Thèse

présentée à

L'Université Paris 7 Denis Diderot

pour obtenir le titre de

Docteur de l'Université Paris 7 Denis Diderot

Spécialité

Informatique Fondamentale

par

Tristan Crolard

Sujet de la thèse

Extension de l'isomorphisme de Curry-Howard au traitement des exceptions

(application d'une étude de la dualité en logique intuitionniste)

Soutenue le 3 décembre 1996 devant le jury composé de :

Gérard	Huet	Président
Serge	Grigorieff	Directeur
Thierry	Coquand	Rapporteur
Giuseppe	Longo	Rapporteur
Christine	Paulin-Mohring	
Guy	Cousineau	
Philippe	de Groote	
Hugo	Herbelin	

Introduction

Des mécanismes de traitement des exceptions existent dans la plupart des langages de programmation de haut niveau (ML, ADA, C++,...). Du point de vue du génie logiciel, ces mécanismes qui ont pour objet la gestion des erreurs matérielles et logicielles, sont indispensables. Pourtant dans ces langages, le traitement des exceptions n'est pas reflété dans le système de types (ML, ADA) ou ne l'est que partiellement (C++).

Dans les langages fonctionnels, l'implantation du traitement des exceptions est basée sur la manipulation des "continuations", où la notion de continuation (*i.e.* la fin d'un calcul) peut être vue comme la notion duale de celle de "valeur" (*i.e.* le début d'un calcul). D'autre part, il s'est averé que ce type de manipulation permet de donner un sens calculatoire aux preuves de la logique classique. Or la dualité est une notion intrinsèque à la logique classique, puisque la négation classique est involutive. Le "vrai" est dual du "faux", la conjonction duale de la disjonction. L'implication est définissable en posant $A \Rightarrow B \equiv \neg A \vee B$, son connecteur dual étant alors $A \wedge \neg B$.

Toutefois ce connecteur dual est rarement explicité. En particulier, dans la logique intuitionniste, qui est le cadre original de l'isomorphisme de Curry-Howard, quel est le sens de ce connecteur ? Peut-il permettre de typer plus "intrinsèquement" les manipulations de continuations, et ainsi rendre apparent le traitement des exceptions dans le typage des programmes ? Nous tentons ici de répondre à ces questions.

Le sujet principal de cette thèse est donc l'étude du connecteur dual de l'implication, appelé "soustraction", en logique intuitionniste, ainsi que l'extension de l'isomorphisme de Curry-Howard au typage du traitement des exceptions par la soustraction dans les cadres classiques et intuitionnistes.

La soustraction est abordée sous quatre angles différents, correspondant chacun à un chapitre :

1. la théorie des catégories,

2. les sémantiques algébriques, topologiques et de Kripke,

3. la théorie de la démonstration,

4. l'isomorphisme de Curry-Howard et l'aspect calculatoire.

Dans le premier chapitre, nous étudions la structure d'une catégorie bi-[cartésienne fermée] (bi-[CCC]), c'est-à-dire une catégorie cartésienne fermée (CCC) dont la duale est aussi une CCC. Plus précisément, dans une bi-[CCC], tout couple d'objets admet un co-produit et une "co-exponentielle". Nous montrons que cette structure est dégénérée (*i.e.* il existe au plus une flèche entre deux objets) et que, par conséquent la catégorie des ensembles n'est pas une bi-[CCC]. Ce phénomène de dégénérescence, conséquence directe du théorème de Joyal, est inattendu dans la mesure où il était jusqu'à présent étroitement associé à la logique classique alors que nous montrons dans le chapitre 2 que la logique correspondant au système de types des bi-[CCC] est conservatif sur la logique intuitionniste. Un corollaire de ce résultat (que nous prouverons aussi directement) est l'absence d'interprétation de la soustraction dans la catégorie des ensembles, autrement dit l'absence de sémantique directe pour ce connecteur. Le preuve de la dégénérescence est étudiée en détail en prenant la définition usuelle des CCC, étendue aux bi-[CCC], sous forme de théorie équationnelle engendrée par un système de réécriture.

L'axiomatique des bi-[CCC] ne vérifie pas la propriété de complétude fonctionnelle présente dans la théorie des CCC. Ceci explique pourquoi A. FILINSKI [18, 17] est amené à ajouter un morphisme supplémentaire (et son dual) à l'axiomatique des bi-[CCC] pour obtenir l'équivalence avec un λ-calcul "symétrique" (où les continuations peuvent être manipulées explicitement au même titre que les valeurs). La raison pour laquelle ces morphismes sont typés "classiquement" (*i.e.* par une généralisation du tiers-exclu) sera donnée dans le chapitre 2. La démarche inverse, consistant à restreindre le λ-calcul qui décore les preuves du système de déduction, est l'objet du chapitre 4 (nous utilisons alors le $\lambda\mu$-calcul de M. PARIGOT dont les propriétés calculatoires sont mieux connues que celles du λ-calcul "symétrique" de A. FILINSKI).

Dans le chapitre 2, nous définissons le "calcul propositionnel catégorique symétrique". Les règles de ce calcul proviennent du système de types des "combinateurs catégoriques symétriques" du chapitre précédent, vu comme un système de déduction logique. Puisque les bi-[CCC] sont dégénérées, leur axiomatique définit aussi une classe particulière de pré-ordres : les algèbres de Heyting-Brouwer. On obtient ainsi trivialement une sémantique algébrique pour le calcul propositionnel catégorique symétrique. Une sémantique plus fine est donnée par les modèles topologiques et le forcing de Kripke qui s'avèrent coïncider en présence de la soustraction. Cette sémantique permet de prouver simplement que cette logique, qui possède un connecteur dual de l'implication (appelé "soustraction") est conservative sur la logique intuitionniste. Ce résultat est connu depuis les travaux de C. RAUSZER (1974–80) sur la dualité en sémantique intuitionniste (où la dualité syntaxique n'est pas envisagée, puisque l'axiomatique utilisée est un système de Hilbert). Une des particularités de la soustraction est qu'elle permet de définir une "négation faible" qui vérifie l'axiome du tiers-exclu, mais pas la règle de la contradiction. Nous utilisons aussi la sémantique de Kripke pour prouver la non-définissabilité de la soustraction à partir de la négation faible. Nous terminons ce chapitre par un lien avec d'autres travaux sur la dualité en logique intuitionniste dus à Y. GUREVICH (1977), et par le dilemme posé par la soustraction au premier ordre : la logique intuitionniste soustractive n'est plus conservative sur la logique intuitionniste (puisque le schéma d'axiomes DIS est prouvable, cf. section 6).

Dans le chapitre 3, nous passons à des systèmes de déduction "raisonnables" pour dériver des preuves (à opposer au calcul catégorique symétrique) : le calcul des séquents et la déduction naturelle, que nous étendons par dualité à la soustraction. Toutefois cette extension par dualité nécessite de multiples conclusions, et les deux systèmes ainsi obtenus sont classiques. Pour retrouver un calcul intuitionniste soustractif, il est connu que, dans le cadre propositionnel, il suffit de restreindre la règle de déchargement d'une hypothèse à une unique conclusion, par dualité nous restreindrons la règle de déchargement d'une conclusion à une hypothèse.

Nous définissons alors une nouvelle traduction, pour le calcul propositionnel (puis du premier ordre), de la logique classique (soustractive) vers la logique intuitionniste soustractive et nous montrons que cette traduction s'étend directement aux preuves sans coupure du calcul des séquents LK de G. GENTZEN. Une particularité de cette traduction est qu'elle traduit des preuves de LK sans coupure en des preuves de LK avec soustraction qui ne contiennent aucune occurrence des règles de déchargement d'hypothèse ou de conclusion : ce sont donc des preuves intuitionnistes soustractives.

La restriction des règles de déchargement à une conclusion (resp. à une hypothèse) n'est pas stable par élimination des coupures. Nous définissons alors une contrainte plus faible, dans le cadre de la déduction naturelle classique CND de M. PARIGOT, qui est stable pour les règles de réductions de preuves de CND. Le principe de cette contrainte consiste à expliciter des dépendances entre occurrences d'hypothèses et occurrences de conclusions dans un séquent dérivé dans CND. Une hypothèse ne pourra alors être déchargée sur une conclusion que si aucune autre conclusion n'en dépend. On obtient par la même occasion un critère permettant de déterminer si une preuve de CND est intuitionniste. Ce critère est très contraignant dans CND : on peut alors montrer qu'au moins une des conclusions de tout séquent dérivé est dérivable dans la déduction naturelle intuitionniste usuelle NJ. On obtient ainsi la conservativité du calcul.

Plus précisément, ce critère est défini à partir de liens de dépendances "non orientés" entre occurrences d'hypothèses et occurrences de conclusions. Il se généralise donc aisément à la soustraction. Toutefois la possibilité d'extraire d'une preuve avec soustraction contrainte une preuve d'une des conclusions en déduction naturelle soustractive disparaît. Nous donnerons néanmoins une preuve de conservativité sur le calcul propositionnel (resp. du premier ordre) catégorique symétrique et donc sur la logique intuitionniste usuelle (resp. augmentée de DIS)

Dans le chapitre 4, nous proposons une interprétation calculatoire de la soustraction (vue comme un constructeur de type) dans les cadres classique et intuitionniste. Pour cela, nous présentons tout d'abord quelques rappels sur la machine à environnement de Krivine et les opérateurs de contrôle. Nous utilisons ensuite le $\lambda\mu$-calcul de M. PARIGOT pour définir un λ-calcul étendu par des opérateurs de contrôle **catch** et **throw**, appelé $\lambda_{\mathbf{ct}}$-calcul, qui soit confluent, à la différence du $\lambda_{\mathbf{c/t}}$-calcul non-déterministe de H. NAKANO.

Nous définissons alors une notion de "variable locale piégée" par un **throw** dans un $\lambda_{\mathbf{ct}}$-terme pur et nous utilisons cette notion pour obtenir une contrainte sur les $\lambda_{\mathbf{ct}}$-termes purs) stable pour les règles de réduction du $\lambda\mu$-calcul (resp. du $\lambda_{\mathbf{ct}}$-calcul. Cette contrainte est (par construction) telle que tout $\lambda_{\mathbf{ct}}$-terme contraint typable soit typé par un théorème intuitionniste. Plus précisément, pour tout $\lambda\mu$-terme (resp. $\lambda_{\mathbf{ct}}$-terme) typé, la contrainte définie dans ce chapitre correspond exactement à la contrainte intuitionniste sur le jugement de typage du terme, vu comme une preuve de CND contrainte à la manière du chapitre 3. On retrouve alors, dans le cadre typé, une contrainte semblable à celle de H. NAKANO [45].

Nous étendons enfin ce $\lambda\mu$-calcul (resp. ce $\lambda_{\mathbf{ct}}$-calcul) à l'interprétation calculatoire de la soustraction, dans les cadres classique puis intuitionniste. Ceci se fait tout d'abord en décorant les dérivations, en logique classique, des règles d'introduction et d'élimination de la soustraction. La soustraction apparaît alors comme un candidat naturel pour typer une instruction proche du **try with** du langage ML. La contrainte intuitionniste sur cette extension du $\lambda_{\mathbf{ct}}$-calcul (dont les propriétés calculatoires ne sont pas investiguées dans cette thèse) se définit à nouveau à partir de l'équivalence entre les notions variables locales piégées et la notion de preuve soustractive contrainte au sens du chapitre 3.

Dans le chapitre 5, qui est indépendant des précédents, nous étudions une propriété de la logique intuitionniste usuelle, qui disparaît en présence de la soustraction (mais aussi en logique classique), à savoir la possibilité d'interpréter "naivement" un programme par la fonction qu'il calcule. Cette interprétation est aussi appelée sémantique directe par opposition à la sémantique indirecte (ou "par continuation") qui permet d'interpréter des "traits impératifs" comme le traitement des exceptions. L'arithmétique de Heyting, par exemple, est un système "idéal" dans le sens où toute preuve représente un programme (par l'isomorphisme de Curry-Howard) qui peut alors être interprété par une fonction calculable (sémantique directe du système T de Gödel). On retrouve ainsi la sémantique bien connue des connecteurs en logique intuitionniste : la sémantique de Brouwer-Heyting-Kolmogoroff (dite BHK).

On peut considérer que la réalisabilité modifiée de G. Kreisel est une formalisation de cette sémantique (la réalisabilité récursive de S. C. Kleene en est une autre). L'objectif de ce dernier chapitre est tout d'abord de comparer cette réalisabilité à la prouvabilité intuitionniste, pour aboutir à la définition d'une théorie des types où ces deux notions coïncident.

Chapitre 1

Catégories bi-[cartésiennes fermées]

Dans ce chapitre, nous étudions la structure d'une catégorie bi-[cartésienne fermée] (bi-[CCC]), c'est-à-dire d'une catégorie cartésienne fermée (CCC) dont la duale est aussi une CCC. Nous montrons que cette structure est dégénérée (*i.e.* il existe au plus une flèche entre deux objets) et que, par conséquent la catégorie des ensembles n'est pas une bi-[CCC]. Ce phénomène de dégénérescence, conséquence directe du théorème de Joyal, est étudié en détail en prenant la définition usuelle des CCC, étendue aux bi-[CCC], sous forme de théorie équationnelle engendrée par un système de réécriture. Nous montrons enfin pourquoi A. Filinski est amené à ajouter un morphisme « classique » (et son dual) à l'axiomatique des bi-[CCC] pour obtenir la complétude fonctionnelle.

Introduction

À toute notion catégorique correspond directement une notion duale. En particulier, on peut définir la co-exponentielle, notion duale de l'exponentielle des catégories cartésiennes fermées (CCC). On aboutit alors à la définition d'une bi-[CCC] (*i.e.* une CCC dont la duale est aussi une CCC). Cette structure semble avoir été étudiée pour la première fois par A. Filinski [18, 17] dans le cadre de la sémantique des langages fonctionnels. A. Filinski montre que la dualité dans les bi-[CCC] peut s'interpréter élégamment comme une dualité entre valeurs et continuations. Il construit pour cela un λ-calcul « symétrique » dans lequel les continuations peuvent être manipulées explicitement (au même titre que les valeurs) et pour lequel il étend le théorème de J. Lambek et P. J. Scott qui exprime l'équivalence entre CCC et λ-calcul simplement typé [33]. Toutefois, pour obtenir ce résultat, A. Filinski est amené à étendre l'axiomatique des bi-[CCC] en ajoutant un nouveau morphisme (et son dual) dont le type est une variante du tiers-exclu. Plus précisément, cette extension est nécessaire pour obtenir la complétude fonctionnelle (propriété qui traduit la possibilité de simuler l'abstraction du λ-calcul) dans les bi-[CCC]. Nous verrons qu'elle se justifie déjà du point de vue du système de déduction issu de l'axiomatique des bi-[CCC] dans le chapitre 2. La démarche inverse, consistant à restreindre le λ-calcul qui décore les preuves du système de déduction, sera étudiée dans le chapitre 4 (nous utiliserons alors le $\lambda\mu$-calcul de M. Parigot dont les propriétés calculatoires sont mieux connues que celle du λ-calcul « symétrique » de A. Filinski).

Nous allons ici nous concentrer sur la théorie des bi-[CCC] (sans morphisme supplémentaire). Le plus surprenant est sans doute de constater que les bi-[CCC] sont dégénérées (il existe au plus une flèche entre deux objets), bien que la logique correspondante soit conservative sur la logique intuitionniste (cf. chapitre 2). Cette dégénérescence était jusqu'alors étroitement associée à la logique classique. Nous donnons une analyse détaillée de ce phénomène en section 3.3.

1 Préliminaires

Nous rappelons tout d'abord les notions de base de la théorie des catégories et la définition d'une catégorie cartésienne fermée (CCC). Plusieurs livres d'introduction au domaine peuvent être conseillés, par exemple [34, 1]. Nous nous appuyons ici sur l'intuition ensembliste pour introduire les notions, afin de mettre en valeur le corollaire 2.3.4. Le lecteur familier avec la théorie des CCC pourra se rendre directement en section 2.

1.1 Catégories

Nous appelons « graphe » une structure G composée d'un ensemble d'*objets* et d'un ensemble de *flèches* muni de deux applications, appelées *source* et *cible*, qui à toute flèche associent un objet. Nous sommes maintenant en mesure de donner la définition formelle d'une catégorie. La notation $f : A \to B$ est une abréviation pour « une flèche f de source A de cible B ».

Définition 1.1.1 *Une catégorie $\mathcal{C}$ est un graphe muni de deux applications :*

- *une application unaire Id : objets($\mathcal{C}$) $\to$ flèches($\mathcal{C}$) qui vérifie :*
  source(Id_A) = cible(Id_A) = A pour tout A de objets($\mathcal{C}$).
- *une application binaire $\circ$: composable(flèches($\mathcal{C}$) $\times$ flèches($\mathcal{C}$)) $\to$ flèches($\mathcal{C}$)*
  où composable(g, f) est défini par : cible(f) = source(g).

Ces applications vérifient de plus :

- $f \circ Id_A = Id_B \circ f = f$*, pour tout* $f : A \to B$.
- $(h \circ g) \circ f = h \circ (g \circ f)$*, pour tout* $f : A \to B$, $g : B \to C$ *et* $h : C \to D$

Notation. Pour toute catégorie $\mathcal{C}$, nous noterons $\mathcal{C}[A, B]$ l'ensemble des flèches (ou « morphismes ») allant de A dans B. De même, nous utiliserons désormais aussi les termes « domaine » et « codomaine » pour désigner respectivement les applications *source* et *cible*.

Exemple. L'exemple clé, qui va nous servir à motiver les constructions de la section suivante est bien sûr la catégorie des ensembles, dont les flèches sont les fonctions totales et où l'identité et la composition sont définies de façon standard.

1.2 Constructions

Dans une catégorie quelconque, les objets ne sont pas forcément des ensembles. Comme aucune hypothèse ne peut être faite sur la structure interne des objets, il faut donc exprimer leurs propriétés à partir des flèches qui les relient. Nous allons donc exprimer certaines notions ensemblistes bien connues dans le langage des catégories, et terminer par la définition d'une catégorie cartésienne fermée.

⋆ Isomorphisme

Dans la catégorie des ensembles, une bijection entre deux ensembles est une application injective et surjective. Autrement dit, c'est une application qui admet une application réciproque. Voici la généralisation de cette notion à une catégorie quelconque :

Définition 1.2.1 *Soit $\mathcal{C}$ une catégorie, une flèche $i : A \to B$ est un* ***isomorphisme*** *s'il existe une flèche $j : B \to A$ telle que $j \circ i = Id_A$ et $i \circ j = Id_B$. Les objets A et B sont alors dit* ***isomorphes****.*

Notation. Nous utiliserons le symbole $\cong$ pour représenter un isomorphisme entre deux objets. Il est clair que $\cong$ est une relation d'équivalence sur les objets.

Remarque. Dans une catégorie $\mathcal{C}$, si A, B et C sont des objets de $\mathcal{C}$ et si $A \cong B$ alors (en notant $\approx$ la bijection ensembliste) :

$$\mathcal{C}[B, C] \approx \mathcal{C}[A, C]$$

Plus précisément, si $i : A \to B$ est un isomorphisme entre A et B, d'inverse $j : B \to A$, cette bijection peut s'écrire $f \longmapsto f \circ i$ où $f \in \mathcal{C}[B, C]$ et sa réciproque est alors $g \longmapsto g \circ j$ où $g \in \mathcal{C}[A, C]$.

⋆ Objet terminal

Un singleton $\{\star\}$ est un ensemble qui a la propriété suivante : pour tout ensemble E, il existe une unique application allant de E dans $\{\star\}$. C'est un objet « terminal » de la catégorie des ensembles.

Définition 1.2.2 *Dans une catégorie $\mathcal{C}$, un objet $\top$ est* ***terminal*** *si pour tout objet A de $\mathcal{C}$, il existe une unique flèche dans $\mathcal{C}[A, \top]$ (on note $\Diamond_A : A \to \top$ cette unique flèche).*

De même que dans la catégorie des ensembles, dans toute catégorie munie d'un objet terminal, celui-ci est unique à isomorphisme près.

Remarque. Une application de $\{\star\}$ dans un ensemble quelconque détermine un élément unique de cet ensemble. Pour cette raison, dans une catégorie, toute flèche allant de l'objet terminal vers un objet C est appelée « point » de C.

⋆ Produit

Étant donnés deux ensembles S et S', le produit cartésien $S \times S'$ est l'ensemble des couples (s, s') où $s \in S$ et $s' \in S'$. En notant π et π' les deux projections, nous avons la propriété suivante : pour tout ensemble E, tout couple $\langle f, g\rangle$ d'applications $f : E \to S$ et $g : E \to S'$ est de la forme $\langle \pi \circ h, \pi' \circ h\rangle$ pour une unique application $h : E \to S \times S'$. Cette propriété justifie la définition suivante :

Définition 1.2.3 *Dans une catégorie $\mathcal{C}$, on appelle* ***produit*** *de deux objets A et B un objet, noté $A \times B$, muni de deux flèches $\pi_{A,B} : A \times B \to A$ et $\pi'_{A,B} : A \times B \to B$ qui vérifie la* *propriété suivante : pour tout objet C, toutes flèches $f : C \to A$ et $g : C \to B$, il existe une unique flèche $h : C \to A \times B$ telle que le diagramme suivant commute :*

$$\begin{tikzcd} & C \arrow[dl, "f"'] \arrow[d, "h"] \arrow[dr, "g"] & \\ A & A \times B \arrow[l, "\pi_{A,B}"] \arrow[r, "\pi'_{A,B}"'] & B \end{tikzcd}$$

On note $\langle f, g\rangle$ cet unique h.

Proposition 1.2.4 *Dans une catégorie $\mathcal{C}$, le produit de deux objets A et B, s'il existe, est unique à isomorphisme près.*

Notation. Soient $f : A \to C$ et $g : B \to D$ deux flèches de $\mathcal{C}$. On note $f \times g$ la flèche (unique par définition de $C \times D$) $\langle f \circ \pi_{A,B}, g \circ \pi'_{A,B}\rangle : A \times B \to C \times D$. Intuitivement, f n'agit que sur la première composante du produit et g n'agit que sur la seconde, d'où la notation. Cette intuition apparaît clairement dans le diagramme commutatif suivant :

$$\begin{tikzcd} C & C \times D \arrow[l, "\pi_{C,D}"'] \arrow[r, "\pi'_{C,D}"] & D \\ A \arrow[u, "f"] & A \times B \arrow[u, "f\times g"] \arrow[l, "\pi_{A,B}"] \arrow[r, "\pi'_{A,B}"'] & B \arrow[u, "g"'] \end{tikzcd}$$

Le produit vérifie bien le propriétés habituelles de commutativité et d'associativité. De plus, l'objet terminal, s'il existe dans la catégorie, en est élément neutre.

Proposition 1.2.5 *Si $\mathcal{C}$ est une catégorie munie d'un objet terminal $\top$, alors pour tous objets A, B, et C, lorsqu'elles sont définies, les équations suivantes sont vérifiées.*

- $A \times \top \cong A$
- $A \times B \cong B \times A$
- $A \times (B \times C) \cong (A \times B) \times C$

Définition 1.2.6 *Une catégorie munie d'un objet terminal et d'un produit pour tout couple d'objets est appelée **catégorie cartésienne**.*

⋆ Exponentielle

Soient A et B deux ensembles, on note B^A l'ensemble des applications de A dans B. Pour tout ensemble C, toute application de $C \times A$ dans B est en correspondance bi-univoque avec une application de C dans B^A : c'est la « curryfication ». La définition suivante généralise cette notion à une catégorie cartésienne quelconque.

