p-calcul et A un type; elle sera notée $x : A$.

Un contexte Γ est un ensemble fini (éventuellement vide) de déclarations de variables distinctes.

$\Gamma, x : A$ représente le contexte Γ augmenté de la déclaration $x : A$.

3.2 Les règles de typage

Etant donnés M un terme du λ_p-calcul, A un type et Γ un contexte, on définit les règles qui permettent de dériver le jugement "t est de type A dans le contexte Γ" (jugement que l'on notera $\Gamma \vdash t : A$ ou simplement $\vdash t : A$ si Γ est vide) comme suit:

$$(var)\ \frac{}{\Gamma \vdash x : \Gamma_x}$$

$$(const)\ \frac{}{\Gamma \vdash c : \alpha_c}$$

$$(abs)\ \frac{\Gamma[x : A] \vdash M : B}{\Gamma \vdash \lambda x.M : A \to B}$$

$$(app)\ \frac{\Gamma \vdash M : A \to B \qquad \Gamma \vdash N : A}{\Gamma \vdash MN : B}$$

$$(restr)\ \frac{\Gamma \vdash M : A \qquad \text{pour tout } N \in D \qquad \Gamma \vdash N : A_N}{\Gamma \vdash M\lceil D : A}$$

Tous ces jugements sont à considérer modulo α-conversion. On dénotera l'ensemble des termes typables dans ce calcul par Λ_t.

3.3 Quelques lemmes

Nous donnons à présent quelques lemmes utiles.

Lemme 16 *affaiblissement: Si* $\Gamma\Phi \vdash M : T$ *et si* $\Gamma\Delta\Phi$ *est un contexte alors* $\Gamma\Delta\Phi \vdash M : T$.

Preuve: Ceci se prouve sans difficulté par induction sur la dérivation de $\Gamma\Phi \vdash M : T$ (en effectuant éventuellement une α-conversion des variables liées de M).

Lemme 17 *substitution: Si* $\Gamma[x : A]\Delta \vdash N : B$, $\Gamma \vdash M : A$ *alors :* $\Gamma\Delta \vdash N[M/x] : B$.

Preuve: Comme précédemment, par induction sur la dérivation de $\Gamma[x : A]\Delta \vdash N : B$. L'hypothèse d'induction résout aisément tous les cas à l'exception de:

$(var)\dfrac{}{\Gamma[x : A]\Delta \vdash x : A}$

Mais alors, de l'hypothèse $\Gamma \vdash M : A$, on déduit par le lemme 16 $\Gamma\Delta \vdash M : A$, ce qui est le résultat cherché. □

Lemme 18 *Si* $\Gamma \vdash M\lceil N : A$ *(resp.*$\Gamma \vdash M\lceil D : A$*) alors* $\Gamma \vdash M : A$ *et il existe* A_N *tel que* $\Gamma \vdash N : A_N$ *(resp. pour tout* $N \in D$*).*

Si $\Gamma \vdash MN : B$ *alors il existe* A *tel que* $\Gamma \vdash M : A \to B$ *et* $\Gamma \vdash N : A$.

Preuve: Trivial, la structure du terme déterminant toujours la dernière règle appliquée . □

3.4 Préservation du typage par réduction

Definition 12 *Soit* $\triangleright$ *une relation binaire sur* Λ_t. *Nous dirons que* $\triangleright$ *passe au contexte si elle est réflexive et si les conditions suivantes sont satisfaites*

1. $M \triangleright N \implies MP \triangleright NP$

2. $M \triangleright M' \Longrightarrow PM \triangleright PN$

3. $M \triangleright N \Longrightarrow \lambda x.M \triangleright \lambda x.N.$

4. $M \triangleright N \Longrightarrow M\lceil D \triangleright N\lceil D.$

5. $M \triangleright N \Longrightarrow P\lceil D \cup \{M\} \triangleright P\lceil D \sqcup \{N\}$

Definition 13 *Nous pouvons étendre les deux fonctions (cf definition 4)* h *et* r *à* Λ_t *pour avoir* $\mathsf{h} : \Lambda_t \to \Lambda_t$ *et* $\mathsf{r} : \Lambda_t \to \Lambda_t$.

$$\begin{array}{ll} \mathsf{h}(x) = x & \mathsf{r}(x) = \emptyset \\ \mathsf{h}(c) = c & \mathsf{r}(c) = \emptyset \\ \mathsf{h}(\lambda x.M) = \lambda x.M & \mathsf{r}(\lambda x.M) = \emptyset \\ \mathsf{h}(MN) = \mathsf{h}(M)\mathsf{h}(N) & \mathsf{r}(MN) = \mathsf{r}(M) \sqcup \mathsf{r}(N) \\ \mathsf{h}(M\lceil D) = \mathsf{h}(M) & \mathsf{r}(M\lceil D) = \mathsf{r}(M) \sqcup D \end{array}$$

Lemme 19 *Si* $\Gamma \vdash M : A$ *alors* $\Gamma \vdash \mathsf{h}(M) : A$ *et pour tout* $N \in \mathsf{r}(M)$, *il existe* A_N *tel que* $\Gamma \vdash N : A_N$.

Preuve: C'est immédiat par induction sur M moyennant le lemme 18. □

Nous allons à présent munir ce calcul de deux réductions, l'une $\to_{\beta_1}$ qui correspond à la réduction usuelle du λ_p-calcul et la seconde qui en est une variante et qui présente l'intérêt de coïncider avec la relation $\to_{\beta_{pl}}$ du précédent chapitre pour les termes communs aux calculs Λ_{pl} et Λ_t.

Definition 14 *Nous définissons la relation binaire* $\to_{\beta_1}$ *sur* Λ_t *comme la plus petite relation qui passe au contexte et telle que si* $\mathsf{h}(P) = \lambda x.M$ *alors* $PQ \to_{\beta_1} M[Q/x]\lceil\{Q\} \sqcup \mathsf{r}(P)$ *La relation* $\to_{\beta_1}^*$ *est la plus petit relation binaire sur* Λ_t, *qui est transitive et qui contient* $\to_{\beta_1}$.

Théorème 3 *Si* $\Gamma \vdash M : A$ *et* $M \to_{\beta_1} M'$ *alors* $\Gamma \vdash M' : A$.

Preuve: Remarquons que $\to_{\beta_1}$ peut être vue comme ce qui est déductible des clauses suivantes:

1. $\mathsf{h}(P) = \lambda x.M \Longrightarrow PQ \to_{\beta_1} M[Q/x]\lceil\{Q\} \sqcup \mathsf{r}(P)$

2. $M \to_{\beta_1} N \Longrightarrow MP \to_{\beta_1} NP$

3. $M \to_{\beta_1} M' \Longrightarrow PM \to_{\beta_1} PN$

4. $M \to_{\beta_1} N \Longrightarrow \lambda x.M \to_{\beta_1} \lambda x.N.$

5. $M \to_{\beta_1} N \Longrightarrow M\lceil D \to_{\beta_1} N\lceil D.$

6. $M \to_{\beta_1} N \Longrightarrow P\lceil D \sqcup \{M\} \to_{\beta_1} P\lceil D \sqcup \{N\}$

On prouve alors le résultat en regardant la dernière clause appliquée:

On a appliqué la clause 1: $\Gamma \vdash PQ : B$ provient de (*app*), donc $\Gamma \vdash P : A \to B$ et $\Gamma \vdash Q : A$. D'après le lemme 19 $\Gamma \vdash \mathsf{h}(P) : A \to B$ et pour tout $N \in \mathsf{r}(P)$, il existe A_N tel que $\Gamma \vdash N : A_N$. Si $\mathsf{h}(P) = \lambda x.M$ alors la dernière règle appliquée est (*abs*), d'où $\Gamma[x : A] \vdash M : B$. On peut appliquer le lemme 17 pour en déduire $\Gamma \vdash M[Q/x] : B$, il suffit dès lors d'appliquer (*restrens*) pour obtenir $\Gamma \vdash M[Q/x]\lceil\{Q\} \cup \mathsf{r}(P) : B$.

les autres cas sont tous immédiats par induction sur le nombre de clauses appliquées. On connaît toujours la dernière règle appliquée , il suffit de la réappliquer après l'hypothèse d'induction .

□

Definition 15 *Nous définissons la relation binaire* $\rightarrow_{\beta_2}$ *sur* Λ_t *comme la plus petite relation qui passe au contexte et telle que si* $\mathsf{h}(P) = \lambda x.M$ *alors* $PQ \rightarrow_{\beta_2} M[\mathsf{h}(Q)/x]\lceil\{Q\} \sqcup \mathsf{r}(P)$

La relation $\overset{*}{\rightarrow}_{\beta_2}$ *est la plus petite relation binaire sur* Λ_t*, qui est transitive et qui contient* $\rightarrow_{\beta_2}$.

Théorème 4 *Si* $\Gamma \vdash M : A$ *et* $M \rightarrow_{\beta_2} M'$ *alors* $\Gamma \vdash M' : A$.

Preuve: Tout à fait similaire à celle du théorème 3. La seule différence est qu'il faut appliquer une nouvelle fois le lemme 19 pour déduire $\Gamma \vdash \mathsf{h}(Q) : A$. □

Lemme 20 *Si* $M = P\lceil(D \cup E) \in red_{\Gamma,A}$ *alors* $N = P\lceil E \in red_{\Gamma,A}$.

En particulier, si $M = N\lceil D \in red_{\Gamma,A}$, nous aurons $N \in red_{\Gamma,A}$.
Preuve : Le point (*CR1*) du lemme 22 montre que M est fortement normalisable. Nous allons par conséquent pouvoir prouver le premier point du lemme par une induction lexicographique sur le couple (A, n), avec $n = \nu(M)$:

- A est atomique: il s'agit alors simplement de s'assurer que N est fortement normalisable, ce qui résulte immédiatement du fait que M est fortement normalisable.

- $A = S \rightarrow T$: Par définition, on voit qu'il s'agit de vérifier que, si $u \in red_{\Gamma,S}$, alors $Nu \in red_{\Gamma,T}$. Nous prouvons ce résultat par induction sur $\nu(u)$, Nu ne peut se réduire qu'en:

  - Nu', alors $\nu(u) \neq 0$ et il suffit d'appliquer l'hypothèse d'induction sur $\nu(u)$ pour obtenir $Nu' \in red_{\Gamma,T}$.

  - $N'u$, alors soit $N' = P'\lceil E$ avec $P\beta P'$, soit $N' = P\lceil F \cup \{Q'\}$ avec $E = F \cup \{Q\}$ et $Q\beta Q'$, alors dans le premier cas, posons $M' = P'\lceil(D \cup E)$ et $M' = P\lceil(D \cup (F \cup \{Q'\})$ dans le second. Alors $M\beta M'$, et par le lemme 22 $M' \in red_{\Gamma,A}$ Appliquons maintenant le lemme par hypothèse d'induction sur n, il s'ensuit que $N' \in red_{\Gamma,S\rightarrow T}$ et donc que $N'u \in red_{\Gamma,T}$.

  - $R[u/x]\lceil\{u\} \cup \mathsf{r}(N)$, avec $\mathsf{h}(N) = \lambda x.R$ et $\beta = \beta'$. Or, par hypothèse sur M, $(P\lceil D \cup E)u \in red_{\Gamma,T}$ et (*CR2*) nous dit alors que $R[u/x]\lceil\{u\} \cup \mathsf{r}(P\lceil D \cup E) \in red_{\Gamma,T}$, comme $\mathsf{r}(P\lceil D \cup E) = \mathsf{r}(P) \cup D \cup E$ et que $\mathsf{r}(P\lceil D) = \mathsf{r}(P) \cup D$, on voit qu'il suffit d'appliquer l'hypothèse d'induction , T étant moins complexe que A, pour obtenir $R[u/x]\lceil\{u\} \cup \mathsf{r}(N) \in red_{\Gamma,T}$

    (si $\beta = \beta_2$, la démonstration est identique, mutatis mutandis, soit en remplaant $R[u/x]$ par $R[\mathsf{h}(u)/x]$).

    Il apparaît ainsi que Nu est un terme neutre dont toutes les réductions en une étape conduisent à un terme de $red_{\Gamma,T}$, donc $Nu \in red_{\Gamma,T}$, et ainsi $N \in red_{\Gamma,S\rightarrow T}$, par définition des termes réductibles d'un type flèche. □

Le lemme suivant est le résultat qui permet de lever la difficulté particulière que nous avons soulevé dans l'introduction de ce chapitre.

Lemme 21 *Si $M \in red_{\Gamma,A}$ et si tous les termes N appartenant à l'ensemble fini D sont fortement normalisables et sont typables dans le contexte Γ d'un type A_N, alors $M\lceil D \in red_{\Gamma,A}$.*

Preuve : Par induction lexicographique sur (A, n), avec $n = \nu(M\lceil D) = \nu(M) + \sum_{N \in D} \nu(N)$.

Remarquons que, sous les hypothèses du lemme, $\Gamma \vdash M\lceil D : A$ se déduit immédiatement par la règle (*restr*).

•si A est un type atomique, alors $M\lceil D$ est fortement normalisable et $M\lceil D \in red_{\Gamma,A}$ s'ensuit de la définition des termes réductibles de type atomique.

si A n'est pas un type atomique, $A = S \to T$,

l'étude se divise en deux cas:

•M est neutre, alors

si $n = 0$, $M\lceil D$ est neutre et normal et il suffit d'appliquer le corollaire 1.

si $n > 0$, alors $M\lceil D$ est neutre, puisque M l'est, ce qui va nous permettre d'utiliser le point (*CR3*) du lemme 22. Or $M\lceil D$ ne peut se réduire qu'en:

o$M'\lceil D$, avec M se réduisant en M' et donc, par le point (*CR2*) du lemme 22, $M' \in red_{\Gamma,A}$ et par hypothèse d'induction sur n, on en déduit que $M'\lceil D \in red_{\Gamma,A}$.

o$M\lceil E \cup \{N'\}$, avec $D = E \cup \{N\}$, N se réduisant en N', et N' demeurant un terme fortement normalisable de même type que N.

On a décrémenté n dans les deux cas, ce qui nous permet d'appliquer l'hypothèse d'induction , celle-ci montrant alors que $M\lceil D$ ne peut se réduire qu'en des termes réductibles de même type. Il ne reste plus qu'à appliquer (*CR3*).

M n'est pas neutre, alors $\mathsf{h}(M) = \lambda x.P$. Il s'agit donc de prouver que pour tout $u \in red_{\Gamma,S}$, $(M\lceil D)u \in red_{\Gamma,T}$. $(M\lceil D)u$ étant neutre, nous allons ici encore utiliser (*CR3*). Procédons par induction sur $\nu(u)$:

$(M\lceil D)u$ ne peut se réduire qu'en:

o$(M\lceil D)u'$, avec $u\beta u'$ et donc par (*CR2*), $u' \in red_{\Gamma,S}$. Par hypothèse d'induction sur $\nu(u)$, on obtient $(M\lceil D)u' \in red_{\Gamma,T}$.

o$(M'\lceil D)u$ avec $M\beta M'$, alors cette fois l'hypothèse d'induction sur n nous dit que $M'\lceil D \in red_{\Gamma,S\to T}$. Par définition des termes réductibles de type $S \to T$, il apparaît que $(M'\lceil D)u \in red_{\Gamma,T}$.

o$(M\lceil E \cup \{N'\})u$ avec $D = E \cup \{N\}$ et $N\beta N'$, alors cette fois l'hypothèse d'induction sur n nous dit que $M\lceil E \cup \{N'\} \in red_{\Gamma,S\to T}$, puisque $\nu(E \cup \{N'\}) = \nu(D) - 1$. Par définition des termes réductibles de type $S \to T$, il apparaît que $(M\lceil E \cup \{N'\})u \in red_{\Gamma,T}$.

oDans ce dernier cas $(M\lceil D)u$ est un rédex que l'on réduit, et la suite de l'étude se divise en deux cas, suivant que la réduction considérée est β' ou β_2, mais qui se résolvent de façon toute similaire, aussi nous traitons que le premier:

Avec $\beta = \beta'$, $(M\lceil D)u$ peut se réduire en $P[u/x]\lceil\{u\} \cup \mathsf{r}(M) \cup D$. Par hypothèse $M \in red_{\Gamma,S\to T}$ et puisque $u \in red_{\Gamma,S}$, il s'ensuit que $Mu \in red_{\Gamma,T}$. Or ceci montre, par (*CR2*), que $P[u/x]\lceil\{u\} \cup \mathsf{r}(M) \in red_{\Gamma,T}$. (*CR1*) nous indique maintenant que u et $P[u/x]\lceil\{u\} \cup \mathsf{r}(M)$ sont fortement normalisables, et par suite que $\mathsf{r}(M)$ l'est également. Le lemme 21 montre que $\mathsf{r}(M)$ est typable. Par suite $\{u\} \cup \mathsf{r}(M) \cup D$ est un ensemble

de termes typables fortement normalisables. Le lemme 20 montre que $P[u/x] \in red_{\Gamma,T}$. On peut maintenant appliquer le lemme par hypothèse d'induction sur le type pour obtenir $P[u/x]\lceil\{u\} \cup r(M) \cup D \in red_{\Gamma,T}$.

Toutes les réductions en une étape du terme neutre $(M\lceil D)u$ conduisent à des termes de $red_{\Gamma,T}$, $(CR3)$ nous indique alors que $(M\lceil D)u \in red_{\Gamma,T}$, quelque soit $u \in red_{\Gamma,S}$. Il s'ensuit, par définition des termes réductibles de type flèche, que $M\lceil D \in red_{\Gamma,A}$. □

3.5 Normalisabilité forte des termes typables

Dans cette partie nous allons considéé une réduction β et nous prendrons indifféremment $\beta = \rightarrow_{\beta_1}$ ou $\beta = \rightarrow_{\beta_2}$. Ici nous appliquons la méthode due à Tait et revisitée par Girard [Girard 90] avec la notion de réductibilité.

Definition 16 *On définit par induction sur le type d'un terme la notion de réductiblité.*

Si $\Gamma \vdash M : A$, *M sera dit* réductible *de type A relativement au contexte* Γ *(ce qu'on notera* $M \in red_{\Gamma,A}$*) s'il vérifie l'une des propriétés suivantes:*

- a_1) *A est un type constant (atomique) et M est fortement normalisable.*
- a_2) *A est de la forme* $B \rightarrow C$ *et pour tout* $U \in red_{\Gamma,B}, MU \in red_{\Gamma,C}$.

Plus simplement un terme M sera dit réductible *s'il existe* Γ *et A tel que* $M \in red_{\Gamma,A}$.

Definition 17 *Un terme M est dit* neutre *si* $h(M)$ *n'est pas une abstraction.*

Autrement dit,M est neutre, si et seulement si, pour tout terme N, MN n'est pas un rédex.

Lemme 22

- *(CR1) Si* $T \in red_{\Gamma,A}$ *alors T est fortement normalisable.*
- *(CR2) Si* $T \in red_{\Gamma,A}$ *et* $T\beta T'$, *alors* $T' \in red_{\Gamma,A}$.
- *(CR3) Si T est neutre ,*$\Gamma \vdash T : A$ *et est tel que pour tout* T' *tel que* $T\beta T'$, $T' \in red_{\Gamma,A}$, *alors* $T \in red_{\Gamma,A}$.

Preuve:

Celle-ci se fait par induction sur le type A.

- Si A est un type constant (atomique) alors:
  - o($CR1$) est vraie par définition.
  - o($CR2$) est vérifiée puisque toute réduction d'un terme fortement normalisable conduit à un terme également fortement normalisable et conserve le type (théorèmes 3 et 4).
  - o($CR3$) Si $T\beta T'$ et $\Gamma \vdash T' : A$, on sait par les théorèmes 3 et 4 que $\Gamma \vdash T : A$, donc T est bien de type A et il est fortement normalisable puisque toutes ses réductions en une étape conduisent à des termes fortement normalisables, donc $T \in red_{\Gamma,A}$.
- Si A est de la forme $B \rightarrow C$:
  - o($CR1$) Soit $T \in red_{\Gamma,B\rightarrow C}$. Si $U \in red_{\Gamma,B}$, alors par définition $TU \in red_{\Gamma,C}$, or C est un type moins complexe que $B \rightarrow C$ donc l'hypothèse d'induction nous dit que TU est fortement normalisable, donc a fortiori T.

- ($CR2$) Si $T \in red_{\Gamma,B\to C}$ et $T\beta T'$. Soit $U \in red_{\Gamma,B}$, alors $TU \in red_{\Gamma,C}$ et $TU\beta T'U$, mais le type C étant moins complexe que $B \to C$ vérifie ($CR2$), d'où $T'U \in red_{\Gamma,C}$, ce qui prouve que $T' \in red_{\Gamma,B\to C}$.
- ($CR3$) Soit T un terme neutre vérifiant les hypothèses de ($CR3$). Soit $U \in red_{\Gamma,B}$, par hypothèse d'induction on sait par ($CR1$) que U est fortement normalisable. Soit $\nu(U)$ la longueur maximale des dérivations partant de U et conduisant à une forme normale. Démontrons par induction sur $\nu(U)$ que $TU \in red_{\Gamma,C}$:

  •Si $\nu(U) = 0$ alors toute réduction de TU est de la forme $TU\beta T'U$ où $T\beta T'$, de plus $T'U \in red_{\Gamma,C}$ par hypothèse. Comme TU est neutre, ($CR3$) s'appliquant au type C permet de déduire que $TU \in red_{\Gamma,C}$.

  •Si $\nu(U) = n + 1$ alors toute réduction de TU est soit de la forme $TU\beta T'U$ où $T\beta T'$ auquel cas $T'U \in red_{\Gamma,C}$, soit de la forme $TU\beta TU'$ avec $\nu(U') = n$ auquel cas par hypothèse d'induction sur $\nu(U)$ on a $TU' \in red_{\Gamma,C}$.

  Par conséquent si $U \in red_{\Gamma,B}$, TU est un terme neutre dont toutes les réductions en une étape conduisent à des termes de $red_{\Gamma,C}$, ($CR3$) étant vraie pour le type C nous en déduisons que $TU \in red_{\Gamma,C}$, ce qui permet d'affirmer que $T \in red_{\Gamma,B\to C}$.

□

Corollaire 1 *Si* $\Gamma \vdash M : A$ *et* M *est neutre et normal alors* $M \in red_{\Gamma,A}$.

Preuve: ceci résulte immédiatement du point ($CR3$) du lemme précédent. □

Lemme 23 *les variables sont réductibles*

Preuve: Ceci n'est qu'un cas particulier du corollaire précédent. □

Lemme 24

Si pour tout N *réductible* $T[N/x]\lceil\{N\}$ *est réductible alors:*

$\lambda x.T$ *est réductible*

Preuve:

L'hypothèse et le lemme 23 nous disent que $T[x/x]\lceil\{x\} = T\lceil\{x\}$ est réductible donc fortement normalisable, et par conséquent T est lui-même fortement normalisable.

Soit $\lambda x.T : \Pi x \in A.B$.

On veut démontrer que pour tout $U \in red_{\Gamma,A}$, $(\lambda x.T)U \in red_{\Gamma,B[U/x]}$. La démonstration se fait par induction sur $\nu(U) + \nu(T)$:

• Si $\nu(U) + \nu(T) = 0$ alors $(\lambda x.T)U$ ne peut se réduire qu'en $T[U/x]\lceil\{U\}$ qui est réductible, et donc, en vertu du point ($CR3$) du lemme 22, $\lambda x.T$ est réductible.

• Si $\nu(U) + \nu(T) = n + 1$

$(\lambda x.T)U$ peut se réduire soit en $T[U/x]\lceil\{U\}$ qui est réductible, soit en $(\lambda x.T')U$ avec $\nu(T') = \nu(T) - 1$, soit en $(\lambda x.T)U'$ avec $\nu(U') = \nu(U) - 1$ qui sont l'un et l'autre réductibles par hypothèse d'induction. Le point ($CR3$) du lemme 22 permet alors de conclure. □

Lemme 25 *Pour tout terme* T *ayant pour variables libres* $x_1, .., x_n$ *et pour tous* $U_1, .., U_n$ *réductibles:*

$T[U_1/x_1, .., U_n/x_n]$ *est réductible.*

Preuve: Par induction sur T:

- Si $T = x$, c'est évident.
- Si $T = (MN)$, par hypothèse d'induction

$$M_1 = M[U_1/x_1, U_2/x_2, .., U_n/x_n]$$

et

$$N_1 = N[U_1/x_1, U_2/x_2, .., U_n/x_n]$$

sont réductibles

Le lemme 18 montre que M_1 et N_1 sont de types respectifs $A \to B$ et A, donc par hypothèse d'induction ils sont réductibles de ces mêmes types et enfin $M_1 N_1$ - à savoir $(MN)[U_1/x_1, U_2/x_2, .., U_n/x_n]$ - l'est aussi.

- Si $T = M\lceil\{N\}$ par hypothèse d'induction:

$M_1 = M[U_1/x_1, U_2/x_2, .., U_n/x_n]$ et $N_1 = N[U_1/x_1, .., U_n/x_n]$ sont réductibles, en particulier par $(CR1)$ N_1 est fortement normalisable, donc $T[U_1/x_1, .., U_n/x_n] = M_1\lceil N_1$ est réductible.

- Si $T = \lambda x.N$

Posons $N_1 = N[U_1/x_1, .., U_n/x_n]$.

Par hypothèse d'induction pour tout U réductible (donc a fortiori fortement normalisable selon $(CR1)$), $N_1[U/x] = N[U/x, U_1/x_1, .., U_n/x_n]$ est réductible tout comme $N_1[U/x]\lceil U = N[U/x, U_1/x_1, .., U_n/x_n]\lceil U$ il s'ensuit par le lemme 24 que $\lambda x.N_1$ est réductible, à savoir $\lambda x.N[U_1/x_1, .., U_n/x_n] = M_1\lceil N_1$.

Théorème 5 *tous les termes typables sont réductibles et donc fortement normalisables.*

Preuve: Le lemme 23 nous dit que les variables sont réductibles. Il suffit de particulariser le lemme précédent en prenant $U_i = x_i$ pour tout i. □

3.6 Caractérisation d'une classe de termes de Λ_{pl} fortement normalisables

Remarque 11 *Il est aisé de remarquer que si un terme est à la fois dans Λ_{pl} et dans Λ_t alors réduire ce terme selon $\to_{\beta_{pl}}$ ou selon $\to_{\beta_2}$ donne le même résultat. En effet les rédex de $\to_{\beta_{pl}}$ le sont aussi selon $\to_{\beta_2}$ et les réduits sont les mêmes.* □

Lemme 26 *Soit M un terme de Λ_{pl} qui est aussi dans Λ_t tel que $\Gamma \vdash M : A$ et un terme N tel que $M \to_S N$, alors $\Gamma \vdash N : A$.*

Preuve: Par induction sur la dérivation de $M \to_S N$, le lemme 19 suffit à prouver le résultat. □

Remarque 12 *Ce dernier lemme va nous permettre de considérer le sous-ensemble de Λ_{pl} des termes typables dans Λ_t que nous noterons $\Lambda_{p't}$, sachant que celui-ci est donc stable par $\to_{\beta_{pl}}$ et $\to_S$ et qu'on peut donc lui appliquer les résultats du chapitre 2.*

Notons que les r'esultats du chapitre précédent montrent que les termes clos de $\Lambda_{p't}$ sont en particulier des termes fortement réductibles à une valeur, qui est d'ailleurs unique modulo $\underline{=}_{Tres}$.

4

Typage

Le but de ce chapitre est la caractérisation des termes égalisables à une valeur dans le λ_p-calcul et des termes clos qui sont réductible à une valeur dans λ_{pl}. Ce résultat s'obtient via le typage des termes égalisables (ou réductibles) à une valeur dans le λ_v-calcul, résultat déjà obtenu par d'autres méthodes par L.Egidi, Furio Honsell et S.Ronchi Della Rocca en 1991 (cf [Egidi& alt.91]). Néanmoins notre démonstration est nouvelle et nous montrons qu'elle s'adapte à une variante du λ_v-calcul que nous envisageons, le Lambda-calcul par valeur de tête (λ_{vh}).

Nous étudions tout d'abord ce calcul et montrons qu'il jouit d'un théorème de standardisation en suivant pour l'essentiel la démonstration analogue de Plotkin pour le λ_v-calcul (*cf* [Plotkin 75]). Nous rappelons quelques résultats sur le λ_v-calcul. Ensuite nous caractérisons par un système de types les termes réductibles en une valeur pour les deux calculs λ_v et λ_{vh}. Nous montrons que ce résultat permet de caractériser les termes clos du λ_p-calcul égaux à une valeur. Nous montrons enfin que ce système de type nous permet également de caractériser les termes clos réductibles à valeur de λ_{pl}.

4.1 Le Lambda-calcul par valeur de tête

Nous allons dans cette partie proposer une nouvelle forme de λ_v-calcul et montrer qu'il possède de bonnes propriétés, notamment en ce qu'il admet un théorème de standardisation. Ensuite nous caractériserons par un système de types les termes réductibles à une valeur dans le λ_v-calcul de Plotkin et dans celui que nous proposons.

Le lambda-calcul par valeur a été d'abord considéré par Plotkin [Plotkin 75]. Il consiste à distinguer une classe de lambda-termes qui seront appelés *les valeurs*, et la nouveauté, par rapport au λ-calcul usuel, réside dans la notion de réduction ($\rightarrow_{\beta_v}$), les rédex n'étant cette fois que les termes de la forme $(\lambda x.M)N$, où N est une valeur, le réduit (ou contractum) restant le même. Ceci correspond aux implémentations usuelles du λ-calcul, où l'application d'un terme - considéré comme un programme - appliqué à un autre - considéré comme argument du précédent - suppose que cet argument est pour le moins partiellement évalué.

Plotkin n'avait considéré qu'une seule forme de valeur, à savoir les termes qui ne sont pas une application, c'est-à-dire ceux qui sont, soit une abstraction, soit une variable, soit une constante. Or nous nous sommes aperçu qu'il existe des variations possibles pour cette notion[1], notamment celle-ci que nous allons étudier (et que nous avons baptisé λ_{vh}-calcul),

[1] Celle que nous considérons est issue d'une reflexion sur la sémantique dénotationnelle du λ_p-calcul (et de ses variations conjointes avec la notion de valeur) et de la question du rôle que pourraient y jouer les formes normales, à l'instar de celui qu'elles jouent dans les "graph"-modèles. Ici la notion de réduction gauche par valeur dépend de la notion de valeur considérée et l'on sait l'importance de celle-ci dans ces sémantiques dénotationnelles. Ce sujet reste à explorer.

dans laquelle les valeurs sont certains termes en forme normale de tête (néanmoins les termes clos en forme normale de tête coïncideront avec les valeurs closes).

A vrai dire, cette variation mériterait une étude plus approfondie que détudier ses aspects syntaxiques, ce à quoi nous nous bornons ici. En particulier d'autres variations sur la notion de valeur ont déjà été envisagées par M.Dezani, F.Honsell et S.Ronchi Della Rocca, en relation avec la sémantique opérationnelle qui ressort des machines abstraites associées à ces variations. Il semble néanmoins possible de définir une machine abstraite appropriée et d'étudier ainsi des fondements plus pragmatiques du λ_{vh}-calcul.

Quelques soient les variations que l'on peut envisager, il demeure que pour obtenir de bonnes propriétés syntaxiques, les valeurs doivent constituer une classe stable par substitution, c'est-à-dire que si P et Q sont des valeurs, alors $P[Q/x]$ doit être encore une valeur, puisque dans le cas contraire la confluence du calcul (propriété de Church-Rosser) n'est pas vérifiée, comme il se voit aisément (les rédex avant et aprés substitution ne seraient plus les "mêmes").

Nous verrons dans la suite de chapitre comment on peut typer de façon analogue les termes réductibles en une valeur par un système de type quasiment identique pour le λ_v-calcul et le λ_{vh}-calcul.

4.2 Définitions et Propriétés du λ_{vh}-calcul.

Les termes considérés comme des valeurs dans le λ_{vh}-calcul sont les termes en forme normale de tête

- dont la variable de tête est liée , i.e. les termes de la forme

  $\lambda x_1 \cdots x_n.x_i M_1 \cdots M_r$,

- ou encore de la forme

  $\lambda x_1 \cdots x_n.y$,

  où y est une variable libre ou une constante.

A prendre $n = 0$ dans ce dernier cas, nous pouvons reconnaître que les variables et les constantes sont notamment considérées comme des valeurs.

Remarquons qu'à l'inverse, les termes qui ne sont pas des valeurs sont de la forme

$$\lambda x_1 \cdot x_n.M M_1 \cdot M_r,$$

où $r \geq 1$ et où M est une variable libre, une constante ou une abstraction.

La proposition suivante établit un résultat qui est manifestement exigible de toute notion de valeur, et qui constituent la contrepartie dans le λ_{vh}-calcul d'une proposition analogue établies par Plotkin pour le λ_v-calcul. Cette exigence conduit tout naturellement à la notion de valeur exposée ci-dessus, à partir du moment où l'on souhaite que les valeurs soient des termes en forme normale de tête.

Proposition 2 *Soit $\mathcal{V}$ l'ensemble des termes de Λ qui sont des valeurs. Alors $\forall P, Q \in \mathcal{V}$ $P[Q/x] \in \mathcal{V}$.*

Preuve :

Si P est de la forme $(\lambda x_1 \cdots x_n.x_i M_1 \cdots M_r)$ ou de la forme $\lambda x_1 \cdots x_n.c$, où c est une constante; alors $P[Q/x]$ est aussi de cette forme et c'est donc aussi une valeur.

Si P est de la forme $\lambda x_1 \cdots x_n.y$, où y est une variable libre , alors il se voit aisément que si Q est de cette même forme, il en est de même pour $P[Q/x]$, et si Q est d'une des formes précédentes, il en va de même pour $P[Q/x]$. □

Le λ_{vh}-calcul se définit par le système formel constitué par les règles suivantes:

(α) $(\lambda x.M) = (\lambda y.M[y/x])$, si $y \notin VL(M)$.