Définition 1.2.7 *Dans une catégorie cartésienne $\mathcal{C}$, on appelle **exponentielle** de deux objets A et B un objet, noté B^A, muni d'une flèche $\epsilon_{A,B} : B^A \times A \to B$ vérifiant la propriété suivante : pour tout objet C, toute flèche $f : C \times A \to B$, il existe une unique flèche $h : C \to B^A$ telle que le diagramme suivant commute :*

$$\begin{tikzcd} C \arrow[d, "h"'] & C \times A \arrow[d, "h\times Id_A"'] \arrow[r, "f"] & B \\ B^A & B^A \times A \arrow[ur, "\epsilon_{A,B}"'] & \end{tikzcd}$$

On note $f^\star$ cet unique h.

Remarque. Intuitivement, l'objet B^A représente l'ensemble des morphismes de $\mathcal{C}[A, B]$, ϵ correspond à l'évaluation et * à l'opération de curryfication. De même que dans la catégorie des ensembles, on peut montrer que l'opération $f \longmapsto f^\star$ est une bijection de l'ensemble des flèches allant de $C \times A$ vers B et l'ensemble des flèches de C vers B^A. Autrement dit, en notant $\approx$ la bijection ensembliste :

$$\mathcal{C}[C \times A, B] \approx \mathcal{C}[C, B^A]$$

Proposition 1.2.8 *Dans une catégorie cartésienne $\mathcal{C}$, l'exponentielle de deux objets A et B, quand elle existe, est unique à isomorphisme près.*

De même que le produit, l'exponentielle vérifie les propriétés attendues, en particulier la distributivité par rapport au produit.

Proposition 1.2.9 *Si $\mathcal{C}$ est une catégorie cartésienne, alors pour tous objets A, B, et C, lorsqu'elles sont définies, les équations suivantes sont vérifiées.*

- $A^\top \cong A$
- $\top^A \cong \top$
- $(A \times B)^C \cong A^C \times B^C$
- $A^{B \times C} \cong (A^B)^C$

Définition 1.2.10 *Une catégorie cartésienne dans laquelle tout couple d'objets admet une exponentielle est appelée* ***catégorie cartésienne fermée****.*

2 Catégories bi-[cartésiennes fermées]

Nous rappelons comment s'exprime la dualité en théorie des catégories et, comme application, nous définissons les notions duales des constructions de la section précédente. Nous terminons en montrant qu'une structure intégrant la totalité de ces constructions, « une catégorie bi-[cartésienne fermée] » est nécessairement dégénérée : il existe au plus une flèche entre deux objets (nous verrons dans le chapitre 2 que la structure obtenue est en fait une algèbre de Heyting-Brouwer).

2.1 Catégorie duale

À partir d'une catégorie $\mathcal{C}$, on peut définir sa catégorie « duale », notée $\mathcal{C}^\perp$, de la manière suivante : les objets de $\mathcal{C}^\perp$ sont les objets de $\mathcal{C}$, les flèches de $\mathcal{C}^\perp$ sont les flèches de $\mathcal{C}$ et les applications *source* et *cible* de $\mathcal{C}^\perp$ sont respectivement les applications *cible* et *source* de $\mathcal{C}$. Autrement dit, les flèches de $\mathcal{C}^\perp$ sont obtenues en « retournant » les flèches de $\mathcal{C}$, ce qui peut aussi s'écrire $\mathcal{C}^\perp[A, B] \equiv \mathcal{C}[B, A]$. On notera $f^\perp$ la flèche de $\mathcal{C}^\perp$ obtenue en retournant la flèche f de $\mathcal{C}$. L'identité dans $\mathcal{C}^\perp$ est la même que dans $\mathcal{C}$. La composée de deux flèches $f^\perp : B \to A$ et $g^\perp : C \to B$ est définie par $f^\perp \circ g^\perp = (g \circ f)^\perp : C \to A$.

Remarquons que cette dualité est bien involutive, *i.e.* pour toute catégorie $\mathcal{C}$, $(\mathcal{C}^\perp)^\perp = \mathcal{C}$. À toute propriété correspond une propriété duale, obtenue en retournant toutes les flèches (et compositions) de la formule qui l'exprime. De même, à toute construction de la théorie des catégories on peut donc associer de façon naturelle une construction duale, qui est le reflet de la même construction dans la catégorie duale. Informellement, pour définir une construction duale, il faut donc passer dans la catégorie duale (en retournant les flèches « argument » de la construction), y appliquer la construction, puis revenir dans la catégorie de départ (en retournant les flèches « résultat » de la construction).

2.2 Application de la dualité

⋆ Objet initial

Nous donnons pour commencer la définition d'un objet « initial », c'est-à-dire terminal dans la catégorie duale (et on sait par dualité que si cet objet existe, il est unique à isomorphisme près). Dans la catégorie des ensembles, l'objet terminal est l'ensemble vide.

Définition 2.2.1 *Dans une catégorie $\mathcal{C}$, un objet $\perp$ est* ***initial*** *si pour tout objet A de $\mathcal{C}$, il existe une unique flèche dans $\mathcal{C}[\perp, A]$ (on note $\square_A : \perp \to A$ cette unique flèche).*

⋆ Objet co-produit

La définition du co-produit, qui correspond à la somme disjointe dans la catégorie des ensembles, est duale de celle du produit. Autrement dit, le co-produit de deux objets A et B dans une catégorie $\mathcal{C}$ peut-être obtenu en prenant le produit de A et B dans $\mathcal{C}^\perp$ (s'il existe) puis en retournant les projections $\pi_{A,B}$ et $\pi'_{A,B}$ (qui sont des flèches de $\mathcal{C}^\perp$).

Définition 2.2.2 *Dans une catégorie $\mathcal{C}$, on appelle* ***co-produit*** *de deux objets A et B un objet, noté $A \oplus B$, muni de deux flèches $\iota_{A,B} : A \to A \oplus B$ et $\iota'_{A,B} : B \to A \oplus B$ qui vérifie la propriété suivante : pour tout objet C, toutes flèches $f : A \to C$ et $g : B \to C$, il existe une unique flèche $h : A \oplus B \to C$ telle que le diagramme suivant commute :*

$$\begin{tikzcd} & C & \\ A \arrow[ur, "f"] \arrow[r, "\iota_{A,B}"'] & A \oplus B \arrow[u, "h"] & B \arrow[ul, "g"'] \arrow[l, "\iota'_{A,B}"] \end{tikzcd}$$

On note $[f, g]$ cet unique h.

Proposition 2.2.3 *Dans une catégorie $\mathcal{C}$, le co-produit de deux objets A et B, s'il existe, est unique à isomorphisme près.*

Notation. Soient $f : A \to C$ et $g : B \to D$ deux flèches de $\mathcal{C}$. On note $f \oplus g$ la flèche (unique par définition de $A \oplus B$) $[\iota_{C,D} \circ f, \iota'_{C,D} \circ g] : A \oplus B \to C \oplus D$. De même que pour le produit, l'intuition est que f n'agit que sur la première composante du co-produit et g n'agit que sur la seconde, ce qui justifie la notation. À nouveau, cette intuition apparaît clairement dans le diagramme commutatif suivant :

$$\begin{tikzcd} C \arrow[r, "\iota_{C,D}"] & C \oplus D & D \arrow[l, "\iota'_{C,D}"'] \\ A \arrow[u, "f"] \arrow[r, "\iota_{A,B}"'] & A \oplus B \arrow[u, "f \oplus g"] & B \arrow[u, "g"'] \arrow[l, "\iota'_{A,B}"] \end{tikzcd}$$

On sait, par dualité, que le co-produit vérifie les propriétés de commutativité, d'associativité et que l'objet initial, s'il existe, est élément neutre.

Proposition 2.2.4 *Si $\mathcal{C}$ est une catégorie munie d'un objet terminal $\top$, alors pour tous objets A, B, et C, lorsqu'elles sont définies, les équations suivantes sont vérifiées.*

- $A \oplus \perp \cong A$
- $A \oplus B \cong B \oplus A$
- $A \oplus (B \oplus C) \cong (A \oplus B) \oplus C$

Définition 2.2.5 *Une catégorie cartésienne (resp. cartésienne fermée) munie d'un objet initial dans laquelle tout couple d'objets admet un co-produit est appelée **catégorie bi-cartésienne** (resp. **bi-cartésienne fermée**).*

$\star$ Co-exponentielle

Toujours en appliquant la dualité, il est possible de définir la notion de « co-exponentielle ». Plus précisément, la co-exponentielle de deux objets A et B dans une catégorie bi-cartésienne $\mathcal{C}$ peut être obtenu en prenant l'exponentielle de A et B dans $\mathcal{C}^{\perp}$ (si elle existe) puis en retournant le morphisme d'évaluation $\epsilon_{A,B}$ (qui est une flèche de $\mathcal{C}^{\perp}$). Nous n'en donnons pas l'intuition car nous verrons que cette définition ne correspond pas à une notion ensembliste (cf. corollaire 2.3.4).

Définition 2.2.6 *Dans une catégorie bi-cartésienne $\mathcal{C}$, on appelle **co-exponentielle** de deux objets A et B un objet, noté B_A, muni d'une flèche $\ni_{A,B} : B \to B_A \oplus A$ vérifiant la propriété suivante : pour tout objet C, toute flèche $f : B \to C \oplus A$, il existe une unique flèche $h : B_A \to C$ telle que le diagramme suivant commute :*

$$\begin{tikzcd}
C & C \oplus A & B \arrow[l, "f"'] \arrow[dl, "\ni_{A,B}"] \\
B_A \arrow[u, "h"] & B_A \oplus A \arrow[u, "h \oplus Id_A"] &
\end{tikzcd}$$

On note $f_\star$ cet unique h.

Proposition 2.2.7 *Dans une catégorie cartésienne $\mathcal{C}$, la co-exponentielle de deux objets A et B, si elle existe, est unique à isomorphisme près.*

La co-exponentielle vérifie les propriétés duales de celle de l'exponentielle, en particulier la distributivité par rapport au co-produit.

Proposition 2.2.8 *Si $\mathcal{C}$ est une catégorie munie d'un objet initial, alors pour tous objets A, B et C, lorsqu'elles sont définies, les équations suivantes sont vérifiées.*

- $A_\perp \cong A$
- $\perp_A \cong \perp$
- $(A \oplus B)_C \cong A_C \oplus B_C$

– $A_{B\oplus C} \cong (A_B)_C$

Définition 2.2.9 *Une catégorie bi-cartésienne fermée dans laquelle tout couple d'objets admet une co-exponentielle est appelée* ***catégorie bi-[cartésienne fermée]****.*

Notation. Pour éviter toute confusion, nous abrégerons « catégorie bi-[cartésienne fermée] » en « bi-[CCC] » puisque « bi-CCC » est l'abréviation couramment employée pour « catégorie bi-cartésienne fermée » (cf. définition 2.2.5).

Remarque. Une bi-[CCC] est une CCC dont la duale est aussi une CCC (et donc une bi-[CCC]). Une question alors naturelle est : « Existe-t-il des instances non triviales de telles structures ? ». La réponse est « non » dans le sens où toute bi-[CCC] est dégénérée (*i.e.* il existe au plus une flèche entre deux objets).

2.3 Les bi-[CCC] sont dégénérées

Il est connu qu'une CCC munie d'une involution est dégénérée (cf. [21] annexe B). De façon plus générale, toute tentative d'extension des CCC à l'interprétation de la logique classique aboutit au même résultat [33, 63]. Nous montrons ici que ce phénomène de dégénérescence apparaît déjà dans un cadre intuitionniste (cf. théorème 3.4.2). Pour cela nous aurons besoin du théorème de Joyal (cf. [33] p. 67).

Lemme 2.3.1 *Dans une CCC, si $\perp$ est un objet initial alors $(\perp \times A)$ aussi.*

Preuve. En effet, car $C[(\perp \times A), B] \approx C[\perp, B^A]$ (cf. la remarque qui suit la définition 1.2.7). □

Théorème 2.3.2 (Joyal) *Dans une CCC, si $\perp$ est initial et si $C[A, \perp]$ est non vide alors A est initial.*

Preuve. Montrons que A est isomorphe à $\perp \times A$. Si f est une flèche de $C[A, \perp]$, alors les morphismes réciproques entre A et $\perp \times A$ sont $\langle f, Id_A\rangle : A \to \perp \times A$ et $\pi'_{\perp,A} : \perp \times A \to A$. En effet, $\pi'_{\perp,A} \circ \langle f, Id_A\rangle = Id_A$ (par définition du produit) et $\langle f, Id_A\rangle \circ \pi'_{\perp,A} = Id_{\perp\times A}$ car $\perp \times A$ est initial (par le lemme 2.3.1). □

Théorème 2.3.3 *Toute bi-[CCC] est dégénérée : il existe au plus une flèche entre deux objets.*

Preuve. Dans toute CCC, puisque $\top \times B \cong B$,

$$C[B, A] \approx C[(\top \times B), A] \approx C[\top, A^B]$$

et donc par dualité :

$$C[A, B] \approx C[A, (\perp \oplus B)] \approx C[A_B, \perp]$$

Le théorème de Joyal nous dit alors que, puisque $\perp$ est initial, $C[A_B, \perp]$ contient au plus une flèche. □

Comme corollaire direct, on sait que dans la catégorie des ensembles, la co-exponentielle de deux ensembles n'est pas toujours définie. Ce corollaire est évident puisque la catégorie des ensembles n'est clairement pas dégénérée. Il est néanmoins possible d'être plus précis.

Rappelons tout d'abord que dans toute catégorie cartésienne, si $\top$ est terminal, les exponentielles $B^{\top}$ et $\top^{A}$ sont définies pour tous objets A, B (en posant $B^{\top} = B$ avec $\epsilon_{\top,B} = \pi_{B,\top}$ et $\top^{A} = \top$ avec $\epsilon_{A,\top} = \diamond_{\top\times A}$). Par dualité, dans toute catégorie bi-cartésienne, si $\perp$ est initial, les co-exponentielles $B_{\perp}$ et $\perp_{A}$sont toujours définies pour tous objets A, B (en posant $B_{\perp} = B$ avec $\ni_{\perp,B}= \iota_{B,\perp}$ et $\perp_{A} = \perp$ avec $\ni_{A,\perp}= \Box_{\perp\oplus A}$). Nous allons montrer que, dans la catégorie des ensembles, la co-exponentielle n'est définie que dans ces deux cas :

Proposition 2.3.4 *Dans la catégorie des ensembles, la co-exponentielle B_A de deux ensembles A et B est définie si et seulement si $A = \emptyset$ ou $B = \emptyset$.*

Preuve. Nous savons que, puisque $\emptyset$ est initial dans la catégorie bi-cartésienne fermée des ensembles, $B_{\emptyset}$ et $\emptyset_A$ sont définis pour tous A, B.

Supposons maintenant que A et B soient non vides et supposons par l'absurde que B_A soit défini. Cela signifie en particulier que $\ni_{A,B}$ est interprété par une fonction totale de B dans $B_A \oplus A$. Cette fonction choisit, pour chaque élément b de B, un « côté » de $B_A \oplus A$. Or cette fonction doit vérifier la propriété suivante : pour tout ensemble C et tout f allant de B dans $C \oplus A$, $(f_\star \oplus Id_A)\circ \ni_{A,B}= f$. Mais puisque $f_\star \oplus Id_A$ « préserve » le côté choisi, il suffit de prendre un ensemble C non vide et une fonction f qui choisit un autre coté que $\ni_{A,B}$ en b pour que la propriété ne soit plus vérifiée, d'où la contradiction. $\Box$

Remarque. Dans cette preuve, nous n'utilisons pas l'unicité de $f_\star$. Cela signifie donc que dans la catégorie des ensembles, la co-exponentielle « faible » (*i.e.* sans la cette propriété d'unicité) de deux ensembles A et B est définie, pour la même raison que ci-dessus, si et seulement si $A = \emptyset$ ou $B = \emptyset$.

3 Combinateurs catégoriques symétriques

Nous analysons ici le phénomène de dégénérescence dans les bi-[CCC]. Nous reprenons donc les preuves de la section précédente en explicitant les équations qui interviennent. En effet, la théorie des bi-[CCC], comme celle des CCC (cf. l'article de G. HUET [27]) , peut être exprimée comme une théorie du premier ordre purement équationnelle (typée). Les constantes et les constructeurs qui interviennent dans cette théorie, ainsi que leur types et les équations qui les définissent, sont récapitulés ci-dessous :

3.1 Règles de typage

$$Id_A : A \to A$$

$$\frac{f : A \to B \qquad g : B \to C}{g \circ f : A \to C}$$

$$\diamond_A : A \to \top \qquad\qquad \square_A : \bot \to A$$

$$\frac{h : C \to A \qquad g : C \to B}{\langle f, g\rangle : C \to A \times B} \qquad\qquad \frac{f : A \to C \qquad g : B \to C}{[f, g] : A \oplus B \to C}$$

$$\pi_{A,B} : A \times B \to A \qquad \pi'_{A,B} : A \times B \to B \qquad\qquad \iota_{A,B} : A \to A \oplus B \qquad \iota'_{A,B} : B \to A \oplus B$$

$$\epsilon_{A,B} : B^A \times A \to B \qquad\qquad \ni_{A,B}: B \to B_A \oplus A$$

$$\frac{f : C \times A \to B}{f^\star : C \to B^A} \qquad\qquad \frac{h : B \to C \oplus A}{h_\star : B_A \to C}$$

3.2 Règles de calcul

$$
\begin{array}{lrcll}
(1) & (h \circ g) \circ f & = & h \circ (g \circ f) & f : A \to B,\ g : B \to C,\ h : C \to D \\
\\
(2) & f \circ Id_A & = & f & f : A \to B \\
(3) & Id_B \circ f & = & f & f : A \to B \\
\\
(4) & g & = & \diamond_A & g : A \to \top \\
(5) & \pi_{A,B} \circ \langle f, g\rangle & = & f & f : C \to A,\ g : C \to B \\
(6) & \pi'_{A,B} \circ \langle f, g\rangle & = & g & f : C \to A,\ g : C \to B \\
(7) & \langle \pi_{A,B} \circ h, \pi'_{A,B} \circ h\rangle & = & h & h : C \to A \times B \\
\\
(8) & f & = & \square_A & f : \bot \to A \\
(9) & [f, g] \circ \iota_{A,B} & = & f & f : A \to C,\ g : B \to C \\
(10) & [f, g] \circ \iota'_{A,B} & = & g & f : A \to C,\ g : B \to C \\
(11) & [h \circ \iota_{A,B}, h \circ \iota'_{A,B}] & = & h & h : A \oplus B \to C \\
\\
(12) & \epsilon_{A,B} \circ (f^\star \times Id_A) & = & f & f : C \times A \to B \\
(13) & (\epsilon_{A,B} \circ (h \times Id_A))^\star & = & h & h : C \to B^A \\
\\
(14) & (f_\star \oplus Id_A) \circ \ni_{A,B} & = & f & f : B \to C \oplus A \\
(15) & ((h \oplus Id_A) \circ \ni_{A,B})_\star & = & h & h : B_A \to C
\end{array}
$$

Remarque. Nous rappelons qu'étant donné $f : A \to C$ et $g : B \to D$, $f \times g$ et $f \oplus g$ sont les abréviations respectives de $\langle f \circ \pi_{A,B}, g \circ \pi'_{A,B}\rangle$ et $[\iota_{C,D} \circ f, \iota'_{C,D} \circ g]$.

Les équations 4, 7, 8, 11, 13 et 15 sont appelées « règles d'extensionalité » puisqu'elles expriment l'unicité de la flèche dans la définition de la construction correspondante (qui terminent toutes par « pour toutes flèches . . . , il existe une *unique* flèche telle que . . . »). Les constructions obtenues en supprimant la règle d'extensionnalité sont dites « faibles ».

3.3 Preuves équationnelles

⋆ Le théorème de Joyal

Le lemme 2.3.1 dit que $C[(\bot \times A), B]$ contient une unique flèche. En fait, on sait nommer cette unique flèche, ce qui permet de traduire ce lemme sous la forme de la règle dérivée :

$$(16) \qquad h \;=\; \epsilon_{A,B} \circ (\Box_{B^A} \times Id_A) \qquad h : (\bot \times A) \to B$$

La preuve équationnelle de (16) s'obtient ainsi :

$$\begin{array}{rcll} h & \overset{12}{=} & \epsilon_{A,B} \circ (h^\star \times Id_A) & h : (\bot \times A) \to B \\ & \overset{8}{=} & \epsilon_{A,B} \circ (\Box_{B^A} \times Id_A) & h^\star : \bot \to B_A \end{array}$$

Le théorème de Joyal dit que si $C[A, \bot]$ est non vide, alors A est initial et donc $C[A, B]$ contient une unique flèche pout tout B. En fait, on sait nommer cette unique flèche à partir d'une flèche de $C[A, \bot]$, ce qui permet de traduire le théorème de Joyal sous la forme de la règle dérivée suivante :

$$(17) \qquad g \;=\; \epsilon_{A,B} \circ (\Box_{B^A} \times Id_A) \circ \langle f, Id_A \rangle \qquad f : A \to \bot,\ g : A \to B$$

La preuve équationnelle de (17) s'obtient ainsi :

$$\begin{array}{rcll} g & \overset{2}{=} & g \circ Id_A & g : A \to B \\ & \overset{6}{=} & g \circ (\pi'_{\bot\times A} \circ \langle f, Id_A \rangle) & f : A \to \bot,\ Id_A : A \to A \\ & \overset{1}{=} & (g \circ \pi'_{\bot\times A}) \circ \langle f, Id_A \rangle & \\ & \overset{16}{=} & \epsilon_{A,B} \circ (\Box_{B^A} \times Id_A) \circ \langle f, Id_A \rangle & g \circ \pi'_{\bot\times A} : (\bot \times A) \to B \end{array}$$

Un conséquence simple de (17) est que $C[A, \bot]$ contient au plus une flèche, ce qui se traduit par la règle dérivée suivante :

$$(18) \qquad g \;=\; f \qquad f : A \to \bot,\ g : A \to \bot$$

La preuve équationnelle de (18) s'obtient ainsi :

$$\begin{array}{rcll} g & \overset{17}{=} & \epsilon_{A,\bot} \circ (\Box_{\bot^A} \times Id_A) \circ \langle f, Id_A \rangle & f : A \to \bot,\ g : A \to \bot \\ & \overset{17}{=} & f & f : A \to \bot,\ f : A \to \bot \end{array}$$

⋆ Pourquoi les bi-[CCC] sont dégénérées

Comme nous allons le voir, la preuve du théorème 2.3.3 utilise comme seule règle d'extensionalité la règle du $\bot$ (règle 8). La preuve du théorème de Joyal n'utilisait que les règles de l'objet initial $\bot$, du produit et de l'exponentielle. Nous utiliserons ici de plus les règles du co-produit et de la co-exponentielle. L'objet terminal n'est donc pas nécessaire, mais il est bien sûr possible d'en donner une preuve duale qui utiliserait les règles de l'objet terminal et non celle de l'objet initial.

Le théorème 2.3.3 dit que $C[A, B]$ contient au plus une flèche pout tout A, B. Pour en donner la preuve équationnelle, nous aurons besoin de la règle dérivée suivante :

$$(19) \qquad g \;=\; [\Box_B, Id_B] \circ ((\iota'_{B,\bot} \circ f)_\star \oplus Id_B) \circ \ni_{B,A} \qquad f : A \to B,\ g : A \to B$$

La preuve équationnelle de (19) s'obtient ainsi :

$$
\begin{array}{rlr}
g \stackrel{3}{=} & Id_B \circ g & g : A \to B \\
\stackrel{10}{=} & ([\Box_B, Id_B] \circ \iota'_{B,\perp}) \circ g & \Box_B : \perp \to B,\ Id_B : B \to B \\
\stackrel{1}{=} & [\Box_B, Id_B] \circ (\iota'_{B,\perp} \circ g) & \\
\stackrel{14}{=} & [\Box_B, Id_B] \circ ((\iota'_{B,\perp} \circ g)_\star \oplus Id_B) \circ \ni_{B,A} & \iota'_{B,\perp} \circ g : A \to \perp \oplus B \\
\stackrel{18}{=} & [\Box_B, Id_B] \circ ((\iota'_{B,\perp} \circ f)_\star \oplus Id_B) \circ \ni_{B,A} & (\iota'_{B,\perp} \circ g)_\star : A_B \to \perp, (\iota'_{B,\perp} \circ f)_\star : A_B \to \perp
\end{array}
$$

Finalement, la dégénérescence des bi-[CCC] se traduit par la règle :

$$
(20) \qquad g \;=\; f \qquad f : A \to B,\ g : A \to B
$$

La preuve équationnelle de (20) s'obtient de la même manière que (18) :

$$
\begin{array}{rlr}
g \stackrel{19}{=} & [\Box_B, Id_B] \circ ((\iota'_{B,\perp} \circ f)_\star \oplus Id_B) \circ \ni_{B,A} & f : A \to B,\ g : A \to B \\
\stackrel{19}{=} & f & f : A \to B,\ f : A \to B
\end{array}
$$

Remarque. L'identification de toutes les flèches d'un même type $A \to B$ provient donc en particulier de la dernière égalité qui fait disparaître g de l'expression. L'équation indirectement en cause est $\Box_A = h$ pour tout $h : \perp \to A$, qui est la seule règle d'extensionnalité utilisée dans cette preuve. On peut d'ailleurs facilement montrer que la théorie des bi-[CCC] ne contenant aucune règle d'extensionnalité n'est pas dégenerée. Pour cela, nous considérons la théorie comme un sytème de réécriture et nous montrons que ce système peut être complété en un système canonique. Plus précisément, l'ensemble de règles (supposées orientées de la gauche vers la droite) suivant :

$$
\begin{array}{rcllr}
(h \circ g) \circ f & = & h \circ (g \circ f) & & f : A \to B,\ g : B \to C,\ h : C \to D \\[1ex]
f \circ Id_A & = & f & & f : A \to B \\
Id_B \circ f & = & f & & f : A \to B \\[1ex]
\pi_{A,B} \circ \langle f, g \rangle & = & f & & f : C \to A,\ g : C \to B \\
\pi'_{A,B} \circ \langle f, g \rangle & = & g & & f : C \to A,\ g : C \to B \\
\langle \pi_{A,B}, \pi'_{A,B} \rangle & = & Id_{A \times B} & & \\[1ex]
[f, g] \circ \iota_{A,B} & = & f & & f : A \to C,\ g : B \to C \\
[f, g] \circ \iota'_{A,B} & = & g & & f : A \to C,\ g : B \to C \\
[\iota_{A,B}, \iota'_{A,B}] & = & Id_{A \oplus B} & & \\[1ex]
\epsilon_{A,B} \circ \langle f^\star \circ \pi_{C,A}, \pi'_{C,A} \rangle & = & f & & f : C \times A \to B \\[1ex]
[\iota_{C,A} \circ f_\star, \iota'_{C,A}] \circ \ni_{A,B} & = & f & & f : B \to C \oplus A
\end{array}
$$

peut être complété par des règles permettant de résoudre les paires critiques engendrées par la règle d'associativité de la composition (ces règles supplémentaires ont été obtenues par l'auteur en exécutant une implantation de l'algorithme de Knuth-Bendix développée à l'INRIA par G. Huet), à savoir :

$$\begin{array}{rcll}
\pi_{A,B} \circ (\langle f, g\rangle \circ h) & = & f \circ h & f : C \to A,\ g : C \to B,\ h : D \to C \\
\pi'_{A,B} \circ (\langle f, g\rangle \circ h) & = & g \circ h & f : C \to A,\ g : C \to B,\ h : D \to C \\
\\
[f, g] \circ (\iota_{A,B} \circ h) & = & f \circ h & f : A \to C,\ g : B \to C,\ h : D \to A \\
[f, g] \circ (\iota'_{A,B} \circ h) & = & g \circ h & f : A \to C,\ g : B \to C,\ h : D \to B \\
\\
\epsilon_{A,B} \circ (\langle f^{\star} \circ \pi_{C,A}, \pi'_{C,A}\rangle \circ g) & = & f \circ g & f : C \times A \to B,\ g : D \to C \times A \\
\\
[\iota_{C,A} \circ f_{\star}, \iota'_{C,A}] \circ (\ni_{A,B} \circ g) & = & f \circ g & f : B \to C \oplus A,\ g : D \to B
\end{array}$$

Malheureusement, le système canonique obtenu ne permet clairement plus de simuler la β-réduction : chaque règle fait diminuer strictement la taille du terme, la forme normale est donc obtenue en un nombre de réduction au plus linéaire en la taille du terme. En fait, une des règles manquante, qui ne vérifie pas cette propriété, est celle permettant de « progager la substitution » dans un terme (appelée *DistrPair* dans [27]), dérivable à partir de la règle d'extensionalité du produit :

$$\langle f, g\rangle \circ h \;=\; \langle f \circ h, g \circ h\rangle \qquad f : C \to A,\ g : C \to B,\ h : D \to C$$

Plusieurs systèmes de réécriture confluents, basés sur les combinateurs catégoriques et permettant de simuler le λ-calcul, sont connus (cf. [27] §4.4). Toutefois, l'extension de ce résultat aux combinateurs des bi-[CCC], nécéssite tout d'abord de résoudre le problème pour les combinateurs des catégories bi-cartésiennes fermées (*i.e.* avec co-produit mais sans co-exponentielle) qui semble déjà poser des difficultés [12].

Enfin, P. A. Mellies [39] a prouvé que le système des combinateurs catégoriques, avec règles d'extensionalité, même typé, ne vérifie pas la propriété de normalisation. La preuve de ce résultat s'obtient en définissant explicitement un point fixe à partir des combinateurs catégoriques, exactement de la même manière que dans certains λ-calculs avec substitutions explicites.

4 Pas de complétude fonctionnelle

La complétude fonctionnelle est le résultat principal permettant de prouver l'équivalence entre CCC et lambda-calcul simplement typé (cf. [33] p. 61) : il exprime la définissabilité de l'abstraction à partir des combinateurs catégoriques. Informellement, il peut s'énoncer ainsi : si à partir d'une flèche « hypothétique » $x : \top \to A$ il est possible de construire une flèche $\phi(x) : \top \to B$, alors il existe une flèche $f : A \to B$ telle que $f \circ x = \phi(x)$.

Un énoncé formel nécéssite la définition de la « catégorie polynomiale » à laquelle appartient le terme $\phi(x)$. Ce formalisme ne sera pas nécessaire pour montrer que ce résultat n'est pas vérifié dans les bi-[CCC]. En effet, nous allons montrer qu'il n'existe pas nécessairement de flèche $f : A \to B$ telle que $f \circ x = \phi(x)$, en montrant que dans certains cas, il n'existe pas du tout de flèche de type $A \to B$.

Plus formellement, le résultat de complétude fonctionnelle admet un corollaire dans la logique définie par les règles de typage des CCC. C'est ce corollaire, appelé « théorème de déduction » dans [33] (p. 51), exprimé dans la logique associée aux bi-[CCC], que nous allons réfuter dans le chapitre 2.

Par contre, en ajoutant deux nouveaux morphismes typés respectivement par $(B \oplus C)^A \rightarrow B^A \oplus C$ et son dual $B \times C_A \rightarrow (B \times C)_A$ et les équations qui les définissent, il est possible de retouver la complétude fonctionnelle (cf. [18]). Il est surprenant de constater que l'interprétation logique du type du premier morphisme, $A \Rightarrow (B \vee C) \vdash (A \Rightarrow B) \vee C$, est une généralisation du tiers-exclu. Nous verrons dans le chapitre 2 que cette extension à la logique classique est nécessaire, puisqu'elle découle du théorème de déduction.

Chapitre 2

Sémantique de la soustraction

Dans ce chapitre, nous définissons le « calcul propositionnel catégorique symétrique ». Les règles de ce calcul proviennent du système de types des « combinateurs catégoriques symétriques » du chapitre précédent, vu comme un système de déduction logique. Ce calcul permet de prouver simplement que cette logique, qui possède un connecteur dual de l'implication (la « soustraction ») est conservative sur la logique intuitionniste. Ce résultat est connu depuis les travaux de C. Rauszer (1974–80) sur la dualité en sémantique intuitionniste. En particulier, nous en rappelons les sémantiques basées sur les modèles topologiques et le forcing de Kripke qui s'avèrent coïncider en présence de la soustraction. Nous utilisons alors cette sémantique pour prouver la non-définissabilité de la soustraction. Nous terminons ce chapitre par un lien avec d'autres travaux sur la dualité en logique intuitionniste dus à Y. Gurevich (1977), et par le dilemme posé par la soustraction au premier ordre.

Introduction

Les catégories dégénérées correspondent exactement aux pré-ordres. Du point de vue logique, on peut montrer que la structure obtenue à partir de l'axiomatique des bi-[CCC] est exactement une algèbre de Heyting-Brouwer. La logique propositionnelle correspondant à ces algèbres, qui est une extension conservative de la logique intuitionniste, a été étudiée en détail par C. Rauszer [54, 55, 56]. Les résultats présentés dans ce chapitre ont été redécouverts indépendamment par l'auteur et recouvrent donc (en particulier la section 3) partiellement les travaux de C. Rauszer. La présentation des sections 1 et 2 s'appuie toutefois sur le « calcul propositionnel catégorique symétrique » (et non sur un système de Hilbert, comme dans [54]), ce qui rend les preuves plus simples. Enfin, la preuve de la non-définissabilité de la soustraction à partir de la négation faible (cf. sous-section 4.5) est une contribution originale.

1 Algèbres de Heyting-Brouwer

Toute catégorie dégénérée correspond exactement à un pré-ordre. En effet, l'existence d'une flèche entre deux objets nous indique s'ils sont comparables. La réflexivité est donnée par l'identité et la transitivité par la composition. Réciproquement, tout pré-ordre peut être vu comme une catégorie dégénérée.

Puisque les bi-[CCC] sont dégénérées, ce sont des pré-ordres particuliers : les constructions d'une catégorie bi-[cartésienne fermée] définissent une pré-algèbre de Heyting (cf. [66] p. 259), leurs constructions duales une pré-algèbre de Brouwer (cf. [8] p. 162), et l'ensemble une pré-algèbre de Heyting-Brouwer (cf. [54] p. 220).

Définition 1.0.1 *Un **treillis** est une structure* $(\mathcal{A}, \leq, \top, \bot, \sqcap, \sqcup)$ *telle que :*

– $(\mathcal{A}, \leq)$ *est un ordre :*

$$x \leq x$$

$$x \leq y, y \leq x \rightarrow x = y$$

$$x \leq y, y \leq z \rightarrow x \leq z$$

– $\bot$ *et* $\top$ *sont respectivement les plus petit et plus grand élément de* $(\mathcal{A}, \leq)$ *:*

$$x \leq \top \qquad \bot \leq x$$

– *toute paire d'élément possède une borne sup et une borne inf (notées* $\sqcup$ *et* $\sqcap$*) :*

$$x \sqcap y \leq x \qquad x \sqcap y \leq y \qquad z \leq x, z \leq y \rightarrow z \leq x \sqcap y$$

$$x \leq x \sqcup y \qquad y \leq x \sqcup y \qquad x \leq z, y \leq z \rightarrow x \sqcup y \leq z$$

Définition 1.0.2 *Une **algèbre de Heyting** est une structure* $(\mathcal{A}, \leq, \top, \bot, \sqcap, \sqcup, \uparrow)$ *telle que* $(\mathcal{A}, \leq, \top, \bot, \sqcap, \sqcup)$ *soit un treillis et pour toute paire d'éléments* x, y*, l'ensemble des* z *tels que* $z \sqcap x \leq y$ *admet un plus grand élément (noté* $y \uparrow x$*) :*

$$(y \uparrow x) \sqcap x \leq y \qquad z \sqcap x \leq y \rightarrow z \leq (y \uparrow x)$$

Définition 1.0.3 *Une **algèbre de Brouwer** est une structure* $(\mathcal{A}, \leq, \top, \bot, \sqcap, \sqcup, \downarrow)$ *telle que* $(\mathcal{A}, \leq, \top, \bot, \sqcap, \sqcup)$ *soit un treillis et pour toute paire d'éléments* x, y*, l'ensemble des* z *tels que* $y \leq z \sqcup x$ *admet un plus petit élément (noté* $y \downarrow x$*) :*

$$y \leq (y \downarrow x) \sqcup x \qquad y \leq z \sqcup x \rightarrow (y \downarrow x) \leq z$$

Définition 1.0.4 *Une **algèbre de Heyting-Brouwer** est une structure* $(\mathcal{A}, \leq, \top, \bot, \sqcap, \sqcup, \uparrow, \downarrow)$ *telle que* $(\mathcal{A}, \leq, \top, \bot, \sqcap, \sqcup, \uparrow)$ *soit une algèbre de Heyting et* $(\mathcal{A}, \leq, \top, \bot, \sqcap, \sqcup, \downarrow)$ *soit une algèbre de Brouwer.*

La définition d'une *pré-algèbre* de Heyting, de Brouwer ou de Heyting-Brouwer est obtenue (comme pour un préordre) en supprimant de la propriété d'anti-symmétrie de la définition de l'algèbre correspondante.

Remarques

– Tout algèbre de Heyting est distributive (cf. [6] p. 45), *i.e.*

$$x \sqcap (y \sqcup z) \leq (x \sqcap y) \sqcup (x \sqcap z) \qquad x \sqcup (y \sqcap z) \geq (x \sqcup y) \sqcap (x \sqcup z)$$

(les inégalités réciproques sont toujours vérifiées dans un treillis). La première de ces deux lois se dérive facilement : par définition de $\sqcup$, $x \sqcap y \leq (x \sqcap y) \sqcup (x \sqcap z)$ d'où $y \leq ((x \sqcap y) \sqcup (x \sqcap z)) \uparrow x$ par définition de $\uparrow$. De même, $z \leq ((x \sqcap y) \sqcup (x \sqcap z)) \uparrow x$, et donc $y \sqcup z \leq ((x \sqcap y) \sqcup (x \sqcap z)) \uparrow x$. D'où $x \sqcap (y \sqcup z) \leq x \sqcap (((x \sqcap y) \sqcup (x \sqcap z)) \uparrow x)$ et le résultat découle de $x \sqcap (((x \sqcap y) \sqcup (x \sqcap z)) \uparrow x) \leq (x \sqcap y) \sqcup (x \sqcap z)$ par définition de $\uparrow$.

La première loi permet alors de dériver la seconde :

$$(x \sqcup y) \sqcap (x \sqcup z) \leq ((x \sqcup y) \sqcap x) \sqcup ((x \sqcup y) \sqcap z) \leq (x \sqcap x) \sqcup (y \sqcap x) \sqcup (x \sqcap z) \sqcup (y \sqcap z) \leq x \sqcup (y \sqcap z)$$

(les deux premières inégalités s'obtiennent en appliquant la première loi, la dernière inégalité est toujours vérifiée dans un treillis).

Par dualité, une algèbre de Brouwer est aussi distributive : la soustraction va permettre de dériver directement la seconde loi, qui va alors entrainer la première. Dans une algèbre de Heyting-Brouwer, ces deux lois admettent donc des dérivations symétriques.

– Tout treillis fini distributif est une algèbre de Heyting, et par dualité c'est aussi une algèbre de Brouwer. C'est donc une algèbre de Heyting-Brouwer.

2 Calcul propositionnel catégorique symétrique

Il est possible (cf. [33] p. 47) de présenter la théorie des CCC (resp. avec objet initial, co-produit) comme un système de déduction du calcul propositionnel minimal (resp. intuitionniste, avec disjonction). Ce point de vue peut s'étendre au calcul avec co-exponentielle, que *nous noterons désormais* « $-$ » *et appellerons* « *soustraction* » (terminologie apparemment due à Skolem, cf. [8] p. 144).

2.1 Le système de déduction

Tout ce qui concerne les flèches dans la définition des bi-[CCC] est supprimé, puisque « non-informatif » (cf. théorème 2.3.3). Le calcul que nous présentons ici servira désormais de système de déduction de « référence » pour interpréter la soustraction. Les formules sont construites à partir des connecteurs habituels et de la soustraction. Les règles sont exactement la traduction des règles de typage des bi-[CCC] de la section 3, et par conséquent elle traduisent aussi exactement l'axiomatique des algèbres de Heyting-Brouwer (cf. sous-section 2.2).

Axiomes d'identité

$$A \vdash A$$

Règle de coupure

$$\frac{A \vdash B \qquad B \vdash C}{A \vdash C}$$

Axiomes du $\perp$ et du $\top$

$$\perp \vdash A \qquad\qquad A \vdash \top$$

Axiomes d'intro. gauche et règle d'intro. droite du $\wedge$

$$A \wedge B \vdash A \qquad A \wedge B \vdash B \qquad\qquad \frac{C \vdash A \qquad C \vdash B}{C \vdash A \wedge B}$$

Régle d'intro. gauche et axiomes d'intro. droite du $\vee$

$$\dfrac{A \vdash C \qquad B \vdash C}{A \vee B \vdash C} \qquad\qquad A \vdash A \vee B \qquad B \vdash A \vee B$$

Axiomes d'intro. gauche et règle d'intro. droite du $\Rightarrow$

$$(B \Rightarrow A) \wedge B \vdash A \qquad\qquad \dfrac{C \wedge B \vdash A}{C \vdash B \Rightarrow A}$$

Règle d'intro. gauche et axiome d'intro. droite du $-$

$$\dfrac{A \vdash B \vee C}{A - C \vdash B} \qquad\qquad A \vdash (A - B) \vee B$$

Remarques

1. Dans ce système, on peut dériver le séquent $A \wedge \neg B \vdash A - B$ (où $\neg B$ est la négation intuitionniste $B \Rightarrow \bot$) :

   $$\dfrac{\dfrac{\dfrac{A \vdash (A-B) \vee B \qquad \neg B \vdash \neg B}{A \wedge \neg B \vdash ((A-B) \vee B) \wedge \neg B}\times}{A \wedge \neg B \vdash (A-B) \vee (B \wedge \neg B)}\alpha \quad \dfrac{A - B \vdash A - B \qquad \dfrac{B \wedge \neg B \vdash \bot \quad \bot \vdash A - B}{B \wedge \neg B \vdash A - B}}{(A-B) \vee (B \wedge \neg B) \vdash A - B}}{A \wedge \neg B \vdash A - B}$$

   où α est la règle dérivée de distributivité dont la preuve est donnée dans l'exemple ci-dessous et $\times$ est la règle dérivée suivante (qui correspond au produit de flèches catégorique $f \times g = \langle f \circ \pi_{A,B}, g \circ \pi'_{A,B} \rangle \ : A \times B \to C \times D$ si $f : A \to C$ et $g : B \to D$) :

   $$\dfrac{\dfrac{A \wedge B \vdash A \quad A \vdash C}{A \wedge B \vdash C} \quad \dfrac{A \wedge B \vdash B \quad B \vdash D}{A \wedge B \vdash D}}{A \wedge B \vdash C \wedge D}$$

2. Le séquent réciproque $A - B \vdash A \wedge \neg B$ n'est pas dérivable (cf. sous-section 4.5) mais il le devient en présence du tiers-exclu, formulé ici sous forme du séquent $\top \vdash \neg B \vee B$:

   $$\dfrac{\dfrac{A \vdash A \vee B}{A - B \vdash A} \quad \dfrac{\dfrac{A \vdash \top \quad \top \vdash \neg B \vee B}{A \vdash \neg B \vee B}}{A - B \vdash \neg B}}{A - B \vdash A \wedge \neg B}$$

   L'équivalence, en logique classique, entre $A - B$ et $A \wedge \neg B$ justifie donc l'utilisation du symbole de la soustraction ($A - B$ signifie « A et non B » et peut donc se lire « A privé de B » ou « A moins B »). De plus, dans l'interprétation usuelle (classique) des connecteurs propositionnels dans l'algèbre de Boole des parties d'un ensemble, la soustraction s'interprète bien par la différence ensembliste.

3. L'utilisation du symbole de la soustraction brise la symétrie « graphique » des règles (la règle d'introduction droite de l'implication préserve l'ordre des formules du séquent, ce qui n'est pas le cas pour la règle d'introduction gauche soustraction). Cette disymétrie, qui n'a aucune signification mathématique, pourrait être supprimée en notant l'implication « $\Leftarrow$ ».

Dans ce système de déduction, chaque connecteur admet un connecteur dual ($\top, \wedge, \Rightarrow$ sont respectivement duaux de $\bot$, $\vee$,$-$). On peut donc associer à toute formule propositionnelle une formule obtenue en remplaçant chaque connecteur de A par son dual (et en permutant les arguments si le connecteur est $-$ ou $\Rightarrow$). La dualité ainsi définie sur les formules ne coïncide pas avec la dualité en logique booléenne classique (représentée par la négation) puisque les atomes sont inchangés par la traduction : nous l'appelerons donc pseudo-dualité (nous verrons en section 5 qu'il est possible de « compléter » cette pseudo-dualité en se donnant une dualité sur les atomes).

Définition 2.1.1 *La formule pseudo-duale d'une formule A, notée $\overline{A}$, est définie par récurrence :*

- $\overline{A} = A$, *si A est un atome propositionnel*
- $\overline{\top} = \bot$, $\overline{\bot} = \top$
- $\overline{A \vee B} = \overline{B} \wedge \overline{A}$
- $\overline{A \wedge B} = \overline{B} \vee \overline{A}$
- $\overline{A \Rightarrow B} = \overline{B} - \overline{A}$
- $\overline{A - B} = \overline{B} \Rightarrow \overline{A}$

Le sequent pseudo-dual de $A \vdash B$ est le séquent $\overline{B} \vdash \overline{A}$.

Remarque. On vérifie facilement que cette pseudo-dualité est involutive, c'est-à-dire que pour toute formule, $\overline{(\overline{A})} = A$.

On peut alors montrer la propriété suivante, dite de « contraposition », pour la pseudo-dualité :

Proposition 2.1.2 *Si le séquent $A \vdash B$ est dérivable, alors le séquent $\overline{B} \vdash \overline{A}$ est aussi dérivable.*

Preuve. En effet, il est clair que pour chaque connecteur, la règle d'introduction gauche (resp. droite) est duale de la règle d'introduction droite (resp. gauche) du connecteur dual. Cette dualité des règles permet d'associer à toute *instance* d'une règle l'instance pseudo-duale, qui s'étend alors directement aux preuves. La preuve pseudo-duale d'une preuve de $A \vdash B$ est bien entendu une preuve de $\overline{B} \vdash \overline{A}$ □

Définition 2.1.3 *On appelle* ***calcul propositionnel catégorique*** *(resp.* ***symétrique*** *) l'ensemble des règles de déduction ci-dessus en excluant (resp. comprenant) les règles de la soustraction.*

Remarque. Il est connu (cf. [33] p. 47) que le calcul propositionnel catégorique correspond à la logique intuitionniste (cf. chapitre 3 pour une présentation de LJ), la restriction à une formule à gauche traduisant un regroupement conjonctif : au séquent intuitionniste usuel $A_1, \ldots, A_n \vdash B$ est associé le séquent $A_1 \wedge \ldots \wedge A_n \vdash B$.