(β) (alias β_{vh}) $(\lambda x.M)N = M[N/x]$, si N est une valeur.

$$(\mu)\ \frac{M = N}{MP = NP} \qquad (\rho)\ M = M$$

$$(\nu)\ \frac{M = N}{PM = PN} \qquad (\sigma)\ \frac{M = N}{N = M}$$

$$(\xi)\ \frac{M = N}{\lambda x.M = \lambda x.N} \qquad (\tau)\ \frac{M = N \quad N = P}{M = P}$$

On notera $\lambda_{vh} \vdash M = N$ si la formule $M = N$ se dérive dans le système précédent, $\lambda_{vh} \vdash M \to_{\beta_{vh}} N$ si elle se dérive sans utiliser les règles σ et τ, et $\lambda_{vh} \vdash M \xrightarrow{*}_{\beta_{vh}} N$ si elle se dérive sans utiliser la règle σ.

Nous allons dans la suite considérer (abusivement) que nous travaillons sur des termes modulo α-équivalence. De fait, les éventuels problèmes que pourrait soulever l'α-conversion se résolvent aisément et ne sont pas essentiels, puisqu'ils pourraient se contourner aisément par une notation à la de Bruijn. Mais cette dernière présenterait le défaut sensible d'obscurir la compréhension des démonstrations.

Théorème 6 *Si $\lambda_{vh} \vdash M = N$ et si L est une valeur alors $\lambda_{vh} \vdash M[L/x] = N[L/x]$*

Preuve : Par une simple induction sur la preuve de $\lambda_{vh} \vdash M = N$, selon la dernière règle appliquée , nous ne traiterons ici que le seul cas intéressant de la règle β:

Avec une éventuelle α-conversion pour s'assurer que $x \neq y$, on a:

$\lambda_{vh} \vdash (\lambda y.P)Q = P[Q/y]$, Q étant une valeur. Par définition de la substitution, on a $(\lambda y.P)Q[L/x] = (\lambda y.P[L/x])Q[L/x]$ et d'autre part la proposition 2 montre que $Q[L/x]$ est une valeur, d'où par (β):

$$(\lambda y.P)Q[L/x] = (\lambda y.P[L/x])Q[L/x] = P[L/x][Q[L/x]/y].$$

Enfin, il reste à voir $P[L/x][Q[L/x]/y] = P[Q/x][L/x]$ qui résulte d'un lemme classique sur la substitution. □

4.3 Le théorème de standardisation

Definition 18 *La réduction gauche $\to_{gh}$ consiste à réduire le rédex le plus à gauche dans un terme. En d'autres termes, c'est la plus petite relation sur Λ vérifiant les clauses suivantes:*

1. $(\lambda x.M)N \to_{gh} M'[N'/x]$, *si N est une valeur.*
2. *si $M \to_{gh} M'$ et si M n'est pas une abstraction, alors $MN \to_{gh} M'N$.*
3. *si $M \to_{gh} M'$ et si N est une abstraction, alors $NM \to_{gh} NM'$.*

4. si $M \to_{gh} M'$, *alors* $\lambda x.M \to_{gh} \lambda x.M'$.

Nous noterons la clôture réflexive et transitive de cette relation $\overset{*}{\to}_{gh}$.

Nous allons montrer dans ce chapître que cette réduction est munie de bonnes propriétés (sans être complète, comme c'est le cas de la réduction gauche dans le λ-calcul), au sens elle permet de réduire un terme à une valeur, si on peut égaler ce terme à une valeur dans le λ_{vh}-calcul. Un outil intéressant notamment pour étudier la confluence du calcul est la réduction parallèle. Pour la définir, nous allons utiliser un système formel dont les formules sont de la forme $M \gg M'$ et dont les règles sont les suivantes:

$$(\beta'_{vh})\ \frac{M \gg M' \qquad N \gg N'}{(\lambda x.M)N \gg M'[N'/x]} \text{ si } N \text{ est une valeur.}$$

$$(\rho)\ M \gg M$$

$$(\xi)\ \frac{M \gg M'}{(\lambda x.M) \gg (\lambda x.M')}$$

$$(app)\ \frac{M \gg M' \qquad N \gg N'}{MN \gg M'N'}$$

Remarque 13 *si* $M \in \mathcal{V}$ *et* $M \gg M'$, *alors* $M' \in \mathcal{V}$, *et idem en substituant* $\gg$ *par* $\to_{\beta_{vh}}$ *ou* $\to_{gh}$.

Preuve : Toutes ces réductions sont incluses dans la réduction β^* du lambda-calcul, or il est facile pour cette dernière d'établir qu'un terme qui a une forme norme normale de tête $\lambda\vec{x}.RM_1 \cdots M_n$, où R est une variable ou une constante, ne peut se réduire qu'en un terme de la forme $\lambda\vec{x}.RN_1 \cdots N_n$. On conclut en notant que ces deux formes sont simultanément des valeurs. □

Lemme 27 $\to_{\beta_{vh}} \subset \gg$ *et* $\gg \subset \overset{*}{\to}_{\beta_{vh}}$, *et par conséquent* $\overset{*}{\to}_{\beta_{vh}} = (\gg)^*$.

Preuve: $\to_{\beta_{vh}} \subset \gg$ est évident.

$\gg \subset \overset{*}{\to}_{\beta_{vh}}$ se démontre en vérifiant $M \gg M' \Longrightarrow M \overset{*}{\to}_{\beta_{vh}} M'$, par induction sur la preuve de $M \gg M'$. □

Théorème 7 *La réduction parallèle* $\gg$ *est fortement confluente. La réduction* $\to_{\beta_{vh}}$ *est confluente.*

Preuve:

Montrons que la réduction parallèle $\gg$ est fortement confluente, soit la propriété suivante: $\forall M \in \Lambda \quad M \gg M_1$ et $M \gg M_2 \Longrightarrow \exists M_3 \quad M_1 \gg M_3$ et $M_2 \gg M_3$. Prouvons cette propriété par induction sur M

- $M = x$ ou $M = c$ ne sont pas des cas possibles.
- Si $M = \lambda x.P$ alors $M_1 = \lambda x.P_1$ et $M_2 = \lambda x.P_2$ avec $P \gg P_1$ et $P \gg P_2$. D'où par hypothèse d'induction , il existe P_3 tel que $P_1 \gg P_3$ et $P_2 \gg P_3$. Il suffit alors de prendre $M_3 = \lambda x.P_3$.

•Si $M = PQ$, alors plusieurs sous-cas sont à distinguer selon la dernière règle appliquée dans les preuves de $M \gg M_1$ et de $M \gg M_2$ (en éliminant les cas symétriques et celui de la règle ρ, dont la résolution est immédiate), la règle ξ étant exclue, il ne reste à examiner que les cas suivants:

∘$M \gg M_1$ et $M \gg M_2$ résultent de la règle *app*. Alors $M_1 = P_1Q_1$ et $M_2 = P_2Q_2$, $P \gg P_i$ et $Q \gg Q_i$ pour $i = 1, 2$. Par hypothèse d'induction il existe P_3 et Q_3 tels que $P_i \gg P_3$ et $Q_i \gg Q_3$ pour $i = 1, 2$, et il suffit de prendre $M_3 = P_3Q_3$.

∘$M = (\lambda x.R)Q \gg R_1[Q_1/x] = M_1$ résulte de la règle β'_{vh} et d'autre part $M = (\lambda x.R)Q \gg (\lambda x.R_2)Q_2 = M_2$ de la règle *app*, on sait alors que $R \gg R_i$, $Q \gg Q_i$, et Q est une valeur. Par hypothèse d'induction , il existe R_3 et Q_3 tels que $R_i \gg R_3$ et $Q_i \gg Q_3$ pour $i = 1, 2$, et puisque Q est une valeur, il en va de même pour Q_3. Par conséquent, par le lemme 28 $M_1 = R_1[Q_1/x] \gg R_3[Q_3/x]$ et par β'_{vh} $M_2 = (\lambda x.R_2)Q_2 \gg R_3[Q_3/x]$. il suffit donc de prendre $M_3 = R_3[Q_3/x]$.

∘$M = (\lambda x.R)Q \gg R_i[Q_i/x] = M_i$, pour $i = 1, 2$ résultent de la règle β'_{vh}, Q étant une valeur. Comme précédemment on déduit R_3 et une valeur Q_3, et le lemme 28 montre qu'il suffit de prendre $M_3 = R_3[Q_3/x]$.

$\gg$ est fortement confluente, et par conséquent est confluente. Par le lemme 27, on conclut à la confluence de $\rightarrow_{\beta_{vh}}$. □

Dans le système qui nous a servi à définir la réduction parallèle $\gg$, il va nous être utile de définir aussi la taille de la preuve d'une formule $M \gg M'$, que nous noterons $p_{(M,M')}$, ou encore plus simplement p_M, lorsqu'il n'y a pas d'ambiguïté, comme étant l'entier qui représente le nombre implicite d'étapes nécessaires pour réduire M en M' par la réduction $\rightarrow_{\beta_{vh}}$. Nous notons $n(x, M)$ le nombre d'occurences libres de x dans M. La taille d'une preuve se définit inductivement selon la dernière règle appliquée comme suit:

$$
\begin{array}{ll}
(\beta'_{vh}) & p_{(\lambda x.M)N} = p_M + n(x, M')p_N + 1 \\
(\rho) & p_M = 0 \\
(\xi) & p_{(\lambda x.M)} = p_M \\
(app) & p_{MN} = p_M + p_N
\end{array}
$$

Definition 19 *Une suite de termes $N_1, \cdots, N_r$ est dite une* séquence de réduction standard *(en abrégé: s.r.s.) si chaque terme de cette séquence (excepté le premier) s'obtient en effectuant une réduction dans le précédent ($N_i \rightarrow_{\beta_{vh}} N_{i+1}$), de sorte qu'à chaque étape le redex réduit soit se trouve à droite de ceux que l'on a réduit avant lui, soit n'existait pas auparavant, parce que le terme en position de fonction ne se trouvait pas être une abstraction ou parce que, quoique le terme en position de fonction fût bien une abtraction, l'argument ne se trouvait pas être une valeur. Il est fort utile de constater que les s.r.s. peuvent se définir par le système des règles suivantes:*

1. *toute constante c ou toute variable x constitue une s.r.s.*
2. *Si $N_2, \cdots, N_r$ est une s.r.s. et $N_1 \rightarrow_{gh} N_2$, alors:*
   $N_1, \cdots, N_r$ est une s.r.s. .
3. *Si $N_1, \cdots, N_r$ est une s.r.s. , alors*
   $(\lambda x.N_1), \cdots, (\lambda x.N_r)$ est une s.r.s.

4. Si $M_1, \cdots, M_s$ *et* $N_1, \cdots, N_r$ *sont des s.r.s. , alors*
$(M_1N_1), \cdots, (M_sN_1), \cdots, (M_sN_r)$ *est une s.r.s. .*

Lemme 28
S'il existe une preuve de $M \gg M'$ *de taille* p_M *et de* $N \gg N'$ *de taille* p_N, *où* N *est une valeur, alors il existe une preuve de* $M[N/x] \gg M'[N'/x]$ *de taille* $\leq p_M + n(x, M')p_N$.

Preuve : Par induction lexicographique sur (p_M, M), selon la dernière règle appliquée:

(β'_{vh}) Alors $M = (\lambda y.M_1)M_2$, $M' = M'_1[M'_2/y]$.

On a une preuve de taille p_{M_1} de $M_1 \gg M'_1$, une autre de taille p_{M_2} de $M_2 \gg M'_2$ et $p_M = p_{M_1} + n(y, M'_1)p_{M_2} + 1$.

Par une éventuelle α-conversion on peut supposer $x \neq y$.

Par hypothèse d'induction, on sait qu'il existe une preuve de

$M_1[N/x] \gg M'_1[N'/x]$, de taille $\leq p_{M_1} + n(x, M'_1)p_N$,

et de $M_2[N/x] \gg M'_2[N'/x]$, de taille $\leq p_{M_2} + n(x, M'_2)p_N$, et le lemme 2 nous indique que $M_2[N/x]$ est une valeur, d'où en appliquant maintenant à ces deux prémisses la règle β'_{vh}, on obtient une preuve de

$(\lambda y.M_1[N/x])M_2[N/x] \gg M'_1[N'/x][M'_2[N'/x]/y]$,

formule qui, en vertu d'un lemme classique sur la commutation de la substitution (on peut supposer que y ne figure ni dans n, ni dans N'), n'est autre que

$$((\lambda y.M_1)M_2)[N/x] \gg M'_1[M'_2/y][N'/x]$$

et la taille p de cette preuve est donc telle que:

$p \leq p_{M_1[N/x]} + n(y, M'_1[N'/x])p_{M_2[N/x]} + 1$ y ne figurant pas dans N, il s'ensuit que $n(y, M'_1[N'/x]) = n(y, M'_1)$, et d'autre part:

$$p_{M_1[N/x]} \leq p_{M_1} + n(x, M'_1)p_N \text{ et } p_{M_2[N/x]} \leq p_{M_2} + n(x, M'_2)p_N$$

d'où

$$\begin{aligned} p &\leq p_{M_1} + n(x, M'_1)p_N + n(y, M'_1)(p_{M_2} + n(x, M'_2)p_N) + 1 \\ &= p_{M_1} + n(y, M'_1)p_{M_2} + 1 + n(x, M'_1)p_N + n(y, M'_1)n(x, M'_2)p_N \\ &= p_M + p_N(n(x, M'_1) + n(y, M'_1)n(x, M'_2)) \end{aligned}$$

remarquons enfin que:

$n(x, M') = n(x, M'_1[M'_2/y]) = n(x, M'_1) + n(y, M'_1)n(x, M'_2)$,

et par conséquent $p \leq p_M + p_N n(x, M')$.

(ρ) $M \gg M = M'$ (et $p_M = 0$). Ici s'impose l'induction sur M:

$M = z \neq x$ $M = c$, c'est alors immédiat.

$M = x$, alors $n(x, M) = 1$, et c'est non moins immédiat puisque on a une preuve de $N \gg N'$ de taille p_N par hypothèse.

$M = (M_1M_2)$:

Par hypothèse d'induction , on sait qu'il existe des preuves de $M_1[N/x] \gg M_1[N'/x]$ et de $M_2[N/x] \gg M_2[N'/x]$ de tailles respectivement plus petites que $n(x, M_1)p_N$ et que $n(x, M_2)p_N$.

D'où, en réappliquant *app* on en déduit une preuve de taille plus petite que:

$n(x, M_1)p_N + n(x, M_2)p_N = n(x, M)p_N$.

$M = \lambda z.M_1$, on peut supposer $z \neq x$. Par hypothèse d'induction il existe une preuve de $M_1[N/x] \gg M_1[N'/x]$ de taille plus petite que $n(x, M_1)p_N$, et en réappliquant la reg ξ, on en déduit une preuve de $M \gg M$ de taille plus petite que $n(x, M_1)p_N = n(x, M')p_N$.

(ξ) $M = (\lambda y.M_1)$ et $M' = (\lambda y.M_1')$, on peut supposer $y \neq x$. Il existe par hypothèse d'induction une preuve de $M_1[N/x] \gg M_1'[N'/x]$ de taille plus petite que

$$p_{M_1} + n(x, M_1')p_N = p_M + n(x, M')p_N,$$

ce dont on déduit la preuve cherchée en réappliquant ξ.

(*app*) Alors $M = (M_1 M_2)$ et $M' = (M_1' M_2')$.

Par hypothèse d'induction , il existe une preuve de

$M_1[N/x] \gg M_1'[N'/x]$, de taille $\leq p_{M_1} + n(x, M_1')p_N$

et une de $M_2[N/x] \gg M_2'[N'/x]$, de taille $\leq p_{M_2} + n(x, M_2')p_N$.

On en déduit donc une preuve de $M[N/x] \gg M'[N'/x]$ de taille inférieure à

$p_{M_1} + p_{M_2} + (n(x, M_1') + n(x, M_2'))p_N = p_M + n(x, M')p_N$. □

A partir d'ici la démonstration diverge sensiblement de celle de Plotkin, puisqu'il devient désormais nécessaire de tenir compte de la forme particulière des termes que nous considérons être des valeurs.

Lemme 29 *1. Si $M \gg \lambda y.P' = M'$ possède une preuve de taille p_M, et si M n'est pas une abstraction (auquel cas $M = \lambda y.P$ et $P \gg P'$), c'est-à-dire - comme il se voit aisément - si M est une application, alors il existe P tel que $M \xrightarrow{+}_{gh} R = \lambda y.P \gg \lambda y.P' = M'$, et la preuve de $R = \lambda y.P \gg \lambda y.P' = M'$ a une taille $< p_M$.*

2. Si $M \gg \lambda \vec{y}.N P_1' \cdots P_n' = M'$ ($\vec{y} = y_1, \cdots, y_r$, $r \geq 0$), possède une preuve de taille p_M, où N est une variable ou une constante, et si M n'est pas de la forme $\lambda \vec{y}.N P_1 \cdots P_r$, alors il existe $P_1, \cdots, P_n$ tels que:

$M \xrightarrow{+}_{gh} R = \lambda \vec{y}.N P_1 \cdots P_n \gg \lambda \vec{y}.N P_1' \cdots P_n' = M'$,

et la preuve de $R \gg M'$ a une taille $< p_M$.

Remarquons en particulier que cette deuxième partie du lemme implique le fait suivant: Si $M \gg M'$, où M' est une valeur et M n'est pas une, alors il existe une valeur R telle que $M \xrightarrow{+}_{gh} R \gg M'$ et $p_R < p_M$.

En effet, les valeurs se trouvent être des termes qui sont de la forme $\lambda \vec{y}.N P_1 \cdots P_r$, où N est une variable ou une constante.

Preuve : Montrons simultanément les deux points du lemme par une induction lexicographique sur (p_M, M):

1 La dernière règle appliquée est nécessairement β'_{vh}, d'où:

$M = (\lambda x.M_1)M_2 \gg M'_1[M'_2/x] = M'$,

sachant que M_2 est une valeur et que d'autre part $M_i \gg M'_i$.

Alors $M = (\lambda x.M_1)M_2 \rightarrow_{gh} M_1[M_2/x]$ et d'autre part le lemme 28 nous fournit une preuve de $M_1[M_2/x] \gg M'_1[M'_2/x] = M'$ dont la taille est $< p_M$. Si $M_1[M_2/x]$ est une abstraction alors il suffit de prendre $R = M_1[M_2/x]$ et on a bien $p_R < p_M$, sinon il suffit d'appliquer l'hypothèse d'induction pour trouver R de la forme requise et qui soit tel que $M_1[M_2/x] \xrightarrow{+}_{gh} R \gg M'$ et ainsi par transitivité $M \xrightarrow{+}_{gh} R \gg M'$ avec $p_R < p_M$.

2 Il va nous falloir distinguer plusieurs cas:

•Si $M' = \lambda \vec{y}.NP'_1 \cdots P'_r$ est une abstraction (c'est-à-dire si $\vec{y}$ n'est pas de dimension nulle), alors différents cas se présentent:

◦soit M n'est pas une abstraction, alors en appliquant le point *1* du lemme, on trouve un terme Q, tel que $M \xrightarrow{+}_{gh} Q \gg M'$ avec $p_Q < p_M$. Si Q est de la forme requise ($Q = \lambda \vec{y}.NP_1 \cdots P_n$), il suffit de prendre $R = Q$, sinon appliquons maintenant le deuxième point du lemme à Q, on trouve ainsi un terme R de la forme requise, tel que $M \xrightarrow{+}_{gh} Q \xrightarrow{+}_{gh} R \gg M'$, $p_R < p_M$ et qui par conséquent convient.

◦soit M est une abstraction, i.e. $M = \lambda y_1.P$. Alors, dans la preuve de $M \gg M'$ on a nécessairement appliqué la règle ξ à la prémisse $P \gg \lambda y_2 \cdots y_r.NP'_1 \cdots P'_n$ et $p_P = p_M$. P étant moins complexe que M et n'étant pas de la forme $\lambda y_2 \cdots y_r.NP_1 \cdots P_n$, on peut lui appliquer l'hypothèse d'induction sur le point *2* du lemme: il existe donc R' tel que $P \xrightarrow{+}_{gh} R' \gg \lambda y_2 \cdots y_r.NP'_1 \cdots P'_n$ avec $p_{R'} < p_P$. Il suffit dès lors de réappliquer la règle ξ, pour vérifier que $R = \lambda y_1.R'$ convient.

•Si M' n'est pas une abstraction ,soit $M' = NP'_1 \cdots P'_r$. Deux cas peuvent se présenter:

◦La dernière règle appliquée est *app*.

Alors $M = PQ$ et $P \gg NP'_1 \cdots P'_{r-1}$, $Q \gg P'_r$, P étant moins complexe que M et n'étant pas de la forme $NP_1 \cdots P_{r-1}$, on peut lui appliquer l'hypothèse d'induction sur le point *2* du lemme:

il existe donc $R' = NP_1 \cdots P_{r-1}$ tel que $P \xrightarrow{+}_{gh} R' \gg NP'_1 \cdots P'_{r-1}$, tel que $p_{R'} < p_P$. On en déduit que $PQ \xrightarrow{+}_{gh} R'Q \gg NP'_1 \cdots P'_{r-1}P'_r$ (remarquer que dans la séquence de réduction $P \xrightarrow{+}_{gh} R'$, aucun terme ne peut être une abstraction, ce qui autorise a appliquer la clause *2* de la définition de $\rightarrow_{gh}$) et $p_{R'Q} = p_{R'} + p_Q < p_P + p_Q = p_M$. Par conséquent $R = R'Q$ convient

◦La dernière règle appliquée est β'_{vh}. La résolution de ce cas est tout à fait similaire à celle du point *1* du lemme. □

Remarque 14 *Dans les deux points du lemme précedent, on peut exiger que le terme R recherché soit le premier terme rencontré dans la réduction gauche issue de M qui aît la forme requise.*

Preuve : Il suffit de remarquer, en suivant la preuve du lemme précédent, que, si cette propriété est exigée du terme R, elle se conserve à chaque pas de l'induction. □

Lemme 30 *Si $M \gg M' \rightarrow_{gh} M''$ et $M \neq M'$ alors il existe un terme P tel que $M \xrightarrow{+}_{gh} P \gg M''$ et $p_P < p_M$.*

(∗) *De plus, si M' n'est pas une abstraction alors que M'' en est une, alors il n'y a aucune abstraction précédant P dans la séquence de réduction $M \xrightarrow{+}_{gh} P$.*

Preuve : Par induction lexicographique sur (p_M, M), selon la dernière règle appliquée:

(β'_{vh}) Alors $M = (\lambda x.M_1)M_2 \gg M_1'[M_2'/x] = M'$, $M_1 \gg M_1'$ $M_2 \gg M_2'$, M_2 étant une valeur. Alors $M \rightarrow_{gh} M_1[M_2/x]$.

•si $M_1[M_2/x] = M'$, alors il suffit de poser $P = M''$, puisqu'alors $M \rightarrow_{gh} M' \rightarrow_{gh} M''$. A supposer en outre que M' n'est pas une abstraction, le point (∗) s'ensuit immédiatement.

•si $M_1[M_2/x] \neq M'$, alors d'après le lemme 28, il existe une preuve de $M_1[M_2/x] \gg M_1'[M_2'/x] = M'$ dont la taille est plus petite que $p_{M_1} + n(x, M_1')p_{M_2}$, et donc strictement plus petite que $p_M = p_{M_1} + n(x, M_1')p_{M_2} + 1$. Il s'ensuit par hypothèse d'induction qu'il existe P tel que $M = (\lambda x.M_1)M_2 \rightarrow_{gh} M_1[M_2/x] \xrightarrow{+}_{gh} P \gg M''$, ce P convient. (∗) s'ensuit alors de l'hypothèse d'induction sur ce même point (∗) et du fait qu'on ne rajoute dans la séquence que le terme M, qui n'est pas une abstraction.

(ρ) ce cas est impossible puisque $M \neq M'$.

(ξ) Alors $M = (\lambda x.M_1) \gg (\lambda x.M_1') = M'$. Par conséquent $M'' = (\lambda x.M_1'')$, et d'autre part $M_1 \gg M_1' \rightarrow_{gh} M_1''$ avec $M_1 \neq M_1'$. Par hypothèse d'induction sur M cette fois, il existe P_1 tel que $M_1 \xrightarrow{+}_{gh} P_1 \gg M_1''$ et $p_{P_1} < p_{M_1}$. Il suffit d'appliquer ξ pour conclure (remarquer $p_P = p_{P_1} < p_{M_1} = p_M$).

(*app*) Ici $M = M_1M_2 \gg M_1'M_2' = M'$, on dispose d'une preuve de $M_i \gg M_i'$ de taille p_{M_i} et $p_M = p_{M_1} + p_{M_2}$.

Analysons la preuve de $M' \rightarrow_{gh} M''$:

•$M' \rightarrow_{gh} M''$ résulte de la clause *1.*

Dans ce cas $M' = (\lambda x.M_3')M_2' \rightarrow_{gh} M_3'[M_2'/x] = M''$, et M_2' est une valeur et $M_1 \gg (\lambda x.M_3')$ deux cas:

∘M_1 est une abstraction: $M_1 = (\lambda x.M_3)$.

Alors il est facile de constater que $M_3 \gg M_3'$. Deux cas se distinguent:

◇M_2 n'est pas une valeur. Alors, par le lemme 29, il existe une valeur R_2, telle que $M_2 \xrightarrow{+}_{gh} R_2 \gg M_2'$ et qui est la première valeur rencontrée dans la séquence de réduction $M_2 \xrightarrow{+}_{gh} R_2$. D'où $(\lambda x.M_3)M_2 \xrightarrow{+}_{gh} (\lambda x.M_3)R_2 \gg M''$, avec $p_{(\lambda x.M_3)R_2} < p_M$. Il ne reste plus qu'à appliquer l'hypothèse d'induction pour trouver P. Pour satisfaire (∗), il suffit de remarquer que la séquence $(\lambda x.M_3)M_2 \xrightarrow{+}_{gh} (\lambda x.M_3)R_2$ ne contient aucune abstraction.

◇M_2 est une valeur. Par le lemme 28, on a:

$$M = (\lambda x.M_3)M_2 \rightarrow_{gh} M_3[M_2/x] \gg M_3'[M_2'/x] = M''$$

et il suffit de prendre $P = M_3[M_2/x]$. (∗) est alors évidente.

∘M_1 n'est pas une abstraction. On a $M_1 \gg \lambda x.M_3'$, d'où par le lemme 29, il existe $R = \lambda x.R'$ tel que $M_1 \xrightarrow{+}_{gh} R \gg \lambda x.M_3'$ avec $p_R < p_{M_1}$. De plus la remarque précedente permet d'affirmer que parmi les termes de la séquence de réduction $M_1 \xrightarrow{+}_{gh} R$, seul R est une abstraction, ce qui nous autorise à en déduire

maintenant que $M_1M_2 \xrightarrow{+}_{gh} RM_2$. Remarquons que $RM_2 \gg M'$ et que $p_{RM_2} < p_M$, ce qui par hypothèse d'induction garantit l'existence d'un terme P tel que $RM_2 \xrightarrow{+}_{gh} P \gg M''$ et on constate par transitivité que P convient. $(*)$ rèsulte alors du fait que $M_1M_2 \xrightarrow{+}_{gh} RM_2$ est une sèquence qui ne comporte aucune abstraction.

- Si $M' \rightarrow_{gh} M''$ résulte de la clause *2*, alors on a $M' = M_1'M_2' \rightarrow_{gh} M_1''M_2' = M''$ avec $M_1' \rightarrow_{gh} M_1''$, M_1' n'ètant pas une abstraction. Alors les prémisses des règles appliquées nous disent que $M_1 \gg M_1' \rightarrow_{gh} M_1''$. On peut ainsi appliquer l'hypothèse d'induction puisque $p_{M_1} \leq p_M$ et que M_1 est moins complexe que M. Il existe donc P_1 tel que $M_1 \xrightarrow{+}_{gh} P_1 \gg M_1''$. Or on peut supposer qu'il n'y a pas d'abstraction précédant P_1 dans la séquence $M_1 \xrightarrow{+}_{gh} P_1$, puisque, soit M_1'' n'est pas une abstraction, auquel cas c'est évident, soit M_1'' est une abstraction, et c'est précisément l'hypothèse d'induction faite sur le point $(*)$.

  Ce dont nous déduisons que $M_1M_2 \xrightarrow{+}_{gh} P_1M_2 \gg M_1''M_2' = M''$ et il suffit de poser $P = P_1M_2$ pour conclure.

- Si $M' \rightarrow_{gh} M''$ résulte de la clause *3*, alors: $M' = M_1'M_2' \rightarrow_{gh} M_1'M_2'' = M''$, $M_2' \rightarrow_{gh} M_2''$, M_2' n'étant pas une valeur, $M_1' = \lambda y.M_3$.

  Les prémisses des règles appliquées nous disent cette fois que $M_2 \gg M_2' \rightarrow_{gh} M_2''$ et d'autre part que $M_1 \gg \lambda y.M_3$. Par le lemme 29, il existe donc $R_1 = \lambda y.R_3$ tel que $M_1 \xrightarrow{*}_{gh} R_1 \gg \lambda y.M_3$, avec $P_{R_1} < P_{M_1}$ et R_1 étant la première abstraction rencontrée dans la séquence $M_1 \xrightarrow{*}_{gh} R_1$, et par hypothèse d'induction il existe R_2, tel que $M_2 \xrightarrow{+}_{gh} R_2 \gg M_2''$, et comme M_2' n'est pas une valeur, il n'y a aucune valeur dans la séquence $M_2 \xrightarrow{+}_{gh} R_2$. D'où:

  $M = M_1M_2 \xrightarrow{*}_{gh} (\lambda y.R_3)M_2 \xrightarrow{+}_{gh} (\lambda y.R_3)R_2 \gg M_1'M_2'' = M''$,

  et on peut donc poser $R = (\lambda y.R_3)R_2$. $(*)$ est évidente. □

Lemme 31 *Si $M_1, \cdots, M_j$ est une s.r.s. et $M \gg M_1$, alors il existe une s.r.s. $N_1, \cdots, N_k$ avec $N_1 = M$ et $N_k = M_j$.*

Preuve : Par une induction lexicographique sur (j, p_M, M) et selon la dernière règle appliquée pour dériver $M \gg M_1$:

(β'_{vh}) Alors $M = (\lambda x.P_1)P_2, M_1 = P_1'[P_2'/x]$, P_2 est une valeur et $P_i \gg P_i'$. $M \rightarrow_{gh} P_1[P_2/x]$, ce dont on par le lemme 28 $P_1[P_2/x] \gg P_1'[P_2'/x] = M_1$, avec $p_{P_1[P_2/x]} < p_M$, d'où par hypothèse d'induction sur p_M, il existe une s.r.s. $R_1, \cdots, R_l$ avec $R_1 = P_1[P_2/x]$ et $R_l = M_j$. Mais comme $M \rightarrow_{gh} R_1$, $M, R_1, \cdots, R_l$ est une s.r.s. qui convient.

(ρ) Ce cas est immédiat.

(ξ) Alors $M = (\lambda x.P)$, $M_1 = (\lambda x.P_1)$, $P \gg P_1$. En examinant la dernière règle appliquée pour montrer que $M_1, \cdots, M_j$ est une s.r.s. , on constate que $M_i = (\lambda x.P_i)$ pour $1 \leq i \leq j$, et que $P_1, \cdots, P_j$ est une s.r.s. Par hypothèse d'induction sur M, il existe une s.r.s. $Q_1, \cdots, Q_k$, telle que $Q_1 = P$ et $Q_k = P_j$ et il apparaît que la s.r.s. $(\lambda x.Q_1), \cdots, (\lambda x.Q_k)$ convient.

(*app*) Alors $M = PQ$, $M_1 = P_1Q_1$, $P \gg P_1$, $Q \gg Q_1$. Deux cas apparaissent selon la dernière règle appliquée pour montrer que $M_1, \cdots, M_j$ est une s.r.s. .

- Si c'est la clause *2*, alors $M \gg M_1 \rightarrow_{gh} M_2$ et $M_2, \cdots, M_j$ est une s.r.s. Par le lemme 30, il existe P tel que $M \xrightarrow{+}_{gh} P \gg M_2$. Par hypothèse d'induction sur j, il existe une s.r.s. dont les extrèmes sont P et M_j. Comme $M \xrightarrow{+}_{gh} P$, appliquer plusieurs fois la clause *2* montre qu'il existe une s.r.s. dont les extrèmes sont M et M_j.
- Si c'est la règle *4*. Alors $j = r + s - 1$ et $M_1, \cdots, M_j = (P_1 Q_1), \cdot, (P_s Q_1), \cdot, (P_s Q_r)$, $P \gg P_1$, $Q \gg Q_1$ $P_1, \cdot, P_s$ et $Q_1, \cdot, Q_r$, étant des s.r.s. , Par hypothèse d'induction il existe deux s.r.s. $P = P'_1, \cdots, P'_{s'} = P_s$ et $Q = Q'_1, \cdots, Q'_{r'} = Q_r$ et il suffit maintenant de réappliquer la clause *4*. □

Théorème 8 *(théorème de standardisation)* $\lambda_{vh} \vdash M \xrightarrow{*}_{\beta_{vh}} N$ *si et seulement si il existe une s.r.s.* $M_1, \cdots, M_j$ *telle que* $M_1 = M$ *et* $M_j = N$.

Preuve : Une s.r.s. est en particulier une séquence de réduction par $\rightarrow_{\beta_{vh}}$, ce dont s'ensuit l'une des implications.

Réciproquement, si $\lambda_{vh} \vdash M \xrightarrow{*}_{\beta_{vh}} N$, par le lemme 27 il existe $P_1, \cdots, P_n$ tels que $P_1 \gg \cdots \gg P_n$, $M = P_1$ et $N = P_n$. Montrons alors le résultat cherché par induction sur n. Si $n = 1$, alors il est facile de voir que $M = N$ constitue une s.r.s. , sinon il existe par hypothèse d'induction une s.r.s. $Q_1, \cdots, Q_m$ avec $Q_1 = P_2$ et $Q_m = P_n$. Comme $M \gg P_2$, il ne reste plus qu'appliquer le lemme 31. □

Corollaire 1 $\lambda_{vh} \vdash M \rightarrow_{\beta_{vh}} N$, *où* N *est une valeur, si et seulement si il existe une valeur* N' *telle que* $M \xrightarrow{*}_{gh} N'$.

Preuve : Si $M \xrightarrow{*}_{gh} N$, où N est une valeur, alors $\lambda_{vh} \vdash M \rightarrow_{\beta_{vh}} N$ et cela conclut l'une des implications. Pour l'autre, remarquons que si $\lambda_{vh} \vdash M \rightarrow_{\beta_{vh}} N$, alors par le théorème de standardisation, il existe une s.r.s. $M_1, \cdots, M_j$ telle que $M_1 = M$ et $M_j = N$. Soit N' la première valeur rencontrée dans cette séquence. Il reste à vérifier que le début de cette s.r.s. jusqu'à N' consiste en une séquence de réduction gauche. Supposons que ce ne soit pas le cas, alors il y aurait un rédex de tête qui n'aurait pas été réduit, celui-ci ne pourrait plus être réduit par la suite et par conséquent tous les termes ultérieurs possèderaient un rédex de tête, ce qui est contradictoire. □

4.4 Quelques rappels sur le λ_v-calcul

Dans le λ_v- calcul, la réduction gauche consiste à réduire le rédex par valeur le plus à gauche dans un terme.

Plotkin a montré la proposition suivante, dont notre corollaire 1 est la contrepartie pour le λ_{vh}-calcul, et qui est de même un corollaire immédiat de son théorème de standardisation:

Proposition 3 *il existe une valeur* N *telle que* $M \overset{\lambda_v}{=} N$ *si et seulement si il existe une valeur* N' *telle que* $M \xrightarrow{*}_{g} N'$.

Si M vérifie la deuxième propriété, il sera dit *réductible à une valeur.* La proposition énonce donc que tout terme égalisable à une valeur est réductible à une valeur.

Nous rappelons que la proposition suivante vaut aussi pour les deux calculs (pour λ_{vh}, c'est la proposition 6):

Proposition 4 *Soit* $\mathcal{V}$ *l'ensemble des valeurs. Alors* $\forall P, Q \in \mathcal{V}$ $P[Q/x] \in \mathcal{V}$.

A partir de maitenant, sauf précision, ce que nous faisons vaut pour les deux calculs.

Proposition 5 *Soit $\mathcal{N}$ l'ensemble des termes réductibles à une valeur.*
Alors $\forall P, Q \in \mathcal{N}\ P[Q/x] \in \mathcal{N}$.

Preuve : Le théorème 28 pour le λ_{vh}-calcul et un théorème qui est son homologue pour le λ_v-calcul montrent que, si $P \xrightarrow{*}_{\beta_v} P'$ et $Q \xrightarrow{*}_{\beta_v} Q'$, alors $P[Q/x] \xrightarrow{*}_{\beta_v} P'[Q'/x]$, puisque, de même que dans le λ-calcul, la clôture transitive de la réduction parallèle est égale à celle de $\rightarrow_{\beta_v}$. Si choisissons $P', Q' \in \mathcal{V}$, $P'[Q'/x] \in \mathcal{V}$ s'ensuit de la proposition précédente et par conséquent $P[Q/x] \in \mathcal{N}$.

Lemme 32 *Si $P[Q/x] \in \mathcal{V}$ alors $P \in \mathcal{V}$*

Preuve : Par l'absurde: Pour le λ_v-calcul:

Si P n'était pas une valeur, alors P serait une application et il en irait de même pour $P[Q/x]$.

Pour le λ_{vh}-calcul:

Si P n'était pas une valeur, alors P serait de la forme $\lambda x_1..x_n.MM_1..M_r$, avec $r \geq 1$ où M serait une variable libre y, une constante ou une abstraction. Il est évident dans les deux derniers cas, et dans le premier si $y \neq x$, qu'il en irait de même pour $P[Q/x]$ qui ne saurait appartenir à $\mathcal{V}$, d'où contradiction.

Si $M = x$, c'est évident aussi sauf si Q est une application. Mais alors Q s'écrit $Q_1..Q_n$ où Q_1 n'est pas une application. Il en ressort que $P[Q/x] = \lambda x_1..x_n.Q_1..Q_nM_1..M_r$ et Q_1 ne saurait bien évidemment pas être une variable liée dans ce terme, d'où $P[Q/x] \notin \mathcal{V}$. □

4.5 Le système $\mathcal{D}*$

Nous allons construire un système de types qui va nous permettre de caractériser les termes réductibles à une valeur. Ce système est voisin du système $\mathcal{D}$ introduit par Coppo et M. Dezani-Ciancaglini (*cf* [Coppo et alt.78] et voir aussi [Krivine 90]) qui a constitué notre source d'inspiration. Ce dernier sert à typer les termes normalisables du λ-calcul. Nous considérons indifféremment λ_v ou λ_{vh}. Un résultat similaire pour le λ_v-calcul a été démontré par des techniques sémantiques par L.Egidi, F.Honsell, S.Ronchi della Rocca (*cf* [Egidi& alt.91]). La démonstration suit celle de Coppo et M. Dezani-Ciancaglini, mais il faut évidemment adapter la notion d'ensemble saturé et le thérème de standardisation s'avère ici capital. Il est à noter également que c'est le fait que l'interpretation de tout type est non vide qui permet d'assurer que les termes typables sont réductibles à une valeur. Le lemme 38 représente la complication essentielle de la réciproque.

Les types sont les formules construites avec:

une constante $*$, représentant le type des valeurs;

des variables $X, Y, ..$;

les connecteurs $\rightarrow$ et $\wedge$.

Une déclaration de variable est, par définition, un couple (x, A), où x est une variable du λ_v-calcul et A un type; elle sera notée $x : A$.

Un contexte Γ est un ensemble fini (éventuellement vide) de déclarations de variables distinctes.

$\Gamma, x : A$ représente le contexte Γ augmenté de la déclaration $x : A$.

Etant donnés M un terme du λ-calcul, A un type et Γ un contexte, on définit les règles qui permettent de dériver le jugement "t est de type A dans le contexte Γ" (jugement que l'on notera $\Gamma \vdash t : A$ ou simplement $\vdash t : A$ si Γ est vide) comme suit:

$$(\mathsf{var})\ \frac{x \text{ variable}}{\Gamma, x : A \vdash x : A} \qquad (\mathsf{app})\ \frac{\Gamma \vdash M : A \to B \quad \Gamma \vdash N : A}{\Gamma \vdash MN : B}$$

$$(\mathsf{abs})\ \frac{\Gamma, x : A \vdash M : B}{\Gamma \vdash \lambda x.M : A \to B} \qquad (\wedge\mathsf{I})\ \frac{\Gamma \vdash M : A \quad \Gamma \vdash M : B}{\Gamma \vdash M : A \wedge B}$$

$$(\wedge\mathsf{E1})\ \frac{\Gamma \vdash M : A \wedge B}{\Gamma \vdash M : A} \qquad (\wedge\mathsf{E2})\ \frac{\Gamma \vdash M : A \wedge B}{\Gamma \vdash M : B}$$

Les dernières règles vont légèrement différer selon que l'on considère le λ_v-calcul ou le λ_{vh}-calcul:

Pour le λ_v-calcul:

$$(*_1)\ \frac{x \text{ n'est pas déclaré dans } \Gamma}{\Gamma \vdash x : *}$$

$$(*_2)\ \frac{M \text{ constante ou abstraction}}{\Gamma \vdash M : *}$$

Pour le λ_{vh}-calcul:

$$(*_1)\ \frac{x \text{ n'est pas déclaré dans } \Gamma}{\Gamma \vdash x : *}$$

$$(*_2)\ \frac{c \text{ est une constante}}{\Gamma \vdash c : *}$$

$$(*_3)\ \frac{r \geq 0 \text{ et } n \geq 1}{\Gamma \vdash \lambda y_1..y_n.y_1 M_1..M_r : *}$$

$$(*_4)\ \frac{\Gamma \vdash N : *}{\Gamma \vdash \lambda y_1..y_n.N : *}$$

Remarque 15 *Si $M \in \mathcal{V}$ alors $\Gamma \vdash M : *$ sauf dans les cas suivant:*

pour le λ_v-calcul, lorsque $M = x$ et $x \in \Gamma$.

pour le λ_{vh}-calcul, lorsque $M = \lambda y_1..y_n.x$ et $x \in \Gamma$.

Néanmoins, on constate que toutes les valeurs peuvent être typées dans un certain contexte par l'une des règles $*$ et notamment dans le contexte vide.

4.6 Les termes typables sont réductibles à une valeur

Etant donnés deux parties $\mathcal{X}$ et $\mathcal{Y}$ de Λ, on définit la partie notée $\mathcal{X} \to \mathcal{Y}$ comme suit:

$$u \in \mathcal{X} \to \mathcal{Y} \Leftrightarrow \forall t \in \mathcal{X} \quad (u)t \in \mathcal{Y}$$

Une partie $\mathcal{X}$ de Λ sera dite saturée si, pour tout terme N réductible à une valeur et pour tous termes $P, N_1, .., N_n$, on a:

$$P[N/x]N_1..N_n \in \mathcal{X} \Longrightarrow (\lambda x.P)NN_1..N_n \in \mathcal{X}.$$

L'intersection de deux parties saturées est également saturée.
Rappelons que $\mathcal{N}$ représente l'ensemble des termes réductibles à une valeur.

Proposition 6 *$\mathcal{N}$ est saturé.*

Preuve: Si N est réductible à une valeur V, alors par la proposition 5:

$$\begin{aligned}\lambda_v(\text{resp.}\lambda_{vh}) \vdash (\lambda x.P)NN_1..N_n &= (\lambda x.P)VN_1..N_n \\ &= P[V/x]VN_1..N_n \\ &= P[N/x]NN_1..N_n\end{aligned}$$

Donc si $P[N/x]NN_1..N_n \in \mathcal{N}$, il est égal une valeur et $(\lambda x.P)NN_1..N_n$ l'est aussi. La proposition 3 montre alors que $(\lambda x.P)NN_1..N_n \in \mathcal{N}$ □

Proposition 7 *Si $\mathcal{Y}$ est saturée alors $\mathcal{X} \to \mathcal{Y}$ est saturée*

Preuve : Supposons que $(M[N/x])N_1..N_n \in \mathcal{X} \to \mathcal{Y}$, où N est réductible à une valeur. Si $P \in \mathcal{X}$, alors $(M[N/x])N_1..N_nP \in \mathcal{Y}$, donc $(\lambda x.M)NN_1..N_nP \in \mathcal{Y}$ et par conséquent $(\lambda x.M)NN_1..N_n \in \mathcal{X} \to \mathcal{Y}$. □

Proposition 8 *Si $\mathcal{Y} \subset \mathcal{N}$ alors $\mathcal{X} \to \mathcal{Y} \subset \mathcal{N}$.*

Preuve : Soient t et u tels que $ut \in \mathcal{N}$. Ceci implique que la réduction gauche de u se termine et forcément sur une abstraction et le résultat est donc évident s'agissant du λ_v-calcul. Pour le λ_{vh}-calcul: si ut a une forme normale de tête alors il en va de même pour u (c'est vrai dans le λ-calcul et a fortiori dans le λ_{vh}-calcul) et celle-ci s'obtient par réduction gauche. Supposons que u soit indéfini, c'est-à-dire ne soit pas une valeur, alors $u = \lambda x_1..x_n.zM_1..M_r$ où z est une variable libre et $r \geq 1, n \geq 0$, le cas $n = 0$ est impossible et t doit être réductible à une valeur t', par conséquent ut se réduit à gauche par valeur en $\lambda x_2..x_n.zM_1[t'/x_1]..M_r[t'/x_1]$ qui n'est pas réductible à une valeur, d'où contradiction. □

Nous allons maintenant définir l'interprétation $[\![A]\!]$ d'un type A, comme un sous-ensemble saturé de $\mathcal{N}$, par induction sur le type A:

si $A = *$ ou si $A = X$ (variable de type), alors $[\![A]\!] = \mathcal{N}$;

si $A = B \to C$, alors $[\![A]\!] = [\![B]\!] \to [\![C]\!]$;

si $A = B \wedge C$, alors $[\![A]\!] = [\![B]\!] \cap [\![C]\!]$.

Lemme 33 *Pour tout type A, $[\![A]\!] \subset \mathcal{N}$ et $[\![A]\!]$ est saturée.*

Preuve : Induction immédiate sur la complexité du type en utilisant la proposition 8. □

Lemme 34 *Soit c une constante. Pour tout type A, $\exists n_A \forall m \geq n_A \quad C_m = \lambda x_1..x_m.c \in [\![A]\!]$.*
En particulier pour tout type A, $[\![A]\!] \neq \emptyset$.

Par induction sur le type A:

$A = X$ ou $A = *$: il suffit de prendre $n_A = 0$.

Si $A = B \to C$ alors par hypothèse d'induction n_B et n_C existent. Pour tout $t \in [\![B]\!]$, $t \in \mathcal{N}$ et donc $(\lambda x_1..x_m.c)t$ se réduit à gauche en $(\lambda x_2..x_m.c)$ qui est dans $[\![C]\!]$ pourvu que $m \geq n_C + 1$. Par conséquent, par définition, $\lambda x_1..x_m.c \in [\![B \to C]\!]$ pour $m \geq n_C + 1 = n_A$.

Si $A = B \wedge C$: il suffit de prendre $n_A = sup(n_B, n_C)$. □

Lemme 35 *Lemme d'adéquation Si* $x_1 : A_1, .., x_n : A_n \vdash M : A$ *et* $N_1 \in [\![A_1]\!], .., N_n \in [\![A_n]\!]$ *alors* $M[N_1/x_1, .., N_n/x_n] \in [\![A]\!]$.

Par induction sur la preuve de $x_1 : A_1, .., x_n : A_n \vdash M : A$, selon la dernière règle utilisée:

Si c'est la règle **var**, alors $A = A_i$ et $M[N_1/x_1, .., N_n/x_n] = N_i$ et on a bien $N_i \in [\![A_i]\!]$.

Si c'est la règle **abs**, alors $A = B \to C$ et $M = \lambda x.P$ et on peut supposer modulo α-conversion que x n'est pas une variable libre de N_i. Par hypothèse d'induction, on a $P[Q/x, N_1/x_1, .., N_n/x_n]$ quelque soit $Q \in [\![B]\!]$. L'hypothèse faite sur x permet de voir que $P[Q/x, N_1/x_1, .., N_n/x_n] = P[N_1/x_1, .., N_n/x_n][Q/x]$. Comme $Q \in \mathcal{N}$ et que $[\![C]\!]$ est saturée, on en déduit $\forall Q \in [\![B]\!](\lambda x.P[N_1/x_1, .., N_n/x_n])Q \in [\![C]\!]$, d'où $(\lambda x.P)[N_1/x_1, .., N_n/x_n] \in [\![B]\!] \to [\![C]\!] = [\![A]\!]$.

Si c'est la règle **app**, alors $M = PQ$, P étant de type $B \to A$ et Q de type B. Par hypothèse d'induction $P[N_1/x_1, .., N_n/x_n] \in [\![B]\!] \to [\![A]\!]$ et $Q[N_1/x_1, .., N_n/x_n] \in [\![B]\!]$, d'où $PQ[N_1/x_1, .., N_n/x_n] \in [\![A]\!]$

Si c'est la règle $\wedge$ I, alors $A = B \wedge C$, alors par hypothèse d'induction $M[N_1/x_1, .., N_n/x_n] \in [\![B]\!] \cap [\![C]\!] = [\![A]\!]$.

Si c'est la règle $\wedge$ E1 ou la règle $\wedge$ E2, l'hypothèse d'induction donne immédiatement le résultat, puisque $[\![A \wedge B]\!] = [\![B \wedge A]\!] \subset [\![A]\!]$.

Pour le λ_v-calcul:

Si c'est la règle $*_1$, Alors $M = x$ et $A = *$ et il s'agit de voir que $x_i[N_1/x_1, .., N_n/x_n] = x \in [\![*]\!]$, ce qui est évident puisque $[\![*]\!] = \mathcal{N}$.

Si c'est la règle $*_2$ alors M est nécessairement une abstraction ou une constante. Mais alors, il en va de même pour $M[N_1/x_1, .., N_n/x_n]$ qui est donc un terme de $\mathcal{N} = [\![*]\!]$.

Pour le λ_{vh}-calcul:

Si c'est l'une des règles $*_1$ ou $*_2$, le raisonnement est le même que pour la règle $*_1$ dans le cas du λ_v-calcul.

Si c'est la règle $*_3$ alors $M = \lambda y_1..y_n.y_1 M_1..M_r$, d'où

$$M[N_1/x_1, .., N_n/x_n] = \lambda y_1..y_n.y_1 M_1[N_1/x_1, .., N_n/x_n]..M_r[N_1/x_1, .., N_n/x_n]$$

est donc un terme de $\mathcal{N} = [\![*]\!]$, puisque c'est une valeur.

Si c'est la règle $*_4$, alors $M = \lambda y_1..y_n.N$ (aucune des variable y_j n'etant libre dans N) et l'hypi nous dit que $N[N_1/x_1, .., N_n/x_n] \in [\![*]\!] = \mathcal{N}$, d'où $M[N_1/x_1, .., N_n/x_n] = \lambda y_1..y_n.N[N_1/x_1, .., N_n/x_n] \in \mathcal{N} = [\![*]\!]$. □

Théorème 9 *Si* M *est un terme clos typable dans* $\mathcal{D}*$*ou encore si* M *est typable dans le contexte vide, alors* M *est réductible à une valeur.*

Preuve : Soit $x_1 : A_1, .., x_n : A_n \vdash M : A$. Nous avons vu que l'interprétation de tout type est non vide, appliquons le lemme d'adéquation à des N_i quelconques choisis dans $[\![A_i]\!]$. On en déduit $M \in [\![A]\!]$, d'où $M \in \mathcal{N}$. □

4.7 Les termes réductibles à une valeur sont typables

Proposition 9 *Soient* Γ *un contexte et* $x_1, .., x_n$ *des variables non déclarées dans* Γ.
Si $\Gamma, x_1 : A_1, .., x_n : A_n \vdash M : A$, *et si* $\Gamma \vdash N_i : A_i$, *pour* $1 \leq i \leq n$, *alors* $\Gamma \vdash M[N_1/x_1, .., N_n/x_n] : A$.

Preuve : Par induction sur la preuve de $\Gamma, x_1 : A_1, .., x_n : A_n \vdash M : A$:

Si c'est la règle **var**, alors soit $M = x_i$, auquel cas c'est immédiat, soit $M = y$, $M \neq x_i$, et alors $y : A$ est déclarée dans Γ et **var** donne le résultat.

Pour le λ_v-calcul:

Si c'est la règle $*_1$, alors $M = x$ et $M[N_1/x_1, .., N_n/x_n] = x$ et alors la conclusion n'est autre que l'hypothèse faite sur N_i, soit $M \neq x$, la conclusion s'obtient en appliquant la règle $*_1$.

Si c'est la règle $*_2$, M est une abstraction ou une constante et évidemment il en est de même pour $M[N_1/x_1, .., N_n/x_n]$. On conclut en lui appliquant la règle $*_2$.

Pour le λ_{vh}-calcul:

Si c'est la règle $*_1$, $M = x$ et $M[N_1/x_1, .., N_n/x_n] = x$. Or par hypothèse $\Gamma \vdash N_i : *$, la conclusion s'obtient en appliquant la règle $*_1$.

Si c'est la règle $*_2$, $M = c$, c étant une constante, donc $M[N_1/x_1, .., N_n/x_n] = M$, on conclut par une application de la règle $*_2$.

Si c'est la règle $*_3$, $M = \lambda y_1..y_n.y_1 M_1..M_r$, d'où :

$$M[N_1/x_1, .., N_n/x_n] = \lambda y_1..y_n.y_1 M_1[N_1/x_1, .., N_n/x_n]..M_r[N_1/x_1, .., N_n/x_n],$$

on conclut en appliquant la règle $*_3$.

Dans les autres cas (en particulier pour $*_4$) appliquer l'hypothèse d'induction aux prémisses de la règle, puis réappliquer cette dernière. □

Un type est dit *premier* s'il n'est pas une conjonction. Tout type est une conjonction de types premiers qui seront dits les *facteurs premiers* de ce type.

Lemme 36 *Si* $\Gamma \vdash \lambda x.P : A$, *où* A *est un type premier* $\neq *$, *alors* $A = B \to C$ *et* $\Gamma, x : B \vdash P : C$.

Preuve : La première fois que l'on a $\Gamma \vdash \lambda x.P : A'$ avec A facteur premier de A' dans la preuve de $\Gamma \vdash \lambda x.P : A$, on a appliqué une règle qui ne peut être que **abs**, d'où $A = A'$. □

Lemme 37 *Si* $\Gamma \vdash MN : A$, *où* A *est un type premier, alors* $\Gamma \vdash M : B \to A'$ *avec* A *facteur premier de* A' *et* $\Gamma \vdash N : B$.

Preuve : La première fois que l'on a $\Gamma \vdash MN : A'$ avec A facteur premier de A' dans la preuve de $\Gamma \vdash MN : A$, on a appliqué une règle qui ne peut être que **app**, d'où le résultat. □

Lemme 38 *1. Si* $x_1 : A_1, x_2 : A_2, \ldots, x_n : A_n \vdash M : A$ *alors*

$$x_1 : A_1 \wedge B, x_2 : A_2, \ldots, x_n : A_n \vdash M : A.$$

2. *Si* $x_1 : A_1, x_2 : A_2, \ldots, x_n : A_n \vdash M : A$ *alors*

$$x_1 : A_1 \wedge B_1, x_2 : A_2 \wedge B_2, \ldots, x_n : A_n \wedge B_n \vdash M : A.$$

3. *Si* $\Gamma \vdash y : A$ *et* $y \notin \{x_i\}_{1 \leq i \leq n}$ *alors*

$$\Gamma, x_1 : A_1, .., x_n : A_n \vdash y : A.$$

Preuve :

1. Par une induction sur la preuve du typage.
2. résulte directement du point précédent.
3. La dernière règle appliquée pour dériver $\Gamma \vdash y : A$ est soit **var**, et alors on dérive $\Gamma, x_1 : A_1, .., x_n : A_n \vdash y : A$ par cette même règle , soit $*_1$ et l'hypothèse faite sur y permet à nouveau d'appliquer $*_1$ pour obtenir le résultat. □

Proposition 10 *Soient* Γ *un contexte et* $x_1, .., x_n$ *des variables non déclarées dans* Γ. *Si* $\Gamma \vdash M[N_1/x_1, .., N_n/x_n] : A$ *et si les* N_i *sont typables dans* Γ *pour* $1 \leq i \leq n$, *alors il existe des types* A_i *tels que* $\Gamma \vdash N_i : A_i$ *et* $\Gamma, x_1 : A_1, .., x_n : A_n \vdash M : A$.

Preuve : D'abord par induction sur la taille de M, puis sur le type A.

Eliminons en premier lieu le cas où $A = B \wedge C$, alors $\Gamma \vdash M[N_1/x_1, .., N_n/x_n] : B$ et$\Gamma \vdash M[N_1/x_1, .., N_n/x_n] : C$ et par hypothèse d'induction il existe des types B_i et C_i tels que $\Gamma \vdash N_i : B_i$ et $\Gamma \vdash N_i : C_i$. Posons $A_i = B_i \wedge C_i$. Le point *2* du lemme 38 permet alors de voir que $\Gamma, x_1 : A_1, .., x_n : A_n \vdash M : B$ et $\Gamma, x_1 : A_1, .., x_n : A_n \vdash M : C$, il suffit dès lors d'appliquer la règle $\wedge$I.

Nous pouvons dès lors supposer que A est un type premier. Examinons les formes possibles de M:

Si $M = x_i$, alors $N_i = M[N_1/x_1, .., N_n/x_n]$ est de type A dans Γ et on peut donc prendre $A_i = A$ et $\Gamma, x_1 : A_1, .., x_n : A_n \vdash x_i : A$ résulte de la règle **var**.

Si $M = y \neq x_i$ ou c (constante), pour $1 \leq i \leq n$, alors on peut choisir A_i quelconque. On conclut par le lemme 38.

Si $M = \lambda y.P$, alors d'après le lemme 36, soit $A = *$, soit $A = B \to C$ et $\Gamma, y : B \vdash P[N_1/x_1, .., N_n/x_n] : C$.

Si $A = *$, on conclut par la règle $*_2$ dans le cas du λ_v-calcul.

Dans le cas du λ_{vh}-calcul, remarquons tout d'abord que la première fois que l'on rencontre $\Gamma \vdash M[N_1/x_1, .., N_n/x_n] : A'$ avec $*$ facteur premier de A' dans la preuve de $\Gamma \vdash M[N_1/x_1, .., N_n/x_n] : *$, on ne peut avoir appliqué que l'une des règles $*_3$ ou $*_4$, d'où $A' = *$ et $\Gamma \vdash M[N_1/x_1, .., N_n/x_n] : *$ résulte donc de l'une de ces règles . On peut éliminer le cas de la règle $*_4$ qui se résout en appliquant l'hypothèse d'induction à la prémisse de la règle, puis en réappliquant cette dernière.

Reste le cas où on a appliqué la règle $*_3$. Alors, en posant $y_1 = y$, $M[N_1/x_1, .., N_n/x_n]$ est de la forme $\lambda y_1..y_r.y_1 M_1..M_n$ et M est nécessairement aussi de cette même forme, on peut donc choisir les A_i quelconques et $\Gamma, x_1 : A_1, .., x_n : A_n \vdash M : *$ s'obtient par $*_3$.

Si $A \neq *$, par hypothèse d'induction il existe des types A_i tels que $\Gamma, y : B \vdash N_i : A_i$ et $\Gamma, y : B, x_1 : A_1, .., x_n : A_n \vdash P : C$, comme on peut supposer, par *alpha*-conversion, que y n'est pas une variable libre de N_i, la proposition 9 montre que $\Gamma \vdash N_i : A_i$. Il suffit maintenant de réappliquer la règle **abs** pour conclure.

Si $M = PQ$, d'après le lemme 36, $\Gamma \vdash Q[N_1/x_1, .., N_n/x_n] : B$ et $\Gamma \vdash P[N_1/x_1, .., N_n/x_n] : B \to A'$, où A' est un facteur premier de A. Par hypothèse d'induction il existe A'_i, A''_i tels que :

$\Gamma \vdash N_i : A'_i$ et $\Gamma, x_1 : A'_1, .., x_n : A'_n \vdash Q : B$

$\Gamma \vdash N_i : A''_i$ et $\Gamma, x_1 : A''_1, .., x_n : A''_n \vdash P : B \to A'$.

D'où en posant $A_i = A'_i \wedge A''_i$, le point *3* du lemme 38 montre que $\Gamma \vdash N_i : A_i$, $\Gamma, x_1 : A_1, .., x_n : A_n \vdash Q : B$ et que $\Gamma, x_1 : A_1, .., x_n : A_n \vdash P : B \to A'$, et il ne reste plus qu'à appliquer **abs** puis $\wedge$I. □

Corollaire 2 *Si* $\Gamma \vdash M[N/x] : A$ *et si* N *est typable dans* Γ, *alors* $\Gamma \vdash (\lambda x.M)N : A$.

Preuve : Appliquer la proposition précédente, puis **abs**. □

Corollaire 3 *Si* $\Gamma \vdash M[N/x]N_1 \cdots N_n : A$ *et si* N *et si les* N_i *sont typables dans* Γ, *alors*

$$\Gamma \vdash (\lambda x.M)NN_1 \cdots N_n : A.$$

Preuve : Par induction sur n. C'est le corollaire précédent si $n = 0$, sinon:

Si A est pas premier, alors le lemme 37 permet alors de déduire que

$$\Gamma \vdash (\lambda x.M)NN_1 \cdots N_{n-1} : B \to A' \text{ et } \Gamma \vdash N_n : B,$$

d'où par **app**:

$$\Gamma \vdash (\lambda x.M)NN_1 \cdots N_n : A',$$

ce dont on déduit aisément le résultat.

Si A n'est pas premier alors $A = A_1 \wedge \ldots \wedge A_r$, alors $\Gamma \vdash M[N/x]N_1 \cdots N_n : A_i$, pour chaque i, d'où $\Gamma \vdash (\lambda x.M)NN_1 \cdots N_n : A_i$ et il ne reste plus alors qu'appliquer la règle $\wedge$I autant de fois que nécessaire. □

Corollaire 4 *Si* $\Gamma \vdash \lambda x_1 \cdots x_m.M[N/x]N_1 \cdots N_n : A$ *et si* N *et les* N_i *sont typables dans* Γ, *alors*

$$\Gamma \vdash \lambda x_1 \cdots x_m.(\lambda x.M)NN_1 \cdots N_n : A.$$

Preuve : Par induction sur m: Si $m = 0$, c'est le corollaire précédent, sinon:

Si A est premier. Par le lemme 36 $A = B \to C$ et

$$\Gamma, x_1 : B \vdash \lambda x_2 \cdots x_m.M[N/x]N_1 \cdots N_n : C,$$

d'où, par hypothèse d'induction :

$$\Gamma, x_1 : B \vdash \lambda x_2 \cdots x_m.(\lambda x.M)N_1 \cdots N_n : C,$$

il ne reste plus qu' à appliquer **abs**.

Si A n'est pas premier, on raisonne comme dans le corollaire précédent. □

Théorème 10 *Si* M *est un terme réductible à une valeur alors* M *est typable dans le contexte vide.*

Preuve: Par la proposition 3, nous savons que M se réduit à gauche en une valeur et nous allons donc raisonner par induction sur la longueur de cette réduction et prouver de plus que si $M \to_g M'$ et $\Gamma \vdash M' : A$ alors $\Gamma \vdash M : A$:

Si celle-ci est nulle, alors $M \in \mathcal{V}$ et il suffit d'appliquer l'une des règles $*$.

Si cette longueur est non nulle:

Pour le λ_v-calcul, considérons le terme M' dans lequel M se réduit à gauche en une étape (M est alors de la forme $(\lambda x.P)NN_1...N_n$). Deux cas sont alors à distinguer:

Si N est une valeur alors $M' = P[N/x]N_1...N_n$. Par hypothèse d'induction M' est typable - ainsi que N et les N_i - de type $*$ dans le contexte vide. Il ne reste plus qu'à appliquer le corollaire 3.

Si N n'est pas une valeur, alors $N \rightarrow_g N'$ et $M' = (\lambda x.P)N'N_1..N_n$. Par hypothèse d'induction M' est typable de type A dans le contexte vide, de même que N' et les termes N_i. La proposition 10 montre qu'il existe alors un type B tel que $\vdash N' : B$ avec $y : B \vdash (\lambda x.P)yN_1..N_n : A$. L'hypothèse d'induction montre que $\vdash N : B$ et il ne reste plus qu'à appliquer la proposition 9.

Pour le λ_{vh}-calcul, le raisonnement est le même, si ce n'est qu'il faut utiliser cette fois le corollaire 4. □

Il en ressort que les termes réductibles à une valeur sont exactement les termes typables (ou encore de type $*$) dans le contexte vide.

4.8 Caractérisation des termes égaux à une valeur dans le λ_p-calcul

Le but de cette section est de montrer comment on peut caractériser les termes égaux à une valeur par l'intermédiaire du système de type $\mathcal{D}*$.

Definition 20 *A tout terme $M \in \Lambda_P$ on peut associer inductivement un terme $cod(M) \in \Lambda$ de la façon suivante:*

$$\begin{aligned} cod(x) &= x, \\ cod(c) &= c, \\ cod(PQ) &= cod(P)cod(Q), \\ cod(P\lceil Q) &= (\lambda z.cod(P))cod(Q), \text{ où } z \text{ est une nouvelle variable,} \\ cod(\lambda x.P) &= \lambda x.cod(P). \end{aligned}$$

Remarque 16 *On vérifie aisément que $\lambda_p \vdash M = cod(M)$.*

Lemme 39 *Soient $P, Q \in \Lambda_P$, alors $cod(P[Q/x]) = cod(P)[cod(Q)/x]$.*

Preuve : Par une induction immédiate sur P. □

Definition 21 *La relation binaire $\sim$ sur les termes de Λ^0 considérés modulo λ_v se définit comme suit: $M \sim N \Leftrightarrow \forall$ contexte clos $C[\,], C[M]\Downarrow \Longleftrightarrow C[N]\Downarrow$*

R.Pino.Pérez a montré dans [Pino 91] (voir aussi [Pino 92]) le résultat suivant qui s'avèrera fort utile pour la démonstration du lemme suivant:

$$M \sim N \Leftrightarrow (\forall P_1, \cdots, P_n \in \Lambda^0, \quad MP_1 \cdots P_n \Downarrow \Longleftrightarrow NP_1 \cdots P_n \Downarrow).$$

Remarque 17 *Si $M \sim N$ alors pour tout contexte clos $C[\,]$, $C[M] \sim C[N]$.*

Preuve: C'est évident, puisque si $D[\,]$ est un contexte clos, $D[C[\,]]$ en est un aussi, d'où $D[C[M]]\Downarrow \Longleftrightarrow D[C[N]]\Downarrow$. □

Lemme 40 *[Pino[2]]*
Soient M et $N \in \Lambda^0$.
Si $\lambda_p \vdash M = N$, où N est égal à une valeur, alors $cod(M) \sim cod(N)$.

Preuve: Prouvons ce résultat par induction sur la taille n de la preuve de $\lambda_p \vdash M = N$:
Si $n = 0$, c'est un axiome, soit l'un des cas suivants:

(ρ) et $M = N$ et c'est trivial.