Nous appelerons donc *logique intuitionniste soustractive* la logique définie par le calcul propositionnel catégorique symétrique. Nous montrerons en effet dans la section 3.4.2 que ce calcul est conservatif sur la logique propositionnelle intuitionniste.

Exemple. Nous avons vu (cf. la remarque de la section 1) que les règles de distributivité sont vérifiées dans toute algèbre de Heyting. Les règles de l'implication permettent de dériver directement la première des deux lois :

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{B \wedge C \vdash B \wedge C}{B \wedge C \vdash (B \wedge C) \vee (B \wedge A)}}{C \vdash B \Rightarrow ((B \wedge C) \vee (B \wedge A))} \quad \dfrac{\dfrac{B \wedge A \vdash B \wedge A}{B \wedge A \vdash (B \wedge C) \vee (B \wedge A)}}{A \vdash B \Rightarrow ((B \wedge C) \vee (B \wedge A))}}{C \vee A \vdash B \Rightarrow ((B \wedge C) \vee (B \wedge A))}}{B \wedge (C \vee A) \vdash B \wedge B \Rightarrow ((B \wedge C) \vee (B \wedge A))}}{B \wedge (C \vee A) \vdash (B \wedge C) \vee (B \wedge A)}$$

(les deux dernières règles appliquées ci-dessus n'appartiennent pas au calcul catégorique, mais se dérivent facilement). La soustraction permet de prouver directement la seconde règle de distributivité :

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{B \vee C \vdash B \vee C}{(B \vee C) \wedge (B \vee A) \vdash B \vee C}}{((B \vee C) \wedge (B \vee A)) - B \vdash C} \quad \dfrac{\dfrac{B \vee A \vdash B \vee A}{(B \vee C) \wedge (B \vee A) \vdash B \vee A}}{((B \vee C) \wedge (B \vee A)) - B \vdash A}}{((B \vee C) \wedge (B \vee A)) - B \vdash C \wedge A}}{((B \vee C) \wedge (B \vee A)) - B \vee B \vdash B \vee (C \wedge A)}}{(B \vee C) \wedge (B \vee A) \vdash B \vee (C \wedge A)}$$

Remarque. Il est possible de dériver la commutativité et l'associativité de la conjonction et de la disjonction à partir des règles de ces seuls connecteurs. Par contre, les règles de la conjonction et de la disjonction ne suffisent pas à dériver les règles de distributivité : ceci est dû à la restriction à une hypothèse et une conclusion dans les séquents (cf. sous-section 1.2 du chapitre 3).

2.2 Validité et complétude

Les résultats que nous présentons dans cette sous-section sont immédiats, il suffit de remarquer que l'axiomatique d'une pré-algèbre de Heyting-Brouwer est exactement la même que celle du calcul propositionnel catégorique symétrique.

Définition 2.2.1 *Étant données une algèbre de Heyting-Brouwer $\mathcal{A}$ et une valuation $\mathcal{V}$ qui associe à chaque atome propositionnel un élément de $\mathcal{A}$, on définit l'interprétation $\mathcal{I}$ d'une formule par récurrence :*

- $\mathcal{I}(A) \equiv \mathcal{V}(A)$ *si A est un atome propositionnel*
- $\mathcal{I}(\top) \equiv \top$, $\mathcal{I}(\bot) \equiv \bot$
- $\mathcal{I}(A \wedge B) \equiv \mathcal{I}(A) \sqcap \mathcal{I}(B)$
- $\mathcal{I}(A \vee B) \equiv \mathcal{I}(A) \sqcup \mathcal{I}(B)$
- $\mathcal{I}(A \Rightarrow B) \equiv \mathcal{I}(B) \uparrow \mathcal{I}(A)$

– $\mathcal{I}(A - B) \equiv \mathcal{I}(A) \downarrow \mathcal{I}(B)$

*Un séquent $A \vdash B$ est **valide** dans $\mathcal{A}$ ssi $\mathcal{I}(A) \leq \mathcal{I}(B)$.*

Remarque. Nous rappelons que la permutation des arguments lors de l'interpétation de l'implication est uniquement dû à nos choix de notation : $A \Rightarrow B$ correspond à la notation B^A des CCC et donc à la notation $B \uparrow A$ des algèbres de Heyting.

Théorème 2.2.2 (validité) *Tout séquent prouvable dans le calcul propositionnel catégorique symétrique est valide dans toute algèbre de Heyting-Brouwer.*

Preuve. L'interprétation de chaque axiome et de chaque règle du calcul propositionnel catégorique symétrique apparaît dans l'axiomatisation des algèbres de Heyting-Brouwer. □

Théorème 2.2.3 (complétude) *Tout séquent valide dans toute algèbre de Heyting-Brouwer est prouvable dans le calcul propositionnel catégorique symétrique.*

Preuve. L'ensemble $\mathcal{F}$ des formules propositionnelles avec soustraction muni de la relation binaire « $A \vdash B$ est un séquent prouvable » forme clairement une pré-algèbre de Heyting-Brouwer. Notons $\leq$ l'ordre engendré par ce préordre (ce qui revient à identifier les formules équivalentes) et considérons un séquent valide dans toutes les algèbres de Heyting-Brouwer. Ce séquent sera en particulier valide dans l'algèbre $(\mathcal{F}, \leq, \top, \bot, \wedge, \vee, \Rightarrow, -)$. Par conséquent, il est prouvable dans le calcul propositionnel catégorique symétrique. □

3 Sémantique bi-topologique

Une classe importante de modèles du calcul propositionnel intuitionniste est formée par les modèles topologiques. C'est en fait une sous-classe des algèbres de Heyting, toutefois suffisante pour obtenir la complétude (théorème 3.1.4). Nous rappelons tout d'abord la sémantique topologique, puis nous présentons une extension de cette sémantique permettant d'interpréter la soustraction.

3.1 Modèles topologiques

Nous rappelons tout d'abord la définition d'un espace topologique, ainsi que les notions habituelles « d'intérieur » et « d'adhérence ».

Définition 3.1.1 *Un **espace topologique** est la donnée d'un ensemble X et d'un sous-ensemble $\mathcal{O}$ des parties de X contenant $\emptyset$ et X, et clos par intersection finie et union quelconque. Un élément S de $\mathcal{O}$ est un « ouvert » et son complémentaire, noté S^c, est un « fermé » de l'espace topologique.*

Définition 3.1.2 *Étant donné un espace topologique $\mathcal{O}$ sur X, et S une partie de X, l'**intérieur** de S, noté $int(S)$, est l'union de tous les ouverts inclus dans S. L'**adhérence** de S, noté $adh(S)$, est par définition $(int(S^c))^c$, c'est-à-dire l'intersection des fermés contenant S.*

Nous pouvons maintenant rappeler la sémantique topologique d'une formule du calcul propositionnel. Chaque formule est interprétée par un ouvert.

Définition 3.1.3 *Étant donnés un espace topologique $\mathcal{O}$ sur X, une valuation $\mathcal{V}$ qui associe à chaque atome propositionnel un ouvert, on définit l'interprétation $[\![A]\!]$ d'une formule A par récurrence :*

- $[\![A]\!] \equiv \mathcal{V}(A)$ *si A est un atome propositionnel*
- $[\![\top]\!] \equiv X$, $[\![\bot]\!] \equiv \emptyset$
- $[\![A \wedge B]\!] \equiv [\![A]\!] \cap [\![B]\!]$
- $[\![A \vee B]\!] \equiv [\![A]\!] \cup [\![B]\!]$
- $[\![A \Rightarrow B]\!] \equiv int([\![A]\!]^c \cup [\![B]\!])$

*Un séquent $A \vdash B$ est **valide** dans $\mathcal{O}$ ssi $[\![A]\!] \subset [\![B]\!]$.*

Notation. En cas d'ambiguïté, nous noterons $[\![A]\!]_{\mathcal{O}}^{\mathcal{V}}$ (*i.e.* l'espace topologique est noté en indice et la valuation en exposant).

Remarque. Tout espace topologique $\mathcal{O}$ donne lieu à l'algèbre de Heyting $(\mathcal{O}, \subset, X, \emptyset, \cup, \cap, \uparrow)$ où $S_1 \uparrow S_2$ est défini par $int(S_1^c \cup S_2)$. Il est facile de vérifier que la sémantique définie dans le cadre des algèbres de Heyting correspond bien à la sémantique ci-dessus.

Théorème 3.1.4 (complétude et validité) *Une formule du calcul propositionnel est prouvable en logique intuitionniste si et seulement si elle est valide dans tout modèle topologique (resp. fini).*

Preuve. Cf. [66] p. 246, par exemple. □

3.2 Modèles bi-topologiques

Pour interpréter l'implication dans les espaces topologiques, nous avons utilisé la notion d'intérieur, qui existe en raison de la clôture de l'ensemble des ouverts par union quelconque. On aimerait que la soustraction, connecteur dual de l'implication, ait une sémantique duale de celle de l'implication. Malheureusement, le dual d'un espace topologique (défini comme l'ensemble de ses fermés) n'est pas un espace topologique puisqu'il n'est pas clos par union quelconque mais par intersection quelconque (c'est en fait une instance d'algèbre de Brouwer). Un moyen simple d'y remédier est de se placer dans des espaces bi-topologiques, c'est-à-dire clos par union *et* intersection quelconque.

⋆ Espaces bi-topologiques

Définition 3.2.1 *Un **espace bi-topologique** est la donnée d'un ensemble X est d'un sous-ensemble $\mathcal{O}$ des parties de X contenant $\emptyset$ et X, et clos par intersection et union quelconque.*

Définition 3.2.2 *Étant donné un espace bi-topologique $\mathcal{O}$ sur X, et S une partie de X, la **couverture** de S, notée $couv(S)$, est l'intersection de tous les ouverts contenant X. L'**inhérence** de S, noté $inh(S)$, est par définition $(couv(S^c))^c$, c'est-à-dire l'union des fermés inclus dans X.*

Remarques

– Pour tout ensemble X, le complémentaire définit une dualité bien connue sur $\mathcal{P}(X)$ vue comme une algèbre de Boole. Cette dualité s'étend aux espaces bi-topologiques, le dual de $(X, \mathcal{O})$ étant $(X, \mathcal{O}^\perp)$ où $\mathcal{O}^\perp$ est défini par :

$$\mathcal{O}^\perp = \{S^c : S \in \mathcal{O}\}$$

Les notions de couverture et d'inhérence se trouvent être exactement les notions duales de celles d'intérieur et d'adhérence, car elles font commuter les diagrammes suivants :

$$\begin{tikzcd} \mathcal{P}(X) \arrow[r, "c"] \arrow[d, "couv_{\mathcal{O}}"'] & \mathcal{P}(X) \arrow[d, "int_{\mathcal{O}^\perp}"] \\ \mathcal{O} \arrow[r, "c"'] & \mathcal{O}^\perp \end{tikzcd} \qquad \begin{tikzcd} \mathcal{P}(X) \arrow[r, "c"] \arrow[d, "inh_{\mathcal{O}}"'] & \mathcal{P}(X) \arrow[d, "adh_{\mathcal{O}^\perp}"] \\ \mathcal{O} \arrow[r, "c"'] & \mathcal{O}^\perp \end{tikzcd}$$

En effet, pour tout ensemble S de $\mathcal{P}(X)$:

$$(int_{\mathcal{O}^\perp}(S^c))^c = adh_{\mathcal{O}^\perp}(S)$$

or l'adhérence de S dans $\mathcal{O}^\perp$ est l'intersection des fermés de $\mathcal{O}^\perp$ qui contiennent S, c'est-à-dire aussi l'intersection des ouverts de $\mathcal{O}$ qui contiennent S, autrement dit $couv_{\mathcal{O}}(S)$. De même pour l'inhérence. Notez bien que ces notions sont distinctes de celles d'intérieur et d'adhérence puisque l'ensemble des ouverts n'est pas clos par complémentarité.

– Tout espace bi-topologique $\mathcal{O}$ donne lieu à l'algèbre de Heyting-Brouwer $(\mathcal{O}, \subset, X, \emptyset, \cup, \cap, \uparrow, \downarrow)$ où $S_1 \uparrow S_2$ est défini par $int(S_1^c \cup S_2)$ et $S_1 \downarrow S_2$ par $couv(S_1 \cap S_2^c)$. La sémantique définie dans le cadre des algèbres de Heyting-Brouwer correspond bien à la sémantique bi-topologique.

⋆ Interprétation de la soustraction

Nous sommes maintenant en mesure d'étendre l'interprétation topologique des formules à la soustraction. Remarquons tout d'abord que la sémantique des connecteurs est définie indépendamment de la valuation (en tant que fonction de $\mathcal{O} \times \mathcal{O}$ dans $\mathcal{O}$ pour les connecteurs $\wedge, \vee, \Rightarrow$ et en tant qu'éléments de $\mathcal{O}$ pour $\top, \bot$). En effet, pour tous ouverts S, T de $\mathcal{O}$:

– $[\![\top]\!]_{\mathcal{O}} = X$, $[\![\bot]\!]_{\mathcal{O}} = \emptyset$

– $[\![\wedge]\!]_{\mathcal{O}}(T, S) = T \cap S$

– $[\![\vee]\!]_{\mathcal{O}}(T, S) = T \cup S$

– $[\![\Rightarrow]\!]_{\mathcal{O}}(T, S) = int(T^c \cup S)$

Notons alors $[\![\ ]\!]_{\mathcal{O}}^\perp$ la sémantique duale d'un connecteur, c'est-à-dire, pour un connecteur Ξ n-aire, l'unique fonction $[\![\Xi]\!]_{\mathcal{O}}^\perp$ qui fait commuter le diagramme suivant :

$$\begin{tikzcd} \mathcal{O}^n \arrow[r, "c^n"] \arrow[d, "{[\![\Xi]\!]_{\mathcal{O}}^\perp}"'] & (\mathcal{O}^\perp)^n \arrow[d, "{[\![\Xi]\!]_{\mathcal{O}^\perp}}"] \\ \mathcal{O} \arrow[r, "c"'] & \mathcal{O}^\perp \end{tikzcd}$$

L'opération qui associe à la sémantique d'un connecteur sa sémantique duale est clairement involutive. En effet, le diagramme suivant commute, et $\mathcal{O}^{\perp\perp} = \mathcal{O}$:

$$\begin{tikzcd}
\mathcal{O}^n \arrow[r, "c^n"] \arrow[d, "{[\![\Xi]\!]^{\perp\perp}_{\mathcal{O}}}"'] & (\mathcal{O}^{\perp})^n \arrow[r, "c^n"] \arrow[d, "{[\![\Xi]\!]^{\perp}_{\mathcal{O}^{\perp}}}"] & (\mathcal{O}^{\perp\perp})^n \arrow[d, "{[\![\Xi]\!]_{\mathcal{O}^{\perp\perp}}}"] \\
\mathcal{O} \arrow[r, "c"'] & \mathcal{O}^{\perp} \arrow[r, "c"'] & \mathcal{O}^{\perp\perp}
\end{tikzcd}$$

Les connecteurs (0-aires) $\top$ et $\perp$ sont bien sémantiquement duaux (*i.e.* $[\![\perp]\!]^{\perp}_{\mathcal{O}} = [\![\top]\!]_{\mathcal{O}}$) puisque l'équation suivante est vérifiée :

$$([\![\top]\!]_{\mathcal{O}})^c = [\![\top]\!]_{\mathcal{O}^{\perp}}$$

De même, les connecteurs (binaires) $\wedge$ et $\vee$ sont aussi sémantiquement duaux (*i.e.* $[\![\wedge]\!]^{\perp}_{\mathcal{O}} = [\![\vee]\!]_{\mathcal{O}}$) puisque le diagramme suivant est commutatif :

$$\begin{tikzcd}
\mathcal{O} \times \mathcal{O} \arrow[r, "{(c,c)}"] \arrow[d, "{[\![\wedge]\!]_{\mathcal{O}}}"'] & \mathcal{O}^{\perp} \times \mathcal{O}^{\perp} \arrow[d, "{[\![\vee]\!]_{\mathcal{O}^{\perp}}}"] \\
\mathcal{O} \arrow[r, "c"'] & \mathcal{O}^{\perp}
\end{tikzcd}$$

On souhaite donc naturellement que les sémantiques de la soustraction et de l'implication soient aussi duales, ce qui s'exprime par l'équation $[\![\Leftarrow]\!]^{\perp}_{\mathcal{O}} = [\![-]\!]_{\mathcal{O}}$. Par conséquent, le diagramme suivant doit commuter :

$$\begin{tikzcd}
\mathcal{O} \times \mathcal{O} \arrow[r, "{(c,c)}"] \arrow[d, "{[\![-]\!]_{\mathcal{O}}}"'] & \mathcal{O}^{\perp} \times \mathcal{O}^{\perp} \arrow[d, "{[\![\Leftarrow]\!]_{\mathcal{O}^{\perp}}}"] \\
\mathcal{O} \arrow[r, "c"'] & \mathcal{O}^{\perp}
\end{tikzcd}$$

Autrement dit, pour tous ouverts S, T de $\mathcal{O}$,

$$([\![-]\!]_{\mathcal{O}}(S,T))^c = [\![\Rightarrow]\!]_{\mathcal{O}^{\perp}}(T^c, S^c)$$

C'est-à-dire :

$$([\![-]\!]_{\mathcal{O}}(S,T))^c = int_{\mathcal{O}^{\perp}}((T^c)^c \cup S^c) = (int_{\mathcal{O}^{\perp}}(T^c \cap S)^c) = (adh_{\mathcal{O}^{\perp}}(T^c \cap S))^c$$

et puisque $adh_{\mathcal{O}^{\perp}}(S) = couv_{\mathcal{O}}(S)$, on obtient donc la sémantique bi-topologique suivante pour la soustraction :

$$[\![A - B]\!] \equiv couv([\![A]\!] \cap [\![B]\!]^c)$$

Remarque. À toute valuation $\mathcal{V}$ sur $\mathcal{O}$ correspond naturellement une « valuation complémentaire » sur $\mathcal{O}^{\perp}$, notée $\mathcal{V}^c$, et définie par $\mathcal{V}^c(A) = \mathcal{V}(A)^c$ (qui interprète bien chaque atome par un fermé de $\mathcal{O}$, et donc un ouvert de $\mathcal{O}^{\perp}$). En notant $\mathcal{A}$ l'ensemble des atomes, le diagramme suivant commute donc :

$$\begin{tikzcd}
\mathcal{A} \arrow[r, "Id"] \arrow[d, "\mathcal{V}"'] & \mathcal{A} \arrow[d, "\mathcal{V}^c"] \\
\mathcal{O} \arrow[r, "c"'] & \mathcal{O}^{\perp}
\end{tikzcd}$$

L'opération $\mathcal{V} \longmapsto \mathcal{V}^c$ permet de compléter sémantiquement l'opération de pseudo-dualité sur les formules $F \longmapsto \overline{F}$. Il est alors possible de chercher la duale de l'interprétation des formules, c'est-à-dire la duale de l'opération $(\mathcal{V}, F) \longmapsto [\![F]\!]^{\mathcal{V}}_{\mathcal{O}}$. On remarque alors que cette opération est « autoduale » : le diagramme suivant est commutatif.

$$\begin{tikzcd}
\mathcal{O}^A \times \mathcal{F} \arrow[r, "(\mathcal{V},F)\mapsto(\mathcal{V}^c,\overline{F})"] \arrow[d, "{[\![\ ]\!]_{\mathcal{O}}}"'] & \mathcal{O}^A \times \mathcal{F} \arrow[d, "{[\![\ ]\!]_{\mathcal{O}^\perp}}"] \\
\mathcal{O} \arrow[r, "c"'] & \mathcal{O}^\perp
\end{tikzcd}$$

Une récurrence simple sur F permet en effet de vérifier que pour toute valuation $\mathcal{V}$:

$$([\![F]\!]^{\mathcal{V}}_{\mathcal{O}})^c = [\![\,\overline{F}\,]\!]^{\mathcal{V}^c}_{\mathcal{O}^\perp}$$

On montre alors que, comme dans les algèbres de Heyting-Brouwer, la dualité introduite dans les espaces topologiques interprète bien la dualité logique des connecteurs (la notion de séquent pseudo-dual a été introduite dans la définition 2.1.1) :

Proposition 3.2.3 *Étant donné un modèle bi-topologique $\mathcal{O}$ et une valuation $\mathcal{V}$ sur $\mathcal{O}$, le séquent $A \vdash B$ est valide, pour la valuation $\mathcal{V}$, dans $\mathcal{O}$ si et seulement si le séquent pseudo-dual $\overline{B} \vdash \overline{A}$ est valide, pour la valuation $\mathcal{V}^c$, dans $\mathcal{O}^\perp$.*

Preuve. En effet, $[\![A]\!]^{\mathcal{V}}_{\mathcal{O}} \subset [\![B]\!]^{\mathcal{V}}_{\mathcal{O}}$ ssi $([\![B]\!]^{\mathcal{V}}_{\mathcal{O}})^c \subset ([\![A]\!]^{\mathcal{V}}_{\mathcal{O}})^c$ ssi $[\![\,\overline{B}\,]\!]^{\mathcal{V}^c}_{\mathcal{O}^\perp} \subset [\![\,\overline{A}\,]\!]^{\mathcal{V}^c}_{\mathcal{O}^\perp}$ par la remarque ci-dessus. □

Ceci nous permet de donner une preuve sémantique de la propriété de contraposition pour la pseudo-dualité :

Corollaire 3.2.4 *Si le séquent $A \vdash B$ est valide dans tout modèle bi-topologique, alors le séquent $\overline{B} \vdash \overline{A}$ aussi.*

Preuve. Il suffit d'appliquer la proposition en remarquant que tout modèle bi-topologique est le dual de son dual. □

Avant d'étudier plus en détail les propriétés de la sémantique bi-topologique, il est naturel de se poser la question suivante : « Existe-t-il des instances non triviales d'espaces bi-topologiques ? ». La réponse est « oui » dans le sens où il existe des espaces bi-topologiques qui ne sont pas des algèbres de Boole (*i.e.* où tout ouvert n'est pas complémenté). Un espace bi-topologique est toutefois dégénéré dans le sens où il est toujours formé des sections finales d'un pré-ordre. Ce théorème est un résultat classique de topologie (cf. [7], p. 48) :

Théorème 3.2.5 *Un espace topologique $(X, \mathcal{O})$ est bi-topologique si et seulement si $\mathcal{O}$ est l'ensemble des section finales (ou l'ensemble des sections initiales) d'un pré-ordre sur X.*

Preuve. Il est facile de voir que l'ensemble des sections finales d'un pré-ordre sur X forme un espace bi-topologique sur X. Réciproquement, on définit la relation suivante sur X :

$$x \leq y \equiv \forall S \in \mathcal{O}(x \in S \Rightarrow y \in S)$$

Cette relation est clairement réflexive et transitive, c'est donc un pré-ordre. Montrons que les sections finales de ce pré-ordre sont exactement les ouverts de $\mathcal{O}$. Par définition du pré-ordre, tout ouvert est une section finale. Considérons maintenant une section finale S du pré-ordre et montrons que S est un ouvert. On note $V(x)$ l'intersection de tous les ouverts qui contiennent x (le « voisinage immédiat » de x). Il est clair que si $y \in V(x)$ alors $y \geq x$ car tout ouvert contenant x contient $V(x)$ et donc y. Par conséquent, pour tout $x \in S$, $V(x) \subset S$. On en déduit que $S = \bigcup_{x \in S} V(x)$ est un ouvert. □

Remarques

– En particulier, le dual d'un espace bi-topologique $\mathcal{O}$ est en correspondance avec le pré-ordre dual de celui qui définit $\mathcal{O}$. En effet,

$$\forall S \in \mathcal{O}(x \in S \Rightarrow y \in S) \text{ ssi } \forall S \in \mathcal{O}(y \notin S \Rightarrow x \notin S) \text{ ssi } \forall T \in \mathcal{O}^{\perp}(y \in T \Rightarrow x \in T)$$

Les ouverts de $\mathcal{O}^{\perp}$, qui sont les sections finales de $(X, \geq)$, sont donc les sections initiales de $(X, \leq)$.

– Dans l'espace bi-topologique défini par un pré-ordre $(X, \leq)$, l'intérieur d'une partie E de X, qui est la plus grande section finale incluse dans E, peut être définie par :

$$x \in \mathit{int}(E) \text{ ssi } \forall y \geq x(y \in E)$$

et par dualité, la couverture d'un ensemble, qui est la plus petite section finale contenant E, peut être définie par :

$$x \in \mathit{couv}(E) \text{ ssi } \exists y \leq x(y \in E)$$

De la même manière, on peut définir l'adhérence et l'inhérence de E par :

$$x \in \mathit{adh}(E) \text{ ssi } \exists y \geq x(y \in E) \qquad x \in \mathit{inh}(E) \text{ ssi } \forall y \leq x(y \in E)$$

3.3 Forcing de Kripke

Nous venons de montrer que tout espace bi-topologique est formé des sections finales d'un pré-ordre. Or il existe une interprétation de la logique intuitionniste dont les modèles (dus à Kripke) sont justement des pré-ordres muni d'une relation de « forcing » (notée $\Vdash$) entre les nœuds du pré-ordre et les formules. Cette relation doit vérifier la propriété suivante : lorsqu'un nœud force une formule, tout nœud plus grand force aussi la formule. Cette propriété de monotonie dit exactement que les formules sont interprétées par des sections finales du pré-ordre. Nous allons voir que le forcing de Kripke correspond exactement à l'interprétation bi-topologique de la section précédente. Voici tout d'abord la définition formelle d'un modèle de Kripke (remarquons qu'on ne se limite pas ici aux arbres, cf. la remarque qui suit le corollaire 4.1.2).

Définition 3.3.1 *Un **modèle de Kripke** est donné par un pré-ordre $(E, \leq)$ et une valuation $\mathcal{V}$ qui associe à tout atome propositionnel une section finale du pré-ordre. La relation de forcing est définie pour tout nœud α par récurrence sur la formule :*

– $\alpha \Vdash A \equiv \alpha \in \mathcal{I}(A)$*, si A est un atome propositionnel*

– $\alpha \Vdash \top$ *et* $\alpha \nVdash \bot$

– $\alpha \Vdash (A \vee B) \equiv \alpha \Vdash A$ *ou* $\alpha \Vdash B$

– $\alpha \Vdash (A \wedge B) \equiv \alpha \Vdash A$ *et* $\alpha \Vdash B$

– $\alpha \Vdash (A \Rightarrow B) \equiv$ *pour tout* $\beta \geq \alpha$ ($\beta \Vdash A$ *implique* $\beta \Vdash B$)

Proposition 3.3.2 *Étant donné un pré-ordre* $(E, \leq)$ *et une valuation* $\mathcal{V}$ *qui associe à tout atome propositionnel une section finale du pré-ordre, alors pour tout nœud* α *et toute formule* A *:*

$$\alpha \Vdash A \textit{ ssi } \alpha \in [\![A]\!]$$

où $[\![A]\!]$ *est l'interprétation de* A *dans l'espace bi-topologique défini par* $(E, \leq)$*.*

Preuve. Par récurrence, le seul cas non trivial etant celui de l'implication. Si l'on considère que l'interprétation des connecteurs du méta-langage est classique, et par hypothèse de récurrence,

$$(\beta \Vdash A \textit{ implique } \beta \Vdash B) \text{ ssi } (\beta \nVdash A \textit{ ou } \beta \Vdash B) \text{ ssi } \beta \in [\![A]\!]^c \cup [\![B]\!]$$

or par définition,

$$\alpha \Vdash (A \Rightarrow B) \equiv \textit{ pour tout } \beta \geq \alpha \ (\beta \Vdash A \textit{ implique } \beta \Vdash B)$$

d'où :

$$\alpha \Vdash (A \Rightarrow B) \text{ ssi } \textit{ pour tout } \beta \geq \alpha \ (\beta \in [\![A]\!]^c \cup [\![B]\!])$$

et par conséquent,

$$\alpha \Vdash (A \Rightarrow B) \text{ ssi } \alpha \in int([\![A]\!]^c \cup [\![B]\!])$$

par la remarque qui suit le théorème 3.2.5. □

⋆ Interprétation de la soustraction

Puisqu'un modèle de Kripke est exactement un modèle bi-topologique, la sémantique de la soustraction est donc définie :

$$\alpha \Vdash (A - B) \text{ ssi } \alpha \in couv([\![A]\!] \cap [\![B]\!]^c)$$

On obtient ainsi la sémantique de Kripke suivante pour la soustraction :

$$\alpha \Vdash (A - B) \equiv \textit{ il existe } \beta \leq \alpha \ (\beta \Vdash A \textit{ et } \beta \nVdash B)$$

Le proposition 3.3.2 s'étend alors par dualité à la soustraction. Autrement dit, les modèles de Kripke permettent toujours d'interpréter la soustraction : c'est de cette propriété que découle la conservativité sur la logique propositionnelle intuitionniste.

3.4 Conservativité sur la logique intuitionniste

Nous sommes maintenant en mesure de prouver le résultat annoncé en introduction du chapitre, à savoir la conservativité du calcul propositionnel catégorique avec soustraction sur le calcul propositionnel intuitionniste.

Lemme 3.4.1 (validité) *Toute formule prouvable dans le calcul propositionnel catégorique symétrique est valide dans tout modèle de Kripke.*

Preuve. Considérons une formule prouvable dans le calcul propositionnel catégorique symétrique. Cette formule est valide dans toute algèbre de Heyting-Brouwer, et en particulier dans tout modèle bi-topologique (*i.e.* dans tout modèle de Kripke). □

Théorème 3.4.2 (conservativité) *Toute formule ne contenant pas de soustraction prouvable dans le calcul propositionnel catégorique symétrique est prouvable en logique intuitionniste.*

Preuve. Considérons une formule prouvable dans le calcul propositionnel catégorique symétrique. Par le lemme 3.4.1, cette formule est valide dans tout modèle de Kripke. Le théorème de complétude pour les modèles de Kripke (cf. [66] p. 254) nous permet de conclure que ce séquent est prouvable en logique intuitionniste. □

Les travaux de C. Rauszer [54] nous permettent de clore cette sous-section sur le théorème de complétude pour les modèles de Kripke finis.

Théorème 3.4.3 (complétude) *Toute formule valide dans tout modèle de Kripke (resp. fini) est prouvable dans le calcul propositionnel catégorique symétrique.*

Preuve. Par le théorème de complétude de C. Rauszer (cf. [54] p. 244) toute formule valide dans tout modèle de Kripke fini est valide dans toute algèbre de Heyting-Brouwer. Le résultat s'obtient alors en appliquant le théorème 2.2.3 de complétude pour ces algèbres. □

3.5 Pas de théorème de déduction

Nous montrons ici le résultat annoncé à la fin du chapitre 1, à savoir que le théorème de déduction, vérifié dans le calcul propositionnel catégorique, ne l'est plus dans le calcul propositionnel catégorique symétrique. Tout d'abord, rappelons l'énoncé de ce théorème :

Théorème 3.5.1 *Dans le calcul propositionnel catégorique, pour toutes formules propositionnelles A et B, s'il existe une dérivation de $\top \vdash B$ dont toute feuille est soit un axiome, soit le séquent $\top \vdash A$, alors il existe une dérivation dans ce calcul de $A \vdash B$ (dont toutes les feuilles sont des axiomes).*

Preuve. Cf. [33] p. 51. □

Pour montrer que ce théorème n'est plus vérifié dans le calcul propositionnel catégorique symétrique, nous montrons qu'en présence de la soustraction, ce théorème permet de dériver le tiers-exclu. Pour cela, nous aurons besoin du lemme suivant (où $\neg A$ désigne $A \Rightarrow \bot$) :

Lemme 3.5.2 *Pour toute formule propositionnelle A, le séquent $\neg(A \vee \neg A) \vdash \bot$ est dérivable en logique intuitionniste.*

Preuve. En voici la preuve dans le calcul propositionnel catégorique :

$$\dfrac{\dfrac{\dfrac{\dfrac{\neg(A\vee\neg A)\wedge A\vdash\neg(A\vee\neg A)\qquad \dfrac{A\vdash A\vee\neg A}{\neg(A\vee\neg A)\wedge A\vdash A\vee\neg A}}{\neg(A\vee\neg A)\wedge A\vdash\bot}}{\neg(A\vee\neg A)\vdash\neg A}}{\neg(A\vee\neg A)\vdash A\vee\neg A}\qquad \neg(A\vee\neg A)\vdash\neg(A\vee\neg A)}{\neg(A\vee\neg A)\vdash\bot}$$

□

Corollaire 3.5.3 *Dans le calcul propositionnel catégorique symétrique, le théorème de déduction n'est pas vérifié.*

Remarque. Dans le calcul propositionnel catégorique symétrique, si le théorème de déduction était vérifié, son dual le serait aussi. En effet, étant donnée une dérivation de $B\vdash\bot$ dont toute feuille est soit un axiome, soit le séquent $A\vdash\bot$, par dualité on obtient une preuve de $\top\vdash\overline{B}$ dont toute feuille est soit un axiome, soit le séquent $\top\vdash\overline{A}$. Par le théorème de déduction, il existe une dérivation de $\overline{B}\vdash\overline{A}$, et donc, à nouveau par dualité, une preuve de $A\vdash B$.

Preuve. Considérons la dérivation suivante, où la preuve de $\neg(A\vee\neg A)\vdash\bot$ est celle donnée dans le lemme ci-dessus :

$$\dfrac{\dfrac{A\vee\neg A\vdash\bot}{\top\vdash\neg(A\vee\neg A)}\qquad \begin{array}{c}\vdots\\ \neg(A\vee\neg A)\vdash\bot\end{array}}{\top\vdash\bot}$$

Par la remarque précédente, si le théorème de déduction était vérifié dans le calcul propositionnel catégorique symétrique, son dual le serait aussi. Appliquons donc ce dernier à la dérivation prédédente de $\top\vdash\bot$ à partir de $A\vee\neg A\vdash\bot$: il devrait exister une dérivation de $\top\vdash A\vee\neg A$, ce qui contredit la conservativité du calcul propositionnel catégorique symétrique sur la logique intuitionniste. □

Remarque. La théorème de déduction et son dual peuvent être formulés sous forme de règles de déduction qui déchargent le séquent hypothèse :

$$\begin{array}{c}[\top\vdash A]\\ \vdots\\ \dfrac{\top\vdash B}{A\vdash B}\end{array}\qquad\qquad \begin{array}{c}[A\vdash\bot]\\ \vdots\\ \dfrac{B\vdash\bot}{B\vdash A}\end{array}$$

Nous avons vu dans la preuve précédente que le calcul obtenu en ajoutant ces règles au calcul propositionnel catégorique symétrique est classique, voici par exemple une preuve directe de $\neg\neg A\vdash A$:

$$\dfrac{\dfrac{\dfrac{\dfrac{[A\vdash\bot]^1}{\top\vdash\neg A}\qquad [\top\vdash\neg\neg A]^2}{\top\vdash\bot}}{\top\vdash A}(1)}{\neg\neg A\vdash A}(2)$$

Ce calcul étend en fait l'isomorphisme de Curry-Howard au lambda-calcul symétrique de A. Filinski. Il est en effet facile d'établir la correspondance entre ces règles de déduction et le système de typage décrit dans [18].

4 Négation faible

4.1 Négation intuitionniste et négation faible

En logique intuitionniste, il est habituel de définir la négation $\neg A$ (nous l'appellerons « négation intuitionniste ») comme $A \Rightarrow \bot$. On obtient alors les règles dérivées du calcul propositionnel catégorique suivantes (l'axiome est aussi appelé « règle de la contradiction ») :

$$\frac{B \wedge A \vdash \bot}{B \vdash \neg A} \qquad A \wedge \neg A \vdash \bot$$

Les sémantiques topologiques et de Kripke de la négation sont elles aussi dérivées à partir de celles de l'implication et de l'absurde :

$$[\![\neg A]\!] \equiv int([\![A]\!]^c) \quad \text{et} \quad \alpha \Vdash \neg A \equiv \forall \beta \geq \alpha (\beta \nVdash A)$$

Par dualité, en logique soustractive, il est possible de définir une nouvelle négation, notée $\sim$ (que nous appellerons négation faible) en posant $\sim A \equiv \top - A$ et dont la sémantique bi-topologique (*i.e.* de Kripke) est donnée par :

$$[\![\sim A]\!] \equiv couv([\![A]\!]^c) \quad \text{et} \quad \alpha \Vdash \sim A \equiv \exists \beta \leq \alpha (\beta \nVdash A)$$

et qui vérifie les règles dérivées (l'axiome est aussi appelé « règle du tiers-exclu pour la négation faible ») duales de celles de la négation intuitionniste :

$$\frac{\top \vdash A \vee B}{\sim A \vdash B} \qquad \top \vdash \sim A \vee A$$

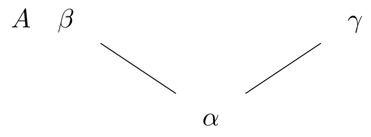
Remarque. Puisque la négation intuitionniste vérifie la règle de la contradiction et la négation faible la règle du tiers-exclu, il suffit d'identifier les deux négations pour obtenir la logique classique. Du point de vue sémantique, identifier l'intérieur et la couverture de $[\![A]\!]^c$ revient à dire que $[\![A]\!]$ est un ouvert-fermé. L'ensemble des ouverts-fermés d'un espace bi-topologique est par définition clos par complémentarité : il forme donc une algèbre de Boole.

Avant de détailler certaines propriétés de la négation faible, nous montrons que ce nouveau connecteur n'est pas définissable à partir des autres connecteurs intuitionnistes. Ce résultat, qui découle directement du corollaire 4.1.2, a bien sûr pour conséquence la non définissabilité de la soustraction (qui peut aussi être obtenue comme corollaire de la non-conservativité au premier ordre, cf. section 6).

Lemme 4.1.1 *Le séquent $\top \vdash \neg\neg A \vee \neg A$ n'est pas dérivable en logique intuitionniste, mais il est valide dans tous les modèles de Kripke ayant un plus grand élément.*

Preuve. Considérons un modèle de Kripke ayant un plus grand élément σ. Dans ce modèle, soit $\sigma \Vdash A$ soit $\sigma \nVdash A$. Dans le premier cas, aucun α ne peut forcer $\neg A$ (puisque $\alpha \leq \sigma$), donc tout α force $\neg\neg A$. Dans le second cas, aucun α ne peut forcer A (puisque $\alpha \leq \sigma$) et donc tout α force $\neg A$. Dans les deux cas, tout α force $\neg\neg A \vee \neg A$.

La formule $\neg\neg A \vee \neg A$ n'est toutefois pas un théorème intuitionniste puisque elle n'est pas valide dans le modèle suivant :



En effet, α ne force pas $\neg A$ puisque β force A, et α ne force pas non plus $\neg\neg A$ puisque γ force $\neg A$. □

Corollaire 4.1.2 (dual du lemme 4.1.1) *Le séquent $\sim\sim A \wedge \sim A \vdash \bot$ n'est pas un théorème de la logique intuitionniste soustractive, mais il est valide dans tous les modèles de Kripke ayant un plus petit élément.*

Preuve. Par la proposition 3.2.3 (puisqu'un modèle de Kripke a un plus petit élément si et seulement si son dual a un plus grand élément). □

Remarque. Les arbres ne suffisent donc plus à obtenir la complétude en présence de la négation faible (et *a fortiori* de la soustraction).

4.2 Propriétés de la négation faible

Nous nous concentrons ici uniquement sur les propriétés de la négation faible. Provisoirement, pour faciliter la lecture, les preuves seront effectuées en déduction naturelle intuitionniste, munie d'une nouvelle négation vérifiant l'axiome du tiers-exclu $A \vee \sim A$. La traduction de ces preuves vers le calcul catégorique symétrique ne pose pas de difficulté : c'est la traduction habituelle vers le calcul propositionnel catégorique, puisque l'axiome du tiers-exclu pour la négation faible n'a pas à être traduit. Cet axiome est équivalent à la règle suivante :

$$\dfrac{\begin{array}{c}[A]\\ \vdots\\ B\end{array} \qquad \begin{array}{c}[\sim A]\\ \vdots\\ B\end{array}}{B}$$

En effet, la preuve de gauche ci-dessous dérive l'axiome à partir de la règle, la preuve de droite dérive la règle à partir de l'axiome (et la règle du $\vee$ en déduction naturelle) :

$$\dfrac{\dfrac{[A]^1}{A \vee \sim A} \qquad \dfrac{[\sim A]^1}{A \vee \sim A}}{A \vee \sim A}(1) \qquad\qquad \dfrac{A \vee \sim A \qquad \begin{array}{c}[A]^1\\ \vdots\\ B\end{array} \qquad \begin{array}{c}[\sim A]^1\\ \vdots\\ B\end{array}}{B}(1)$$

Nous justifions tout d'abord la terminologie « négation faible » en montrant qu'elle est effectivement plus faible que la négation intuitionniste (*i.e.* $\neg A \vdash \sim A$). La justification sémantique est évidente, puisque pour tout ensemble E, $int(E) \subset couv(E)$. En voici une preuve syntaxique :

$$\dfrac{\dfrac{\dfrac{[A]^1 \qquad \neg A}{\bot}}{\sim A} \qquad\qquad [\sim A]^1}{\sim A}\,(1)$$

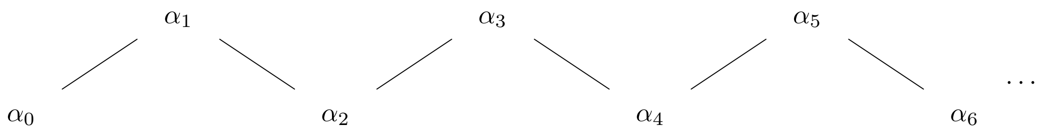
Comme conséquence, (et en appliquant la dualité) on obtient les implications suivantes :

$$\neg\sim A \vdash \sim\sim A \vdash A \vdash \neg\neg A \vdash \sim\neg A$$

Ces implications sont strictes. En effet, il est connu que $A \vdash \neg\neg A$ est une implication stricte (sa réciproque donne la logique classique), sa duale $\sim\sim A \vdash A$ l'est donc aussi. D'autre part, si l'équivalence $\neg\neg A \Leftrightarrow \sim\neg A$ était vérifiée, puisque $\neg\neg\neg A \Leftrightarrow \neg A$, toutes les formules de la chaîne (infinie) d'implications suivante seraient équivalentes :

$$\ldots \vdash \neg\sim\neg\sim\neg\sim A \vdash \neg\sim\neg\sim A \vdash \neg\sim A$$

Or *ces implications sont toutes strictes*. En effet, considérons le modèle de Kripke infini (à droite) suivant :



La sémantique de $\neg\sim A$ est $int(couv(\llbracket A\rrbracket^c)^c) = int(inh(\llbracket A\rrbracket))$. Si on prend comme interprétation de A l'ensemble de tous les nœuds sauf α_0 (qui est bien une section finale), l'inhérence de cet ensemble est obtenu en supprimant le nœud α_1, puis l'intérieur de l'ensemble restant est obtenu en supprimant le α_2. En continuant ainsi à prendre alternativement l'inhérence puis l'intérieur de l'ensemble restant, on supprime à chaque fois l'élément supérieur puis l'élément inférieur gauche. Chaque formule de la chaîne ci-dessus est donc interprétée par un ensemble contenant deux éléments de moins que l'interprétation de la formule qui est à sa droite. Ces implications sont donc toutes strictes.

⋆ Négation faible et implication intuitionniste

Nous allons maintenant utiliser la négation faible pour encadrer l'implication intuitionniste. La conséquence $(\neg A \vee B) \vdash (A \Rightarrow B)$ est un résultat intuitionniste habituel, en voici une preuve :

$$\dfrac{\neg A \vee B \qquad \dfrac{\dfrac{\dfrac{[\neg A]^1 \quad [A]^2}{\bot}}{B}}{A \Rightarrow B}\,(2) \qquad \dfrac{[B]^1}{A \Rightarrow B}}{A \Rightarrow B}\,(1)$$

La négation faible permet de compléter l'encadrement par $(A \Rightarrow B) \vdash (\sim A \vee B)$ dont voici la preuve :

$$\dfrac{\dfrac{\dfrac{A \Rightarrow B \quad [A]^1}{B}}{\sim A \vee B} \qquad \dfrac{[\sim A]^1}{\sim A \vee B}}{\sim A \vee B}\,(1)$$

Par conséquent,

$$(\neg A \vee B) \vdash (A \Rightarrow B) \vdash ({\sim}A \vee B)$$

Remarque. Si l'on revient au calcul propositionnel soustractif, par dualité, on obtient l'encadrement suivant :

$$(B \wedge \neg A) \vdash (B - A) \vdash (B \wedge {\sim}A)$$

À nouveau, ces encadrements sont stricts puisque d'une part il est clair (prendre $A = B$) que $(A \Rightarrow B) \nvdash (\neg A \vee B)$ et d'autre part la soustraction n'est pas définissable à partir des autres connecteurs, donc $(B - A) \nvdash (B \wedge \neg A)$. Les autres conséquences sont strictes par dualité.

4.3 Application de la négation faible

Nous considérons maintenant la logique classique définie en ajoutant, à la déduction naturelle intuitionniste habituelle une nouvelle négation, notée $^\perp$, vérifiant les règles du tiers-exclu et de la contradiction :

$$\dfrac{\begin{matrix}[A] \\ \vdots \\ B\end{matrix} \qquad \begin{matrix}[A^\perp] \\ \vdots \\ B\end{matrix}}{B} \qquad\qquad \dfrac{A \qquad A^\perp}{\perp}$$

Puisque nous avons par ailleurs à notre disposition un calcul intuitionniste muni d'une négation faible qui vérifie la règle du tiers-exclu et de la négation intuitionniste qui vérifie la règle de la contradiction :

$$\dfrac{\begin{matrix}[A] \\ \vdots \\ B\end{matrix} \qquad \begin{matrix}[{\sim}A] \\ \vdots \\ B\end{matrix}}{B} \qquad\qquad \dfrac{A \qquad \neg A}{\perp}$$

Nous pouvons alors tenter de traduire directement une preuve de la logique classique en une preuve intuitionniste avec négation faible simplement en remplacant la négation classique par la négation adéquate (intuitionniste ou faible) selon son occurence dans la preuve (règle de la contradiction ou règle du tiers-exclu), puis en propageant le remplacement dans la preuve. Cette propagation peut bien sûr échouer si la même négation classique apparaît à la fois dans une instance de la règle de la contradiction et dans une instance de la règle du tiers-exclu. Voici deux exemples d'application de cet algorithme informel, un cas de succès et un cas d'échec.