(β_P) $\lambda_p \vdash M = (\lambda x.P)Q = P[Q/x]\lceil Q = N$: $cod((\lambda x.P)Q) = (\lambda x.cod(P))cod(Q)$, et donc, si $cod((\lambda x.P)Q)P_1 \cdots P_n \Downarrow$, alors $cod(Q) \Downarrow$, et par conséquent, en vertu du lemme 39 et par β_V-réduction, $\lambda_v \vdash cod((\lambda x.P)Q)P_1 \cdots P_n = cod(P)[cod(Q)/x]P_1 \cdots P_n = cod(P[Q/x]\lceil Q)P_1 \cdots P_n$, d'où $cod((\lambda x.P)Q) \sim cod(P[Q/x]\lceil Q)$.

Si on rajoute à λ_p la règle (η_P):

(η_P) $\lambda_p \vdash M = (\lambda x.Px)\lceil P = \lambda x.P = N$.

Puisque N est égal à une valeur, il s'ensuit que P l'est aussi, d'où:

$$\lambda_v \vdash cod((\lambda x.P)\lceil P) = (\lambda z.\lambda x.cod(P))cod(P) = \lambda x.cod(P)$$

.

Sinon, examinons la dernière règle appliquée :

La règle de symétrie (σ) et la règle de transitivité (τ) sont immédiates par hypothèse d'induction .

(μ) $\lambda_p \vdash P = Q \Longrightarrow \lambda_p \vdash M = PR = QR = N$: $[\ ]cod(R)$ est un contexte clos, et il suffit d'appliquer l'hypothèse d'induction et la remarque 17.

(ν) $\lambda_p \vdash P = Q \Longrightarrow \lambda_p \vdash M = PR = QR = N$: analogue au cas précédent.

(ξ) immédiat par hypothèse d'induction .

(γ) $\lambda_p \vdash P = Q \Longrightarrow \lambda_p \vdash M = P\lceil R = Q\lceil R = N$: $cod(P\lceil R) = (\lambda z.cod(P))cod(R)$, si $cod(P\lceil R)P_1 \cdots P_n \Downarrow$, alors $cod(R) \Downarrow$, d'où $cod(P\lceil R)P_1 \cdots P_n = cod(P)P_1 \cdots P_n$ et donc, puisque par hypothèse d'induction $cod(P) \sim cod(Q)$, on a $cod(Q)P_1 \cdots P_n \Downarrow$, d'où $cod(P\lceil R) \sim cod(Q\lceil R)$.

(δ) $\lambda_p \vdash P = Q \Longrightarrow \lambda_p \vdash M = R\lceil P = R\lceil Q = N$: $(\lambda x.cod(R))[\ ]$ est un contexte clos, et il suffit d'appliquer l'hypothèse d'induction et la remarque 17.

□

Théorème 11 *Soit $M \in \Lambda_P{}^0$,*

(M est égal à une valeur dans le λ_p-calcul.)

$\Leftrightarrow$

($cod(M)$ est égal (ou réductible) à une valeur dans le λ_v-calcul.)

[2]En fait, R.Pino Pérez a montré un résultat plus fort puisqu'il montre ce résultat pour tous M et $N \in \Lambda_P$.

($\Rightarrow$) Il est immédiat de s'assurer que $cod(N)$ est une valeur lorsque N est une valeur et donc $cod(N) \Downarrow$. Or, d'après le lemme 40, $cod(M) \sim cod(N)$. [ ] est un contexte clos, et donc on déduit par la definition de $\sim$ que $cod(M) \Downarrow$, et donc $cod(M)$ est réductible à une valeur dans le λ_v-calcul.

($\Leftarrow$) Si $cod(M)$ est réductible en une valeur dans le λ_v-calcul, alors M l'est dans le λ_p-calcul, puisque d'une part,toute égalité prouvable dans λ_v l'est aussi dans λ_p, toute valeur de λ_v étant une valeur de λ_p, et que d'autre part $\lambda_p \vdash M = cod(M)$. □

Corollaire 5 *Soit* $M \in {\Lambda_P}^0$,

(M est égal à une valeur dans le λ_p-calcul.)

$\Leftrightarrow$

($cod(M)$ est typable dans le système $\mathcal{D}*$)

Il est vraisemblable que le théorème 11 puisse se conserver si l'on remplace "M est égal à une valeur dans le λ_p-calcul" par "M est réductible à une valeur dans le λ_p-calcul". En effet la réduction de $cod(M)$ en une valeur dans λ_v semble pouvoir se relever en une réduction de M en une valeur dans λ_p, mais la démonstration doît être assez technique.

4.9 Caractérisation des termes clos (fortement) normalisables de λ_{pl}

Le but de cette partie est de montrer que dans λ_{pl}, un terme clos est normalisable si et seulement si il est fortement normalisable et que c'est équivalent au fait que $cod(M)$ soit typable dans le système $\mathcal{D}*$. Comme tout terme de Λ_{pl} est un terme de Λ_P, et que nous avons vu que cette propriété caractérise les termes égalisables à une valeur et que d'autre part la restriction de $T\lceil$ à Λ_{pl} n'est autre que T_{res}.

Nous allons tout d'abord introduire des variantes du λ_v-calcul, le λ_{vl}-calcul (le λ_v-calcul paresseux (ou lazy)) et le λ_{vl+}-calcul (dont l'intérêt est sa parenté avec le λ_{pl}-calcul), et montrer notamment que dans la première de ces deux variantes les termes clos normalisables et fortement normalisables coïncident, puis montrer que tout terme λ_v-fortement normalisable est λ_{vl+}-fortement normalisable, avant d'en déduire la propriété annoncée pour λ_{pl}.

le λ_{vl}-calcul est le λ_v-calcul moins la règle (ξ). Le λ_{vl}-calcul se définit ainsi par le système formel constitué par les règles suivantes:

(α) $(\lambda x.M) = (\lambda y.M[y/x])$, si $y \notin VL(M)$.

(β_{vl}) $(\lambda x.M)N = M[N/x]$, si N est une valeur.

$$(\mu)\ \frac{M = N}{MP = NP} \qquad (\rho)\ M = M \qquad (\nu)\ \frac{M = N}{PM = PN}$$

$$(\sigma)\ \frac{M = N}{N = M} \qquad (\tau)\ \frac{M = N \quad N = P}{M = P}$$

On notera $\lambda_{vl} \vdash M = N$ si la formule $M = N$ se dérive dans le système précédent; $\lambda_{vh} \vdash M \rightarrow_{vl} N$, si elle se dérive sans utiliser les règles σ et τ, et $\lambda_{vh} \vdash M \stackrel{*}{\rightarrow}_{vl} N$ si elle se dérive sans utiliser la règle σ.

De même qu'au chapitre 2, nous définirons un contexte factoriel (un contexte factoriel étant toujours un terme avec une nouvelle variable qu'on représente par un trou ([ ]) et où aucune occurence de cette variable-trou ne se trouve sous la portée d'un λ) inductivement comme suit:

Contexte factoriel = [ ],
M, où M est un terme quelconque,
CC', où C et C' sont des contextes factoriels.

Rappelons qu'un contexte factoriel sera dit *propre* s'il y a au moins une occurence de [ ], *simple* s'il y en a une et une seule, *multiple* sinon. Un terme u sera dit factoriel en x si $u[[\,]/x]$ est factoriel; en particulier ce sera le cas si $x \notin VL(u)$.

Une variante du λ_{vl}-calcul qui s'avère très proche de λ_{pl} et que nous baptisons λ_{vl+} est celui où l'on remplace la règle (β_{vl}) par la règle (β_{vl+}): $(\lambda x.M)N = M[N/x]$, si N est une valeur, *ou*[3] *si M est un contexte factoriel en x.*

On définit de façon analogue les notations $\lambda_{vl+} \vdash M = N$, $\rightarrow_{vl+}$ et $\stackrel{*}{\rightarrow}_{vl+}$.

Il est également à noter que nous aurions pu tout aussi bien définir $\rightarrow_{vl}$ (resp. $\rightarrow_{vl+}$) comme la plus petite relation passant au contexte factoriel contenant β_{vl} (resp. β_{vl+}).

Remarque 18 *Il est à noter que la réduction gauche du λ_v-calcul coïncide avec celle du λ_{vl}-calcul, puisque l'on ne pratique aucune réduction sous un λ. Les termes clos et normaux pour (β_{vl}) coïncident avec ceux qui le sont pour $\stackrel{*}{\rightarrow}_g$, et les termes réductibles en une valeur dans λ_v coïncident avec ceux qui le sont dans λ_{vl} par le théoreme de standardisation.*

La réduction gauche de λ_{vl+}, par contre, diffère de $\rightarrow_g$ puisque les β_v-rédex et β_{vl+} ne coïncident plus.

Théorème 12 *Les termes clos normalisables sont fortement normalisables dans λ_{vl}. Plus précisément, si M est normalisable en une valeur V en n étapes alors toutes les séquences de réductions en n étapes conduisent à la même valeur V.*

Preuve: La remarque 18 nous montre que nous pouvons raisonner par induction sur la longueur de la réduction gauche de M.

Si M est normal alors il est une abstraction ou une constante et le résultat va de soi.

Sinon M est la forme $(\lambda x.U)V_1...V_n$. Il faut distinguer deux cas selon que V_1 est une valeur ou non:

- si V_1 est une valeur, alors $M \rightarrow_g M' = U[V_1/x]V_2...V_n$. Il va s'agir premièrement de montrer que toute autre réduction diminue la longueur de la rlocal.lifl.systeméduction gauche d'une unité.

  Supposons que nous ayons réduit V_i en V_i' dans M (V_1 étant une valeur on a $i \geq 2$), soit $M \rightarrow_{vl} M'' = (\lambda x.U)V_1..V_i'..V_n$. Alors la réduction gauche de M' consiste d'abord à réduire $U[V_1/x]V_2..V_{i-1}$ en une valeur V (en p étapes), tandis que celle de M'' consiste d'abord à réduire $(\lambda x.U)V_1..V_{i-1}$ en $U[V_1/x]V_2..V_{i-1}$ puis de réduire ce terme en V, donc en $p+1$ étapes. Ensuite dans la réduction de M', il s'agit de réduire V_i en une valeur W_i, tandis que celle de M'' il s'agit de réduire V_i' en une valeur W_i'. Mais par hypothèse d'induction comme la longueur de réduction gauche de V_i est strictement

[3] Il est à noter que nous aurions pu également considérer un calcul où la règle de de réduction aurait été la règle (β_{vl++}): $(\lambda x.M)N_1...N_n = M[N_1/x_1,..,N_2/x_2]$, si pour tout i, N_i est une valeur ou bien M est un contexte factoriel en x_i. Mais celui-ci se serait apparenté à un autre λ_{pl}-calcul que celui que nous avons considéré, et que le lecteur pourra d'ailleurs facilement extrapoler, et qui aurait été d'ailleurs plus expressif que celui que nous considérons, en ce qu'il permettrait d'identifier plus de termes indéfinis et apparaitrait intermédaire entre le λ_{pl}-calcul et le λ_p-calcul. Mais tous les résulats que nous obtenons ne devraient présenter aucune difficulté particulière à se relever à cet autre calcul. Ce calcul pourrait faire l'objet d'une étude ultérieure au présent travail.

plus courte que celle de M, il s'avère que $W_i = W_i'$ et que si la réduction de V_i' en W_i se fait en q étapes, celle V_i en W_i' se fait en $q+1$ étapes.

Il apparaît ainsi, qu'après $p+q+1$ étapes, les séquences de réduction gauche de M' et de M'' coïncident et conduisent donc à la même valeur.

- si V_1 n'est pas une valeur, la réduction gauche de M consiste à réduire d'abord V_1 et le raisonnement est tout à fait analogue. □

Dans tout ce qui suit nous ne considérerons plus que des termes clos.

L'étape que nous voulons maintenant franchir est de montrer que les termes (fortement) normalisables pour λ_{vl} le sont aussi pour λ_{vl+}. La difficulté est de prouver que l'on ne peut pas faire une infinité de réductions $\rightarrow_{vl+}$ dans un terme λ_{vl}-(fortement) normalisable, alors qu'il est vrai cependant qu'une réduction $\rightarrow_{vl+}$ donne certes un terme qui est à nouveau λ_{vl}-(fortement) normalisable, mais dont la longueur des chemins de λ_{vl}-normalisation est plus grande que celle du terme dont nous sommes partis. Nous allons tout de même pouvoir définir une mesure qui garantit la finitude du processus.

Soit un terme M normalisable. Il existe donc une unique valeur (close: c'est-à-dire une abstraction) telle que $M \stackrel{*}{\rightarrow}_{vl} V$.

Nous allons introduire un système de preuves dans lequel on déduit des formules de la forme $M \overset{\lambda_{vl}}{\vdash} V$, où V est une valeur, et nous montrerons en particulier que $M \overset{\lambda_{vl}}{\vdash} V$ implique $M \stackrel{*}{\rightarrow}_{vl} V$ (en fait, il est possible dre montrer que c'est équivalent).

Nous allons proposer un système comprenant essentiellement deux règles et montrer qu'en fait une seule de ces deux règles peut toujours être préférée à l'autre au cas où les deux peuvent s'appliquer, ce qui présentera l'agrément de nous fournir une preuve unique, standard, du fait que le terme M est réductible à une valeur V. Nous montrerons enfin que la réduction $\rightarrow_{vl+}$ fait décroître la taille de cette preuve ce qui garantira le résultat annoncé.

Voici le système de preuves:

$$(axiome)\ \frac{V \text{ est une abstraction (valeur close)}}{V \overset{\lambda_{vl}}{\vdash} V}$$

$$(1)\ \frac{M \overset{\lambda_{vl}}{\vdash} \lambda x.u \qquad N \overset{\lambda_{vl}}{\vdash} V \qquad u[V/x] \overset{\lambda_{vl}}{\vdash} W}{MN \overset{\lambda_{vl}}{\vdash} W}$$

$$(2)\ \frac{M \overset{\lambda_{vl}}{\vdash} \lambda x.u \qquad N \overset{\lambda_{vl}}{\vdash} V \qquad u \text{ factoriel en } x \qquad u[N/x] \overset{\lambda_{vl}}{\vdash} W}{MN \overset{\lambda_{vl}}{\vdash} W}$$

On notera $M \overset{\lambda_{vl}1}{\vdash} V$ si on n'a pas utilisé la règle 2 et $M \overset{\lambda_{vl}2}{\vdash} V$ *si on n'a pas utilisé la règle 1*:

$$(1)\ \frac{M \overset{\lambda_{vl}}{\vdash} \lambda x.u \qquad N \overset{\lambda_{vl}}{\vdash} V \qquad u[V/x] \overset{\lambda_{vl}}{\vdash} W}{MN \overset{\lambda_{vl}}{\vdash} W}$$

avec u un contexte factoriel en x. Nous verrons alors ultérieurement que dans ce cas on peut toujours utiliser la règle 2.

Lemme 41 $M \stackrel{*}{\rightarrow}_{vl} V \iff M \overset{\lambda_{vl}1}{\vdash} V$.

Preuve:

Prouvons $M \xrightarrow{*}_{vl} V \Longrightarrow M \overset{\lambda_{vl}1}{\vdash} V$ par induction sur la longueur de la réduction $M \xrightarrow{*}_{vl} V$.

Si M est λ_{vl}-normalisable alors soit M est une valeur, d'où $M = V$, soit M est une application et par conséquent $M = NP$, or cette application doit pouvoir s'éliminer, d'où $N \xrightarrow{*}_{vl} V_N = \lambda x.Q$ et $P \xrightarrow{*}_{vl} V_P$ avec $Q[V_P/x] \xrightarrow{*}_{vl} V$, qui sont trois réduction strictement plus courte que $M \xrightarrow{*}_{vl} V$. Donc par hypothèse d'induction $N \overset{\lambda_{vl}}{\vdash} V_N = \lambda x.Q$ et $P \overset{\lambda_{vl}}{\vdash} V_P$ et $Q[V_P/x] \overset{\lambda_{vl}}{\vdash} V$ et nous pouvons appliquer la règle 1.

Dans l'autre sens, raisonnons par induction sur la preuve de $M \overset{\lambda_{vl}}{\vdash} V$. L'axiome ne pose aucun problème.

Si on a appliqué la règle 1 alors par hypothèse d'induction $M \xrightarrow{*}_{vl} \lambda x.u$, $N \xrightarrow{*}_{vl} V$ et $u[V/x] \xrightarrow{*}_{vl} W$; d'où $MN \xrightarrow{*}_{vl} (\lambda x.u)V \xrightarrow{*}_{vl} u[V/x] \xrightarrow{*}_{vl} W$, puisque $\xrightarrow{*}_{vl}$ passe au contexte factoriel. □

L'idée est maintenant de transformer toute preuve de $M \overset{\lambda_{vl}}{\vdash} V$ en une preuve de $M \overset{\lambda_{vl}2}{\vdash} V$, c'est-à-dire en une preuve où l'on n'a pas utilisé la règle 1 lorsqu'on aurait pu utiliser la règle 2 (et nous verrons que c'est toujours le cas si u était factoriel en x dans l'utilation de la règle 1). Ce processus tend à accroître la taille de la preuve, et tout le problème serait de voir que cet accroissement est borné si l'on voulait définir un système de réécriture des preuves, dont il s'agirait alors de prouver la confluence et la nœthérianité. Mais il s'avère que l'on peut directement définir la forme normale par induction sur la preuve et obtenir ainsi une preuve standard. C'est l'objet des deux lemmes suivants:

Lemme 42 *Soit C un contexte factoriel, un terme N et deux valeurs V et W tels que $C[V] \overset{\lambda_{vl}2}{\vdash} W$ et $N \overset{\lambda_{vl}2}{\vdash} V$, alors il est possible de dériver $C[N] \overset{\lambda_{vl}2}{\vdash} W$.*

Preuve:

Nous allons prouver ce fait par induction sur la preuve de $C[V] \overset{\lambda_{vl}2}{\vdash} W$ en distinguant les cas selon la structure de C:

- si $C = [\ ]$ alors $C[V] = V$ d'où $V = W$ et dans ce cas $C[N] \overset{\lambda_{vl}2}{\vdash} W$ n'est autre que $N \overset{\lambda_{vl}2}{\vdash} V$.
- si $C = M\ ([\ ] \notin VL(C))$ alors $C[N] = C[V]$ et $C[N] \overset{\lambda_{vl}2}{\vdash} W$ n'est autre que $C[V] \overset{\lambda_{vl}2}{\vdash} W$.
- si $C = C_1C_2$, considérons la dernière règle appliquée pour dériver $C[V] \overset{\lambda_{vl}2}{\vdash} W$.

  si c'est la règle 1:

$$(1)\ \frac{C_1[V] \overset{\lambda_{vl}2}{\vdash} \lambda x.u \qquad C_2[V] \overset{\lambda_{vl}2}{\vdash} V' \qquad u[V'/x] \overset{\lambda_{vl}2}{\vdash} W}{C[V] \overset{\lambda_{vl}2}{\vdash} W}$$

  où u n'est pas factoriel en x. Alors par hypothèse d'induction on dispose de $C_1[N] \overset{\lambda_{vl}2}{\vdash} \lambda x.u$ et de $C_2[N] \overset{\lambda_{vl}2}{\vdash} V'$, ce qui fournit en réappliquant la règle 1:

$$(1)\ \frac{C_1[N] \overset{\lambda_{vl}2}{\vdash} \lambda x.u \qquad C_2[N] \overset{\lambda_{vl}2}{\vdash} V' \qquad u[V'/x] \overset{\lambda_{vl}2}{\vdash} W}{C[N] \overset{\lambda_{vl}2}{\vdash} W}.$$

  si c'est la règle 2:

$$(2)\ \frac{C_1[V] \overset{\lambda_{vl}2}{\vdash} \lambda x.u \quad C_2[V] \overset{\lambda_{vl}2}{\vdash} V' \ u \text{ factoriel en } x \quad u[C_2[V]/x] \overset{\lambda_{vl}2}{\vdash} W}{C[V] \overset{\lambda_{vl}2}{\vdash} W}$$

Cette fois on dispose par induction de $C_1[N] \overset{\lambda_{vl}2}{\vdash} \lambda x.u$ et $C_2[N] \overset{\lambda_{vl}2}{\vdash} V'$, mais aussi de $u[C_2[N]/x] \overset{\lambda_{vl}2}{\vdash} W$ puisque $u[C_2[\ ]/x]$ est un contexte factoriel (comme il se voit aisément) et la preuve de $u[C_2[V]/x] \overset{\lambda_{vl}2}{\vdash} W$ est plus courte que celle de $C[V] \overset{\lambda_{vl}2}{\vdash} W$. Il suffit dès lors de réappliquer la règle 2:

$$(2)\ \frac{C_1[N] \overset{\lambda_{vl}2}{\vdash} \lambda x.u \quad C_2[N] \overset{\lambda_{vl}2}{\vdash} V' \ u \text{ factoriel en } x \quad u[C_2[N]/x] \overset{\lambda_{vl}2}{\vdash} W}{C[N] \overset{\lambda_{vl}2}{\vdash} W}$$

□

Lemme 43 *Si* $M \overset{\lambda_{vl}}{\vdash} V$ *alors* $M \overset{\lambda_{vl}2}{\vdash} V$.

Preuve: Le seul cas intéressant est évidemment celui où la dernière règle appliquée est la règle 1:

$$(1)\ \frac{M \overset{\lambda_{vl}}{\vdash} \lambda x.u \qquad N \overset{\lambda_{vl}}{\vdash} V \qquad u[V/x] \overset{\lambda_{vl}}{\vdash} W}{MN \overset{\lambda_{vl}}{\vdash} W}$$

avec u un contexte factoriel.

Alors par hypothèse d'induction on sait que $M \overset{\lambda_{vl}2}{\vdash} \lambda x.u$, $N \overset{\lambda_{vl}2}{\vdash} V$ et que $u[V/x] \overset{\lambda_{vl}2}{\vdash} W$. Or l'hypothèse d'induction sur 1 nous permet d'en déduire $u[N/x] \overset{\lambda_{vl}2}{\vdash} W$. Il suffit alors d'appliquer la règle 2. □

Il se voit aisément que la preuve de $M \overset{\lambda_{vl}2}{\vdash} V$ est unique. Ainsi chaque terme λ_{vl}-normalisable se voit asssocier une unique preuve de $M \overset{\lambda_{vl}2}{\vdash} V$.

Lemme 44 *Les termes* λ_v*-normalisables sont fortement normalisables dans* λ_{vl+}.

Preuve: Nous allons de fait prouver que si M est un terme λ_{vl}-normalisable et que $M \to_{vl+} M'$, alors on a $M = C[(\lambda x.u)N]$ et $M' = C[u[N/x]]$, avec N étant une valeur ou u un contexte factoriel en x et $C[\ ]$ un contexte factoriel simple.

Si N est une valeur, alors on a $M \to_{vl} M'$ et si u est factoriel en x alors comme N est un facteur de M on a nécessairement $N \overset{*}{\to}_{vl} V$ où V est une valeur et ainsi $M' \overset{*}{\to}_{vl} C[u[V/x]]$. De même $M \overset{*}{\to}_{vl} C[(\lambda x.u)V] \to_{vl} C[u[V/x]]$. Le théoreme 12 montre que si $M \overset{*}{\to}_{vl} W$ alors $M' \overset{*}{\to}_{vl} W$. Le lemme 41 montre que $M \overset{\lambda_{vl}}{\vdash} W$ et $M' \overset{\lambda_{vl}}{\vdash} W$ et le lemme 43 que $M \overset{\lambda_{vl}2}{\vdash} W$ et $M' \overset{\lambda_{vl}2}{\vdash} W$.

Nous allons prouver que la preuve de $M' \overset{\lambda_{vl}2}{\vdash} W$ est plus courte que celle de $M \overset{\lambda_{vl}2}{\vdash} W$.

Nous allons en fait prouver que *pour tout contexte factoriel propre* $C[\ ]$, la preuve de $C[(\lambda x.u)N] \overset{\lambda_{vl}2}{\vdash} W$ (si elle existe) est plus longue que celle de $C[u[N/x] \overset{\lambda_{vl}2}{\vdash} W$ (qui alors existe), d'abord par induction sur la preuve de $C[(\lambda x.u)N] \overset{\lambda_{vl}2}{\vdash} W$ et selon le contexte $C[\ ]$.

- Si $C = [\ ]$ alors $M = (\lambda x.u)N$,

  osi la preuve de $M \overset{\lambda_{vl}2}{\vdash} W$ se termine par le règle 1, alors u n'est pas factoriel en x et $N = V$, on a donc:

$$(1)\ \frac{\lambda x.u \overset{\lambda_{vl}2}{\vdash} \lambda x.u \qquad V \overset{\lambda_{vl}2}{\vdash} V \qquad u[V/x] \overset{\lambda_{vl}2}{\vdash} W}{(\lambda x.u)V \overset{\lambda_{vl}}{\vdash} W}$$

et ainsi la preuve de $M' = u[V/x] \overset{\lambda_{vl}2}{\vdash} W$ est plus courte que celle de $M \overset{\lambda_{vl}2}{\vdash} W$.

osi la preuve de $M \overset{\lambda_{vl}2}{\vdash} W$ se termine par le règle 2, alors u est pas factoriel en x on a donc:

$$(2)\ \frac{\lambda x.u \overset{\lambda_{vl}2}{\vdash} \lambda x.u \quad N \overset{\lambda_{vl}2}{\vdash} V \quad u \text{ factoriel en } x \quad u[N/x] \overset{\lambda_{vl}2}{\vdash} W}{(\lambda x.u)N \overset{\lambda_{vl}2}{\vdash} W}$$

et la preuve de $M' = u[N/x] \overset{\lambda_{vl}2}{\vdash} W$ est donc encore cette fois plus courte.

- Le dernier cas $C = C_1C_2$ se résout simplement par induction. Nous laissons au lecteur le soin de cette vérification[4]. □

Remarque 19 *Si $M \in \Lambda$ est un terme clos λ_{pl}*-normalisable, alors $\lambda_{pl} \vdash \mathrm{M} = \mathrm{V}$, d'où $\lambda_p \vdash M = V$ et $cod(M)$ est un terme clos typable dans le système $\mathcal{D}*$ (dans le contexte vide) et est donc λ_v-réductible à une valeur, soit encore λ_{vl}-normalisable et par suite λ_{vl+}-fortement normalisable. Nous allons montrer que M est alors λ_{pl}-fortement normalisable.

Appelons pour les distinguer $\nu_{pl}(M)$ et $\nu_+(M)$ les longueurs respectives éventuellement infinies des plus longs chemins de réduction de M par les réductions $\rightarrow_{\beta_{pl}}$ et $\rightarrow_{vl+}$.

Lemme 45 *Si M est un terme pur λ_{vl}-(fortement) normalisable ($M \in \Lambda$), alors M est λ_{pl}*-fortement normalisable et $\nu_{pl}(M) \leq 2^{\nu_+(M)}\nu_+(M)$.

Preuve: Nous allons le prouver par induction sur la longueur du chemin de normalisation de M par $\rightarrow_{vl+}$. Si celle-ci est 0, alors M est une abstraction et le résultat est évident.

Sinon, Il se vérifie aisément que, si $M \rightarrow_{\beta_{pl}} M'$, alors $M \rightarrow_{vl+} \mathsf{h}(M')$ et $\mathsf{r}(M') = \{N\}$ où N un facteur propre de M, d'où $\nu_+(\mathsf{h}(M')) \leq \nu_+(M) - 1$ et $\nu_+(N) \leq \nu_+(M) - 1$. Or par le lemme 10 $\nu_{pl}(M') = \nu_{pl}(\mathsf{h}(M')) + \nu_{pl}(\mathsf{r}(M'))$ et par hypothèse d'induction on a $\nu_{pl}(\mathsf{h}(M')) \leq 2^{\nu_+(M)-1}(\nu_+(M) - 1)$ et $\nu_{pl}(\mathsf{r}(M')) \leq 2^{\nu_+(M)-1}(\nu_+(M) - 1)$, d'où $\nu_{pl}(M')2^{\nu_+(M)}\nu_+(M)$. □

Lemme 46 *Soit M un terme clos de Λ_{pl}, alors il existe $M' \in \Lambda_{pl}$ tel que $cod(M) \overset{*}{\rightarrow}_{\beta_{pl}} M'$ et $\mathsf{h}(M') = \mathsf{h}(M)$, $\nu_{pl}(\mathsf{r}(M')) = \nu_{pl}(\mathsf{r}(M))$. En particulier le lemme 10 montre que $\nu_{pl}(cod(M)) \geq \nu_{pl}(M)$*

Preuve:

- si $M = PQ$ alors $cod(M) = cod(P)cod(Q) \overset{*}{\rightarrow}_{\beta_{pl}} P'Q'$, avec $\mathsf{h}(P') = \mathsf{h}(P)$, $\nu_{pl}(\mathsf{r}(P')) = \nu_{pl}(\mathsf{r}(P))$, $\mathsf{h}(Q') = \mathsf{h}(Q)$, $\nu_{pl}(\mathsf{r}(Q')) = \nu_{pl}(\mathsf{r}(Q))$.
  Alors il se vérifie immédiatement que $M' = P'Q'$ convient.

- $M = P\lceil Q$, alors
$$cod(M) = (\lambda z.cod(P))cod(Q) \rightarrow_{\beta_{pl}} cod(P)\lceil cod(Q) \overset{*}{\rightarrow}_{\beta_{pl}} P'\lceil Q' = M'$$
  et ce M' convient.

[4]Notons que c'est ici que l'on pourra voir l'intérêt de garder trace dans la règle 2 de $N \overset{\lambda_{vl}}{\vdash} V'$, où N est l'argument dans le terme. En effet si u est un contexte où ne figure pas x, la taille de la preuve n'aurait alors pas forcément diminuée.

- $M = \lambda x.P$ Ce cas est évident puisqu'alors $M = cod(M)$. □

Corollaire 6 *Les termes clos* $M \in \Lambda_{pl}$ λ_{pl}-normalisables et λ_{pl}-(fortement) normalisables coïncident avec les termes clos $M \in \Lambda_{pl}$ tels que $cod(M)$ soit typable dans le système $\mathcal{D}*$. Comme ceci caractérise les termes égalisables en une valeur, il s'ensuit que pour un terme clos de Λ_{pl} les notions de réductible à une valeur, de normalisable et de fortement normalisable coïncident.

Si $cod(M)$ est typable dans le système $\mathcal{D}*$ alors par le lemme 45 $cod(M)$ est λ_{pl}-fortement normalisable et donc par le lemme 46 M est lui-même fortement normalisables.

Réciproquement, si M est un terme clos, nous avons vu par la remarque 19 que $cod(M)$ est typable dans le système $\mathcal{D}*$. □

5

Algèbres combinatoires avec éléments partiels

Le but de cette partie est de résoudre la conjecture de Moggi en caractérisant par un nombre fini d'équations les algèbres combinatoires qui correspondent au λ_p-calcul. La technique est inspirée de celle du résultat homologue dans le cas total que nous devions à Curry (*cf* [Barendregt]). Les résultats de cette section ont été en partie exposés à la 12^{eme} Conférence internationale de la Société Chilienne d'Informatique Théorique à Santiago du Chili (*cf* [Even& Pino 92]) et complétés depuis. Les algèbres combinatoires avec éléments partiels trouvent leur cadre dans une logique partielle qui offre certaines complications et nous préférons présenter cette logique sous forme d'un système formel. Nous verrons d'ailleurs ultérieurement qu'à modifier très légèrement le λ_p-calcul tous les résultats que nous sont conservés mais avec une notion d'algèbre combinatoire qui se laisse définir dans un cadre classique. La technique suit celle de Curry pour le cas total. Nous définissons les λ_p-algèbres et montrons que celles-ci peuvent s'axiomatiser par un nombre fini d'équations. Nous formulons enfin un théorème d'équivalence entre la logique des algèbres combinatoires avec éléments partiels et le λ_p-calcul.

Nous considérons le lambda calcul partiel $\Lambda(V, C)$ construit sur V et C.

5.1 Lien entre le λ_p-calcul et les algèbres combinatoires

Definition 22 *Considérons les termes du premier ordre construits sur la signature* $(\cdot, S, K, I)$ *où* $\cdot$ *est un symbole de fonction binaire. Soit* $\Downarrow$ *un symbole de prédicat unaire.*

$u\lceil v$ *dénotera* Kuv.

Nous appellerons Logique combinatoire avec éléments partiels *(en abrégé* LCEP *), l'ensemble des formules atomiques dérivables dans le système*[1] *suivant:*

t, u, v *dénotent des termes,* x *dénote une variable.*

Schémas d'axiomes:

$$x \Downarrow \qquad S \Downarrow \qquad K \Downarrow \qquad I \Downarrow \qquad Stuv = tv(uv) \qquad Iu = u \qquad t\lceil t = t$$

$$(t\lceil u)\lceil v = (t\lceil v)\lceil u \qquad t\lceil(u\lceil v) = (t\lceil u)\lceil v \qquad t(u\lceil v) = tu\lceil v \qquad (t\lceil v)u = tu\lceil v \qquad u = u$$

[1] Une autre solution serait de considérer une logique partielle dans l'esprit de celles introduites par Scott et Beeson (*cf* [Scott79, Beeson81, Beeson85]). Celle de Scott semblerait plus proche de celle qu'il nous faudrait formuler, par le fait que la quantification est bornée au termes totaux. Néanmoins aucune de ces deux approches ne convient ici, parce qu'elles identifient tous les termes indéfinis. Cette approche reste à formuler. Quoiqu'il en soit, il est ici plus pratique de raisonner sur un système formel. Nous montrerons ultérieurement comment on peur se ramener à une logique classique et obtenir des résultats similaires.

Schémas de règles d'inférence:

$$\frac{t\Downarrow}{St\Downarrow} \qquad \frac{t\Downarrow}{Kt\Downarrow} \qquad \frac{t\Downarrow \quad u\Downarrow}{Stu\Downarrow} \qquad \frac{tu\Downarrow}{u\Downarrow} \qquad \frac{tu\Downarrow}{t\Downarrow} \qquad \frac{t=u \quad t\Downarrow}{u\Downarrow}$$

$$\frac{u\Downarrow}{t\lceil u=t} \qquad \frac{t=u \quad v=w}{tv=uw} \qquad \frac{t=u \quad u=v}{t=v} \qquad \frac{t=u}{u=t}$$

Soit $\mathcal{A}=<A,S,K,I,\cdot,\Downarrow>$.

Nous dirons que $\mathcal{A}\models\Phi(\vec{x})$, si pour toute valuation $\rho: V\to A$, telle que $\forall x\in V, \rho(x)\Downarrow$, $\mathcal{A}\models\Phi(\vec{\rho})$.

Un modèle de LCEP sera dit *une algèbre combinatoire avec éléments partiels* .

Il est à noter tout d'abord qu'on n'a pas, pour tout élément a de $\mathcal{A}$, $a\Downarrow$.

Remarque 20 *Si $t=u$ est dérivable dans ce système alors il ne présente aucune difficulté de prouver, par induction sur la preuve de $t=u$, que pour tout terme v tel que $v\Downarrow$, l'équation $t[v/x]=u[v/x]$ se dérive également dans ce système.*

On peut remarquer par ailleurs que le prédicat $\Downarrow$ peut se définir à partir de l'égalité par $u\Downarrow \Longleftrightarrow I\lceil u=I$, comme il se voit aisément.

Definition 23 *Soit $\mathcal{T}$ l'ensemble des termes construit sur la signature $<\cdot,S,K,I>$ à partir d'un ensemble de variables V et d'un ensemble de constantes totales C (i.e. $c\in C\Longrightarrow c\Downarrow$).*

Pour chaque variable x, nous définissons une application $[x]:\mathcal{T}\to\mathcal{T}$ comme suit:

$$\begin{aligned}
&[x]x = I,\\
&[x]c = Kc,\\
&[x]y = Ky,\ si\ y\neq x,\\
&[x]uv = S([x]u)([x]v).
\end{aligned}$$

Nous définissons deux applications $()_{CL}:\Lambda_P\to\mathcal{T}$ et $()_\lambda:\mathcal{T}\to\Lambda_P$ comme suit:

$$\begin{array}{ll}
x_{CL}=x & x_\lambda=x\\
c_{CL}=c & c_\lambda=c\\
MN_{CL}=M_{CL}N_{CL} & (uv)_\lambda=u_\lambda v_\lambda\\
(M\lceil N)_{CL}=M_{CL}\lceil N_{CL} & S_\lambda=\lambda xyz.xz(yz)\\
\lambda x.N_{CL}=[x]N_{CL} & K_\lambda=\lambda xy.x\\
 & I_\lambda=\lambda x.x
\end{array}$$

Remarque 21 *Il est facile de constater que ces traductions conservent les variables libres. L'ensemble des variables d'un terme $M\in\Lambda_P$ (resp. $u\in\mathcal{T}$) sera noté $VL(M)$ (resp. $VL(u)$).*

Notation 1 *Soient $M,N\in\lambda_p$ alors on écrira* LCEP $\vdash M=N$ *pour* LCEP $\vdash M_{CL}=N_{CL}$.

Lemme 47

$$\forall u\in\mathcal{T} \qquad \lambda_p\vdash([x]u)_\lambda=\lambda x.u_\lambda.$$

Preuve : Par induction sur u:

$u=x, ([x]x)_\lambda=I_\lambda=\lambda x.x$

$u = c$, alors $([x]c)_\lambda = (Kc)_\lambda = K_\lambda c$, or $\lambda_p \vdash K_\lambda c = (\lambda x.c)\lceil c = \lambda x.c$, puisque les constantes sont totales.

$u = S, K, I, y \neq x$, similaire au cas précédent.

$u = vw$,

$$\begin{array}{rcl} ([x]vw)_\lambda & = & S_\lambda([x]v)_\lambda([x]w)_\lambda \\ & \overset{induction}{=} & \lambda x.((\lambda x.v_\lambda)x)(\lambda x.w_\lambda)x \\ & = & \lambda x.(v_\lambda w_\lambda) = \lambda x.u_\lambda \end{array}$$ □

Lemme 48 *On a, pour tout* $M \in \lambda_p$: $\lambda_p \vdash M_{CL,\lambda} = M$.

Preuve : Par induction sur M:

$M = x, c$, évident.

$M = NP$, immédiat par induction.

$N\lceil P$, en utilisant l'hypothèse d'induction et les règles de la restriction, on a $(N\lceil P)_{CL,\lambda} = K_\lambda N_{CL,\lambda} P_{CL,\lambda} = K_\lambda NP = N\lceil P$.

$M = \lambda x.N$: alors $(\lambda x.N)_{CL,\lambda} = ([x]N_{CL})_\lambda$, et par le lemme précédent:

$\lambda_p \vdash ([x]N_{CL})_\lambda = \lambda x.N_{CL,\lambda}$, et il suffit dès lors d'appliquer l'hypothèse d'induction et la régle ξ. □

Definition 24 *λ_p-algèbre*

Une algèbre combinatoire avec éléments partiels $\mathcal{A}$ sera dite une λ_p-algèbre *si on a la propriété suivante:*

$$\forall u, v \in \mathcal{T} \qquad \lambda_p \vdash u_\lambda = v_\lambda \Longrightarrow \mathcal{A} \models u = v.$$

Lemme 49 *Une algèbre combinatoire avec éléments partiels $\mathcal{A}$ est une λ_p-algèbre si et seulement si les deux conditions suivantes sont vérifiées:*

1. $\mathcal{A} \models S_{\lambda,CL} = S$ *et* $K_{\lambda,CL} = K$.
2. $\lambda_p \vdash M = N \Longrightarrow \mathcal{A} \models M_{CL} = N_{CL}$

Preuve :

($\Rightarrow 1$) Remarquons que le lemme 48 dit en particulier que pour tout $u \in \mathcal{T}$ LCEP $\vdash u_{\lambda,CL,\lambda} = u_\lambda$. $\mathcal{A}$ étant une λ_p-algèbre, il s'ensuit que $\mathcal{A} \models u_{\lambda,CL} = u$.

($\Rightarrow 2$) Si $\lambda_p \vdash M = N$, alors, par le lemme 48 et par transivité, $\lambda_p \vdash M_{CL,\lambda} = N_{CL,\lambda}$. Ainsi, puisque $\mathcal{A}$ est une λ_p-algèbre $\mathcal{A} \models M_{CL} = N_{CL}$.

($\Leftarrow$) Il est aisé de montrer par induction sur $u \in \mathcal{T}$ que si $\mathcal{A} \models S_{\lambda,CL} = S$ et $K_{\lambda,CL} = K$ (remarquons que $I_{\lambda,CL} = I$ est toujours vrai) alors $\mathcal{A} \models u_{\lambda,CL} = u$, donc:

$$\begin{array}{rcl} \lambda_p \vdash u_\lambda = v_\lambda & \Longrightarrow & \mathcal{A} \models u_{\lambda,CL} = v_{\lambda,CL} \\ & \Longrightarrow & \mathcal{A} \models u = v \end{array}$$ □

Nous allons maintenant caractériser équationnellement les algèbres combinatoires avec éléments partiels qui sont des λ_p-algèbres et pour cela nous allons établir un certain nombre de lemmes préparatoires:

Lemme 50 $\forall t \in \mathcal{T}, [x]t \Downarrow$.

Preuve : Par une induction immédiate sur t. □

Lemme 51 $\forall t, u \in \mathcal{T}$ LCEP $\vdash ([x]t)u = t\lceil u$

Preuve : Par induction sur t:

$t = x, ([x]x)u = Iu = u = u\lceil u,$

$t = y,\ y \neq x : ([x]y)u = Kyu = y\lceil u,$

$t = c$, pour $c \in C$, ou $c \in \{S, K, I\} : ([x]c)u = Kcu = c\lceil u,$

$t = vw, ([x]vw)u = S([x]v)([x]w)u = ([x]v)u(([x]w)u) \overset{induction}{=} (v\lceil u)(w\lceil u) = vw\lceil u.$ □

Lemme 52 *Si nous enrichissons le système avec le schéma d'axiome suivant:*

$$S(Ku)(Kv)\lceil uv = K(uv) \tag{5.1}$$

alors $\forall u \in \mathcal{T}$ $y \notin VL(u)$, *le schéma d'axiome* $([y]u)\lceil u = Ku$ *est dérivable dans le nouveau système.*

Preuve : Nous allons montrer ce résultat par induction sur u:

Si $u = z$, alors $z \neq y$ et c'est évident.

Si $u = c, S, K, I$, c'est tout aussi simple.

Si $u = vw$, alors

$$\begin{array}{rcl} ([y]vw)\lceil u & = & S([y]v)([y]w) \\ \text{par induction} & = & S(Kv)(Kw)\lceil vw \\ \text{par le schéma} & = & K(vw)\lceil vw \end{array}$$ □

Corollaire 2 *Dans le nouveau système, si* $\forall t \in \mathcal{T}$ $t \Downarrow$ *alors* $[x]t = Kt$, *et par conséquent* $[x]t \Downarrow$.

Lemme 53 *Maitenant, si nous enrichissons le système avec les schémas d'axiome suivants:*

$$S(S(KK)I)(Ka) = I\lceil a \tag{5.2}$$

$$S(S(KK)(Kx))(Ka) = Kx\lceil a \tag{5.3}$$

$$S(S(KK)(Sab))(Kc) = Sab\lceil c \tag{5.4}$$

alors $\forall v, w \in \mathcal{T}$ $y \notin VL(w)$ *le schéma d'axiome* $([y]v\lceil w)\lceil w = ([y]v)\lceil w$ *est dérivable dans le nouveau système.*

Par induction sur v:

$v = y,$

$$\begin{array}{rcl} ([y]y\lceil w)\lceil w & = & ([y]Kyw)\lceil w \\ & = & (S(S(KK)I)([y]w))\lceil w \\ \text{par les règles} & = & (S(S(KK)I)(([y]w)\lceil w))\lceil w \\ \text{par le lemme 52} & = & (S(S(KK)I)(Kw))\lceil w \\ \text{par le schéma 5.2} & = & I\lceil w \end{array}$$

$v = z$,

$$\begin{aligned}
([y]z\lceil w)\lceil w &= ([y]Kzw)\lceil w \\
&= (S(S(KK)(Kz))([y]w))\lceil w \\
\text{par les règles} &= (S(S(KK)(Kz))(([y]w)\lceil w))\lceil w \\
\text{par le lemme 52} &= (S(S(KK)(Kz))(Kw))\lceil w \\
\text{par le schéma 5.3} &= Kz\lceil w
\end{aligned}$$

$v = c, S, K, I$, analogue au cas précédent.

$v = ut$,

$$\begin{aligned}
([y]ut\lceil w)\lceil w &= ([y]K(ut)w)\lceil w \\
&= (S(S(KK)(S([y]u)([y]t)))([y]w))\lceil w \\
\text{par les règles} &= (S(S(KK)(S([y]u)([y]t)))(([y]w)\lceil w))\lceil w \\
\text{par le lemme 52} &= (S(S(KK)(S([y]u)([y]t)))(Kw))\lceil w \\
\text{par le schéma 5.4} &= (S([y]u)([y]t))\lceil w
\end{aligned}$$

Lemme 54 *Soient t et u deux termes de λ_p, tels que $x \notin VL(u)$, alors on peut dériver dans le dernier système le schéma d'axiome suivant:*

$$t[u/x]_{CL}\lceil u_{CL} = t_{CL}[u_{CL}/x]\lceil u_{CL}$$

Preuve : Par induction sur t:

si $t = x$, ou $t = y, y \neq x$, c'est élémentaire.

si $t = MN$ ou $M\lceil N$, il suffit d'appliquer l'hypothèse d'induction et les règles de manipulation des restrictions.

si $t = \lambda y.N$. On peut supposer $y \neq x$ - sinon on ne fait aucune substitution et c'est immédiat - et aussi $y \notin VL(u)$ (pour effectuer les substitutions sans avoir à faire d'α-conversion). Procédons par induction sur N:

$N = x$,

$$\begin{aligned}
(\lambda y.x)[u/x]_{CL} &= (\lambda y.u)_{CL} \\
&= [y]u_{CL} \\
(\lambda y.x)_{CL}[u_{CL}/x] &= Ku_{CL}
\end{aligned}$$

Ce que l'on cherche à montrer a fait l'objet du lemme 52.

$N = y$ ou $N = z$, on ne fait aucune substitution et c'est immédiat.

$N = PQ$,

$$\begin{aligned}
((\lambda y.PQ)[u/x])_{CL}\lceil u_{CL} &= (\lambda y.P[u/x]Q[u/x])_{CL}\lceil u_{CL} \\
&= ([y]P[u/x]_{CL}Q[u/x]_{CL})\lceil u_{CL} \\
&= S([y]P[u/x]_{CL})([y]Q[u/x]_{CL})\lceil u_{CL} \\
&= S((\lambda y.P)[u/x]_{CL})((\lambda y.Q)[u/x]_{CL})\lceil u_{CL} \\
\text{par les règles} &= S((\lambda y.P)[u/x]_{CL}\lceil u_{CL})((\lambda y.Q)[u/x]_{CL})\lceil u_{CL})
\end{aligned}$$

$$\begin{aligned}
\text{par induction} &= S(([y]P_{CL})[u_{CL}/x]\lceil u_{CL})(([y]Q_{CL})[u_{CL}/x]\lceil u_{CL}) \\
\text{par les règles} &= S(([y]P_{CL})[u_{CL}/x])(([y]Q_{CL})[u_{CL}/x])\lceil u_{CL} \\
&= S([y]P_{CL})([y]Q_{CL})[u_{CL}/x]\lceil u_{CL} \\
&= ([y]P_{CL}Q_{CL})[u_{CL}/x]\lceil u_{CL} \\
&= (\lambda y.PQ)_{CL}[u_{CL}/x]\lceil u_{CL}
\end{aligned}$$

$N = P\lceil Q$, similaire au cas précédent.

$N = \lambda z.P$, comme $\lambda z.P[u/x]_{CL} \Downarrow$ et $\lambda z.P_{CL}[u_{CL}/x] \Downarrow$ et que $y \notin VL(u_{CL})$ on va pouvoir utiliser le lemme 53:

$$\begin{aligned}
(\lambda yz.P[u/x])_{CL}\lceil u_{CL} &= ([y](\lambda z.P[u/x])_{CL})\lceil u_{CL} \\
\text{par le lemme 53} &= [y]((\lambda z.P[u/x])_{CL}\lceil u_{CL})\lceil u_{CL} \\
\text{par induction} &= [y]((\lambda z.P)_{CL}[u_{CL}/x]\lceil u_{CL})\lceil u_{CL} \\
\text{par le lemme 53} &= ([y](\lambda z.P)_{CL}[u_{CL}/x])\lceil u_{CL}) \\
&= (\lambda yz.P)_{CL}[u_{CL}/x]\lceil u_{CL}
\end{aligned}$$

□

Lemme 55 *Dans le dernier système le schéma d'axiome suivant (β) est dérivable:*

$$(\lambda x.M)N = M[N/x]\lceil N$$

.

Preuve : d'après le lemme 51 on a:

$$\mathsf{LCEP} \vdash ([x]M_{CL})N_{CL} = M_{CL}[N_{CL}/x]\lceil N_{CL},$$

d'où:

$$\begin{aligned}
((\lambda x.M)N)_{CL} &= ([x]M_{CL})N_{CL} \\
&= M_{CL}[N_{CL}/x]\lceil N_{CL} \\
\text{par le lemme 54} &= (M[N/x])_{CL}\lceil N_{CL} \\
&= (M[N/x]\lceil N)_{CL}
\end{aligned}$$

□

Lemme 56 *Soit le dernier système enrichi par les schémas d'axiome suivants:*

$$S(S(KK)I)I = I \tag{5.5}$$

$$S(S(KK)(Ku))I = Ku \tag{5.6}$$

$$S(S(KK)(Stu))I = Stu \tag{5.7}$$

alors, pour tout terme $u \in \mathcal{T}$, le schéma d'axiome $[x](u\lceil x) = [x]u$ est dérivable dans le nouveau système.

Preuve : On a pour tout terme $u \in \mathcal{T}, [x](u\lceil x) = (S(KK)([x]u))I$ et le résultat se montre aisément selon la forme de u:

$u = x$ se résout par 5.5.

$u = y \neq x, c, S, K, I$, par 5.6.

$u = vw$, par 5.7

Ainsi $(\lambda x.M\lceil x)_{CL} = [x](KM_{CL}x) = S(S(KK)([x]M_{CL}))I = [x]M_{CL} = (\lambda x.M)_{CL}$. □

On dira qu'un système satisfait la ξ-règle pour un schéma d'axiome de la forme $t = u$, si le schéma d'axiome $[x]t = [x]u$ est dérivable dans ce système.

Lemme 57 *Soit le système du lemme précedent enrichi par les schémas suivants:*

$$S(S(KK)(Sab))(Sab) = Sab \tag{5.8}$$
$$S(S(KK)(S(S(KK)t)u))v = S(S(KK)(S(S(KK)t)v))u \tag{5.9}$$
$$S(S(KK)t)(S(S(KK)u)v) = S(S(KK)(S(S(KK)t)u))v \tag{5.10}$$
$$St(S(S(KK)u)v) = S(S(KK)(Stu))v \tag{5.11}$$
$$S(S(S(KK)t)v)u = S(S(KK)(Stu))v \tag{5.12}$$

Alors ce système satisfait la ξ-règle pour les schémas d'axiome suivants:

$t\lceil t = t$

$(t\lceil u)\lceil v = (t\lceil v)\lceil u$

$t\lceil(u\lceil v) = (t\lceil u)\lceil v$

$t(u\lceil v) = tu\lceil v$

$(t\lceil v)u = tu\lceil v$

Preuve:

- $t\lceil t = t$: Il s'agit de prouver que $[x]t\lceil t = [x]t$. Or $[x]t\lceil t = S(S(KK)([x]t))([x]t)$, et en examinant la forme de t, on constate que:

  $t = x$ se résout par 5.5.

  $t = y \neq x, c, S, K, I$, par 5.3 et les règles .

  $t = vw$, par 5.8

- $(t\lceil u)\lceil v = (t\lceil v)\lceil u$: il s'agit de vérifier $[x]K(Ktu)v = [x]K(Ktv)u$ soit

  $$S(S(KK)(S(S(KK)[x]t)[x]u)))[x]v = S(S(KK)(S(S(KK)([x]t))[x]v))[x]u$$

  ce qui est donné par le schéma 5.9.

- $t\lceil(u\lceil v) = (t\lceil u)\lceil v$ est donné de même par le schéma 5.10

- $t(u\lceil v) = tu\lceil v$ est donné par le schéma 5.11.

- $(t\lceil v)u = tu\lceil v$ est donné par le schéma 5.12. □

Lemme 58 *Si nous ajoutons maintenant le schéma d'axiome suivant:*

$$S(S(S(KS)a)b)c = S(Sac)(Sbc) \tag{5.13}$$

alors le système obtenu satisfait la règle ξ pour le schéma d'axiome $Stuv = tv(uv)$.

Preuve : On a

$$[x]Suvw = S(S(S(KS)([x]u))([x]v))([x]w)$$

et

$$[x]uw(vw) = S(S([x]u)([x]w))(S([x]v)([x]w)),$$

il est donc suffisant d'avoir le schéma (5.13). □

Lemme 59 *Soient le schéma suivant:*

$$S(S(KK)(Stv))(Ku) = Stv\lceil u \tag{5.14}$$

Soit SCHEM *l'ensemble des schémas de (5.1) à (5.14). Alors si* LCEP $\vdash u\Downarrow$, *alors* LCEP + SCHEM $\vdash [x]t\lceil u = [x]t$

Preuve :

$$\begin{aligned} [x](t\lceil u) &= [x](Ktu) \\ &= S(S(KK)([x]t)([x]u) \\ \text{puisque } u\Downarrow &= S(S(KK)([x]t)(([x]u)\lceil u) \\ \text{par le lemme 52} &= S(S(KK)([x]t)(Ku) \end{aligned}$$

Afin de prouver $S(S(KK)([x]t)(Ku) = [x]t$, il suffit d'examiner la forme de t:

$t = x$ se résout par 5.2 en utilisant le fait que $u\Downarrow$.

$t = y \neq x, c, S, K, I$, par 5.3 et les règles .

$t = vw$, par 5.15. □

Lemme 60 *Tout ce qui est dérivable dans* LCEP +SCHEM *est dérivable dans* LCEP *enrichi par l'ensemble des équations closes suivantes:*

$$[u][v]S(Ku)(Kv)\lceil uv = [u][v]K(uv) \tag{5.1}$$
$$[u]S(S(KK)I)(Ku) = [u](I\lceil u) \tag{5.2}$$
$$[u][v]S(S(KK)(Kv))(Ku) = [u][v]Kv\lceil u \tag{5.3}$$
$$[u][v][w]S(S(KK)(Suv))(Kw) = [u][v][w]Suv\lceil w \tag{5.4}$$
$$S(S(KK)I)I = I \tag{5.5}$$
$$[u]S(S(KK)(Ku))I = [u]Ku \tag{5.6}$$
$$[u][t]S(S(KK)(Stu))I = [u][t]Stu \tag{5.7}$$
$$[u][v]S(S(KK)(Suv))(Suv) = [u][v]Suv \tag{5.8}$$
$$[t][u][v]S(S(KK)(S(S(KK)t)u))v = [t][u][v]S(S(KK)(S(S(KK)t)v))u \tag{5.9}$$
$$[t][u][v]S(S(KK)t)(S(S(KK)u)v) = [t][u][v]S(S(KK)(S(S(KK)t)u))v \tag{5.10}$$
$$[t][u][v]St(S(S(KK)u)v) = [t][u][v]S(S(KK)(Stu))v \tag{5.11}$$
$$[t][u][v]S(S(S(KK)t)v)u = [t][u][v]S(S(KK)(Stu))v \tag{5.12}$$
$$[u][v][w]S(S(S(KS)u)v)w = [u][v][w]S(Suw)(Svw) \tag{5.13}$$
$$[t][u][v]S(S(KK)(Stv))(Ku) = [t][u][v]Stv\lceil u \tag{5.14}$$

Preuve : Il suffit de montrer que chaque schéma d'axiome de SCHEM se dérive dans LCEP enrichi par l'ensemble des equations closes précédentes.

Souvenons-nous que le lemme 51 énonce que $\forall t \in \mathcal{T}$ LCEP $\vdash ([x]t)u = u[u/x]\lceil u$.

Ainsi pour dériver une équation close $A(u, v, w) = B(u, v, w)$, où A et B sont des termes dépendant par exemple des variables u, v, w, il suffit d'avoir l'équation close

$$[u][v][w]A(u, v, w) = [u][v][w]B(u, v, w),$$

puisqu'il suffit alors d'appliquer ces termes aux variables u, v, w, et d'obtenir ainsi (via le lemme 51)

$$A(u,v,w)\lceil u\lceil v\lceil w = B(u,v,w)\lceil u\lceil v\lceil w,$$

les schémas d'axiome concernant la restriction conduisant alors aisément à $A(u,v,w) = B(u,v,w)$. □

Soient les schémas suivants:

$$[x][y][z]xz(yz) = S \tag{5.15}$$

$$[x][y]x = K \tag{5.16}$$

$$S(KI)I = I \tag{5.17}$$

On notera EQ l'ensemble des équations closes numérotées de (5.1) à (5.17).

Lemme 61 *L'ensemble* EQ *est vérifié dans toute* λ_p*-algèbre .*

Preuve :

La définition d'une λ_p-algèbre nous montre qu'il est suffisant de prouver $\lambda_p \vdash A_\lambda = B_\lambda$ pour toute équation $A = B$ de EQ.

Pour abréger cette vérification, il est utile de remarquer que pour une équation de la forme $[x_1][x_2]\cdots[x_n]C = [x_1][x_2]\cdots[x_n]D$ de EQ, il suffit de vérifier $\lambda_p \vdash C_\lambda = D_\lambda$, puisque d'une part λ_p verifie la règle ξ, d'où:

$$\lambda_p \vdash \lambda x_1 x_2 \cdots x_n.A_\lambda = \lambda x_1 x_2 \cdots x_n.B_\lambda,$$

et que d'autre part le lemme 47 montre que

$$\lambda_p \vdash ([x_1][x_2]\cdots[x_n]u)_\lambda = \lambda x_1 x_2 \cdots x_n.u_\lambda$$

La satisfaction de l'équation close s'ensuit par la définition des λ_p-algèbres et par transitivité de l'égalité.

A partir de ces constatations la vérification n'est plus qu'une excursion relativement monotone dans le lambda calcul partiel, ce pourquoi nous la renvoyons en annexe. □

Lemme 62 $\forall M, N \in \Lambda_P{}^0, \quad \lambda_p \vdash M = N \implies \mathsf{LCEP} + \mathsf{EQ} \vdash M_{CL} = N_{CL}$

Preuve : Prouvons le par induction sur la preuve de $\lambda_p \vdash M = N$.

Pour les axiomes:

$M\lceil x = M$ résulte de l'axiome $x \Downarrow$.

$M\lceil c = M$, résulte de $c \Downarrow$, puisque les constantes ont étés choisies totales,

$M\lceil \lambda x.P$, résulte du lemme 50.

$M\lceil M = M$, $M(N\lceil P) = MN\lceil P$, $(M\lceil P)N = MN\lceil P$, $M\lceil(N\lceil P) = (M\lceil N)\lceil P$, $(M\lceil N)\lceil P = (M\lceil P)\lceil N$, résultent des règles qui sont leur contreparties dans LCEP .

$\beta : (\lambda x.M)N = M[N/x]\lceil N$ a fait l'objet du lemme 55.

Pour les règles d'inférence: Toutes les règles, à l'exception de ξ, ne posant aucun problème, nous n'étudierons que ce cas:

Or il suffit de montrer que, si $\mathsf{LCEP} + \mathsf{EQ} \vdash u = t$ alors $\mathsf{LCEP} + \mathsf{EQ} \vdash [x]u = [x]t$, ce que nous faisons par induction sur la preuve de $\mathsf{LCEP} + \mathsf{EQ} \vdash u = t$:

Les lemmes 57 et 59 règlent la plupart des cas

Le cas de l'axiome $u = u$ est évident, aussi bien que le cas de $Iu = u$ qui résulte directement du schéma (5.17). Le cas de la règle d'inférence $t = v \quad u = w \Longrightarrow tu = vw$ se résout par une simple induction. Pour les équations de EQ: elles n'impliquent que des termes clos et totaux (*cf* le lemme 50). Ainsi, Si $t_1 = t_2$ est un équation de EQ, par le lemme 52 $[x]t_i = Kt_i$, d'où $[x]t_1 = [x]t_2$. □

Théorème 13 *Une algèbre combinatoire avec éléments partiels est une λ_p-algèbre si et seulement si elle est un modèle de* LCEP + EQ.

Preuve : Que toute λ_p-algèbre soit un modèle de LCEP + EQ est conséquence immédiate du lemme 61.

Voyons maitenant que tout modèle de LCEP + EQ est une λ_p-algèbre :

Remarquons tout d'abord que $\mathsf{LCEP} + \mathsf{EQ} \vdash S_{\lambda,CL} = S$ et que $\mathsf{LCEP} + \mathsf{EQ} \vdash K_{\lambda,CL} = S$, puisque ce sont les schémas (5.15) et (5.16). Par le lemme 62 $\lambda_p \vdash M = N \Longrightarrow \mathsf{LCEP} + \mathsf{EQ} \vdash M_{CL} = N_{CL}$, il ne reste plus qu'à appliquer le lemme 49. □

Théorème 14 λ_p *et* LCEP + EQ *sont équivalentes au sens suivant:*

1. $\lambda_p \vdash M_{CL,\lambda} = M$.
2. $\mathsf{LCEP} + \mathsf{EQ} \vdash u_{\lambda,CL} = u$.
3. $\lambda_p \vdash M = N \Leftrightarrow \mathsf{LCEP} + \mathsf{EQ} \vdash M_{CL} = N_{CL}$.
4. $\mathsf{LCEP} + \mathsf{EQ} \vdash u = v \Leftrightarrow \lambda_p \vdash u_\lambda = v_\lambda$.
5. $\lambda_p \vdash I\lceil N = I \Longrightarrow \mathsf{LCEP} + \mathsf{EQ} \vdash N_{CL} \Downarrow$.
6. $\mathsf{LCEP} + \mathsf{EQ} \vdash u \Downarrow \Longrightarrow \lambda_p \vdash I\lceil u_\lambda = I$.

Preuve :

1. objet du lemme 48.

2. par induction en utilisant les équations (5.15) et (5.16).

3. ($\Rightarrow$) c'est le lemme 62.

4. ($\Rightarrow$), par induction sur la preuve de $\mathsf{LCEP} + \mathsf{EQ} \vdash u = v$

3. ($\Leftarrow$), par *4* ($\Rightarrow$) $\lambda_p \vdash M_{CL,\lambda} = N_{CL,\lambda}$, d'où par *1* $\lambda_p \vdash M = N$.

4 ($\Leftarrow$), par *3* ($\Rightarrow$) $\mathsf{LCEP} + \mathsf{EQ} \vdash M_{\lambda,CL} = N_{\lambda,CL}$, d'où par *2* $\mathsf{LCEP} + \mathsf{EQ} \vdash M = N$

5. et *6.* résultent de *3.* et *4.* puisque $\mathsf{LCEP} + \mathsf{EQ} \vdash u \Downarrow \Longleftrightarrow \mathsf{LCEP} + \mathsf{EQ} \vdash I\lceil u = I$. □

Remarque 22 *La η_p-règle est la règle suivante: $(\lambda x.Mx)\lceil M = M$, si $x \notin VL(M)$). Remarquons que celle-ci peut s'axiomatiser de la même façon. En effet, pour que s'assurer que celle-ci soit satisfaite, il faut vérifier dans l'algèbre combinatoire avec éléments partiels que l'on a bien $S([x]M_{CL})I\lceil M_{CL} = M_{CL}$. Or $S([x]M_{CL})I\lceil M_{CL} = S(([x]M_{CL})\lceil M_{CL})I = S(KM_{CL})I$ se vérifie par le lemme 52 pour tout terme clos M et il suffit par conséquent d'ajouter aux équations la clôture du schéma $S(Ka)I = a$, qui est vraie dans toute $\lambda_p{+}\eta_p$-algèbre puisque $\lambda_p \vdash S_\lambda(K_\lambda a_\lambda)I_\lambda = (\lambda x.(a_\lambda x))\lceil a_\lambda$. Soit donc* EQ' *l'ensemble* EQ *augmenté de léquation close $[a]S(Ka)I = [a]a$.*

Théorème 15 LCEP + EQ' *et $\lambda_p{+}\eta_p$ sont équivalentes au sens du théorème précédent.*

La preuve en est identique.

Théorème 16

Soit $\mathcal{A} = \Lambda_P/\lambda_p$ (resp. $\mathcal{A}' = \Lambda_P/\lambda_p + \eta_p$) muni des constantes $S = \lambda xyz.xz(yz)$, $K = \lambda xy.x, I = \lambda x.x$ et de l'application définie par l'application des λ_p-termes. Alors $\mathcal{A}$ est une λ_p-algèbre (resp. $\mathcal{A}'$ est une $\lambda_p + \eta_p$-algèbre).

Preuve: Remarquons que $[\![M_{CL}]\!]_{\mathcal{A}} = M_{CL,\lambda}$ Nous allons utiliser le lemme 49:

1. $[\![S]\!]_{\mathcal{A}} = S_\lambda$, $[\![S_{\lambda,CL}]\!]_{\mathcal{A}} = S_{\lambda,CL,\lambda}$. Or, d'après le lemme 48, $\lambda_p \vdash S_\lambda = S_{\lambda,CL,\lambda}$, d'où $\mathcal{A} \models S = S_{\lambda,CL}$, et de même pour K.

2. Nous devons prouver que si $\lambda_p \vdash M = N$ alors $\mathcal{A} \models M_{CL} = N_{CL}$, soit $[\![M_{CL}]\!]_{\mathcal{A}} = [\![N_{CL}]\!]_{\mathcal{A}}$. Or $[\![M_{CL}]\!]_{\mathcal{A}} = M_{CL,\lambda}$ et le lemme 48 prouve donc que $[\![M_{CL}]\!]_{\mathcal{A}} = M$, et de même pour N. Mais alors $\lambda_p \vdash M = N$ donne le résultat souhaité.

Le raisonnement est exactement identique pour prouver que $\mathcal{A}'$ est une $\lambda_p + \eta_p$-algèbre. □

Conclusion

Le théorème d'équivalence que nous avons prouvé montre que les termes clos du λ_p-calcul constituent une λ_p-algèbre avec plusieurs éléments indéfinis (*cf* [Pino 91] pour des exemples de λ_p-algèbres avec un un seul point indéfini). Remarquons également que toute λ-algèbre est un cas particulier et dégénérée de λ_p-algèbre. Les "vraies" équations de Curry impliquent donc les équations qui caractérisent les λ_p-algèbres.

Une question qui demeure ouverte serait de construire une autre λ_p-algèbre que l'algèbre des termes ayant au moins deux éléments indéfinis distincts, question qui pourrait sans doute être abordée par des techniques issues de la sémantique dénotationnelle.

6

Approches catégoriques

Nous allons montrer que la donnée d'une catégorie cartésienne fermée partielle munie d'un objet réflexif permet de construire un modèle du λ_p-calcul en ce qu'elle fournit une λ_p-algèbre. Nous montrons ensuite que, réciproquement la donnée d'une λ_p-algèbre permet de construire une telle catégorie. Nous montrons enfin comment partant d'une λ_p-algèbre , construisant la PCCC associée et extrayant de cette dernière une λ_p-algèbre , à quelle condition cette λ_p-algèbre est isomorphe à la première. Ceci est à rapprocher de résultats analogues pour le λ-calcul (*cf* [Scott80, Koymans 82, Barendregt], et de l'interprétation catégorique dans les C-monoïdes partiels $PCCL$ de [Pino 87].