Nous donnerons une traduction formelle des preuves (qui n'échoue jamais) dans le chapitre 3 section 3. La négation classique de A est notée $\mathrm{A}^\perp$. La traduction de la preuve de $A^{\perp\perp} \Rightarrow A$ nous donne une preuve de $\neg{\sim}A \Rightarrow A$ (dont nous savions déjà que c'était un théorème).

$$\dfrac{\dfrac{[A]^2 \qquad \dfrac{\dfrac{[A^\perp]^2 \qquad [A^{\perp\perp}]^1}{\perp}}{A}}{A}\,(2)}{A^{\perp\perp} \Rightarrow A}\,(1) \quad \rightsquigarrow \quad \dfrac{\dfrac{[A]^2 \qquad \dfrac{\dfrac{[{\sim}A]^2 \qquad [\neg{\sim}A]^1}{\perp}}{A}}{A}\,(2)}{\neg{\sim}A \Rightarrow A}\,(1)$$

La preuve de l'axiome de Pierce est bien sûr intraduisible puisque cet axiome ne contient aucune négation et que la logique soustractive est conservative sur la logique intuitionniste.

L'algorithme échoue, pour la preuve suivante, en raison d'une occurence (numérotée (2) en exposant) de $(A \Rightarrow B)^\perp$ à la fois dans une instance de la règle de la contradiction et dans l'instance de la règle du tiers-exclu indicée par (2).

$$\dfrac{\dfrac{\dfrac{[(A \Rightarrow B) \Rightarrow A]^1 \qquad [A \Rightarrow B]^2}{A} \qquad \dfrac{[A]^3 \qquad \dfrac{\dfrac{[(A \Rightarrow B)^\perp]^2 \qquad \dfrac{\dfrac{\dfrac{[A^\perp]^3 \qquad [A]^4}{\perp}}{B}}{A \Rightarrow B}(4)}{\perp}}{A}}{A}(3)}{A}(2)}{((A \Rightarrow B) \Rightarrow A) \Rightarrow A}$$

4.4 Équiprouvabilité *versus* équivalence

Nous savons que l'axiome $A \vdash \neg{\sim}A$ mène à la logique classique puisque $\neg{\sim}A \vdash {\sim}{\sim}A$ est dérivable (et $A \vdash {\sim}{\sim}A$ est le dual de $\neg\neg A \vdash A$). Pourtant, la propriété remarquable suivante est vérifiée dans le calcul propositionnel catégorique symétrique :

Proposition 4.4.1 *Dans le calcul propositionnel catégorique symétrique, si $A \Rightarrow B$ est un théorème alors $\neg(A - B)$ est aussi un théorème. En particulier, si B est un théorème, $\neg{\sim}B$ est aussi un théorème.*

Preuve. Nous aurons besoin de la règle dérivée suivante :

$$\dfrac{\top \vdash A \Rightarrow B}{A \vdash B}$$

qui peut se dériver ainsi :

$$\dfrac{\dfrac{A \vdash \top \qquad A \vdash A}{A \vdash \top \wedge A} \qquad \dfrac{\dfrac{\top \wedge A \vdash \top \qquad \top \vdash A \Rightarrow B}{\top \wedge A \vdash A \Rightarrow B} \qquad \top \wedge A \vdash A}{\top \wedge A \vdash B}}{A \vdash B}$$

Voici maintenant la dérivation de $\neg(A - B)$, où la dérivation de $A \vdash B$ est celle donnée ci dessus :

$$\dfrac{\dfrac{\top \wedge (A - B) \vdash (A - B) \qquad \dfrac{\dfrac{\begin{array}{c}\vdots \\ A \vdash B\end{array} \qquad B \vdash \perp \vee B}{A \vdash \perp \vee B}}{(A - B) \vdash \perp}}{\top \wedge (A - B) \vdash \perp}}{\top \vdash \neg(A - B)}$$

En particulier, en prenant $A = \top$, dans cette dernière preuve, on obtient que si B est un théorème alors $\neg{\sim}B$ est aussi un théorème. Etant donné que le séquent $\neg{\sim}B \vdash B$ est toujours dérivable (cf. ci-dessus), on voit que B et $\neg{\sim}B$ sont équiprouvables. □

Corollaire 4.4.2 *Dans le calcul propositionnel catégorique symétrique, les formules $A \Rightarrow B$ et $\neg(A - B)$ sont équiprouvables (en particulier, les formules B et $\neg{\sim}B$ sont équiprouvables).*

Preuve. En effet, le séquent $\neg(A - B) \vdash A \Rightarrow B$ est dérivable : :

$$\dfrac{\dfrac{A \vdash (A - B) \vee B}{\neg(A - B) \wedge A \vdash B}}{\neg(A - B) \vdash A \Rightarrow B}$$

où la première règle appliquée est une instance de la règle dérivée :

$$\dfrac{A \vdash B \vee C}{\neg B \wedge A \vdash C}$$

qui peut se prouver ainsi :

$$\dfrac{\dfrac{\neg B \wedge A \vdash \neg B \qquad \dfrac{\neg B \wedge A \vdash A \qquad A \vdash (B \vee C)}{\neg B \wedge A \vdash (B \vee C)}}{\neg B \wedge A \vdash \neg B \wedge (B \vee C)} \qquad \neg B \wedge (B \vee C) \vdash C}{\neg B \wedge A \vdash C}$$

où le séquent $\neg B \wedge (B \vee C) \vdash C$ se dérive en utilisant la règle de distributivité vue en sous-section 2.1 :

$$\dfrac{\neg B \wedge (B \vee C) \vdash (\neg B \wedge B) \vee (\neg B \wedge C) \qquad \dfrac{\dfrac{\neg B \wedge B \vdash \bot \qquad \bot \vdash C}{\neg B \wedge B \vdash C} \qquad \neg B \wedge C \vdash C}{(\neg B \wedge B) \vee (\neg B \wedge C) \vdash C}}{\neg B \wedge (B \vee C) \vdash C}$$

□

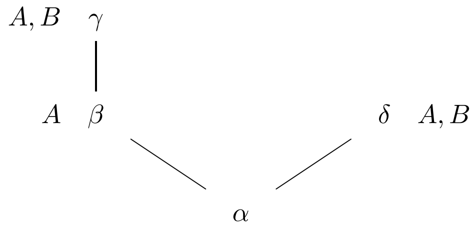
Remarque. Par dualité, on obtient que si $A \vdash \bot$ alors $\neg{\sim}A \vdash \bot$, d'où il résulte que A et $\neg{\sim}A$ sont « équiréfutables ».

4.5 Non définissabilité de la soustraction

Nous avons déjà vu que la soustraction n'est pas définissable à partir des autre connecteurs intuitionnistes (puisque la négation faible ne l'est pas, cf. corollaire 4.1.2). Nous montrons ici que la soustraction n'est pas définissable en logique intuitionniste à partir des autres connecteurs *et de la négation faible*. La preuve de ce résultat qui s'appuie sur le lemme de validité dans les modèles de Kripke est un corollaire direct de la proposition suivante :

Proposition 4.5.1 *Il existe un modèle de Kripke interprétant deux atomes propositionnels A et B dans lequel toute formule construite à partir de $\top$, $\bot$, A, B et des connecteurs $\vee$, $\wedge$, $\Rightarrow, \neg, \sim$ a une sémantique différente de celle de $A - B$.*

Preuve. Considérons le modèle de Kripke suivant (les nœuds sont étiquetés par les atomes forcés) :



La sémantique de $A - B$ dans ce modèle est :

$$[\![A - B]\!] = couv(\{\beta,\gamma,\delta\} \cap \{\gamma,\delta\}^c) = couv(\{\beta,\gamma,\delta\} \cap \{\alpha,\beta\}) = couv(\{\beta\}) = \{\beta,\gamma\}$$

Notons N l'ensemble de tous les nœuds. Nous allons montrer qu'aucune autre formule construite à partir de $\top$, $\bot$, A, B et des connecteurs $\vee$, $\wedge$, $\Rightarrow, \neg, \sim$ n'est interprétée par $\{\beta,\gamma\}$. Pour cela, il suffit de montrer que l'ensemble de sections finales suivant :

$$\mathcal{E} = \{\emptyset, \{\gamma,\delta\}, \{\beta,\gamma,\delta\}, N\}$$

(qui sont respectivement les interprétations de $\bot$, B, A et $\top$) est clos par les sémantiques des connecteurs $\vee,\wedge,\Rightarrow,\neg,\sim$. La négation intuitionniste est un cas particulier de l'implication. D'autre part, l'ensemble $\mathcal{E}$ est clairement clos par union et intersection. Il reste donc à vérifier que pour tous éléments U, V, W de $\mathcal{E}$, $int(U^c \cup V)$ et $couv(W^c)$ sont aussi des éléments de $\mathcal{E}$. C'est clair dans les cas $U = V, U = \emptyset, U = N, V = N, W = \emptyset, W = N$. Les cas restants sont :

- $int(\{\gamma,\delta\}^c \cup \emptyset) = int(\{\gamma,\delta\}^c) = int(\{\alpha,\beta\}) = \emptyset \in \mathcal{E}$
- $int(\{\beta,\gamma,\delta\}^c \cup \emptyset) = int(\{\beta,\gamma,\delta\}^c) = int(\{\alpha\}) = \emptyset \in \mathcal{E}$
- $int(\{\gamma,\delta\}^c \cup \{\beta,\gamma,\delta\}) = int(\{\alpha,\beta\} \cup \{\beta,\gamma,\delta\}) = int(N) = N \in \mathcal{E}$
- $int(\{\beta,\gamma,\delta\}^c \cup \{\gamma,\delta\}) = int(\{\alpha\} \cup \{\gamma,\delta\}) = int(\{\alpha,\gamma,\delta\}) = \{\gamma,\delta\} \in \mathcal{E}$
- $couv(\{\gamma,\delta\}^c) = couv(\{\alpha,\beta\}) = N \in \mathcal{E}$
- $couv(\{\beta,\gamma,\delta\}^c) = couv(\{\alpha\}) = N \in \mathcal{E}$

□

5 Dualité explicite

En logique propositionnelle intuitionniste soustractive, à chaque connecteur correspond un connecteur dual, ce qui permet de définir les notions de formule et de séquent pseudo-dual (cf. la définition 2.1.1 de la sous-section 2.1). On peut alors montrer syntaxiquement, ou sémantiquement (cf. corollaire 3.2.4), la propriété dite de contraposition :

$$A \vdash B \quad \text{implique} \quad \overline{B} \vdash \overline{A}$$

Toutefois, cette pseudo-dualité ne reflète pas la dualité booléenne classique (représentée par la négation) puisque les atomes sont inchangés par cette traduction (en effet, si l'on note $^\perp$ la négation classique et si A et B sont deux atomes, on a $(A \vee B)^\perp \equiv A^\perp \wedge B^\perp$). Il est facile de résoudre ce problème en supposant définie une dualité sur les atomes. La dualité des atomes s'étend alors par récursivement formules, en exploitant la dualité des connecteurs.

Cette dualité syntaxique s'interprète de plus directement par une dualité sémantique. Il est donc naturel de chercher à axiomatiser (*i.e.* internaliser) cette dualité. C'est l'objet de cette section, dont le résultat principal est la conservativité sur le calcul propositionnel catégorique soustractif (cf. corollaire 5.3.2).

5.1 Dualité sur les atomes

On considère maintenant une dualité σ (*i.e.* une involution) sur l'ensembles des atomes propositionnels. Cette dualité permet alors d'associer à toute valuation $\mathcal{V}$, sur un espace bi-topologique $\mathcal{O}$, une valuation duale notée $\mathcal{V}^{\perp}$ simplement en posant :

$$\mathcal{V}^{\perp}(A) = \mathcal{V}(\sigma(A))$$

Cette valuation duale permet alors de définir récursivement la sémantique duale, notée $[\![\ ]\!]^{\perp}$, d'une formule (à partir de la sémantique duale des connecteurs introduite en 3.2) :

- $[\![A]\!]^{\perp}_{\mathcal{O}} = \mathcal{V}^{\perp}(A)$, si A est un atome propositionnel
- $[\![A \wedge B]\!]^{\perp}_{\mathcal{O}} = [\![\wedge]\!]^{\perp}_{\mathcal{O}}([\![A]\!]^{\perp}_{\mathcal{O}}, [\![B]\!]^{\perp}_{\mathcal{O}})$
- $[\![A \vee B]\!]^{\perp}_{\mathcal{O}} = [\![\vee]\!]^{\perp}_{\mathcal{O}}([\![A]\!]^{\perp}_{\mathcal{O}}, [\![B]\!]^{\perp}_{\mathcal{O}})$
- $[\![A \Rightarrow B]\!]^{\perp}_{\mathcal{O}} = [\![\Rightarrow]\!]^{\perp}_{\mathcal{O}}([\![B]\!]^{\perp}_{\mathcal{O}}, [\![A]\!]^{\perp}_{\mathcal{O}})$
- $[\![A - B]\!]^{\perp}_{\mathcal{O}} = [\![-]\!]^{\perp}_{\mathcal{O}}([\![B]\!]^{\perp}_{\mathcal{O}}, [\![A]\!]^{\perp}_{\mathcal{O}})$

Notation. Nous rappelons que la permutation des arguments, pour l'implication et la soustraction, lors du passage au dual, est uniquement due à la notation employée.

Il est bien sûr aussi possible de définir syntaxiquement la notion de formule duale (et non plus pseudo-duale) obtenue en remplaçant chaque connecteur par son dual et chaque atome par son dual. Mieux encore, cette définition peut être internalisée dans le système formel : on obtient alors un opérateur de dualité explicite.

5.2 Opérateur de dualité explicite

On considère toujours la dualité σ définie sur l'ensembles des atomes propositionnels. Cette dualité présupposée va permettre d'amorcer la définition par récurrence (*i.e.* l'axiomatisation) de l'opérateur de dualité explicite (noté *) :

Définition 5.2.1 *Le* ***calcul propositionnel catégorique à dualité explicite*** *est défini à partir du calcul catégorique symétrique et d'un nouveau connecteur unaire, noté * et appelé opérateur de dualité explicite, axiomatisé par :*

- $A^* \dashv\vdash \sigma(A)$, *si A est un atome propositionnel*
- $(A \wedge B)^* \dashv\vdash A^* \vee B^*$
- $(A \vee B)^* \dashv\vdash A^* \wedge B^*$
- $(A \Rightarrow B)^* \dashv\vdash B^* - A^*$
- $(A - B)^* \dashv\vdash B^* \Rightarrow A^*$

où $A \dashv\vdash B$ est une abréviation pour les deux axiomes $A \vdash B$ et $B \vdash A$.

Remarque. Une récurrence facile montre que l'opérateur de dualité est involutif (*i.e.* pour tout A, $A^{**} \dashv\vdash A$). D'autre part, les connecteurs $\top$, $\wedge$, $\Rightarrow$ et * suffisent clairement à définir les autres connecteurs (en particulier la soustraction est définissable).

Cet opérateur de dualité explicite vérifie bien sûr la propriété de contraposition :

Proposition 5.2.2 *Dans le calcul propositionnel catégorique à dualité explicite, si le séquent $A \vdash B$ est dérivable, alors le séquent dual $B^* \vdash A^*$ est aussi dérivable.*

Preuve. Par les axiomes ci-dessus, on montre que B^* et A^* sont respectivement équivalentes à des formules B' et A' obtenues en substituant dans $\overline{B}$ et $\overline{A}$ tous les atomes par leurs duaux. Le séquent dual $B' \vdash A'$ et donc dérivable en tant qu'instance du séquent pseudo-dual $\overline{B} \vdash \overline{A}$. $\square$

5.3 Sémantique

Soit $\mathcal{V}$ une valuation des atomes propositionnels dans un espace bi-topologique $\mathcal{O}$. On souhaite étendre la sémantique $[\![\ ]\!]_O$ vue en section 3 au nouveau connecteur * de dualité explicite de sorte à ce que l'on ait :

$$[\![A^*]\!]_{\mathcal{O}} = [\![A]\!]^{\perp}_{\mathcal{O}}$$

Cela nous amène à définir les deux sémantiques $[\![\ ]\!]_{\mathcal{O}}$ et $[\![\ ]\!]^{\perp}_{\mathcal{O}}$ par une récurrence simultanée :

– $[\![A]\!]_{\mathcal{O}} = \mathcal{V}(A)$, si A est un atome	– $[\![A]\!]^{\perp}_{\mathcal{O}} = \mathcal{V}^{\perp}(A)$, si A est un atome
– $[\![A \wedge B]\!]_{\mathcal{O}} = [\![\wedge]\!]_{\mathcal{O}}([\![A]\!]_{\mathcal{O}}, [\![B]\!]_{\mathcal{O}})$	– $[\![A \wedge B]\!]^{\perp}_{\mathcal{O}} = [\![\wedge]\!]^{\perp}_{\mathcal{O}}([\![A]\!]^{\perp}_{\mathcal{O}}, [\![B]\!]^{\perp}_{\mathcal{O}})$
– $[\![A \vee B]\!]_{\mathcal{O}} = [\![\vee]\!]_{\mathcal{O}}([\![A]\!]_{\mathcal{O}}, [\![B]\!]_{\mathcal{O}})$	– $[\![A \vee B]\!]^{\perp}_{\mathcal{O}} = [\![\vee]\!]^{\perp}_{\mathcal{O}}([\![A]\!]^{\perp}_{\mathcal{O}}, [\![B]\!]^{\perp}_{\mathcal{O}})$
– $[\![A \Rightarrow B]\!]_{\mathcal{O}} = [\![\Rightarrow]\!]_{\mathcal{O}}([\![B]\!]_{\mathcal{O}}, [\![A]\!]_{\mathcal{O}})$	– $[\![A \Rightarrow B]\!]^{\perp}_{\mathcal{O}} = [\![\Rightarrow]\!]^{\perp}_{\mathcal{O}}([\![B]\!]^{\perp}_{\mathcal{O}}, [\![A]\!]^{\perp}_{\mathcal{O}})$
– $[\![A - B]\!]_{\mathcal{O}} = [\![-]\!]_{\mathcal{O}}([\![B]\!]_{\mathcal{O}}, [\![A]\!]_{\mathcal{O}})$	– $[\![A - B]\!]^{\perp}_{\mathcal{O}} = [\![-]\!]^{\perp}_{\mathcal{O}}([\![B]\!]^{\perp}_{\mathcal{O}}, [\![A]\!]^{\perp}_{\mathcal{O}})$
– $[\![A^*]\!]_{\mathcal{O}} = [\![A]\!]^{\perp}_{\mathcal{O}}$	– $[\![A^*]\!]^{\perp}_{\mathcal{O}} = [\![A]\!]_{\mathcal{O}}$

Remarque. La dualité de ces sémantiques s'exprime par le diagramme commutatif suivant :

$$\begin{tikzcd} \mathcal{F} \arrow[r, "*"] \arrow[d, "{[\![\ ]\!]^{\perp}_{\mathcal{O}}}"'] & \mathcal{F} \arrow[d, "{[\![\ ]\!]_{\mathcal{O}}}"] \\ \mathcal{O} \arrow[r, "Id"'] & \mathcal{O} \end{tikzcd}$$

Il reste à vérifier, par récurrence sur la formule, que cette sémantique valide bien les axiomes qui définissent * :

Proposition 5.3.1 (validité) *Les équations suivantes sont vérifiées :*

– $[\![(A)^*]\!]_{\mathcal{O}} = [\![\sigma(A)]\!]_{\mathcal{O}}$*, si A est un atome propositionnel*

– $[\![(A \wedge B)^*]\!]_{\mathcal{O}} = [\![A^* \vee B^*]\!]_{\mathcal{O}}$

– $[\![(A \vee B)^*]\!]_{\mathcal{O}} = [\![A^* \wedge B^*]\!]_{\mathcal{O}}$

– $[\![(A \Rightarrow B)^*]\!]_{\mathcal{O}} = [\![B^* - A^*]\!]_{\mathcal{O}}$

– $[\![(A - B)^*]\!]_{\mathcal{O}} = [\![B^* \Rightarrow A^*]\!]_{\mathcal{O}}$

Puisque tout modèle peut être étendu de manière à interpréter l'opérateur de dualité, en vérifiant les axiomes de la définition 5.2.1, on obtient le corollaire du théorème de complétude de la logique intuitionniste soustractive suivant :

Corollaire 5.3.2 (conservativité) *Une formule propositionnelle ne contenant pas l'opérateur de dualité est prouvable dans le calcul propositionnel à dualité explicite ssi elle est prouvable en logique propositionnelle intuitionniste soustractive.*

Remarque. Les interprétations des atomes A et $\sigma(A)$ ne sont, par construction, *absolument pas liées*. En particulier, l'opérateur de dualité n'est donc pas une négation. En effet, la règle $A \vee A^* = \top$ n'est pas vérifiée en général et sa duale $A \wedge A^* = \bot$ non plus. Cet opérateur est seulement involutif. Nous reviendrons sur cette question en sous-section 5.4.

Proposition 5.3.3 (complétude) *Toute formule du calcul propositionnel à dualité explicite valide dans tout espace bi-topologique est prouvable.*

Preuve. Toute formule A du calcul propositionnel à dualité explicite est sémantiquement équivalente (par la proposition 5.3.1) à une formule A' ne contenant pas l'opérateur de dualité (mais pouvant contenir des atomes négatifs). Par la remarque ci-dessus et le théoreme 3.4.3 de complétude pour les modèles bi-topologiques, la formule A' est prouvable dans le calcul catégorique symétrique. Or A' est par construction prouvablement équivalente à A dans le calcul propositionnel à dualité explicite. □

5.4 Négation forte et dualité explicite

Nous n'avons jusqu'alors supposé aucun lien entre les formules atomiques A et A^*. En particulier, les formules A^* et $\neg A$ ainsi que les formules A^* et $\sim A$ ne sont pas comparables. Nous tentons ici d'y remédier.

Rappelons tout d'abord que le forcing de Kripke peut-être lu comme « l'accumulation d'information » d'un mathématicien idéalisé (cf. [66] p. 246). Une formule A peut être, à un moment donné (*i.e.* un nœud du préordre), soit « incertaine », soit « certainement vraie ». Il est alors naturel d'interpréter A^* comme A est « certainement fausse » (cette interprétation est due à Y. Gurevich [24]). La négation obtenue, appelée « négation forte », joue alors bien le rôle d'un opérateur de dualité (elle est en effet plus forte que $\neg A$, car si une formule est certainement fausse, elle ne sera jamais vraie). Pour des informations plus détaillées concernant la négation forte, cf. [53] p. 276). Malheureusement, les règles de la négation forte sont incompatibles avec celles du calcul propositionnel à dualité explicite dans le sens où le calcul obtenu est classique. En effet, si $^\perp$ est une négation forte, alors $A^\perp \vdash \neg A$, et la règle duale (qui est aussi une règle de définition de la négation forte, cf. [24]) $(\neg A)^\perp \vdash A^{\perp\perp}$ (et donc $\sim A \vdash A^\perp$) nous donne l'encadrement suivant, qui aboutit à la logique classique (cf. section 4).

$$\sim A \vdash A^\perp \vdash \neg A$$

L'encadrement réciproque ($\neg A \vdash A^{\perp} \vdash {\sim}A$) mériterait d'être investigué puisqu'il se rapproche davantage de l'interprétation sémantique :

$$[\![\neg A]\!] = int([\![A]\!]^c) \subset [\![A]\!]^c \subset couv([\![A]\!]^c) = [\![{\sim}A]\!]$$

Nous reviendrons sur cette question dans la section 6 consacrée à la logique soustractive du premier ordre.