Pour les notions catégoriques nous renvoyons à [Mac Lane, Lambek& Scott].

6.1 Les catégories cartésiennes fermées partielles

La notion de catégorie cartésienne fermée est issue de l'observation de la catégorie *PSET*, dont les objets sont les ensembles et les flèches les applications partielles.

C'est une structure de catégorie enrichie d'une structure d'ordre, puisque les "Hom" sont munis d'un ordre: si $f, g \in Hom(a, b)$

$$(f \leq g) \Leftrightarrow (Dom(f) \subset Dom(g) \text{ et } f\lceil Dom(f) = g\lceil Dom(f))$$

Cet ordre sera appelé *(ordre de la) restriction.*

On observe dans *PSET* les faits suivants:

Les applications totales correspondent aux flèches maximales dans $Hom(a, b)$ pour l'ordre de la restriction,

Les identités sont totales,

La composition est croissante dans ses deux arguments et conserve la "totalité".

Le singleton $\{*\}$ est un objet terminal, et la flèche terminale $!_a$ ($!_a = \lambda x.* : a \to \{*\}$) associée à tout objet a est totale,

Si $f : a \to b$, alors la flèche $!_b \circ f$ représente le domaine de définition de f,

Soient les applications partielles $f : a \to b, g : a \to c$, alors l'application partielle: $< f, g >$: $a \to b \times c$ est définie si et seulement si f et g le sont. On a ainsi un produit partiel où les projections sont totales. De plus $Fst \circ < f, g >$ correspond à la restriction de f au domaine de g, ce qui nous fournit un autre moyen de parler du domaine d'une application.

Les applications partielles d'un ensemble dans un autre forment un nouvel ensemble, et on dispose ainsi d'une exponentielle partielle. Ses propriétés diffèrent du cas total,

puisqu'une exégèse catégorique de PSET montrerait qu'il s'agit ici de lax-limites, et non de limites et de même les adjonctions qui expriment les relations entre produit et exponentielle demanderaient à être formulées dans un cadre différent que celui des adjonctions entre foncteurs. Nous ne développerons pas cet aspect. Remarquons seulement que la donnée d'une application partielle de $a \times b \to c$ correspond à une flèche totale de a dans l'ensemble des flèches partielles de b dans c. L'abstraction par rapport à la première composante est totale, autrement dit "les abstractions sont totales", d'où s'aperçoit une affinité avec le λ_p-calcul.

6.2 Définition

Les considérations précédentes conduisent ainsi à généraliser la structure rencontrée dans *PSET*, et c'est ce qu'ont proposé P.L. Curien et A.Obtułówicz [Curien-Obtu.89]. Une catégorie C enrichie d'une structure d'ordre sera dite une catégorie cartésienne fermée partielle (en abrégé PCCC) si elle a en outre la structure suivante:

1. Il existe un objet distingué T et, pour tout objet a, une flèche $!_a$ maximale dans $a \to T$.
2. Les flèches $f : a \to b$ qui vérifient $!_b \circ f = !_a$ (et qui sont dites *totales*) sont maximales.
3. Les identités sont totales, la composition est croissante dans ses deux arguments:
$$f \leq g \Rightarrow f \circ h \leq g \circ h \text{ et } h \circ f \leq h \circ g.$$
4. Si $f, f', g : a \to b$ vérifient $f, f' \leq g$ et $!_b \circ f \leq !_b \circ f'$, alors $f \leq f'$.
5. Si $h, h' : a \to b$ vérifient $h \leq h'$, alors pour tout $g : b \to T$ $\quad g \circ h = (g \circ h') \cap (!_b \circ h)$.
6. Pour tous objets a, b, un objet $a \times b$ est donné et est dit *produit partiel* de a et de b, ainsi que des flèches totales $Fst_{a,b} : a \times b \to a$ et $Snd_{a,b} : a \times b \to b$ et une opération binaire associant la flèche $< f, g >: a \to b \times c$ aux flèches $f : a \to b$, $g : a \to c$. On posera $f \lceil g = Fst \circ < f, g >$. $f \lceil g$ est donc une expression croissante en f et en g.
7. $< \ >$ est croissante dans ses deux arguments:
$$f \leq g \Rightarrow < f, h > \leq < g, h > \text{ et } < h, f > \leq < h, g > .$$
8. $\forall h : a \to b \times c, \quad < Fst_{b,c} \circ h, Snd_{b,c} \circ h > = h.$
9. $\forall f : a \to b, g : a \to c, Fst_{a,b} \circ < f, g > \leq f$ et $Snd_{a,b} \circ < f, g > \leq g$.
10. $\forall f : a \to b, g : a \to c, h : d \to a, < f, g > \circ h = < f \circ h, g \circ h >$.
11. $\forall f : a \to b, g : a \to c, !_{b \times c} \circ < f, g > = (!_b \circ h) \cap (!_c \circ g)$.
12. Il existe une opération binaire (*exponentiation partielle*) $\to_P$ telle pour tous objets a, b, c, il existe des bijections $\lambda_{a,b,c}$ (que nous noterons plus simplement Λ s'il n'y a pas d'ambigüité) de $a \times b \to c$ vers $a \to_T (b \to_P c)$ ($a \to_T b$ dénotant le sous-ensemble des flèches totales de $a \to b$), telles que, pour tous $f : d \to a, g : a \times b \to c$,
$$\Lambda(g) \circ f \leq \Lambda(g \circ < f \circ Fst, Snd >).$$
On peut alors définir une nouvelle flèche $ev_{a,b} = \lambda_{a,b,a \times b}(id_{a \times b})$.

6.3 Présentation équationnelle

Curien et Obtulowicz ([Curien-Obtu.89]) ont montré qu'une catégorie enrichie d'une structure d'ordre et munie d'opérations de même signature que les précédentes est une PCCC si et seulement si les équations suivantes sont satisfaites (lorsqu'elles ont un sens):

$$
\begin{aligned}
&(6.1) && f\lceil f &&= f\\
&(6.2) && f\lceil(g\lceil h) &&= f\lceil(h\lceil g)\\
&(6.3) && f\lceil(g\lceil h) &&= (f\lceil g)\lceil h\\
&(6.4) && f\lceil(g\lceil h) &&= f\lceil<g,h>\\
&(6.5) && (f\circ g)\lceil h &&= f\circ(g\lceil h)\\
&(6.6) && <f,g>\lceil h &&= <f\lceil h, g\lceil h>\\
&(6.7) && <f,g>\lceil h &&= <f, g\lceil h>\\
&(6.8) && <f,g>\circ h &&= <f\circ h, g\circ h>\\
&(6.9) && <Fst, Snd> &&= Id\\
&(6.10) && Fst\circ <f,g> &&= Snd\circ <g,f>\\
&(6.11) && Id\lceil ! &&= Id\\
&(6.12) && !\lceil f &&= f \qquad (f : a \to t)\\
&(6.13) && !\circ\Lambda(f) &&= !\\
&(6.14) && ev\circ <\Lambda(f)\circ Fst, Snd> &&= f\\
&(6.15) && \Lambda(ev\circ <f\circ Fst, Snd>)\lceil f &&= f
\end{aligned}
$$

On notera $f \times g = <f\circ Fst, g\circ Snd>$, ce qui permet de reformuler l'équation (6.14) par:

$$(6.14) \qquad ev\circ(\Lambda(f) \times Id) = f$$

et l'équation (6.15) par:

$$(6.15) \qquad \Lambda(ev\circ(f \times Id))\lceil f = f.$$

6.4 Propriétés

Nous allons ici établir un certain nombre de propriétés qui vont nous être utiles pour la suite. Nous nous appuierons sur un certain nombre de résultats de [Curien-Obtu.89] que nous ne redémontrons pas ici.

Nous admettrons notamment que $a\lceil b = Fst\circ <a,b> = Snd\circ <b,a>$, que $a\lceil <b,c> = a\lceil b\lceil c = a\lceil c\lceil b$ et qu'un crochet de flèches totales est est une flèche totale.

Lemme 63 $(f \times g)\circ <a,b> = <f\circ a, g\circ b>$

Preuve:

$$
\begin{aligned}
(f \times g)\circ <a,b> &= <f\circ Fst, g\circ Snd> \circ <a,b>\\
&= <f\circ Fst\circ <a,b>, g\circ Snd\circ <a,b>>\\
&= <f\circ(a\lceil b), g\circ(b\lceil a)>\\
\text{par 6.5, 6.6 et 6.7} \quad &= <f\circ a, g\circ b> \lceil b\lceil a\\
\text{idem} \quad &= <f\circ(a\lceil a), g\circ(b\lceil b)>\\
\text{par 6.1} \quad &= <f\circ a, g\circ b>
\end{aligned}
$$

□

Lemme 64 $(f \times g)\circ(f' \times g') = (f\circ f') \times (g\circ g')$

Preuve :

$$\begin{aligned}(f \times g)\circ(f' \times g') &= (f \times g)\circ < f'\circ Fst, g'\circ Snd > \\ &= f'\circ f\circ Fst, g\circ g'\circ Snd \\ &= (f\circ f') \times (g\circ g')\end{aligned}$$ □

Lemme 65 $\Lambda(g)\circ f = \Lambda(g\circ(f \times id))\lceil f$

Preuve : Par 12 $(\Lambda(g)\circ f)\lceil f \leq \Lambda(g\circ < f\circ Fst, Snd >)\lceil f$, d'où:

$$\Lambda(g)\circ f \leq \Lambda(g\circ < f\circ Fst, Snd >)\lceil f.$$

Pour voir l'inégalité dans l'autre sens, nous pouvons utiliser la propriété 4:

$!\circ\Lambda(g)\circ f = !\circ f$ et $!\circ\Lambda(g\circ(f \times id))\lceil f = !\lceil f$, par 2, puisque $\Lambda(h)$ est totale pour toute flèche h. Nous sommes donc ramenés à vérifier $!\circ f \leq !\lceil f$. Utilisons cette fois la propriété 5: $Id\circ(!\circ f) = (Id\circ(!\lceil f)) \cap (!\circ!\circ f)$, soit $!\circ f = (!\lceil f) \cap (!\circ f)$, c'est-à-dire $!\circ f \leq !\lceil f$. □

Lemme 66 $\Lambda(f\lceil(g\circ Fst))\lceil g = \Lambda(f)\lceil g$

Preuve : Remarquons tout d'abord que $f\lceil g = (f\circ Id)\lceil g = f\circ(Id\lceil g)$, d'où:

$$\begin{aligned}\Lambda(f)\lceil g &= \Lambda(f)\circ(Id\lceil g)\lceil g \\ \text{par le lemme 65} \quad &= \Lambda(f\circ(Id\lceil g) \times Id)\lceil g \\ &= \Lambda(f\circ(Fst\circ < Id, g >) \times Id)\lceil g \\ &= \Lambda(f\circ < Fst\circ < Id, g > \circ Fst, Snd >)\lceil g \\ &= \Lambda(f\circ < Fst\circ < Fst, g\circ Fst >, Snd >)\lceil g \\ &= \Lambda(f\circ < Fst\lceil(g\circ Fst), Snd >)\lceil g \\ &= \Lambda(f\circ < Fst, Snd > \lceil(g\circ Fst))\lceil g \\ &= \Lambda(f\circ Id\lceil(g\circ Fst))\lceil g \\ &= \Lambda(f\lceil(g\circ Fst))\lceil g\end{aligned}$$ □

Lemme 67 $ev\circ < \Lambda(f), g >= f\circ < Id, g >$

Preuve :

$$\begin{aligned}& ev\circ < \Lambda(f), g > \\ \text{par le lemme 63} \quad &= ev\circ(\Lambda(f) \times Id)\circ < Id, g > \\ \text{par 6.14} \quad &= f\circ < Id, g >\end{aligned}$$ □

Lemme 68 *Si $F\circ G$ est totale alors G est totale.*

Preuve : $!_{U^U}\circ F \leq !_U$, d'où $!_{U^U}\circ F\circ G = !_{U^U}\circ Id_{U^U} = !_{U^U} \leq !_U\circ G$, ce qui par maximalité de $!_{U^U}$ montre que $!_{U^U} = !_U\circ G$, ce dont on déduit par 2 que G est totale. □

Remarquons également que si g est totale alors $f\lceil g = f$ (utiliser 4).

6.5 Interprétation des termes

Soit une PCCC muni d'un objet réflexif U, c'est-à-dire un objet U tel qu'il existe deux flèches $F : U \to U^U$ et $G : U^U \to U$ vérifiant $F\circ G = Id_{U^U}$.

Remarquons que la remarque 68 montre que G est une flèche totale,

Soit la flèche $App = ev_{U,U}\circ(F \times Id_U) : U^2 \to U$.

Soient $f, g : A \to U$, on notera $f._A g = App \circ < f, g >$.

$._T$ fait de $|U|$ une structure applicative.

Posons $U^0 = T, U^{n+1} = U^n \times U$. Soit $\Delta = x_1, ..., x_n$ une suite de variables distinctes. On posera $U^\Delta = U^n$.

$\pi^\Delta_{x_i} : U^\Delta \to U$ est la i^{eme} projection canonique. C'est une flèche totale.

Remarquons que $\pi^\Delta_{x_i} = Snd_{U^i} \circ Fst_{U^{i+1}} \circ ... \circ Fst_{U^n}$.

Si $f_1, ..., f_n : A \to U$, alors $< f_1, ..., f_n >$ est défini par induction:

$$< >=!_A,$$
$$< f_1, ..., f_{n+1} >=<< f_1, ..., f_n >, f_{n+1} >$$

Lemme 69 *Soient $f_1, ..., f_n : A \to U$, alors*

$$\pi^\Delta_{x_i} \circ < f_1, ..., f_n >= f_i \lceil < f_1, ..., f_n >$$

Souvenons-nous que par définition $h \lceil g = Fst \circ < h, g >= Snd \circ < g, h >$ et que $g \circ (f \lceil h) = (g \circ f) \lceil h$.

$$\begin{aligned}
& \pi^\Delta_{x_i} \circ < f_1, ..., f_n > \\
= \; & Snd_{U^i} \circ Fst_{U^{i+1}} \circ ... \circ Fst_{U^n} \circ << f_1, ..., f_{n-1} >, f_n > \\
= \; & (Snd_{U^i} \circ Fst_{U^{i+1}} \circ ... \circ < f_1, ..., f_{n-1} >) \lceil f_n \\
= \; & ... \\
= \; & (...(Snd_{U^i} \circ << f_1, ..., f_{i-1} >, f_i >) \lceil f_{i+1}) ...) \lceil f_n. \\
= \; & (f_i \lceil < f_1, ..., f_{i-1} >) \lceil f_{i+1}) ...) \lceil f_n.
\end{aligned}$$

Il suffit alors de se rappeler que $f \lceil f = f$ et d'utiliser les diverses règles concernant la restriction dans une PCCC pour conclure. □

Soit deux contextes $\Gamma = y_1, ..., y_m$ et $\Delta = x_1, ..., x_n$ tels que $\{y_1, ..., y_n\} \subset \{x_1, ..., x_n\}$, on définit alors Π^Δ_Γ par:

$$\Pi^\Delta_\Gamma =< \pi^\Delta_{y_1}, ..., \pi^\Delta_{y_m} >: U^\Delta \to U^\Gamma$$

Cette flèche correspond à l'affaiblissement (partiel):

$$\lambda < x_1, ..., x_n > . < y_1, ..., y_n > \lceil < x_1, ..., x_n >: U^\Delta \to U^\Gamma$$

et c'est un crochet de flèches totales, donc une flèche totale.

Remarque 23 *Les relations suivantes nous seront fort utiles:*

1. $\Pi^{\Gamma,x}_\Gamma = Fst_{U^{n+1}}$
2. $\Pi^{\Delta,x}_{\Gamma,x} = \Pi^\Delta_\Gamma \times Id$

Preuve :

1.

$$\begin{aligned}
\Pi^{\Gamma,x}_\Gamma &= < \pi^{\Gamma,x}_1, .., \pi^{\Gamma,x}_n > \\
&= < Snd_{U^i} \circ Fst_{U^{i+1}} \circ ... \circ Fst_{U^{n+1}} >_{i=1,..,n} \\
\text{par l'équation 6.8} \quad &= (< Snd_{U^i} \circ Fst_{U^{i+1}} \circ ... \circ Fst_{U^n} >_{i=1,..,n}) \circ Fst_{U^{n+1}} \\
&= \Pi^\Gamma_\Gamma \circ Fst_{U^{n+1}} \\
&= Id \circ Fst_{U^{n+1}} = Fst_{U^{n+1}}
\end{aligned}$$

2.

$$\begin{aligned}\Pi^{\Delta,x}_{\Gamma,x} &= <<\Pi^{\Delta,x}_{\gamma_1},..,\Pi^{\Delta,x}_{\gamma_n}>,\Pi^{\Delta,x}_{x}> \\ &= <\Pi^{\Delta,x}_{\Gamma},\Pi^{\Delta,x}_{x}> \\ &= <\Pi^{\Delta}_{\Gamma}\circ Fst, Snd> \\ &= \Pi^{\Delta}_{\Gamma}\times Id\end{aligned}$$

□

Posons $|U|_T = T \to_T U$, on suppose s'être donné pour chaque constante c du lambda calcul partiel un élément a_c de $|U|_T$.

L'interprétation d'un terme M relativement à un contexte Δ est une flèche $[\![M]\!]_\Delta : U^\Delta \to U$ définie inductivement comme suit:

$$\begin{aligned}[\![x]\!]_\Delta &= \pi^\Delta_x; \\ [\![c]\!]_\Delta &= a_c\circ !_{U^\Delta} \\ [\![PQ]\!]_\Delta &= [\![P]\!]_\Delta\cdot_{U^\Delta}[\![Q]\!]_\Delta; \\ [\![\lambda x.P]\!]_\Delta &= G\circ\Lambda([\![P]\!]_{\Delta,x}), x\notin\Delta; \\ [\![P\lceil Q]\!]_\Delta &= [\![P]\!]_\Delta\lceil[\![Q]\!]_\Delta.\end{aligned}$$

Nous allons maintenant préciser l'interprétation des termes relativement à une valuation comme un élément de $|U|$. Notons que les variables libres, de même que les constantes, représentent implicitement des termes totaux, il convient de les interpréter par des termes totaux, c'est-à-dire dans $|U|_T$.

Soit une valuation $\rho : V \to |U|_T$, définissons l'interprétation d'un terme M relativement à ρ comme l'élément de $|U|_T$ défini par $[\![M]\!]_\rho = [\![M]\!]_\Delta\circ\rho^\Delta$, avec:

$$\rho^\Delta = \rho^{x_1,...,x_n} = <\rho(x_1),...,\rho(x_n)> .$$

Lemme 70

1. *Soient* $\Delta \supset \Gamma \supset VL(M)$. *Alors*

$$[\![M]\!]_\Delta = [\![M]\!]_\Gamma\circ\Pi^\Delta_\Gamma$$

2. *Soient* $\Delta = \{x_1,...,x_n\} \supset VL(M)$ *et des termes* $N_1,...,N_n$ *dont toutes les variables libres sont dans un contexte* Γ. *Alors*

$$[\![M[\vec{N}/\vec{x}]]\!]_\Gamma\lceil<[\![\vec{N}]\!]_\Gamma> = [\![M]\!]_\Delta\circ<[\![\vec{N}]\!]_\Gamma>,$$

à savoir:

$$[\![M[N_1/x_1...,N_n/x_n]\lceil\{N_1,...,N_n\}]\!]_\Gamma = [\![M]\!]_\Delta\circ<[\![N_1]\!]_\Gamma,...,[\![N_n]\!]_\Gamma>$$

3. *Soient* $\Delta \supset VL(\lambda x.M), \Gamma \supset VL((\lambda x.M)N)$ *et tels que* $\Delta \subset \Gamma$. *Alors*

$$[\![M[N/x]\lceil N]\!]_\Gamma = [\![M]\!]_{\Delta,x}\circ<\Pi^\Delta_\Gamma,[\![N]\!]_\Gamma>$$

Preuve : (*1*) Par induction sur M:

$M = x$:

$$\begin{aligned}[\![x]\!]_\Gamma\circ\Pi^\Delta_\Gamma &= \pi^\Gamma_x\circ<\pi^\Delta_{y_1},...,\pi^\Delta_{y_m}> \\ &= \pi^\Delta_x\lceil<\pi^\Delta_{y_1},...,\pi^\Delta_{y_m}> \\ \text{les } \pi^\Delta_{y_j} \text{ étant totales} &= \pi^\Delta_x = [\![x]\!]_\Delta\end{aligned}$$

$M = c$,

$$\begin{aligned} [\![c]\!]_\Gamma \circ \Pi_\Gamma^\Delta &= a_c \circ !_{U\Gamma} \circ \Pi_\Gamma^\Delta \\ \Pi_\Gamma^\Delta \text{ étant totale} &= a_c \circ !_{U\Delta} = [\![c]\!]_\Delta \end{aligned}$$

$M = PQ$,

$$\begin{aligned} [\![PQ]\!]_\Gamma \circ \Pi_\Gamma^\Delta &= ([\![P]\!]_\Gamma \cdot_{U\Gamma} [\![Q]\!]_\Gamma) \circ \Pi_\Gamma^\Delta \\ &= App \circ \langle [\![P]\!]_\Gamma, [\![Q]\!]_\Gamma \rangle \circ \Pi_\Gamma^\Delta \\ &= App \circ \langle [\![P]\!]_\Gamma \circ \Pi_\Gamma^\Delta, [\![Q]\!]_\Gamma \circ \Pi_\Gamma^\Delta \rangle \\ \text{par hypothèse d'induction} &= App \circ \langle [\![P]\!]_\Delta, [\![Q]\!]_\Delta \rangle = [\![PQ]\!]_\Delta \end{aligned}$$

$M = P \lceil Q$,

$$\begin{aligned} [\![P \lceil Q]\!]_\Gamma \circ \Pi_\Gamma^\Delta &= Fst \circ \langle [\![P]\!]_\Gamma, [\![Q]\!]_\Gamma \rangle \circ \Pi_\Gamma^\Delta \\ \text{par hypothèse d'induction} &= Fst \circ \langle [\![P]\!]_\Delta, [\![Q]\!]_\Delta \rangle = [\![P \lceil Q]\!]_\Delta \end{aligned}$$

$M = \lambda x.P$,

$$\begin{aligned} [\![\lambda x.P]\!]_\Gamma \circ \Pi_\Gamma^\Delta &= G \circ (\Lambda([\![P]\!]_{\Gamma,x} \circ (\Pi_\Gamma^\Delta \times Id)) \lceil \Pi_\Gamma^\Delta \\ \Pi_\Gamma^\Delta \text{ étant totale } [\![\lambda x.P]\!]_\Gamma \circ \Pi_\Gamma^\Delta &= G \circ (\Lambda([\![P]\!]_{\Gamma,x} \circ (\Pi_\Gamma^\Delta \times Id)) \\ \text{par la remarque 23} &= G \circ (\Lambda([\![P]\!]_{\Gamma,x}) \circ \Pi_{\Gamma,x}^{\Delta,x}) \\ \text{par hypothèse d'induction} &= G \circ (\Lambda[\![P]\!]_{\Delta,x}) = [\![\lambda x.P]\!]_\Delta \end{aligned}$$

(*2*) Par induction sur M:

$M = x_i$:

$$\begin{aligned} [\![x_i[\vec{N}/\vec{x}]]\!]_\Gamma \lceil \langle [\![\vec{N}]\!]_\Gamma \rangle &= [\![N_i]\!]_\Gamma \lceil \langle [\![\vec{N}]\!]_\Gamma \rangle \\ \text{par le lemme 69} &= \pi_x^\Delta \circ \langle [\![N_1]\!]_\Gamma, \ldots, [\![N_n]\!]_\Gamma \rangle \\ &= [\![x_i]\!]_\Delta \circ \langle [\![\vec{N}]\!]_\Gamma \rangle \end{aligned}$$

$M = c$:

$$\begin{aligned} [\![c]\!]_\Delta \lceil \langle [\![\vec{N}]\!]_\Gamma \rangle &= a_c \circ !_{U\Delta} \lceil \langle [\![\vec{N}]\!]_\Gamma \rangle \\ \text{par l'égalité } !_b \circ f = !_a \lceil f, \text{ pour } f : a \to b &= a_c \circ !_{U\Gamma} \circ \langle [\![\vec{N}]\!]_\Gamma \rangle \\ &= [\![c[\vec{N}/\vec{x}]]\!]_\Gamma \circ \langle [\![\vec{N}]\!]_\Gamma \rangle \end{aligned}$$

$M = PQ$:

$$\begin{aligned} [\![PQ[\vec{N}/\vec{x}]]\!]_\Gamma \lceil \langle [\![\vec{N}]\!]_\Gamma \rangle &= ([\![P[\vec{N}/\vec{x}]]\!]_\Gamma \cdot_{U\Gamma} [\![Q[\vec{N}/\vec{x}]]\!]_\Gamma) \lceil \langle [\![\vec{N}]\!]_\Gamma \rangle \\ &= (App \circ \langle [\![P[\vec{N}/\vec{x}]]\!]_\Gamma, [\![Q[\vec{N}/\vec{x}]]\!]_\Gamma \rangle) \lceil \langle [\![\vec{N}]\!]_\Gamma \rangle \\ \text{par les règles} &= App \circ \langle [\![P[\vec{N}/\vec{x}]]\!]_\Gamma \lceil \langle [\![\vec{N}]\!]_\Gamma \rangle, [\![Q[\vec{N}/\vec{x}]]\!]_\Gamma \lceil \langle [\![\vec{N}]\!]_\Gamma \rangle \rangle \\ \text{par induction} &= App \circ \langle [\![P]\!]_\Delta \circ \langle [\![\vec{N}]\!]_\Gamma \rangle, [\![Q]\!]_\Delta \circ \langle [\![\vec{N}]\!]_\Gamma \rangle \rangle \\ &= App \circ \langle [\![P]\!]_\Delta, [\![Q]\!]_\Delta \rangle \circ \langle [\![\vec{N}]\!]_\Gamma \rangle \\ &= [\![PQ]\!]_\Delta \circ \langle [\![\vec{N}]\!]_\Gamma \rangle \end{aligned}$$

$M = P\lceil Q$:

$$\begin{aligned}
[\![(P\lceil Q)[\vec{N}/\vec{x}]]\!]_\Gamma\lceil <[\![\vec{N}]\!]_\Gamma> &= ([\![P[\vec{N}/\vec{x}]]\!]_\Gamma\lceil[\![Q[\vec{N}/\vec{x}]]\!]_\Gamma)\lceil <[\![\vec{N}]\!]_\Gamma> \\
\text{par les règles} &= ([\![P[\vec{N}/\vec{x}]]\!]_\Gamma\lceil <[\![\vec{N}]\!]_\Gamma>)\lceil([\![Q[\vec{N}/\vec{x}]]\!]_\Gamma)\lceil <[\![\vec{N}]\!]_\Gamma>) \\
\text{par hypothèse d'induction} &= ([\![P]\!]_\Delta \circ <[\![\vec{N}]\!]_\Gamma>)\lceil([\![Q]\!]_\Delta)\circ <[\![\vec{N}]\!]_\Gamma>) \\
&= Fst\circ <[\![P]\!]_\Delta \circ <[\![\vec{N}]\!]_\Gamma>, [\![Q]\!]_\Delta \circ <[\![\vec{N}]\!]_\Gamma>> \\
&= Fst\circ <[\![P]\!]_\Delta, [\![Q]\!]_\Delta> \circ <[\![\vec{N}]\!]_\Gamma> \\
&= [\![P\lceil Q]\!]_\Delta \circ <[\![\vec{N}]\!]_\Gamma>
\end{aligned}$$

$M = \lambda x.P$:

$$\begin{aligned}
[\![(\lambda x.P)[\vec{N}/\vec{x}]]\!]_\Gamma\lceil <[\![\vec{N}]\!]_\Gamma> &= [\![(\lambda x.P[\vec{N}/\vec{x}])]\!]_\Gamma\lceil <[\![\vec{N}]\!]_\Gamma> \\
&= G\circ\Lambda([\![P[\vec{N}/\vec{x}]]\!]_{\Gamma,x})\lceil <[\![\vec{N}]\!]_\Gamma> \\
\text{par le lemme 66} &= G\circ\Lambda([\![P[\vec{N}/\vec{x}]]\!]_{\Gamma,x}\lceil <[\![\vec{N}]\!]_\Gamma> \circ Fst)\lceil <[\![\vec{N}]\!]_\Gamma>, \\
[\![x]\!]_{\Gamma,x} = Snd \text{ étant totale} &= (G\circ\Lambda([\![P[\vec{N}/\vec{x}]]\!]_{\Gamma,x}\lceil <[\![\vec{N}]\!]_{\Gamma,x}, [\![x]\!]_{\Gamma,x}>))\lceil <[\![\vec{N}]\!]_\Gamma> \\
&= (G\circ\Lambda([\![P[\vec{N}/\vec{x}, x/x]]\!]_{\Gamma,x}\lceil <[\![\vec{N}, x]\!]_{\Gamma,x}>))\lceil <[\![\vec{N}]\!]_\Gamma> \\
\text{par induction} &= (G\circ\Lambda([\![P]\!]_{\Delta,x}\circ <[\![\vec{N}, x]\!]_{\Gamma,x}>))\lceil <[\![\vec{N}]\!]_\Gamma> \\
&= (G\circ\Lambda([\![P]\!]_{\Delta,x}\circ <[\![\vec{N}]\!]_\Gamma> \times Id))\lceil <[\![\vec{N}]\!]_\Gamma> \\
&= G\circ\Lambda([\![P]\!]_{\Delta,x})\circ <[\![\vec{N}]\!]_\Gamma> \\
&= [\![\lambda x.P]\!]_\Delta \circ <[\![\vec{N}]\!]_\Gamma>
\end{aligned}$$

(*3*) Il suffit d'appliquer (*2*) à $\Delta' = \Delta, x$ et à Γ (avec $\{\Delta\} = \{\vec{x}_\Delta\}$):

$$\begin{aligned}
[\![M[N/x]\lceil N]\!]_\Gamma &= [\![M[N/x]]\!]_\Gamma\lceil[\![N]\!]_\Gamma \\
\text{par définition de } <\ >: &= [\![M[\vec{x}_\Delta/\vec{x}_\Delta, N/x]]\!]_\Gamma\lceil <[\![N]\!]_\Gamma> \\
\text{par } (2) &= [\![M]\!]_\Delta \circ <[\![\vec{x}_\Delta]\!]_\Gamma, [\![N]\!]_\Gamma> \lceil[\![N]\!]_\Gamma \\
\text{par définition de } \Pi^\Gamma_\Delta \text{ et par les règles sur } \lceil &= [\![M]\!]_\Delta \circ <\Pi^\Gamma_\Delta, [\![N]\!]_\Gamma>
\end{aligned}$$

□

Remarque 24 $[\![M]\!]_\rho$ *se définit par* $[\![M]\!]_\rho = [\![M]\!]_\Delta\circ\rho^\Delta = [\![M]\!]_\Delta \circ < \rho(x_1), ..., \rho(x_n) >$ *indépendamment de* Δ *pourvu que* $VL(M) \subset \Delta$.

Preuve : En effet d'après le point (*1*) du lemme 70, si $VL(M) \subset \{\Gamma\} \subset \{\Delta\}$, alors

$$\begin{aligned}
[\![M]\!]_\Delta\circ\rho^\Delta &= [\![M]\!]_\Gamma\circ\Pi^\Delta_\Gamma\circ\rho^\Delta \\
&= [\![M]\!]_\Gamma\circ(\rho^\Gamma\lceil\rho^\Delta) \\
\rho^\Delta \text{ étant totale} &= [\![M]\!]_\Gamma\circ\rho^\Gamma
\end{aligned}$$

□

Théorème 17 *Si* $VL(M)\cup VL(N) \subset \{\Delta\}$ *alors* $\lambda_p \vdash M = N \Longrightarrow [\![M]\!]_\Delta = [\![N]\!]_\Delta$
Si de plus $G\circ F = Id_U$, *alors* $\lambda_p + \eta_p \vdash M = N \Longrightarrow [\![M]\!]_\Delta = [\![N]\!]_\Delta$

Preuve : Par induction sur la preuve de $\lambda_p \vdash M = N$:

$M\lceil x = M$, $[\![x]\!]_\Delta = \pi^\Delta_x$ est totale, d'où $[\![M\lceil x]\!]_\Delta = [\![M]\!]_\Delta\lceil\pi^\Delta_x = [\![M]\!]_\Delta$.

$M\lceil c = M$, $[\![c]\!]_\Delta = a_c \circ !_{U\Delta}$ est la composée de deux flèches totales et est donc totale, on conclut donc comme ci-dessus.

$M\lceil(\lambda x.P) = M$, $[\![\lambda x.P]\!]_\Delta = G\circ\Lambda([\![P]\!]_{\Delta,x})$ est totale puisque G et $\Lambda([\![P]\!]_{\Delta,x})$ le sont, on conclut comme dans les deux cas précédents.

$M\lceil M = M$, résulte directement de l'équation (6.1).

$P(Q\lceil N) = (PQ)\lceil N$, résulte directement de l'équation (6.7).

$(P\lceil N)Q = (PQ)\lceil N$, résulte directement de l'équation (6.6).

$P\lceil(Q\lceil N) = P\lceil(N\lceil Q)$, résulte directement de l'équation (6.3).

$P\lceil(Q\lceil N) = (P\lceil N)\lceil Q$, résulte directement de l'équation (6.2).