Remarque. L'implication ${\sim}A \vdash \neg A$ est clairement équivalente au tiers-exclu classique $\top \vdash A \vee \neg A$ (il suffit d'utiliser la règle d'introduction de la négation faible pour obtenir ${\sim}A \vdash \neg A$ à partir du tiers-exclu). De plus, la décidabilité des atomes suffit à déduire la décidabilité de toute *formule propositionnelle.* La justification sémantique a déjà été donnée en remarque dans la section 4 : les ouvert-fermés (*i.e.* les ouverts complémentés) d'un espace bi-topologique forment une algèbre de Boole, et puisque l'intérieur ou la couverture d'un ouvert-fermé est l'ouvert-fermé lui-même, les sémantiques bi-topologiques des connecteurs coïncident avec leur sémantiques booléenes.

6 La soustraction au premier ordre

L'extension du calcul propositionnel catégorique au premier ordre ne pose pas de difficulté, il suffit de se donner les règles habituelles d'introduction droite et gauche de quantificateurs, restreintes à une formule à droite et à gauche (les termes sont définis de manière standard à partir d'une signature).

6.1 Les règles des quantificateurs

Règle d'intro. droite du $\exists$

$$\frac{A \vdash B[t/x]}{A \vdash \exists x B}$$

Règle d'intro. gauche du $\exists$ (où x n'apparaît pas dans B)

$$\frac{A \vdash B}{\exists x A \vdash B}$$

Règle d'intro. droite du $\forall$ (où x n'apparaît pas dans A)

$$\frac{A \vdash B}{A \vdash \forall x B(x)}$$

Règle d'intro. gauche du $\forall$

$$\frac{A[t/x] \vdash B}{\forall x A \vdash B}$$

Remarquez que les règles du $\exists$ sont bien duales de celles du $\forall$. Par conséquent, la dualité des connecteurs propositionnels s'étend au premier ordre en échangeant les quantificateurs $\exists$ et $\forall$ (comme nous allons le voir ce n'est pas le cas en logique intuitionniste usuelle). Cela signifie en particulier que l'on conserve au premier ordre la propriété de contraposition (*i.e.* si $A \vdash B$ est dérivable alors $\overline{B} \vdash \overline{A}$ aussi). Par conséquent, puisque le séquent $\exists x A(x) \wedge B \vdash \exists x (A(x) \wedge B)$ est dérivable :

$$\dfrac{\dfrac{\dfrac{\dfrac{A(x) \wedge B \vdash A(x) \wedge B}{A(x) \wedge B \vdash \exists x(A(x) \wedge B)}}{A(x) \vdash B \Rightarrow \exists x(A(x) \wedge B)}}{\exists x A(x) \vdash B \Rightarrow \exists x(A(x) \wedge B)}}{\exists x A(x) \wedge B \vdash \exists x(A(x) \wedge B)}$$

son dual (traditionnellement appelé DIS) $\forall x(A(x) \vee B) \vdash \forall x A(x) \vee B$ est aussi dérivable (il suffit de prendre la preuve duale) :

$$\dfrac{\dfrac{\dfrac{\dfrac{A(x) \vee B \vdash A(x) \vee B}{\forall x(A(x) \vee B) \vdash A(x) \vee B}}{\forall x(A(x) \vee B) - B \vdash A(x)}}{\forall x(A(x) \vee B) - B \vdash \forall x A(x)}}{\forall x(A(x) \vee B) \vdash \forall x A(x) \vee B}$$

Pourtant, il est connu que cet axiome n'est pas intuitionniste. Au premier ordre, la logique intuitionniste soustractive n'est donc pas conservative sur la logique intuitionniste. C. Rauszer a toutefois montré qu'elle est conservative sur la logique intuitionniste + DIS.

Théorème 6.1.1 *Une formule du premier ordre A est prouvable dans le calcul des prédicats intuitionniste avec les règles de la soustraction si et seulement si elle est prouvable dans la théorie du premier ordre contenant uniquement le schéma d'axiomes* DIS.

Preuve. Cf. [56], p. 57. □

6.2 Sémantique de Kripke

La non-conservativité sur la logique intuitionniste a pour conséquence que la sémantique de Kripke habituelle au premier ordre ne s'étend pas à la soustraction. Avant d'en examiner la raison, rappelons cette sémantique (nous appelons Σ la signature du langage des termes).

À chaque nœud α du préordre on associe une Σ-algèbre $\mathcal{A}_\alpha$ non vide (*i.e.* un domaine $dom(\mathcal{A}_\alpha)$ et une interprétation $\mathcal{A}_\alpha(f) \in dom(\mathcal{A}_\alpha)^n \to dom(\mathcal{A}_\alpha)$ de chaque symbole f de fonction n-aire de Σ) de façon telle que si $\alpha \leq \beta$ alors $\mathcal{A}_\alpha$ est une restriction de $\mathcal{A}_\beta$ (*i.e.* $dom(\mathcal{A}_\alpha) \subset dom(\mathcal{A}_\beta)$ et pour tout symbole n-aire f de Σ, $\forall \vec{a} \in dom(\mathcal{A}_\alpha)^n$, $\mathcal{A}_\alpha(\vec{a}) = \mathcal{A}_\beta(\vec{a})$)

La relation de forcing, notée $\alpha \Vdash_v F$, est indicée par une substitution v. Plus précisément, le symbole $\Vdash$ désigne maintenant une relation ternaire entre un nœud α du préordre, une formule F, et une substitution qui associe à toute variable libre de F un élément de $dom(\mathcal{A}_\alpha)$. La notation $\alpha \Vdash_v F$ n'a un sens que pour les substitution v « bien typées », c'est-à-dire définies pour toutes les variables libres de F et telle que l'image de chaque variable libre soit bien un élément de $dom(\mathcal{A}_\alpha)$.

La relation de forcing sur les formules atomiques est définie pour chaque nœud α par une interprétation $\mathcal{I}$ (qui correspond à la valuation dans le calcul propositionnel) qui associe à chaque symbole de prédicat n-aire une partie de $dom(\mathcal{A}_\alpha)^n$ de façon telle que $\forall \vec{a} \in dom(\mathcal{A}_\alpha)^n$ si $\vec{a} \in \mathcal{I}_\alpha(P)$ et $\alpha \leq \beta$ alors $\vec{a} \in \mathcal{I}_\beta(P)$.

Pour résumer, l'information peut croître de deux manières : notre connaissance du monde (représentée par les Σ-algèbres) et notre connaissance de la vérité (représentée par l'interprétation des formules atomiques) peuvent tout deux s'étendre.

Définition 6.2.1 *La relation de forcing pour une formule quelconque est définie par récurrence (où $\hat{v}$ est l'extension usuelle de v aux termes) :*

- $\alpha \Vdash_v P(t_1, \ldots, t_n) \equiv (\hat{v}(t_1), \ldots, \hat{v}(t_n)) \in \mathcal{I}_\alpha(P)$, *si P est un prédicat n-aire,*
- $\alpha \Vdash_v \top$ *et* $\alpha \nVdash_v \bot$
- $\alpha \Vdash_v (A \wedge B) \equiv \alpha \Vdash_v A$ *et* $\alpha \Vdash_v B$
- $\alpha \Vdash_v (A \vee B) \equiv \alpha \Vdash_v A$ *ou* $\alpha \Vdash_v B$
- $\alpha \Vdash_v (A \Rightarrow B) \equiv$ *pour tout* $\beta \geq \alpha$ ($\beta \Vdash_v A$ *implique* $\beta \Vdash_v B$)
- $\alpha \Vdash_v \forall x.F \equiv$ *pour tout* $\beta \geq \alpha, \forall a \in dom(\mathcal{A}_\beta), \beta \Vdash_{v,x=a} F$
- $\alpha \Vdash_v \exists x.F \equiv$ *il existe* $a \in dom(\mathcal{A}_\beta), \alpha \Vdash_{v,x=a} F$

Remarque. On peut vérifier dans chaque équation de la définition ci-dessus que la notation $\alpha \Vdash_v F$ est correctement utilisée (*i.e.* les valuations sont bien typées dans le membre droit de chaque équation à condition qu'elles soient bien typées dans le membre gauche). En particulier, pour l'implication et la quantification universelle, c'est vrai uniquement parce que le domaine ne peut que croître.

On voit maintenant clairement apparaître le problème posé par l'interprétation de la soustraction au premier ordre. On souhaiterait naturellement utiliser l'interprétation donnée dans le cadre propositionnel :

$$\alpha \Vdash_v (A - B) \equiv \text{ \textit{il existe} } \beta \leq \alpha \ (\beta \Vdash_v A \text{ \textit{et} } \beta \nVdash_v B)$$

mais, en raison de la croissance éventuelle du domaine, cette interprétation n'est plus nécessairement bien définie. En effet, dans le membre droit de la définition, la valuation peut être mal typée : elle peut associer à une variable libre de A ou de B un élément de $dom(\mathcal{A}_\alpha)$ qui n'appartient pas à $dom(\mathcal{A}_\beta)$.

Une solution à ce problème qui permet de plus de restaurer la dualité dans la sémantique consiste à exiger que le domaine soit fixé une fois pour toutes. L'interprétation des quantificateurs devient alors :

- $\alpha \Vdash_v \forall x.F \equiv$ *pour tout* $a \in dom(\mathcal{A})$ $(\alpha \Vdash_{v,x=a} F)$
- $\alpha \Vdash_v \exists x.F \equiv$ *il existe* $a \in dom(\mathcal{A})$ $(\alpha \Vdash_{v,x=a} F)$

et l'interprétation de la soustraction est celle donnée ci-dessus. Il est possible de prouver, pour cette sémantique, les résultats de complétude et de validité au premier ordre. Pour plus de détail nous renvoyons aux travaux de C. Rauszer [54, 56].

Remarque. Les règles des quantificateurs, universels et existentiels, sont valides dans la catégorie des ensembles, en les interprétant respectivement par les notions de produit et somme dépendants (nous reviendrons en détail sur cette interprétation dans le chapitre 5 consacré à la réalisabilité). Autrement dit, la sémantique ensembliste (au sens catégorique du chapitre 1, et non au sens classique des algèbres de Boole) des connecteurs s'étend au premier ordre, alors qu'elle ne s'étend pas à la soustraction. La sémantique topologique usuelle, par contre, s'étend à la soustraction, mais ne valide pas les règles des quantificateurs. L'adaptation de cette sémantique permettant de valider ces règles nous éloigne de la sémantique ensembliste (dans le sens catégorique) puisque cette adaptation valide DIS. Il faut donc choisir, au premier ordre, entre l'intuition ensembliste (à nouveau, dans le sens catégorique) et l'intuition apportée par la dualité.

6.3 Dualité dans une théorie

L'absence de théorème de déduction (cf. sous-section 3.5) nous amène à distinguer deux notions de théorème, qui ne sont plus équivalentes en présence de la soustraction. Considérons un ensemble de formules Γ.

– La première définition, qui correspond à la notion habituelle, consiste à appeler « théorème » de Γ une formules φ pour laquelle on peut dériver $\Gamma_1 \wedge \ldots \wedge \Gamma_n \vdash \varphi$ où $\Gamma_1, \ldots, \Gamma_n$ sont des formules de Γ.

– La seconde définition consiste à appeler « théorème » de Γ une formule φ pour laquelle on peut construire une dérivation du séquent $\top \vdash \varphi$ dont les feuilles sont soit des axiomes identité, soit des séquents de la forme $\top \vdash \Gamma_i$ où Γ_i est une formule de Γ.

Un théorème au sens de la première définition est bien sûr un théorème au sens de la seconde, puisqu'il suffit de couper avec $\top \vdash \Gamma_1 \wedge \ldots \wedge \Gamma_n$ (qui est un théorème au sens la seconde définition). Par contre, la réciproque n'est pas vraie puisque la théorie formée de l'unique formule $A \wedge {\sim}A$, où A est un atome propositionnel, forme une théorie contradictoire au sens de la seconde, mais pas au sens de la première. En effet, il est possible de dériver $\top \vdash \bot$ à partir de $\top \vdash A \wedge {\sim}A$ (la preuve duale a été donnée en sous section 3.5). Pourtant, le séquent $A \wedge {\sim}A \vdash \bot$ n'est pas dérivable dans le calcul propositionnel catégorique symétrique (son dual est le tiers-exclu). Enfin, un théorème, au sens de la seconde définition, dont la preuve ne fait pas apparaître la soustraction est un théorème au sens de première définition, puisque le théorème de déduction est alors applicable.

⋆ Théories où les formules atomiques sont décidables

Considérons une théorie Γ où les formules atomiques sont décidables au sens intuitionniste, c'est-à-dire $\Gamma \vdash A \vee \neg A$ pour toute formule atomique A, éventuellement non close, ce qui est le cas par exemple de l'arithmétique de Heyting. La négation devient alors un candidat naturel pour interpréter la dualité sur les formules atomiques, et l'opérateur de dualité est défini comme précédemment par récurrence (cf. définition 5.2.1). On obtient bien ainsi une dualité involutive dans Γ (*i.e.* $\Gamma \vdash A^{**} \Leftrightarrow A$ pour toute formule A). Toutefois, l'autre propriété essentielle de la dualité est donnée par la propriété de contraposition. Dans une théorie, pour que cette propriété soit toujours valide, il faut qu'elle le soit en particulier pour les axiomes. Autrement dit, pour chaque axiome $A \vdash B$ de la théorie Γ, l'axiome $B^* \vdash A^*$ doit aussi être dérivable dans la théorie Γ (*i.e.* la théorie Γ prouve sa théorie duale). Dans ce cas la propriété de contraposition est à nouveau vérifiée, *i.e.* le séquent dual de tout séquent dérivable dans la théorie Γ est lui aussi dérivable dans Γ.

Remarque. Si A est atomique, et si les formules atomiques sont décidables dans Γ, on peut montrer que si le séquent $\top \vdash A$ est dérivable dans Γ alors le séquent dual $A^* \vdash \bot$ l'est aussi (où $A^* \equiv \neg A$ puisque A est atomique). En effet, ce dernier séquent se ramène à $\top \vdash \neg\neg A$ et donc à $\top \vdash A$ puisque A est décidable dans Γ (et donc $\neg\neg A \Leftrightarrow A$). De même, si $\top \vdash \neg A$ est dérivable dans Γ, le séquent dual $(\neg A)^* \vdash \bot$ l'est aussi. En effet, $(\neg A)^* \vdash \bot$ se ramène à $\top \vdash \neg{\sim}\neg A$ et donc à $\top \vdash \neg A$ puisque A est décidable dans Γ (et donc $\neg{\sim} A \Leftrightarrow A$).

$\star$ Exemple : l'arithmétique de Heyting « soustractive »

Dans l'arithmétique de Heyting, les seules fomules atomiques sont de la forme $u = v$ où u, v sont des termes formés à partir des variables et de $0, S, +, \times$. On peut montrer facilement la formule $\forall m, n(m = n) \vee \neg(m = n)$ (appliquer deux récurrences imbriquées sur m et n). Puisque les formules atomiques, éventuellement non closes, sont décidables, il est possible de définir l'opérateur de dualité explicite comme décrit ci-dessus, c'est-à-dire en posant $(u = v)^* = \neg(u = v)$ puis récursivement pour les connecteurs et les quantificateurs.

Montrons maintenant que l'arithmétique de Heyting prouve sa théorie duale. C'est clair pour les axiomes de la forme $\top \vdash A$ où $\top \vdash \neg A$, où A est une formule atomique, par la remarque ci-dessus. Il reste donc à montrer que le dual du schéma de récurrence est dérivable. Rappelons le schéma de récurrence, exprimé sous forme de séquent :

$$\varphi(0) \wedge \forall n(\varphi(n) \Rightarrow \varphi(Sn)) \vdash \forall n \varphi(n)$$

Son dual est donc :

$$\exists n \varphi(n) \vdash \varphi(0) \vee \exists n(\varphi(Sn) - \varphi(n))$$

Montrons alors par récurrence sur m la formule $\phi(m)$, plus générale, suivante :

$$\varphi(m) \Rightarrow (\varphi(0) \vee \exists n \leq m(\varphi(Sn) - \varphi(n)))$$

- Cas de base : $\phi(0)$ est un axiome de la disjonction.
- Cas de récurrence : supposons $\phi(m)$ et montrons $\phi(Sm)$, ou encore, supposons $\phi(m)$ et $\varphi(Sm)$ et montrons $\varphi(0) \vee \exists p \leq Sm(\varphi(Sp) - \varphi(p))$. Puisque :

  $$\varphi(Sm) \vdash (\varphi(Sm) - \varphi(m)) \vee \varphi(m)$$

  on peut raisonner par cas :

  - Premier cas, $\varphi(Sm) - \varphi(m)$: *a fortiori* $\exists p \leq Sm(\varphi(Sp) - \varphi(p))$ (prendre $p = m$) et donc aussi $\varphi(0) \vee \exists p \leq Sm(\varphi(Sp) - \varphi(p))$.
  - Second cas, $\varphi(m)$: à partir de $\phi(m)$ et $\varphi(m)$ on déduit alors $\varphi(0) \vee \exists n \leq m(\varphi(Sn) - \varphi(n))$ et *a fortiori* $\varphi(0) \vee \exists n \leq Sm(\varphi(Sn) - \varphi(n))$.

$\star$ Application : admissibilité de la règle de Markov

Il est connu que dans HA, la règle de Markov est admissible, c'est-à-dire si A est une formule sans quantificateurs et si $\neg\neg\exists x A$ est prouvable dans HA alors $\exists x A$ est aussi prouvable dans HA. La présence de la dualité en arithmétique de Heyting soustractive permet de donner une preuve directe de ce résultat.

Proposition 6.3.1 *Dans* HA *soustractif, pour toute formule A sans quantificateur, si $\neg\neg\exists x A$ est prouvable alors $\exists x A$ est aussi prouvable.*

Preuve. Remarquons tout d'abord (cf. la remarque de la sous-section 5.4) que si A est une formule sans quantificateurs alors $\neg A \Leftrightarrow A^\perp \Leftrightarrow \sim A$. Maintenant supposons que $\top \vdash \neg\neg\exists x A$ sout prouvable dans HA soustractif, comme $\neg\exists x A$ est équivalent (en logique intuitionniste) à $\forall x \neg A$, on en déduit que $\top \vdash \neg\forall x\neg A$ et donc $\forall x \neg A \vdash \perp$ sont prouvables dans HA soustractif. Par dualité, $\top \vdash \exists x(\neg A)^\perp$ est donc prouvable dans HA soustractif. Or puisque A est sans quantificateurs, $(\neg A)^\perp \Leftrightarrow A^{\perp\perp} \Leftrightarrow A$. □

7 Annexe : à propos de la dualité

Dans cette section, nous proposons une formalisation de la dualité que nous avons largement exploitée jusqu'ici, dans les catégories bi-cartésiennes fermées, les algèbres de Heyting-Brouwer, les espaces bi-topologiques et enfin le calcul propositionnel catégorique symétrique. Cette formalisation a pour but d'unifier ces différentes dualités et de mettre en lumière les deux grandes classes de constructions duales utilisées précédemment.

⋆ Définition

Soit Σ une famille de structures ; une dualité sur Σ est la donnée de deux applications Φ, Ψ telles que :

- $\Phi : \Sigma \to \Sigma$ et Φ involutive : $\Phi(\Phi(S)) = S$ pour tout $S \in \Sigma$
- Ψ est de domaine Σ et si $S \in \Sigma$ alors $\Psi(S) : S \to \Phi(S)$ est une application bijective entre les deux structures S et $\Phi(S)$ de Σ dont l'inverse est $\Psi(\Phi(S))$.

Il est usuel de noter $S^\perp$ la structure $\Phi(S)$ et $\perp_{S,S^\perp}$ (ou encore simplement $\perp$) l'application $\Phi(S)$ (dont l'inverse est $\perp_{S^\perp,S}$).

⋆ Constructions duales

Soient Σ et Θ deux familles de structures, chacune munie d'une dualité.

1. Si ξ est une fonction (ou une construction ou un foncteur) de domaine $D_\xi \subseteq \Sigma \times \Theta$ qui à chaque couple $(S,T) \in D_\xi$ de structures dans Σ et Θ associe une fonction $\xi_{S,T} : S \to T$, alors on appelle duale de ξ et l'on note $\xi^\perp$ l'unique fonction de domaine $D_{\xi^\perp} = \{(S,T) : (S^\perp, T^\perp) \in D\}$ telle que pour chaque couple (S,T) de $D_{\xi^\perp}$ le diagramme suivant soit commutatif :
$$\begin{tikzcd} S \arrow[r, "\perp_{S,S^\perp}"] \arrow[d, "\xi^\perp_{S,T}"'] & S^\perp \arrow[d, "\xi_{S^\perp,T^\perp}"] \\ T \arrow[r, "\perp_{T,T^\perp}"'] & T^\perp \end{tikzcd}$$
Autrement dit, $\xi^\perp_{S,T}$ vérifie l'équation $\perp_{T,T^\perp} \circ \xi^\perp_{S,T} = \xi_{S^\perp,T^\perp} \circ \perp_{S,S^\perp}$ et puisque $\perp_{T,T^\perp}$ a $\perp_{T^\perp,T}$, $\xi^\perp_{S,T}$ est *définie* (d'où l'unicité) par l'équation :
$$\xi^\perp_{S,T} = \perp_{T^\perp,T} \circ \xi_{S^\perp,T^\perp} \circ \perp_{S,S^\perp}$$
On remarque alors que $\xi^{\perp\perp} = \xi$. En effet :
$$\xi^{\perp\perp}_{S,T} = \perp_{T^\perp,T} \circ \xi^\perp_{S^\perp,T^\perp} \circ \perp_{S,S^\perp} = \perp_{T^\perp,T} \circ \perp_{T,T^\perp} \circ \xi_{S,T} \circ \perp_{S^\perp,S} \circ \perp_{S,S^\perp} = \xi_{S,T}$$

Par exemple, c'est cette construction que a permis de définir la somme puis la co-exponentielle comme duales du produit et de l'exponentielle en théorie de catégories.

2. Soit $\Omega^{\Sigma,\Theta} = \{T^S : (S,T) \in \Sigma \times \Theta\}$. Les dualités sur les familles Σ et Θ permettent de définir une dualité canonique sur $\Omega^{\Sigma,\Theta}$ en posant :

$$(T^S)^\perp = (T^\perp)^{S^\perp}$$

et si $f \in T^S$ alors $f^\perp$ est l'unique fonction rendant commutatif le diagramme suivant :

$$\begin{tikzcd} S \arrow[r, "\perp_{S,S^\perp}"] \arrow[d, "f"'] & S^\perp \arrow[d, "f^\perp"] \\ T \arrow[r, "\perp_{T,T^\perp}"'] & T^\perp \end{tikzcd}$$

À nouveau, c'est l'involutivité de la dualité qui implique l'unicité, puisqu'elle permet de définir $f^\perp$ par :

$$f^\perp = \perp_{T,T^\perp} \circ f \circ \perp_{S^\perp,S}$$

Bien sûr, on vérifie alors que $f^{\perp\perp} = f$.

3. La même construction peut se faire avec les applications partielles. Notons $(Y^X)_{partiel}$ l'ensemble des fonctions de X dans Y, c'est-à-dire des applications de domaine inclus dans X à valeurs dans Y. Soit $\Omega^{\Sigma,\Theta}_{partiel} = \{(T^S)_{partiel} : (S,T) \in \Sigma \times \Theta\}$. Les dualités sur Σ et Θ permettent de définir comme ci-dessus une dualité canonique sur $\Omega^{\Sigma,\Theta}_{partiel}$.