$\beta : (\lambda x.P)Q = P[Q/x]\lceil Q$,

$$
\begin{aligned}
[\![(\lambda x.P)Q]\!]_\Delta &= G\circ\Lambda([\![P]\!]_{\Delta,x})) \cdot_{U\Delta} [\![Q]\!]_\Delta \\
&= App \circ \langle G\circ\Lambda([\![P]\!]_{\Delta,x}), [\![Q]\!]_\Delta \rangle \\
&= ev \circ (F \times Id_U) \circ \langle G\circ\Lambda([\![P]\!]_{\Delta,x}), [\![Q]\!]_\Delta \rangle \\
&= ev \circ \langle F\circ Fst, Snd \rangle \circ \langle G\circ\Lambda([\![P]\!]_{\Delta,x}), [\![Q]\!]_\Delta \rangle \\
&= ev \circ \langle (F\circ G\circ\Lambda([\![P]\!]_{\Delta,x}))\lceil [\![Q]\!]_\Delta, [\![Q]\!]_\Delta \lceil G\circ\Lambda([\![P]\!]_{\Delta,x}) \rangle \\
\text{par les règles} \quad &= ev \circ \langle F\circ G\circ\Lambda([\![P]\!]_{\Delta,x}), [\![Q]\!]_\Delta \rangle \\
&= ev \circ \langle \Lambda([\![P]\!]_{\Delta,x}) \circ Id, Id \circ [\![Q]\!]_\Delta \rangle \\
&= ev \circ \Lambda([\![P]\!]_{\Delta,x}) \times Id) \circ \langle Id_{U\Delta}, [\![Q]\!]_\Delta \rangle \\
\text{par (6.14)} \quad &= [\![P]\!]_{\Delta,x} \circ \langle \Pi^\Delta_\Delta, [\![Q]\!]_\Delta \rangle \\
\text{par le lemme 70.(3)} \quad &= [\![P[Q/x]\lceil Q]\!]_\Delta
\end{aligned}
$$

Les règles d'inférence se vérifient sans problème.

Si de plus $G\circ F = Id_U$, alors

$$
\begin{aligned}
[\![(\lambda x.Mx)\lceil M]\!]_\Delta &= G\circ\Lambda([\![Mx]\!]_{\Delta,x})\lceil [\![M]\!]_\Delta \\
&= G\circ\Lambda([\![M]\!]_{\Delta,x} \cdot_{U^{\Delta,x}} [\![x]\!]_{\Delta,x})\lceil [\![M]\!]_\Delta \\
&= G\circ\Lambda(ev \circ (F \times Id) \circ \langle [\![M]\!]_{\Delta,x}, [\![x]\!]_{\Delta,x} \rangle)\lceil [\![M]\!]_\Delta
\end{aligned}
$$

Remarquons que $[\![M]\!]_{\Delta,x} = [\![M]\!]_\Delta \circ \Pi^{\Delta,x}_\Delta$ par le lemme 70, et $\Pi^{\Delta,x}_\Delta = Fst$ par la remarque 23, d'où $\langle [\![M]\!]_{\Delta,x}, [\![x]\!]_{\Delta,x} \rangle = [\![M]\!]_\Delta \times Id$, donc:

$$
\begin{aligned}
[\![(\lambda x.Mx)\lceil M]\!]_\Delta &= G\circ\Lambda(ev \circ (F \times Id) \circ ([\![M]\!]_\Delta \times Id))\lceil [\![M]\!]_\Delta \\
\text{par le lemme 64} \quad &= G\circ\Lambda(ev \circ (F\circ [\![M]\!]_\Delta) \times Id)\lceil [\![M]\!]_\Delta \\
&= G\circ\Lambda(ev \circ (F\circ [\![M]\!]_\Delta) \times Id)\lceil G\circ F\circ [\![M]\!]_\Delta \\
&= G\circ\Lambda(ev \circ (F\circ [\![M]\!]_\Delta) \times Id)\lceil G\circ F\circ [\![M]\!]_\Delta \lceil F\circ [\![M]\!]_\Delta \\
&= G\circ\Lambda(ev \circ (F\circ [\![M]\!]_\Delta) \times Id)\lceil [\![M]\!]_\Delta \lceil F\circ [\![M]\!]_\Delta \\
&= G\circ\Lambda(ev \circ (F\circ [\![M]\!]_\Delta) \times Id)\lceil F\circ [\![M]\!]_\Delta \\
\text{par l'équation 6.15} \quad &= G\circ F\circ [\![M]\!]_\Delta \\
&= [\![M]\!]_\Delta
\end{aligned}
$$

□

6.6 Une PCCC avec un objet réflexif fournit une λ_p-algèbre

Le but de cette partie est de montrer qu'on peut extraire d'une PCCC munie d'un objet réflexif non seulement une structure applicative, mais mieux encore, une λ_p-algèbre .

Rappelons que si $F \circ G = Id_{U^U}$ alors G est totale.

Théorème 18 *Soit* $\mathbf{C}$ *une* PCCC *munie d'un objet réflexif* U, *avec* $F : U \to U^U$ *et* $G : U^U \to U$ *telles que* $F \circ G = Id_{U^U}$. *Alors* $\mathbf{C}$ *fournit une algèbre combinatoire avec éléments partiels , que nous noterons* $\mathcal{M}(\mathbf{C}, U, F, G)$, *en posant:*

- $S = [\![\lambda xyz.xz(yz)]\!]^{\mathbf{C}}$.
- $K = [\![\lambda xy.x]\!]^{\mathbf{C}}$.
- $I = [\![\lambda x.x]\!]^{\mathbf{C}}$.
- $ab = a._U b = ev_{U,U} \circ (F \times Id_U) \circ < a, b >$.
- $a \Downarrow$ *ssi* $a \in U_T$.

Nous allons d'abord établir quelques lemmes utiles pour la démonstration de ce théoreme:

Lemme 71 *Soit* $h : U^\Delta \to U$, *alors* $(a._{U^\Delta} b) \circ h = (a \circ h)._U (b \circ h)$

Preuve:

$$\begin{aligned} (a._{U^\Delta} b) \circ h &= ev_{U,U} \circ (F \times Id_U) \circ < a, b > \circ h \\ &= ev_{U,U} \circ (F \times Id_U) \circ < a \circ h, b \circ h > \\ &= (a \circ h)._U (b \circ h) \end{aligned}$$

□

Lemme 72 *La restriction dans* $\mathcal{M}(\mathbf{C}, U, F, G)$, *coïncide avec la restriction dans la catégorie; à savoir* $t \lceil_{\mathcal{M}} u = (K._U t)._U u = t \lceil_{\mathbf{C}} u$

Preuve:

$$\begin{aligned} (K._U t)._U u &= ([\![\lambda xy.x]\!]_x \circ < Id, t >)._U u \\ &= [\![x]\!]_{x,y} \circ << Id, t >, u > \\ &= Snd \circ Fst \circ << Id, t >, u > \\ &= t \lceil_{\mathbf{C}} u \end{aligned}$$

□

Lemme 73 *1.* $[\![\lambda x.P]\!]_{\Delta \cdot U^\Delta} u = [\![P]\!]_{\Delta,x} \circ < Id_{U^\Delta}, u >$

2. $(..(([\![\lambda x_1..x_n.P]\!]_\Delta .t_1).t_2)..).t_n = [\![P]\!]_{\Delta,x_1,..,x_n} \circ << .. << Id, t_1 >, t_2 >, .. >, t_n >$

Preuve:

1.

$$\begin{aligned} [\![\lambda x.P]\!]_{\Delta \cdot U^\Delta} u &= ev_{U,U} \circ (F \times Id_U) \circ < G \circ \Lambda([\![P]\!]_{\Delta,x}), u > \\ &= ev_{U,U} \circ < F \circ G \circ \Lambda([\![P]\!]_{\Delta,x}), u > \\ &= ev_{U,U} \circ < \Lambda([\![P]\!]_{\Delta,x}), u > \\ &= ev_{U,U} \circ (\Lambda([\![P]\!]_{\Delta,x}) \times Id_U) \circ < Id_{U^\Delta}, u > \\ &= [\![P]\!]_{\Delta,x} \circ < Id_{U^\Delta}, u > \end{aligned}$$

2. Généralisation évidente de *1.* □

Preuve du théorème 18:

Vérifions que les axiomes et les règles d'inférence d'une algèbre combinatoire avec éléments partiels sont satisfaites.

- Les règles concernant l'égalité sont évidemment satisfaites.
- $S\Downarrow, K\Downarrow, I\Downarrow$ $[\![\lambda x.P]\!]_\Delta = G\circ\Lambda([\![P]\!]_{\Delta,x})$ est une flèche totale puisque c'est la composée de deux flèches totales. Ceci garantit que les axiomes $S\Downarrow, K\Downarrow, I\Downarrow$ sont satisfaits, puisque par définition S, K, I sont de la forme $[\![\lambda x.P]\!]_\Delta$.
- $Stuv = tv(uv)$:

$$\begin{aligned}
&((S.t).u).v\\
&\quad = ((\Lambda([\![xz(yz)]\!]_{x,y,z}).t).u).v\\
&\text{par le lemme 73}\\
&\quad = [\![xz(yz)]\!]_{x,y,z}\circ <<<Id,t>,u>,v>\\
&\quad = (([\![x]\!]_{x,y,z}.[\![z]\!]_{x,y,z}).([\![y]\!]_{x,y,z}.[\![z]\!]_{x,y,z}))\circ <<<Id,t>,u>,v>\\
&\text{par le lemme 71}\\
&\quad = (([\![x]\!]_{x,y,z}\circ <<<Id,t>,u>,v>).([\![z]\!]_{x,y,z}\circ <<<Id,t>,u>,v>))\\
&\qquad .(([\![y]\!]_{x,y,z}\circ <<<Id,t>,u>,v>).([\![z]\!]_{x,y,z}\circ <<<Id,t>,u>,v>))\\
&\quad = ((t\lceil\{u,v\}).(v\lceil\{t,u\})).((u\lceil\{t,v\}).(v\lceil\{t,u\}))\\
&\text{par les équations portant sur la restriction}\\
&\quad = (t.v).(u.v)
\end{aligned}$$

- $Iu = u$:

$$\begin{aligned}
&[\![\lambda x.x]\!]_{\Delta.U}u\\
\text{par le lemme 71}\quad &= [\![x]\!]_x\circ <Id,u>\\
&= Snd\circ <Id,u>\\
&= u
\end{aligned}$$

- Axiomes et règles d'inférence portant sur la restriction:

  Le lemme 72 montre que la restriction dans l'algèbre combinatoire avec éléments partiels coïncide avec la restriction dans la catégorie. Du coup toutes les règles concernant la restriction sont quasi-évidentes, traitons par exemple la règle $t._U(u\lceil v) = (t._U u)\lceil v$:

$$\begin{aligned}
t._U(u\lceil v) &= ev\circ(F\times Id)\circ <t,u\lceil v>\\
&= ev\circ(F\times Id)\circ <t,u>\lceil v\\
&= (t._U u)\lceil v
\end{aligned}$$

- $t\Downarrow, u\Downarrow \Longrightarrow Kt\Downarrow, St\Downarrow, Stu\Downarrow$ le lemme 73 permet de vérifier les égalités suivantes:

$$\begin{aligned}
K._U t &= [\![\lambda xy.x]\!]_x\circ <Id,t>\\
S._U t &= [\![\lambda yz.xz(yz)]\!]_x\circ <Id,t>\\
(S._U t)._U u &= [\![\lambda z.xz(yz)]\!]_{x,y}\circ <<Id,t>,u>
\end{aligned}$$

  Or, si t et u sont des flèches totales, il en va de de même de $<Id,t>$ et de $<<Id,t>,u>$. Enfin la composée de flèches totales étant totales, il s'ensuit que $Kt\Downarrow, St\Downarrow, Stu\Downarrow$.

- $tu \Downarrow \Longrightarrow t\Downarrow, u\Downarrow$:

  $t._U u = ev_{U,U} \circ (F \times Id_U) \circ \langle t, u \rangle$, d'où

  $$! = !\circ(t._U u) = !\circ ev_{U,U} \circ (F \times Id_U) \circ \langle t, u \rangle$$

  Or $!\circ ev_{U,U} \circ (F \times Id_U) \leq !$ par maximalité de !, et ainsi, la composition étant croissante: $! \leq !\circ \langle t, u \rangle$.

  $\langle t, u \rangle$ est donc une flèche totale, et il en va donc de même pour t et u (cf ??). □

Théorème 19 *$\mathcal{A} = \mathcal{M}(\mathbf{C}, U, F, G)$ est une λ_p-algèbre .*
Si de plus $G \circ F = Id$ alors $\mathcal{A}$ est une $\lambda\eta_p$-algèbre.

Preuve: Pour éviter toute confusion, notons $[\![M]\!]^{\mathbf{C}}_\rho$, l'interprétation de M telle que nous l'avons définie dans la section 6.5 et $[\![M_{CL}]\!]_{\mathcal{A}}$ l'interprétation de M dans l'algèbre combinatoire $\mathcal{A}$.

Utilisons le lemme de caractérisation 49:

Le premier point de ce lemme est immédiat à vérifier puisque les égalités $S_{\lambda,CL} = S$ et $K_{\lambda,CL} = K$ résultent de la définition de K et de S dans notre algèbre.

Nous avons constaté dans le théorème 17 que, pour tous termes clos M et N:

$$\lambda_p \vdash M = N \Longrightarrow [\![M]\!]^{\mathbf{C}}_\rho = [\![N]\!]^{\mathbf{C}}_\rho$$

(resp. si $G \circ F = Id$, $\lambda + \eta_p \vdash M = N \Longrightarrow [\![M]\!]^{\mathbf{C}}_\rho = [\![N]\!]^{\mathbf{C}}_\rho$).

Pour vérifier le deuxième point du lemme 49, il nous suffit par conséquent de montrer que, pour tout terme clos M, $[\![M]\!]^{\mathbf{C}}_\rho = [\![M_{CL}]\!]^{\rho}_{\mathcal{A}}$.

Or, d'après le lemme 48, $\lambda_p \vdash M_{CL,\lambda} = M$ et, d'autre part, en vertu du théorème 17, $[\![M]\!]^{\mathbf{C}}_\rho = [\![M_{CL,\lambda}]\!]^{\mathbf{C}}_\rho$. Il nous suffit donc de vérifier que $[\![M_{CL}]\!]^{\rho}_{\mathcal{A}} = [\![M_{CL,\lambda}]\!]^{\mathbf{C}}_\rho$ et nous allons montrer en fait quelque chose de plus fort, à savoir que pour tout terme u dans l'algèbre combinatoire avec éléments partiels engendrée par S, K, I, on a $[\![u]\!]^{\rho}_{\mathcal{A}} = [\![u_\lambda]\!]^{\mathbf{C}}_\rho$. Nous le montrons par induction sur u:

c'est évident par définition pour $u = S, K, I$, puisque l'interprétation ne dépend pas de ρ.

Si $u = vw$,

$$\begin{array}{rcl}
[\![u]\!]^{\rho}_{\mathcal{A}} & = & [\![v]\!]^{\rho}_{\mathcal{A}} \cdot_U [\![w]\!]^{\rho}_{\mathcal{A}} \\
& \overset{induction}{=} & [\![v_\lambda]\!]^{\mathbf{C}}_\rho \cdot_U [\![w_\lambda]\!]^{\mathbf{C}}_\rho \\
& = & [\![v_\lambda]\!]_\Delta \circ \rho^\Delta \cdot_U [\![w_\lambda]\!]_\Delta \circ \rho^\Delta \\
& = & ev_{U,U} \circ (F \times Id_U) \circ \langle [\![v_\lambda]\!]_\Delta \circ \rho^\Delta, [\![w_\lambda]\!]_\Delta \circ \rho^\Delta \rangle \\
& = & ev_{U,U} \circ (F \times Id_U) \circ \langle [\![v_\lambda]\!]_\Delta, [\![w_\lambda]\!]_\Delta \rangle \circ \rho^\Delta \\
& = & [\![u_\lambda]\!]_\Delta \circ \rho^\Delta \\
& = & [\![u_\lambda]\!]^{\mathbf{C}}_\rho
\end{array}$$

□

6.7 PCCC construite à partir d'une λ_p-algèbre

Soit $\mathcal{A}$ une λ_p-algèbre. Nous appellerons $\mathcal{U}$ la sous-λ_p-algèbre de $\mathcal{A}$ engendrée par S, K, I. Comme tout terme clos du λ_p-calcul se traduit en un terme de cette algèbre, nous adopterons une λ-notation: un terme de $\mathcal{U}$ sera désigné par le terme dont il est la traduction. Remarquons que dans une telle λ_p-algèbre tous les termes sont traductions d'un terme clos de λ_p. Ainsi I pourra être noté $\lambda x.x$, puisque $(\lambda x.x)_{CL} = I$. Comme nous sommes dans une λ_p-algèbre, nous serons fondés à effectuer les calculs comme si nous travaillions dans le λ_p-calcul.

Une intuition utile pour comprendre ce qui suit est de penser qu'un terme a va servir à représenter implicitement l'ensemble des termes M vérifiant $aM = M$, ce qui permet d'exprimer que ces termes M sont d'une certaine forme. Cette intuition permet de reconstruire ce qui suit, aussi bien que de comprendre la technique qui a dû vraisemblablement inspiré Scott pour obtenir un résultat analogue dans le λ-calcul pur (ce que nous appelons aussi "le cas total"), dont Barendregt attribue la paternité à Karoubi. Pour cette construction, une bonne référence est [Lambek& Scott].

Lemme 74 *Pour $a, b \in \mathcal{U}$, posons $a \circ b = \lambda z.a(bz)$.*
$\mathcal{U}$ permet de construire une catégorie $\mathbf{C}(\mathcal{U})$ comme suit:

- *les objets de $\mathbf{C}(\mathcal{U})$ sont les termes idempotents $a \in \mathcal{U}$, à savoir ceux qui vérifient $a \circ a = a$. Remarquons que a idempotent $\Longrightarrow a \Downarrow$, puisqu'alors a est la traduction d'un terme total.*
- *Les flèches de $a \to b$ sont les termes f tels que $b \circ f \circ a = f$.*
- *La composée de deux flèches représentées par f et g est représentée par le terme $f \circ g$ (d'où la notation).*
- *$Id_a = a$.*

Preuve: Il ne s'agit ici que de vérifications immédiates et que nous omettons, elles ne diffèrent pas du cas total. □

Remarque 25 *Remarquons que si f est un terme tel que $b \circ f \circ a = f$, alors $f = \lambda z.b(f(az))$, et donc $\lambda t.ft = \lambda t.(\lambda z.b(f(az)))t = \lambda t.b(f(at)) = f$.*

Le théoreme suivant est l'homologue de celui établi par Scott dans le cas du λ-calcul pur ([Scott79]).

Théorème 20 *On définit une* PCCC $\mathbf{C}(\mathcal{U})$, *en posant:*

- $T = KI = \lambda z.I$
- $\pi_1 = \lambda u.u(\lambda vw.v)$
- $\pi_2 = \lambda u.u(\lambda vw.w)$
- $!_a = \lambda y.I\lceil ay$
- $[a, b] = (\lambda y.yab)\lceil a, b$
- $< f, g >= \lambda z.[fz, gz]$
- $Fst_{a,b} = a \circ \pi_1$
- $Snd_{a,b} = b \circ \pi_2$
- $a \times b = \lambda z.[a(\pi_1 z), b(\pi_2 z)]$
- $b^a = \lambda z.bozoa$
- $ev_{a,b} = \lambda z.b((\pi_1 z)(a(\pi_2 z)))$
- $\Lambda_{a,b,c}(f) = \lambda x.(\lambda y.f[x, y])\lceil ax$

Preuve: Les vérifications nécessaires pour s'assurer que les flèches que nous venons de définir appartiennent aux "Hom" où elles doivent se trouver sont directes, vérifions par exemple $\Lambda_{a,b,c}(f) \in a \to c^b$, en effet:

$$\begin{aligned} c^b \circ \Lambda_{a,b,c}(f) \circ a &= \lambda x.(\lambda z.c \circ z \circ b)((\lambda x.(\lambda y.f[x,y])\lceil ax)ax) \\ &= \lambda x.(\lambda z.c \circ z \circ b)(\lambda y.f[ax,y])\lceil a(ax)) \\ &= \lambda x.c \circ (\lambda y.f[ax,y]) \circ b\lceil ax \\ &= \lambda x.(\lambda z.c((\lambda y.f[ax,y])(bz)))\lceil ax \\ &= \lambda x.(\lambda z.c(f[ax,bz])\lceil ax \end{aligned}$$

enfin $fu = (c \circ f \circ (a \times b))u = c(f[a(\pi_1 u), b(\pi_2 u)])$, d'où $f[x,z] = c(f[ax,bz])$, et par conséquent:

$\lambda x.\lambda z.c(f[ax,bz])\lceil ax = \lambda x.(\lambda z.f[x,z])\lceil ax = \Lambda_{a,b,c}(f)$.

Rappelons que $f\lceil g = Fst_{a,b} \circ < f,g >$.

Etablissons quelques lemmes et résultats qui vont simplifier la suite:

$$\begin{aligned} \pi_1[a,b] &= \lambda u.u(\lambda vw.v)((\lambda y.yab)\lceil a,b) \\ &= (\lambda y.yab)(\lambda vw.v)\lceil a,b \\ &= (\lambda vw.v)ab\lceil a,b \\ &= (\lambda vw.v)ab \\ &= a\lceil b \end{aligned}$$

De même on prouve $\pi_1[a,b] = b\lceil a$.

Lemme 75 *Si* $f : c \to a, g : c \to b$, *alors* $f\lceil g = \lambda t.ft\lceil gt$

Preuve:

$$\begin{aligned} f\lceil g &= (a \circ \lambda u.u(\lambda vw.v)) \circ (\lambda z.[fz,gz]) \\ &= \lambda t.(a \circ \lambda u.u(\lambda vw.v))[ft,gt] \\ &= \lambda t.a([ft,gt](\lambda vw.v))\lceil\{ft,gt\} \\ &= \lambda t.a((\lambda y.y(ft)(gt))(\lambda vw.v))\lceil\{ft,gt\} \\ &= \lambda t.a(ft)\lceil gt \end{aligned}$$

Remarquons enfin que par hypothèse $f = a \circ f \circ c$, d'où $ft = a(f(ct))$ et donc $a(ft) = a(a(f(ct)) = (a \circ a)(f(ct)) = a(f(ct)) = ft$, puisque a est idempotent. □

Soient $f : a \to c, g : b \to d$, alors:

$$\begin{aligned} f \times g &= < f \circ Fst_{a,b}, g \circ Snd_{a,b} > \\ &= \lambda z.[f(Fstz), g(Sndz)] \\ &= \lambda z.[f(a(\pi_1 z)), g(b(\pi_2 z))] \end{aligned}$$

Pour établir le théorème 20, nous allons utiliser la présentation équationnelle des PCCC établie par Curien et Obtulówicz que nous avons déjà introduite dans la section 6.3.

6.8 Vérification de la présentation équationnelle:

1. $f\lceil f = f$

$$\begin{aligned} f\lceil f &= \lambda a.fa\lceil fa \\ \text{par les règles} &= \lambda a.fa \\ \text{par la remarque 25} &= \lambda a.fa = f \end{aligned}$$

2. $f\lceil(g\lceil h) = f\lceil(h\lceil g)$

$$\begin{aligned} f\lceil(g\lceil h) &= \lambda z.fz\lceil(gz\lceil hz) \\ &= \lambda z.(fz\lceil gz)\lceil hz \\ &= f\lceil(h\lceil g) \end{aligned}$$

3. $f\lceil(g\lceil h) = (f\lceil g)\lceil h$ Analogue au cas précédent.

4. $f\lceil(g\lceil h) = f\lceil < g, h >$

$$\begin{aligned} f\lceil < g,h > &= \lambda z.fz\lceil[gz, hz] \\ &= \lambda z.fz\lceil((\lambda yy(gz)(hz))\lceil\{gz, hz\} \\ &= \lambda z.fz\lceil\{gz, hz\} \\ &= f\lceil(g\lceil h) \end{aligned}$$

5. $(f \circ g)\lceil h = f \circ (g\lceil h)$

$$\begin{aligned} (f\circ g)\lceil h &= \lambda z.f(gz)\lceil hz \\ &= \lambda z.f(gz\lceil hz) \\ &= f\circ(g\lceil h) \end{aligned}$$

6. $< f, g > \lceil h = < f\lceil h, g >$

$$\begin{aligned} < f\lceil h, g > &= \lambda z.[fz\lceil hz, gz] \\ &= \lambda z.\lambda y.y(fz\lceil hz)(gz)\lceil\{fz\lceil hz, gz\} \\ &= \lambda z.\lambda y.y(fz\lceil hz)(gz)\lceil\{fz, gz, hz\} \\ &= \lambda z.\lambda y.y(fz)(gz)\lceil\{fz, gz\}\lceil hz \\ &= \lambda z.[fz, gz]\lceil hz \\ &= < f, g > \lceil h \end{aligned}$$

7. $< f, g > \lceil h = < f, g\lceil h >$ Analogue au cas précédent.

8. $< f, g > \circ h = < f\circ h, g\circ h >$

$$\begin{aligned} < f, g > \circ h &= \lambda z.(\lambda y[fy, gy])(hz) \\ &= \lambda z.[(f\circ h)z, (g\circ h)z] \\ &= < f\circ h, g\circ h > \end{aligned}$$

9. $< Fst, Snd >= Id$. Soit $< Fst_{a,b}, Snd_{a,b} >= Id_a \times b$

$$\begin{aligned} < Fst_{a,b}, Snd_{a,b} > &= \lambda z.[a(\pi_1 z), b(\pi_2 z)] \\ &= a \times b = Id_a \times b \end{aligned}$$

10. $Fst \circ \langle f, g \rangle = Snd \circ \langle g, f \rangle$ D'après le lemme 75, $Fst \circ \langle f, g \rangle = f \lceil g = \lambda f z \lceil g z$. Un calcul analogue à la preuve de ce lemme montre que l'on trouve la même chose pour $Snd \circ \langle g, f \rangle$.

11. $Id \lceil ! = Id$. Soit $Id_a \lceil !_a = Id_a$

$$\begin{aligned} Id_a \lceil !_a &= a \lceil (\lambda y. I \lceil a y) \\ &= \lambda z. a z \lceil (I \lceil a z) \\ &= \lambda z. a z \\ \text{par la remarque 25} \quad &= a = Id_a \end{aligned}$$

12. $! \lceil f = f$

$f : a \to t$ et donc $f = KI \circ f \circ a$ d'où $f = \lambda t. KI(f(at)) = \lambda t. I \lceil f(at)$ et donc $ft = I \lceil f(at)$, il s'ensuit donc:

$$\begin{aligned} ! \lceil f &= \lambda t. KI((\lambda y. I \lceil a y)t) \lceil ft \\ &= \lambda t. KI(I \lceil at) \lceil ft \\ &= \lambda t. I \lceil \{at, ft\} \\ &= \lambda t. I \lceil \{at, I \lceil f(at)\} \\ &= \lambda t. I \lceil f(at) = f \end{aligned}$$

13. $! \circ \Lambda(f) = !$ Comme $\Lambda(f) : a \to c^b$, il s'agit plus précisément de vérifier $!_{b^c} \circ \Lambda(f) = !_a$

$$\begin{aligned} !_{b^c} \circ \Lambda(f) &= \lambda z. !_{c^b}((\Lambda(f))z) \\ &= \lambda z. (\lambda y. I \lceil (c^b) y)((\lambda x. (\lambda y. f[ax, by]) \lceil ax)z) \\ &= \lambda z. (\lambda y. I \lceil \{(\lambda z. b \circ z \circ c)y)(\lambda y. f[z, y]), az\} \\ &= \lambda z. (\lambda y. I \lceil \{b \circ y \circ c)(\lambda y. f[z, y]), az\} \\ &= \lambda z. I \lceil \{b \circ (\lambda y. f[z, y]) \circ c, az\} \\ b \circ (\lambda y. f[z, y]) \circ c \text{ étant total} \quad &= \lambda z. I \lceil az \\ &= !_a \end{aligned}$$

14. $ev \circ (\Lambda(f) \times Id) = f$

$f : (a \times b) \to c$ et il s'agit plus précisément de vérifier $ev_{b,c} \circ (\Lambda_{a,b,c}(f) \times Id_b) = f$. Remarquons que $f : (a \times b) \to c$ entraîne $fz = (c \circ f \circ (a \times b))z = c(f[a(\pi_1 z), b(\pi_2 z)]$.

$$\begin{aligned} ev_{b,c} \circ (\Lambda_{a,b,c}(f) \times Id_b) &= \lambda z. ev_{b,c}[(\lambda y. f[a(\pi_1 z), y]) \lceil a(\pi_1 z), b(\pi_2 z)]) \\ &= \lambda z. (\lambda z. c(\pi_1 z)(b(\pi_2 z)))[(\lambda y. f[a(\pi_1 z), y]) \lceil a(\pi_1 z), b(\pi_2 z)] \\ &= \lambda z. c((\lambda y. f[a(\pi_1 z), y])(b(\pi_2 z))) \lceil a(\pi_1 z) \\ &= \lambda z. c(f[a(\pi_1 z), b(\pi_2 z)] \\ &= \lambda z. f z \\ &= f \end{aligned}$$

15. $\Lambda(ev \circ (f \times Id)) \lceil f = f$

$f : a \to c^b$ et il s'agit plus précisément de vérifier $\Lambda_{a,b,c}(ev_{b,c} \circ (f \times Id_b)) \lceil f = f$.

$$\begin{aligned} ev_{b,c} \circ (f \times Id_b) &= \lambda z. (\lambda x. c(\pi_1 x(b(\pi_2 x)))[f(a(\pi_1 z)), b(\pi_2 z)] \\ &= \lambda z. c(f(a(\pi_1 z))(b(\pi_2 z))) \end{aligned}$$

$$\begin{aligned} ev_{b,c}\circ(f \times Id_b)[x,y] &= (\lambda z.c(f(a(\pi_1 z))(b(\pi_2 z))))[x,y] \\ &= c(f(ax)(by)) \end{aligned}$$

d'où

$$\begin{aligned} \Lambda_{a,b,c}(ev_{b,c}\circ(f \times Id_b)) &= \lambda x.(\lambda y.(ev_{b,c}\circ(f \times Id_b))[x,y])\lceil ax \\ &= \lambda x.(\lambda y.c(f(ax)(by)))\lceil ax \\ &= \lambda x.(\lambda y.fxy)\lceil ax \end{aligned}$$

puisque $f : a \to c^b$, $f = c^b \circ f \circ a$, et que, par conséquent, $f = \lambda xy.c(f(ax)(by))$. Enfin

$$\begin{aligned} \Lambda(ev\circ(f \times Id))\lceil f &= \lambda t.(\lambda x.(\lambda y.fxy)\lceil ax)t\lceil ft \\ &= \lambda t.(\lambda y.fty)\lceil at\lceil ft \\ &= \lambda x.(\lambda y.fxy)\lceil fx\lceil at \end{aligned}$$

Or $f = \lambda xy.c(f(ax)(by))$, d'où $ft = \lambda y.c(f(at)(by))$ et par la remarque 25 $\lambda y.fty = ft$

$$\begin{aligned} \Lambda(ev\circ(f \times Id))\lceil f &= \lambda t.ft\lceil at \\ &= \lambda t.c^b(f(at))\lceil at \\ &= \lambda t.c^b(f(at)) \\ &= \lambda t.ft \\ \text{par la remarque 25} &= f \end{aligned}$$

□

Remarque 26 *$f : t \to I$ est totale si $\lambda z.I\lceil fz = I$.*

Preuve: $t = KI$, donc $f : t \to I$ est totale si $!_I \circ f = !_t$, or $!_I = \lambda y.I\lceil Iy = KI$ et $!_t = \lambda y.I\lceil ty = \lambda y.I\lceil KIy = I$, c'est-à-dire f est totale si $\lambda z.KI(fz) = \lambda z.I\lceil fz = I$. □

Posons $\mathbf{1} = \lambda zy.zy$

Lemme 76 *I est un objet réflexif dans $\mathbf{C}(\mathcal{U})$ via $F = G = \mathbf{1}$.*

Remarquons que $F\circ G = G\circ F = \mathbf{1}$. D'où $F\circ G = Id_{\mathbf{1}}$. De plus, si $\mathcal{A}$ est une $\lambda_p + \eta_p$-algèbre, alors $\mathbf{1} = \lambda zy.zy = \lambda z.z = I = Id_I$

La vérification en est sans problème, remarquons:

$$\begin{aligned} I^I &= \lambda z.I\circ z\circ I \\ &= \lambda z.\lambda y.I(z(Iy)) \\ &= \lambda zy.zy \\ &= \mathbf{1} \end{aligned}$$

□

6.9 D'une λ_p-algèbre à une PCCC et retour

Nous allons à présent établir des résultats qui relèvent au cas partiel ceux de Koymans dans le cas total (cf [Barendregt, Koymans 82]), à savoir qu'une λ_p-algèbre permet de construire une PCCC avec objet réflexif, qui à son tour fournit une λ_p-algèbre qui n'est pas sans rapport avec celle dont nous sommes partis. Donnons tout d'abord une définition préliminaire:

Definition 25 *Un morphisme ϕ d'algèbre combinatoire avec éléments partiels est un morphisme de structure applicative qui envoie les définis sur des définis ($a \Downarrow \Longrightarrow \phi(a) \Downarrow$).*

Le lemme et le théorème suivants sont les homologues dans le cas partiel de ceux établis par Koymans dans le cas classique (cf [Koymans 82]).

Lemme 77 *Soit* $M \in \lambda_p$, *alors* $[\![M]\!]^{\mathbf{C}(\mathcal{U})}_{\vec{x}} = \lambda z.M[x_1, \cdots, x_n := \pi^{\vec{x}}_{x_1} z, \cdots, \pi^{\vec{x}}_{x_n} z]$

Preuve: Par induction sur M:

- $M = x_i$:

$$\begin{array}{rcl} [\![M]\!]_{\vec{x}} & = & \pi^{\vec{x}}_{x_i} \\ \pi^{\vec{x}}_{x_i} \text{ étant une abstraction} & = & \lambda z.\pi^{\vec{x}}_{x_i} z \\ & = & \lambda z.x_i[x_1, \cdots, x_n := \pi^{\vec{x}}_{x_1} z, \cdots, \pi^{\vec{x}}_{x_1} z] \end{array}$$

- $M = PQ$:

$$\begin{array}{rcl} [\![M]\!]_{\vec{x}} & = & [\![P]\!]_{\vec{x}} \cdot_{U^{\vec{x}}} [\![Q]\!]_{\vec{x}} \\ & = & \lambda z.[\![P]\!]_{\vec{x}} z([\![Q]\!]_{\vec{x}} z) \\ \text{par induction} & = & \lambda z.P[x_1, \cdots, x_n := \pi^{\vec{x}}_{x_1} z, \cdots, \pi^{\vec{x}}_{x_1} z] Q[x_1, \cdots, x_n := \pi^{\vec{x}}_{x_1} z, \cdots, \pi^{\vec{x}}_{x_1} z] \\ & = & \lambda z.M[x_1, \cdots, x_n := \pi^{\vec{x}}_{x_1} z, \cdots, \pi^{\vec{x}}_{x_1} z] \end{array}$$

- $M = \lambda y.P$:

$$\begin{array}{rcl} [\![M]\!]_{\vec{x}} & = & G \circ \Lambda([\![P]\!]_{\vec{x},y}) \\ [\![P]\!]_{\vec{x},y} : I^n \times I \to I & = & 1 \circ (\lambda a.(\lambda b.[\![P]\!]_{\vec{x},y}[a,b]) \lceil I^n a) \\ & = & (\lambda uv.uv) \circ (\lambda a.(\lambda b.[\![P]\!]_{\vec{x},y}[a,b]) \lceil I^n a) \end{array}$$

Il se vérifie aisément que la restriction par $I^n x$ se simplifie en une restriction par a, laquelle s'élimine puisque a est liée:

$$\begin{array}{rcl} [\![M]\!]_{\vec{x}} & = & \lambda z.(\lambda uv.uv)(\lambda a.(\lambda b.[\![P]\!]_{\vec{x},y}[a,b]))z \\ & = & \lambda z.(\lambda v.(\lambda b.[\![P]\!]_{\vec{x},y}[z,b])v) \\ & \overset{induction}{=} & \lambda zv.(\lambda bz.P[x_1, \cdots, x_n, y := \pi^{\vec{x},y}_{x_1} z, \cdots, \pi^{\vec{x}}_{x_{n+1}} z][z,b])v \\ & = & \lambda uv.(\lambda z.P[x_1, \cdots, x_n, y := \pi^{\vec{x},y}_{x_1} z, \cdots, \pi^{\vec{x},y}_{y} z])[u,v] \\ & = & \lambda uv.P[x_1, \cdots, x_n, y := \pi^{\vec{x}}_{x_1} \circ Fst[u,v], \cdots, \pi^{\vec{x}}_{x_n} \circ Fst[u,v], \pi_2[u,v]] \\ & = & \lambda uv.P[x_1, \cdots, x_n, y := \pi^{\vec{x}}_{x_1}(I^n u), \cdots, \pi^{\vec{x}}_{x_n}(I^n u), v] \\ & = & \lambda uy.P[x_1, \cdots, x_n, y := \pi^{\vec{x}}_{x_1} u, \cdots, \pi^{\vec{x}}_{x_n} u, y] \\ & = & \lambda u.(\lambda y.P)[x_1, \cdots, x_n := \pi^{\vec{x}}_{x_1} u, \cdots, \pi^{\vec{x}}_{x_n} u] \end{array}$$

puisque $\pi^{\vec{x}}_{x_i} : I^n \to I$ et que par conséquent $\pi^{\vec{x}}_{x_i}(I^n u) = \pi^{\vec{x}}_{x_i} u$.

- $M = P\lceil Q$: immédiat par hypothèse d'induction . □

Théorème 21 $\mathcal{M}(\mathbf{C}(\mathcal{U}), I, 1, 1) \cong \mathcal{U}$

Remarque 27 *Ceci montre en particulier que la* PCCC $\mathbf{C}(\mathcal{U})$ *n'est pas dégénérée (toute* CCC *étant une* PCCC *dégénérée), est vraiment "partielle", au sens où elle contient au moins autant de flèches non totales qu'il y a d'éléments indéfinis dans la sous-algèbre* $\mathcal{U}$.

Preuve du théorème 21: Soient x une nouvelle variable et $\phi : \mathcal{U} \to \mathcal{M}$, $a \mapsto \phi(a) = [x]a$ ($= \lambda x.a$ en λ-notation), alors ϕ est un morphisme d'algèbre combinatoire avec éléments partiels :

$$\begin{aligned} \phi(S) &= [x]S \\ \text{par définition de } [x] &= KS \\ \mathcal{U} \text{ étant une } \lambda_p\text{-algèbre} &= KS_{\lambda,CL} \\ &= K(\lambda xyz.xz(yz)) \\ \text{d'après le lemme 77} &= [\![\lambda xyz.xz(yz)]\!]_{\mathcal{M}} \\ \text{par définition de } S_{\mathcal{M}} &= S_{\mathcal{M}} \end{aligned}$$

et il en va de même pour K, I.

Pour $a, b : T \to I$, nous avons:

$$\begin{aligned} a._I b &= ev_{I,I} \circ (F \times Id_I) \circ \langle a, b \rangle \\ &= (\lambda z.\pi_1 z(\pi_2 z)) \circ (\lambda z.[\mathbf{1}(\pi_1 z), \pi_2 z)]) \circ (\lambda z.[az, bz]) \\ &= \lambda z.(\lambda xy.xy)(az)(bz) \\ &= \lambda z.az(bz) \end{aligned}$$

d'où: $a._I b = Sab$, a et b étant totales. Il s'ensuit que $\phi(uv) = [x]uv = S([x]u)([x]v) = \phi(u)._I \phi(v)$.

On peut remarquer que c'est la définition de $[x]$ dans le cas partiel qui fait que tout marche bien ici.

D'autre part, ϕ préserve les définis: Les définis de $\mathcal{M}$ sont les flèches totales de $T \to I$, c'est-à-dire celles qui sont représentées par un terme F tel que $!_I \circ F = !_T$, or:

$$\begin{aligned} !_I \circ F &= (\lambda y.I\lceil Iy) \circ F \\ &= \lambda z.(\lambda y.I)(Fz) \\ &= \lambda z.I\lceil Fz \end{aligned}$$

et $!_T = \lambda z.I\lceil KIz = \lambda z.I = KI = T$

Or $!_I \circ \phi(a) = \lambda z.I\lceil(\lambda x.a)z = \lambda z.I\lceil a$, et si $a \Downarrow$, comme $\mathcal{U}$ vérifie la règle ξ, on a donc $\lambda z.I\lceil a = T$, d'où $\phi(a) \Downarrow$.

ϕ admet un morphisme inverse $\psi : \mathcal{M} \to U$, $a \mapsto \psi(a) = aI$:

En effet

$\psi(S_{\mathcal{M}}) = S_{\mathcal{M}} I = KSI = S$ et de même pour K, I.

$\psi(a._{\mathcal{M}} b) = (\lambda z.az(bz))I = aI(bI) = \psi(a).{}_{\mathcal{U}} \psi(b)$

si a est défini dans $\mathcal{M}$ alors $\lambda z.I\lceil az = KI$, d'où $(\lambda z.I\lceil az)I = KII$, et par conséquent $I\lceil aI = I$, soit $aI \Downarrow$.

enfin ϕ et ψ sont bien inverses l'un de l'autre: $\psi(\phi(a)) = ([x]a)I = a$ et $\phi(\psi(a)) = [x](aI) = S(\lambda x.a)I = \lambda z.az = a$ puisque a est une abstraction. □

6.10 Une possible construction de modèles du λ-calcul partiel

Il peut paraître décevant que toutes les λ_p-algèbres ne puissent apparaître comme résultant de la donnée d'une PCCC munie d'un objet réflexif. En effet, dans le cas classique on obtient

un tel résultat: toutes les λ-algèbres (et par suite tous les modèles du λ-calcul) résultent de la donnée d'un objet réflexif dans une CCC et ceci fournit d'ailleurs une technique pour construire des modèles, dont le modèle de Scott D_∞.

Nous pouvons nous apercevoir à reprendre notre démonstration que ceci tient au fait que nous pouvons pas enrichir le λ_p-calcul de nouvelles constantes qui représenteraient tous les termes d'une λ_p-algèbre $\mathcal{A}$ donnée de manière à obtenir une λ-notation pour tous les éléments de $\mathcal{A}$, car nous avons supposé que les constantes étaient totales lorsque nous avons défini les λ_p-algèbres . Or ceci ne tenait qu'à permettre d'obtenir le lemme 47.

Nous allons voir qu'une autre issue peut être donnée à ce problème :

Considérons un nouveau λ_p-calcul, que nous noterons $\lambda_p 1$, qui ne diffère de λ_p que par les modifications suivantes:

- Les termes considérés doivent être explicitement restreints par les variables libres et les constantes qui figurent dans ce terme (c'est-à-dire que si $x \in VL(M)$ alors $x \in \mathsf{r}(M)$, où r est définie comme au chapitre 3).

  Par exemple, le terme $(\lambda xy.zc)\lceil\{z,c\}$, où z est une variable et c une constante, sera un terme que nous considérerons, mais pas le terme $\lambda x.xy$ puisque y n'apparaît pas dans $\mathsf{r}(\lambda x.xy) = \emptyset$

  Autrement dit, la règle de formation d'une abstraction peut être exprimée comme suit:

$$\frac{M \in \lambda_p 1}{\lambda x.M\lceil\{VL(\lambda x.M)\} \cup \{\text{constantes figurant dans } M\} \in \lambda_p 1}$$

- Les variables libres, ni les constantes ne seront plus considérées comme totales. Autrement dit, l'axiome $M\lceil c = M$ est supprimé et l'axiome $M\lceil x = M$ est remplacé par $\lambda x.M\lceil x = \lambda x.M$.

La réduction de notre nouveau calcul reste inchangée, puisque il se voit aisément qu'elle laisse stable $\lambda_p 1$.

Si c'est au prix de modifier le λ_p-calcul, il n'en demeure pas moins que cela va simplifier un certain nombre de choses. Remarquons d'ailleurs que pour les termes clos sans constantes, c'est-à-dire les plus intéressants, le calcul va demeurer inchangé.

Voyons maintenant quels sont les bénéfices obtenus en reprenant notre démonstration au chapitre 5. Nous allons devoir modifier un certain nombre de définitions, mais la trame reste la même. Supprimons dans la logique combinatoire l'axiome $x \Downarrow$ et ne supposons plus que les constantes soient totales. Les modèles seront dès lors des modèles au sens classique (les valuations sont quelconques).

Le lemme 47 est conservé grâce à la condition que nous avons portée sur la formation d'une abstraction et à partir de là, tous les résultats de ce chapitre 5 demeurent valides.

L'interprétation catégorique des termes doit maintenant se reformuler un peu différemment (section 6.5):

Une valuation n'a plus à être supposée totale, autrement dit c'est une application $\rho : V \to |U|$.

Soit un terme M tel que $VL(M) = \{x_1, .., x_n\}$.

Nous allons interpréter M comme nous aurions fait du terme $M[c_{x_1}/x_1, .., c_{x_n}/x_n]$, c'est-à-dire en considérant que les variables libres se comportent comme des constantes c_{x_i} associées aux éléments $\rho(x_i)$. Les mêmes démonstrations montrent que pour toute valuation ρ, $\lambda_p \vdash M = N \Longrightarrow [\![M]\!]_\rho = [\![N]\!]_\rho$.

Il ne nous reste plus qu'à vérifier que nous obtenons bien l'amélioration annoncée du théorème 21.

Si $\mathcal{A}$ une $\lambda_p 1$-algèbre, alors elle le demeure si l'on enrichit le $\lambda_p 1$-calcul considéré d'autant de constantes qu'il y a d'éléments de $\mathcal{A}$ (ou plus précisément si on associe à chaque élément a de $\mathcal{A}$ une nouvelle constante c_a, en posant $(c_a)_{CL} = a$, c'est ce que nous ne pouvions pas faire auparavant). Dès lors chaque élément de a admet une λ-notation (au moins la notation c_a). Il s'ensuit que tout ce que nous avions fait pour une sous-algèbre $\mathcal{U}$ est cette fois valable pour l'algèbre $\mathcal{A}$. D'où le

Théorème 22 *Soit $\mathcal{A}$ une $\lambda_p 1$-algèbre alors:* $\mathcal{M}(\mathbf{C}(\mathcal{A}), I, \mathbf{1}, \mathbf{1}) \cong \mathcal{A}$

Ceci nous laisse espérer pouvoir construire des modéles du $\lambda_p 1$-calcul en trouvant des catégories cartésiennes fermées partielles munies d'un objet réflexif.

7

Annexe: Fin de la démonstration du lemme 61

Ici on fera les calculs dans λ_p et on écrira S pour S_λ, etc...pour simplifier. On donnera seulement les étapes essentielles des calculs sans autrement préciser les règles utilisées.

5.1: $[u][v]S(Ku)(Kv)\lceil uv = [u][v]K(uv)$

$$\begin{aligned} S(Ku)(Kv)\lceil uv &= (\lambda x.Kux(Kvx))\lceil\{Ku, Kv, uv\} \\ &= (\lambda x.uv\lceil x)\lceil\{\lambda x.u, \lambda x.v, u, v, uv\} \\ &= (\lambda x.uv)\lceil\{uv\} \\ &= K(uv) \end{aligned}$$

5.2: $[a]S(S(KK)I)(Ka) = [a]I\lceil a$

$$\begin{aligned} S(S(KK)I)(Ka) &= (\lambda x.S(KK)Ix(Kax))\lceil\{S(KK)I, Ka\} \\ &= (\lambda x.KKx(Ix)a)\lceil\{\lambda x.a, a\} \\ &= (\lambda x.Kxa)\lceil a \\ &= (\lambda x.x\lceil a)\lceil a \\ &= (\lambda x.x)\lceil a \\ &= I\lceil a \end{aligned}$$

5.3: $[a][b]S(S(KK)(Kb))(Ka) = [a][b]Kb\lceil a$

$$\begin{aligned} S(S(KK)(Kb))(Ka) &= (\lambda x.S(KK)(Kb)x(Kax))\lceil\{S(KK)(Kb), Ka\} \\ &= (\lambda x.(KKx)(Kbx)a)\lceil\{KK, Kb, a\} \\ &= (\lambda x.Kba\lceil x)\lceil\{K, b, a\} \\ &= (\lambda x.b\lceil a)\lceil\{b, a\} \\ &= (\lambda x.b)\lceil\{b, a\} \\ &= Kb\lceil a \end{aligned}$$

5.4: $[t][u][v]S(S(KK)(Stv))(Ku) = [t][u][v]Stv\lceil u$

$$\begin{aligned} S(S(KK)(Stv))(Ku) &= (\lambda z.(S(KK)(Stv)z)(Kuz))\lceil\{u, v, t\} \\ &= (\lambda z.(KKz)(Stvz)(Kuz))\lceil\{u, v, t\} \\ &= (\lambda z.K(Stvz)u\lceil z)\lceil\{u, v, t\} \\ &= (\lambda z.Stvz)\lceil\{u, v, t\} \\ &= (\lambda z.(tz)(vz))\lceil\{u, v, t\} \\ &= Stv\lceil u \end{aligned}$$

5.5: $S(S(KK)I)I = I$

$$S(S(KK)I)I = (\lambda x.S(KK)Ix(Ix))\lceil\{S(KK)I, I\}$$

$$\begin{aligned} &= (\lambda x.(KKx)(Ix)x)\lceil\{\lambda x.KKx(Ix), KK, I\} \\ &= (\lambda x.Kxx)\lceil\{\lambda x.K, K\} \\ &= (\lambda x.x) = I \end{aligned}$$

5.6: $[u]S(S(KK)(Ku))I = [u]Ku$

$$\begin{aligned} S(S(KK)(Ku))I &= (\lambda x.S(KK)(Ku)x(Kux)) \\ &\quad \lceil\{S(KK)(Ku), Ku\} \\ &= (\lambda x.S(KK)(Ku)xu)\lceil\{KK, Ku, u\} \\ &= (\lambda x.(KKx)(Kux)u)\lceil u \\ &= (\lambda x.Kuu)\lceil u \\ &= (\lambda x.u\lceil u)\lceil u \\ &= (\lambda x.u)\lceil u \\ &= Ku\lceil u = Ku \end{aligned}$$

5.7: $[u][t]S(S(KK)(Stu))I = [u][t]Stu$

$$\begin{aligned} S(S(KK)(Stu))I &= (\lambda x.S(KK)(Stu)x(Ix)) \\ &\quad \lceil\{S(KK)(Stu), I\} \\ &= (\lambda x.KKx(Stux)x)\lceil\{KK, Stu\} \\ &= (\lambda x.Stux)\lceil\{t, u\} \\ &= (\lambda x.tx(ux))\lceil\{t, u\} \\ &= Stu \end{aligned}$$

5.8: $[a][b]S(S(KK)(Sab))(Sab) = [a][b]Sab$

$$\begin{aligned} S(S(KK)(Sab))(Sab) &= (\lambda z.S(KK)(Sab)z(Sabz))\lceil\{a, b\} \\ &= (\lambda z.(KKz)(Sabz)(Sabz))\lceil\{a, b\} \\ &= (\lambda z.Sabz)\lceil\{a, b\} \\ &= (\lambda z.az(bz))\lceil\{a, b\} \\ &= Sa \end{aligned}$$

5.9: $[t][u][v]S(S(KK)(S(S(KK)t)u))v = [t][u][v]S(S(KK)(S(S(KK)t)v))u$

$$\begin{aligned} S(S(KK)(S(S(KK)t)u))v &= (\lambda z.S(KK)(S(S(KK)t)u)z(vz)) \\ &\quad \lceil\{S(KK)(S(S(KK)t)u), v\} \\ &= (\lambda z.S(KK)(S(S(KK)t)u)z(vz)) \\ &\quad \lceil\{S(S(KK)t)u, v\} \\ &= (\lambda z.KKz(S(S(KK)t)uz)(vz)) \\ &\quad \lceil\{S(KK)t, u, v\} \\ &= (\lambda z.K(S(KK)tz(uz))(vz)) \\ &\quad \lceil\{t, u, v\} \\ &= (\lambda z.(KKz)(tz)(uz)\lceil vz) \\ &\quad \lceil\{t, u, v\} \\ &= (\lambda z.K(tz)(uz)\lceil vz) \\ &\quad \lceil\{t, u, v\} \\ &= (\lambda z.tz\lceil\{uz, vz\}) \\ &\quad \lceil\{t, u, v\} \end{aligned}$$

ce qui donne bien une expression symétrique en u et v.

5.10: $[t][u][v]S(S(KK)t)(S(S(KK)u)v) = [t][u][v]S(S(KK)(S(S(KK)t)u))v$

$$S(S(KK)t)(S(S(KK)u)v) = (\lambda z.S(KK)tz(S(S(KK)u)vz))$$

$$
\begin{array}{rcl}
 & & \lceil\{S(KK)t, S(S(KK)u)v\} \\
 & = & (\lambda z.(KKz)(tz)(S(KK)uz(vz))) \\
 & & \lceil\{KK, t, S(KK)u, v\} \\
 & = & (\lambda z.K(tz)((KKz)(uz)(vz)) \\
 & & \lceil\{KK, t, S(KK)u, v\} \\
 & = & \lceil\{t, u, v\} \\
 & = & (\lambda z.K(tz)(K(uz)(vz))) \\
 & & \lceil\{t, u, v\} \\
 & = & (\lambda z.tz\lceil uz\lceil vz) \\
 & & \lceil\{t, u, v\} \\
 & = & (\lambda z.tz\lceil\{uz, vz\}) \\
 & & \lceil\{t, u, v\} \\
S(S(KK)(S(S(KK)t)u))v & = & (\lambda z.S(KK)(S(S(KK)t)u)z(vz)) \\
 & & \lceil\{S(KK)(S(S(KK)t)u), v\} \\
 & = & (\lambda z.KKz(S(S(KK)t)uz)(vz))) \\
 & & \lceil\{S(S(KK)t)u, v\} \\
 & = & (\lambda z.K(S(KK)tz(uz))(vz)) \\
 & & \lceil\{S(KK)t, u, v\} \\
 & = & (\lambda z.K(K(tz)(uz))(vz)) \\
 & & \lceil\{t, u, v\} \\
 & = & (\lambda z.tz\lceil\{uz, vz\}) \\
 & & \lceil\{t, u, v\}
\end{array}
$$

5.11: $[t][u][v]St(S(S(KK)u)v) = [t][u][v]S(S(KK)(Stu))v$

$$
\begin{array}{rcl}
St(S(S(KK)u)v) & = & (\lambda z.tz(S(S(KK)u)vz))\lceil\{t, S(S(KK)u)v\} \\
 & = & (\lambda z.tz(S(KK)uz(vz)))\lceil\{t, S(KK)u, v\} \\
 & = & (\lambda z.tz(K(uz)(vz)))\lceil\{t, u, v\} \\
 & = & (\lambda z.tz(uz\lceil vz))\lceil\{t, u, v\} \\
S(S(KK)(Stu))v & = & (\lambda z.S(KK)(Stu)z(vz))\lceil\{S(KK)(Stu), v\} \\
 & = & (\lambda z.K(tz(uz))(vz))\lceil\{Stu, v\} \\
 & = & (\lambda z.tz(uz)\lceil vz)\lceil\{t, u, v\}
\end{array}
$$

Or $tz(uz\lceil vz) = tz(uz)\lceil vz$ et il suffit d'appliquer ξ.

5.12 $[t][u][v]S(S(S(KK)t)v)u = [t][u][v]S(S(KK)(Stu))v$

$$
\begin{array}{rcl}
S(S(S(KK)t)v)u & = & (\lambda z.S(S(KK)t)vz(uz))\lceil\{S(S(KK)t)v, u\} \\
 & = & (\lambda z.S(KK)tz(vz)(uz))\lceil\{S(KK)t, v, u\} \\
 & = & (\lambda z.K(tz)(vz)(uz))\lceil\{t, v, u\} \\
 & = & (\lambda z.(tz\lceil vz)(uz))\lceil\{t, v, u\}
\end{array}
$$

de même que dans le cas précédent on conclut en remarquant que $(tz\lceil vz)(uz) = tz(uz)\lceil vz$.

5.13: $[a][b][c]S(S(S(KS)a)b)c = [a][b][c]S(Sac)(Sbc)$

$$
\begin{array}{rcl}
S(S(S(KS)a)b)c & = & (\lambda x.S(S(KS)a)bx(cx)) \\
 & & \lceil\{S(S(KS)a)b, c\} \\
 & = & (\lambda x.S(KS)ax(bx)(cx) \\
 & & \lceil\{S(KS)a, b, c\} \\
 & = & (\lambda x.(KSx)(ax)(bx)(cx)) \\
 & & \lceil\{KS, a, b, c\} \\
 & = & (\lambda x.S(ax)(bx)(cx))
\end{array}
$$

$$\begin{array}{rcl} & & \lceil\{S,a,b,c\} \\ & = & (\lambda x.(ax)(cx)((bx)(cx))) \\ & & \lceil\{a,b,c\} \\ S(Sac)(Sbc) & = & (\lambda x.Sacx(Sbcx)) \\ & & \lceil\{Sac,Sbc\} \\ & = & (\lambda x.(ax)(cx)((bx)(cx)))\lceil\{a,b,c\} \end{array}$$

5.14: $[a][b][c]S(S(KK)(Sab))(Kc) = [a][b][c]Sab\lceil c$

$$\begin{array}{rcl} S(S(KK)(Sab))(Kc) & = & (\lambda x.S(KK)(Sab)x(Kcx))\lceil\{S(KK)(Sab),Kc\} \\ & = & (\lambda x.KKx(Sabx)(Kcx))\lceil\{KK,Sab,c\} \\ & = & (\lambda x.K(ax(bx))c)\lceil\{a,b,c\} \\ & = & (\lambda x.ax(bx)\lceil c)\lceil\{a,b,c\} \\ & = & (\lambda x.ax(bx))\lceil\{a,b,c\} \\ & = & Sab\lceil c \end{array}$$

5.17: $S(KI)I = I$

$$\begin{array}{rcl} S(KI)I & = & (\lambda x.KIx(Ix))\lceil\{KI,I\} \\ & = & (\lambda x.Ix\lceil x)\lceil\lambda x.I \\ & = & (\lambda x.x) = I \end{array}$$

8

Conclusion et perspectives de travail

8.1 Conclusion

Nous avons essayé de porter quelques lumières sur l'outil de raisonnement sur l'équivalence de programme que représente le λ-calcul.

Nous avons donc exploré le problème de la confluence de λ_p en montrant comment le résoudre dans un calcul simplifié, ceci nous a permis d'éclaicir les difficultés soulevées par ce problème dans toute sa généralité.

Nous avons dégagé une variante du λ_v-calcul et montré ainsi que différentes approches raisonnables (au moins les deux que nous avons considérées) peuvent être faites de cette notion, et notamment qu'un certain nombre de résultats sont simultanément vérifiés pour ces calculs, notamment celui que les termes réductibles à une valeur peuvent être caractérisés par un système de types.

Enfin nous avons défini des algèbres combinatoires (λ_p-algèbre) qui répondent à l'esprit du λ_p-calcul, à savoir de discerner les termes indéfinis.

Nous avons pour finir montré quels étaient les liens de ces structures avec les catégories cartésiennes fermées partielles munies d'un objet réflexif.

8.2 Perspectives de travail

Confluence

Notre résultat de confluence pour un calcul où il n'y a en fait qu'une seule restriction nous a laissé apercevoir quels sont les problèmes que soulève la recherche d'un tel résultat pour le calcul tout entier:

Il semble absolument indispensable de trouver une forme normale par rapport à la théorie de la restriction, puisqu'il s'avère que nous ne disposons que peu de techniques pour des réductions modulo une relation d'équivalence. Néanmoins la recherche de cette forme normale, à supposer que toutes les informations globales soient aussi disponibles à un niveau local, impose de saturer le terme des informations qu'on peut déduire à partir de lui. Ce processus de saturation a déjà été utilisé par Pino-Pérez pour prouver la décidabilité de la théorie de la restriction. Nous pensons qu'à considérer les objets infinitaires qu'engendre ce processus, nous serions susceptibles de prouver la confluence de tout le calcul.

Etude du λ_{pl}-calcul

Le lambda-calcul paresseux que nous avons étudier au chapitre 2 mériterait une étude plus approfondie. Ne serait-il un calcul au moins aussi pertinent que λ_p, voire dans ses amliorations que nous avons évoquées. Ce serait à examiner par des méthodes sémantiques. Il semblerait que dans ce calcul les termes réductibles à une valeur le soient fortement. Peut-être des connexions sont à examiner avec les réseaux de preuves issus de la logique linéaire.

Etude du λ_{vh}-calcul

Le lambda calcul par valeur de tête que nous avons proposé mériterait d'être exploré. Il est évidemment associé à un autre λ_p-calcul que celui que nous avons considéré ici. Il serait intéressant de savoir si on peut lui appliquer les constructions d'interprétation standard comme celle de Pino-Pérez (*cf* [Pino 92]), aussi bien que celles des interprétations catégoriques (*cf* [Pino 87], ou celle du chapitre 6).

Il serait également intéressant de définir une machine abstraite appropriée au λ_{vh}-calcul en relation avec la sémantique opérationnelle qui ressort des machines abstraites associées à ces variations et d'étudier ainsi des fondements plus concrets du λ_{vh}-calcul.

Construction de λ_p-algèbres

Le problème qui se pose naturellement est celui de la construction d'une λ_p-algèbre non dégénéée (avec plusieurs indéfinis) qui soit non syntaxique, ce qui revient à construire une *PCCC* avec un objet réflexif. Pourrait-on, par exemple, reprendre la démarche que Scott a adopté pour construire D_∞ comme un objet réflexif dans la catégorie des CPO. Les formes normales de tête jouent-elles ici le même rôle que dans le cas classique.

Il semblerait également possible d'établir un lien de ces λ_p-algèbres avec les modèles algorithmiques proposés par Pino-Pérez (*cf* [Pino 92]).

Logique partielle

Nous avons évoqué la formulation d'une logique partielle qui pourrait donner un cadre aux algèbres combinatoires avec éléments partiels, et qui serait dans l'esprit de celle qu'a proposée Scott (*cf* [Scott79]), à la différence que l'égalité de deux n'implique pas qu'ils soient définis. Il reste à montrer que les deux formulations sont équivalentes.

Sémantique dénotationelle

L'élaboration d'une sémantique dénotationnelle, notamment de modèles fonctionnels suivant une construction analogue à celle de Krivine (*cf* [Krivine 90]) reste à explorer.

Références

[Barendregt] H.BARENDREGT, *The Lambda Calculus: Its Syntax and Semantics*, North-Holland, Amsterdam. 1984.

[Beeson81] M.BEESON, Formalizing constructive Mathematics: why and how?, in F.Richman, editor, *Constructive Mathematics: Proceedings, New Mexico*,1980, pp 146-190, *Lecture Notes in Mathematics*, Vol 873, Springer, Berlin 1981.

[Beeson85] M.BEESON, Foundations of Constructive Mathematics, Springer Verlag Berlin, 1985.

[Church] A.CHURCH, *The calculi of Lambda Conversion*, Princeton University Press, 1941.

[Coppo et alt.78] COPPO, M. DEZANI-CIANCAGLINI, A new type assignement for λ-terms. *Archiv für Math. Logik und Grundlagenfurschung* 19, 1978.

[Curien et alt.] P.L.CURIEN, M.MAUNY, A.SUÁREZ, The categorical Abstract Machine.

[Curien-Obtu.89] P.L.CURIEN, A.OBTUŁÓWICZ (1989) Partiality, Cartesian Closedness and Toposes. *Information and Computation* Vol 80 n^o 1, 1989.

[Dezani& alt.86] M.DEZANI, F.HONSELL, S.RONCHI DELLA ROCCA, Models for functions strictly depending on their arguments, in *Journal of Symbolic Logic*, Vol. 51, 1986. pp 845-846.

[Egidi& alt.91] L.EGIDI, F.HONSELL, S.RONCHI DELLA ROCCA, The lazy call-by-value λ-calculus, in *Proceedings of the Symposium of Computer Science 91*, Kazimierz Dolny, Poland. *Lecture Notes in Computer Science*, Vol. 520, Springer Verlag Berlin, 1991.

[Even& Pino 92] C.EVEN, R. PINO PÉREZ, λ_p-calculus and algebras with partial elements, in *Proceedings of the XII^{th} International Conference of the SCCC*, S.Mujica, A.Viña, editors. Santiago, Chili, 1992, pp 71-84.

[Girard 90] J.Y.GIRARD, Y.LAFONT, P.TAYLOR, *Proof Theory and types.*

[Girard 90] J.Y.GIRARD, *Interprétation fonctionnelle et l'élimination des coupures dans l'arithmétique d'ordre supérieur*, Thèse d'Etat, Université Paris VII, 1972.

[FH-SRDR 87] F.HONSELL, S.RONCHI DELLA ROCCA, Operational and denotational semantics for some untyped λ-calculi, in *Journal of Symbolic Logic*, Vol. 52, 1987. pp 1066-1067.

[Huet 80] G.HUET, *Confluent Reductions: Abstract properties and Application to term rewriting systems*, JACM, vol.27, $n°4$, October 1980, pp 797-821.

[Koymans 82] K.KOYMANS, Models of the lambda-calculus, *Inform. Control,* 52, pp. 306-332.

[Krivine 90] J.L.KRIVINE, Lambda-calcul, types et modèles. Masson, Paris, 1990.

[Lambek& Scott] J.LAMBEK, P.J.SCOTT, Introduction to higher order Logic, Cambridge University Press.

[Mac Lane] S.MAC LANE, Categories for the working Mathematician, Springer Verlag, New York, 1971.

[Moggi 88] E.MOGGI, *The Partial Lambda Calculus*, Ph.D.Thesis, University of Edinburgh, 1988.

[Ong] C-H.L.ONG Lazy Lambda Calculus: An Investigation into the Foundations of Functional Programming, Ph.D.Thesis, University of London, 1988.

[Parigot] PARIGOT M.Parigot, Programming with Proofs: a second order type theory, in *ESOP'88*, *Lecture Notes in Computer Science*, Springer Verlag, Nancy, 1988.

[Pino 87] R.PINO PÉREZ, *Le lambda calcul partiel*, Rapport LIENS $n°11$, ENS, Paris.

[Pino 89] R.PINO PÉREZ, *Decidability of the restriction equationnal theory in the partial lambda calculus*, TCS, vol.67,1989, pp 129-139. North-Holland.

[Pino 91] R.PINO PÉREZ, A strict partial combinatory algebra based on Lambda terms, in Proceedings of the Symposium Mathematical Foundations of Computer Science 91. September 9-13,1991. Kazimierz Dolny, Poland, *Lecture Notes in Computer Science*. Vol.520. Sringer-Verlag, Berlin, 1991.

[Pino 92] R.PINO PÉREZ, *Contribution à l'étude du Lambda Calcul Partiel*, Thèse de Doctorat, Université Paris VII,1992.

[Plotkin 75] G.PLOTKIN, (1975), Call-by-name, call-by-value and the λ-calculus. *Theoretical Computer Science* 1, 1975.

[Scott79] D.S.SCOTT, Identity and existence imn Intuitionistic Logic. In "Application of Sheaves". M.P Fourman, C.J.Mulvey, D.S.Scott, editors. *Lecture Notes in Mathematics*, vol.753.Springer-Verlag, 1979.

[Scott80] D.S.SCOTT, Lambda-calculus, some models, some philosophy, in Barwise et alt., *The Kleene Symposium*, Studies in Logic 101, North Holland, Amsterdam, 1980.

[Szylberach] C.SZYLBERACH Syntaxe et sémantique de la facilité en λ-calcul, Thèse de Doctorat, Université Paris VII, 1991.

UNIVERSITE DES Sc. et Techn. BIBLIOTHEQUE UNIVERSITAIRE LILLE