Les deux notions de construction duale définies en (1) d'une part et en (2), (3) d'autre part diffèrent sur un point essentiel : dans la première notion, la construction duale a même domaine et co-domaine que la construction de départ, dans la seconde notion, la construction duale appartient à l'espace dual (ses domaines et co-domaines sont les duaux de ceux de la constructions de départ). Il est toutefois possible de considérer la première notion comme un cas particulier de la seconde (plus précisément celle définie en 3).

Posons $\Sigma' = \{\Sigma\}$ et $\Theta' = \{\Theta\}$. La dualité sur Σ induit une dualité sur Σ' : $\Sigma^\perp = \Sigma$ et si $S \in \Sigma$, alors le dual de S, en tant qu'élément de la « structure » Σ de la famille Σ' n'est autre que le dual $\mathrm{S}^\perp$ de S en tant que structure de la famille Σ. Idem pour Θ.

Ces dualités sur Σ' et Θ' induisent clairement une dualité sur $\Sigma' \times \Theta' = \{(\Sigma, \Theta)\}$.

Soit $Z = \cup\Omega = \{f : f \in T^S$ pour un couple (S,T) de $\Sigma \times \Theta\}$ et $Z' = \{Z\}$. La dualité sur Z' est définie par $Z^\perp = Z$ et si $f : U \to V$ est dans Z alors $f^{\perp_{Z,Z}}$ est le $f^\perp$ défini par la construction (2) :

$$f^{\perp_{Z,Z}} = \perp_{V,V^\perp} \circ f \circ \perp_{U^\perp,U}$$

Considérons alors la dualité $\xi \mapsto \xi^\perp$ introduite en (1). On a :

$$\begin{array}{rcll} \xi^\perp_{S,T} & = & \perp_{T^\perp,T} \circ \xi_{S^\perp,T^\perp} \circ \perp_{S,S^\perp} & \\ & = & (\xi_{S^\perp,T^\perp})^{\perp_{Z,Z}} & \quad \text{(prendre } U = S^\perp \text{ et } V = T^\perp) \\ & = & \xi_{(S,T)^{\perp_{\Sigma\times\Theta,\Sigma\times\Theta}}} & \end{array}$$

D'où,

$$\xi^\perp_{S,T} \quad = \quad \perp_{Z,Z} \circ \xi \circ \perp_{\Sigma\times\Theta,\Sigma\times\Theta}$$

ce qui prouve que le diagramme suivant est commutatif, ce qui exprime exactement que $\xi^\perp$ se définit à la façon de (3) (puisque ξ est partielle) :

$$\begin{tikzcd} \Sigma \times \Theta \arrow[r, "\perp_{\Sigma\times\Theta,\Sigma\times\Theta}"] \arrow[d, "\xi"'] & \Sigma \times \Theta \arrow[d, "\xi^\perp"] \\ Z \arrow[r, "\perp_{Z,Z}"'] & Z \end{tikzcd}$$

⋆ **Exemples**

1. Dans un espace bi-topologique $\mathcal{O}$ sur X, la couverture est duale de l'intérieur au sens de la construction (1), où $\mathcal{P}(X)^\perp = \mathcal{P}(X)$:

$$\begin{tikzcd} \mathcal{P}(X) \arrow[r, "c"] \arrow[d, "couv_{\mathcal{O}}=int^\perp_{\mathcal{O}}"'] & \mathcal{P}(X) \arrow[d, "int_{\mathcal{O}^\perp}"] \\ \mathcal{O} \arrow[r, "c"'] & \mathcal{O}^\perp \end{tikzcd}$$

2. Dans un espace bi-topologique $\mathcal{O}$ sur X, l'inhérence est duale de l'intérieur au sens de la construction (2) :

$$\begin{tikzcd} \mathcal{P}(X) \arrow[r, "c"] \arrow[d, "int_{\mathcal{O}}"'] & \mathcal{P}(X) \arrow[d, "inh_{\mathcal{O}^\perp}"] \\ \mathcal{O} \arrow[r, "c"'] & \mathcal{O}^\perp \end{tikzcd}$$

3. Dans un espace bi-topologique $\mathcal{O}$ sur X, la sémantique du $\vee$ est duale de celle du $\wedge$ au sens de la construction (1) :

$$\begin{tikzcd} \mathcal{O} \times \mathcal{O} \arrow[r, "(c,c)"] \arrow[d, "\llbracket\vee\rrbracket_{\mathcal{O}}"'] & \mathcal{O}^\perp \times \mathcal{O}^\perp \arrow[d, "\llbracket\wedge\rrbracket_{\mathcal{O}^\perp}"] \\ \mathcal{O} \arrow[r, "c"'] & \mathcal{O}^\perp \end{tikzcd}$$

Toujours au sens de la construction (1), le $-$ est dual du $\Leftarrow$:

$$\begin{tikzcd} \mathcal{O} \times \mathcal{O} \arrow[r, "(c,c)"] \arrow[d, "\llbracket-\rrbracket_{\mathcal{O}}"'] & \mathcal{O}^\perp \times \mathcal{O}^\perp \arrow[d, "\llbracket\Leftarrow\rrbracket_{\mathcal{O}^\perp}"] \\ \mathcal{O} \arrow[r, "c"'] & \mathcal{O}^\perp \end{tikzcd}$$

Chapitre 3

La contrainte intuitionniste

Dans ce chapitre, nous étudions certains liens entre les preuves de la logique classique et les preuves en logique intuitionniste soustractive. Nous définissons tout d'abord une nouvelle traduction, pour le calcul propositionnel, de la logique classique (soustractive) vers la logique intuitionniste soustractive et nous montrons que cette traduction s'étend directement aux preuves du calcul des séquents LK sans coupures. Nous présentons des contraintes sur les règles de la déduction naturelle classique CND qui permettent de définir une extension conservative de la logique intuitionniste avec des séquents à plusieurs conclusions. Ces contraintes peuvent aussi être vues globalement comme un critère permettant de déterminer si une preuve de CND est intuitionniste. Ce critère qui s'applique aux preuves, et non uniquement à la forme des séquents ou à la règle d'introduction de l'implication, se généralise facilement à la soustraction. On obtient alors une déduction naturelle intuitionniste soustractive, où les séquents ont plusieurs hypothèses et plusieurs conclusions.

Introduction

Dans la déduction naturelle classique (CND) de M. Parigot [47], comme en calcul des séquents classique (LK) de G. Gentzen [62], il est possible de dériver la preuve suivante du tiers-exclu :

$$\dfrac{\dfrac{A \vdash A}{A \vdash \bot, A}}{\vdash \neg A, A}$$

Cette preuve est « immorale » (d'un point de vue intuitionniste) pour la raison suivante : on décharge une hypothèse (A) sur une des conclusions ($\bot$) alors qu'une autre conclusion (A) dépend clairement de l'hypothèse que l'on est en train de décharger.

La règle en cause est le déchargement d'une hypothèse, lors de l'introduction droite de l'implication (ou de la négation intuitionniste), en présence de multiples conclusions. Dans la section 3.3, nous montrons que la négation faible (étudiée dans le chapitre précédent) permet de définir une « implication faible » pour laquelle justement, la règle d'introduction droite en présence de multiples conclusions est *dérivable en logique intuitionniste soustractive.* Nous définissons par ailleurs (en section 3) une traduction, obtenue par le biais de la sémantique bi-topologique, de la logique propositionnelle classique vers la logique propositionnelle intuitionniste soustractive. Nous montrons alors que cette traduction s'étend directement aux preuves sans coupure du calcul des séquents LK. On constate alors que toutes les occurrences de la règle d'introduction droite de l'implication (resp. négation) classique sont traduites en occurrences de la règle d'introduction droite de l'implication (resp. négation) faible.

Après traduction de la preuve, toutes les occurrences de l'introduction droite de l'implication (intuitionniste) ont disparues : la preuve obtenue est donc clairement intuitionniste. Toutefois l'absence d'occurrence de la règle d'introduction de l'implication est une contrainte très forte puisque toutes les instances de cette règle ne sont pas nécessairement « classiques ». En effet, il est connu que si l'on restreint le séquent hypothèse à une conclusion dans toutes les occurrences de cette règle, le calcul obtenu est toujours intuitionniste [13, 14]. Malheureusement, cette contrainte n'est pas stable par élimination des coupures. Nous l'affaiblirons en section 4 pour retrouver cette stabilité en rendant explicite la dépendance évoquée ci-dessus ; la règle de déchargement d'une hypothèse pourra alors être bridée afin de respecter cette contrainte. Cette contrainte sera alors étendue par dualité à la soustraction, le calcul ainsi obtenu sera exploité (comme système de types) dans le chapitre suivant. Nous montrerons aussi que le système ainsi obtenu est bien conservatif sur la logique intuitionniste soustractive.

1 Préliminaires

Nous rappelons tout d'abord les règles du calcul des séquents LK de Gentzen (cf. [22] par exemple). À nouveau les règles de la soustraction s'obtiennent par dualité (le calcul obtenu est appelé SLK). Toutefois, on peut montrer ici qu'il existe une seule négation « classique » (*i.e.* la négation intuitionniste et la négation faible sont prouvablement équivalentes) et que la soustraction est définissable à partir de la négation classique, notée $A^\perp$, par $A - B \equiv A \wedge B^\perp$ (de même que l'implication est définissable par $A \Rightarrow B \equiv A^\perp \vee B$).

1.1 Le calcul classique LK

Axiome

$$A \vdash A$$

Règle de coupure

$$\frac{\Gamma \vdash \Delta, A \qquad \Gamma', A \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta'}$$

Règles de contraction

$$\frac{\Gamma \vdash \Delta, A, A}{\Gamma \vdash \Delta, A} \qquad \frac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}$$

Règles d'affaiblissement

$$\frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, A} \qquad \frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}$$

Règles d'échange

$$\frac{\Gamma \vdash \Delta, A, B, \Delta'}{\Gamma \vdash \Delta, B, A, \Delta'} \qquad \frac{\Gamma, A, B, \Gamma' \vdash \Delta}{\Gamma, B, A, \Gamma' \vdash \Delta}$$

Règles du $\bot$ et du $\top$

$$\frac{\Gamma \vdash \Delta, \bot}{\Gamma \vdash \Delta, A} \qquad \frac{\Gamma, \top \vdash \Delta}{\Gamma, A \vdash \Delta}$$

Règles de la négation classique

$$\frac{\Gamma \vdash \Delta, A}{\Gamma, A^{\bot} \vdash \Delta} \qquad \frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \Delta, A^{\bot}}$$

Règles d'intro. gauche du $\wedge$

$$\frac{\Gamma, A \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta} \qquad \frac{\Gamma, B \vdash \Delta}{\Gamma, A \wedge B \vdash \Delta}$$

Règle d'intro. droite du $\wedge$

$$\frac{\Gamma \vdash \Delta, A \qquad \Gamma' \vdash \Delta', B}{\Gamma, \Gamma' \vdash \Delta, \Delta', A \wedge B}$$

Règles d'intro. droite du $\vee$

$$\frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, A \vee B} \qquad \frac{\Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \vee B}$$

Règle d'intro. gauche du $\vee$

$$\frac{\Gamma, A \vdash \Delta \qquad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \vee B \vdash \Delta, \Delta'}$$

Règle d'intro. gauche du $\Rightarrow$

$$\frac{\Gamma, B \vdash \Delta \qquad \Gamma' \vdash \Delta', A}{\Gamma, \Gamma', A \Rightarrow B \vdash \Delta, \Delta'}$$

Règle d'intro. droite du $\Rightarrow$

$$\frac{\Gamma, A \vdash \Delta, B}{\Gamma \vdash \Delta, A \Rightarrow B}$$

Remarque. Il est possible de donner d'autres formulations des règles qui exigent que les séquents hypothèses aient les mêmes contextes. Par exemple, la règle d'introduction gauche de l'implication est alors formulée ainsi :

$$\frac{\Gamma, B \vdash \Delta \qquad \Gamma \vdash \Delta, A}{\Gamma, A \Rightarrow B \vdash \Delta}$$

Il est bien connu que cette formulation est équivalente à celle présentée ci-dessus, il suffit d'appliquer, les règles d'affaiblissement et de contraction pour montrer l'équivalence. Nous utiliserons indifféremment les deux formulations.

$\star$ **Les règles des quantificateurs**

Les termes sont définis de manière standard à partir d'une signature.

Règle d'intro. droite du $\exists$

$$\frac{\Gamma \vdash \Delta, B[t/x]}{\Gamma \vdash \Delta, \exists x B}$$

Règle d'intro. gauche du $\exists$ (si x n'apparaît pas libre dans Γ, Δ)

$$\frac{\Gamma, A \vdash \Delta}{\Gamma, \exists x A \vdash \Delta}$$

Règle d'intro. droite du $\forall$ (si x n'apparaît pas libre dans Γ, Δ)

$$\frac{\Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, \forall x B(x)}$$

Règle d'intro. gauche du $\forall$

$$\frac{\Gamma, A[t/x] \vdash \Delta}{\Gamma, \forall x A \vdash \Delta}$$

1.2 Le calcul classique soustractif SLK

Les règles de la soustraction sont obtenue par dualité à partir des règles de l'implication. On obtient bien ainsi des « règles de calcul des séquents », puisqu'il s'agit de règles d'introduction gauche et droite.

Règle d'intro. droite du $-$

$$\frac{\Gamma \vdash \Delta, A \qquad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta', A - B}$$

Règle d'intro. gauche du $-$

$$\frac{\Gamma, A \vdash \Delta, B}{\Gamma, A - B \vdash \Delta}$$

Remarque. La propriété bien connue du calcul des séquents LK qui exprime que les règles introduisant les connecteurs définissent en fait un unique connecteur à équivalence près s'étend à la soustraction. Voici donc la preuve de l'unicité du connecteur défini par les règles de l'implication, et sa duale qui est une preuve de l'unicité de la soustraction (toujours à équivalence près). Supposons que $\Rightarrow_1, \Rightarrow_2$ et $-_1, -_2$ soient des connecteurs vérifiant respectivement les règles de l'implication et de la soustraction :

$$\frac{\dfrac{B \vdash B \qquad A \vdash A}{A, A \Rightarrow_1 B \vdash B}}{A \Rightarrow_1 B \vdash A \Rightarrow_2 B} \qquad\qquad \frac{\dfrac{A \vdash A \qquad B \vdash B}{A \vdash A -_1 B, B}}{A -_1 B \vdash A -_2 B}$$

D'autre part, de même que l'implication « internalise » le symbole de déduction « $\vdash$ » à droite (comme conclusion), la soustraction « internalise » le symbole de déduction « $\vdash$ » à gauche (comme hypothèse). Autrement dit, les règles suivantes sont dérivables :

$$\frac{\Gamma \vdash A \Rightarrow B, \Delta}{\Gamma, A \vdash B, \Delta} \qquad\qquad \frac{\Gamma, A - B \vdash \Delta}{\Gamma, A \vdash B, \Delta}$$

En effet,

$$\dfrac{\Gamma \vdash A \Rightarrow B, \Delta \qquad \dfrac{B \vdash B \qquad A \vdash A}{A, A \Rightarrow B \vdash B}}{\Gamma, A \vdash B, \Delta} \qquad\qquad \dfrac{\dfrac{A \vdash A \qquad B \vdash B}{A \vdash B, A - B} \qquad \Gamma, A - B \vdash \Delta}{\Gamma, A \vdash B, \Delta}$$

⋆ **A propos des négations**

En posant, comme précédemment $\neg A \equiv A \Rightarrow \bot$ et $\sim A \equiv \top - A$ on obtient les règles dérivées suivantes, qui sont exactement les mêmes que celles de la négation classique :

Règles dérivées d'intro. du $\neg$

$$\dfrac{\Gamma \vdash \Delta, A}{\Gamma, \neg A \vdash \Delta} \qquad \dfrac{\Gamma, A \vdash \Delta}{\Gamma \vdash \Delta, \neg A}$$

Règles dérivées d'intro. du $\sim$

$$\dfrac{\Gamma \vdash \Delta, A}{\Gamma, \sim A \vdash \Delta} \qquad \dfrac{\Gamma, A \vdash \Delta}{\Gamma \vdash \Delta, \sim A}$$

L'équivalence entre les négations découle de l'unicité, à équivalence près, du connecteur vérifiant ces règles d'introduction gauche et droite. Plus précisément, si $\sharp_G$ et $\sharp_D$ désignent respectivement des négations vérifiant la règle d'introduction gauche et la règle d'introduction droite, on a :

$$\dfrac{\dfrac{A \vdash A}{\vdash \sharp_D A, A}}{\sharp_G A \vdash \sharp_D A}$$

⋆ **A propos des séquents à multiples conclusions**

Le calcul des séquents a été introduit par G. Gentzen dans un but très précis : prouver le théorème d'élimination des coupures (cf. la thèse de H. Herbelin [26] pour plus de détails). Ce système a eu beaucoup de succès par la suite (aussi en partie dû à la logique linéaire [19]) puisqu'il permet une analyse fine des mécanismes de calcul, en particulier en logique classique (les systèmes LC [20], et LK^{tq} [9] en témoignent). On attribue aussi souvent un caractère « atomique » à cette présentation : chaque connecteur est défini par ses règles d'introduction, indépendamment des autres connecteurs. Ce n'est pas le cas dans le calcul catégorique où l'implication est définie à partir de la conjonction (et la soustraction à partir de la disjonction). Toutefois, *le caractère minimaliste de cette présentation est trompeuse.* En effet, si l'on considère par exemple le fragment du calcul ne contenant que la conjonction et la disjonction, il est possible de dériver la preuve suivante :

$$\dfrac{\dfrac{\dfrac{\dfrac{B \vdash B \qquad C \vdash C}{B, C \vdash B \wedge C}}{B, C \vdash (B \wedge C) \vee (B \wedge A)} \qquad \dfrac{\dfrac{B \vdash B \qquad A \vdash A}{B, A \vdash B \wedge A}}{B, A \vdash (B \wedge C) \vee (B \wedge A)}}{B, C \vee A \vdash (B \wedge C) \vee (B \wedge A)}}{B \wedge (C \vee A) \vdash (B \wedge C) \vee (B \wedge A)}$$

alors qu'il est connu que tout treillis n'est pas forcément distributif. Les règles de la conjonction et de la disjonction de LK sont donc plus puissantes que celles du calcul catégorique (à nouveau, uniquement si l'on se restreint aux fragments des deux calculs ne contenant que la conjonction et la disjonction). En réalité, nous avons utilisé une règle de distributivité implicite contenue dans les règles structurelles qui axiomatisent la « virgule de gauche » (qui représente aussi une conjonction). Remarquez que cette preuve est intuitionniste (les séquents n'ont qu'une seule conclusion) et est transposable en déduction naturelle.

Autre exemple, DIS que l'on peut prouver grâce à la soustraction mais qui n'est pas un théorème intuitionniste (cf. chapitre 2, section 6) est maintenant prouvable sans la soustraction.

Remarque. Nous utiliserons dans la preuve de DIS les règles dérivées du $\vee$ suivantes :

$$\frac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, A \vee B} \qquad \frac{\Gamma \vdash \Delta, A \vee B}{\Gamma \vdash \Delta, A, B}$$

qui peuvent se prouver ainsi :

$$\dfrac{\dfrac{\dfrac{\Gamma \vdash \Delta, A, B}{\Gamma \vdash \Delta, A \vee B, B}}{\Gamma \vdash \Delta, A \vee B, A \vee B}}{\Gamma \vdash \Delta, A \vee B} \qquad \dfrac{\Gamma \vdash \Delta, A \vee B \qquad \dfrac{A \vdash A \qquad B \vdash B}{A \vee B \vdash A, B}}{\Gamma \vdash \Delta, A, B}$$

Proposition 1.2.1 *Le schéma d'axiomes DIS est prouvable dans LK en utilisant uniquement les règles structurelles, les règles du $\vee$ et du $\forall$.*

Preuve.

$$\dfrac{\dfrac{\dfrac{\dfrac{A(x) \vee B \vdash A(x) \vee B}{\forall x(A(x) \vee B) \vdash A(x) \vee B}}{\forall x(A(x) \vee B) \vdash A(x), B}}{\forall x(A(x) \vee B) \vdash \forall x A(x), B}}{\forall x(A(x) \vee B) \vdash \forall x A(x) \vee B}$$

□

Un autre « effet de bord », plus flagrant et qui est d'ailleurs l'objet de ce chapitre, dû à la présence de multiples conclusions est bien sûr la possibilité de dériver la règle du tiers-exclu pour la négation intuitionniste (qui devient alors équivalente à la négation classique) :

$$\dfrac{\dfrac{A \vdash A}{\vdash A, \neg A}}{\vdash A \vee \neg A}$$

2 Restrictions intuitionnistes de LK

Le calcul LK peut être restreint en un calcul intuitionniste de plusieurs manières. La restriction la plus connue est obtenue en limitant les séquents à au plus une conclusion : c'est le calcul LJ (cf. [62], par exemple). Toutefois les règles de la soustraction contraintes de cette manière :

$$\frac{\Gamma \vdash A \qquad \Gamma', B \vdash}{\Gamma, \Gamma' \vdash A - B} \qquad \frac{\Gamma, A \vdash B}{\Gamma, A - B \vdash}$$

ne définissent plus la soustraction étudiée dans le chapitre précédent. En effet, on peut alors dériver :

$$\dfrac{\dfrac{\dfrac{B \vdash B}{B, A \vdash B}}{B, A - B \vdash}}{A - B \vdash \neg B}$$

et en particulier $\sim B \vdash \neg B$ qui n'est pas valide en logique intuitionniste soustractive. La dualité suggère la contrainte à imposer pour rester intuitionniste : il faut *restreindre les séquents à au plus une hypothèse.* Mais cette restriction nous donne un calcul dégénéré ou nous ramène au calcul catégorique symétrique.

Une restriction plus faible, mais qui n'est pas stable par élimination des coupures (cf. sous-section 4.4), consiste à restreindre uniquement la règle d'introduction droite de l'implication à une formule à droite dans le séquent hypothèse. Il est connu [43] que ce calcul propositionnel est *conservatif sur la logique propositionnel intuitionniste*, nous l'appellerons LK^i. Au premier ordre, par contre, où il est possible de prouver DIS (cf. la proposition 1.2.1), le calcul LK^i est conservatif sur la logique intuitionniste soustractive (cf. corollaire 2.2.4).

Par dualité, la restriction s'étend bien sûr à la règle d'introduction gauche de la soustraction : le séquent hypothèse ne doit posséder qu'une seule formule à gauche. Le calcul obtenu, cette fois-ci *conservatif sur la logique intuitionniste soustractive* (cf. proposition 2.2.3), sera appelé SLK^i.

Ces contraintes seront encore affaiblies dans la section 4 pour retrouver la stabilité par élimination des coupures (nous nous placerons alors dans le cadre de la déduction naturelle classique). Mais cette question n'est pas pertinente pour le moment, puisque nous allons voir que la traduction des preuves sans coupure de LK (resp. SLK) ne nécessite aucune règle d'introduction droite de l'implication (resp. ni aucune règle d'introduction gauche de la soustraction).

2.1 Le calcul intuitionniste SLK^i

Les règles de ce calcul sont celles de SLK où les règles de l'implication et de la soustraction sont remplacées par les suivantes (et la négation classique supprimée) :

Règle d'intro. du $\Rightarrow$

$$\frac{\Gamma, B \vdash \Delta \qquad \Gamma' \vdash \Delta', A}{\Gamma, \Gamma', A \Rightarrow B \vdash \Delta, \Delta'} \qquad \frac{\Gamma, B \vdash A}{\Gamma \vdash B \Rightarrow A}$$

Règle d'intro. du $-$

$$\frac{A \vdash \Delta, A}{A - B \vdash \Delta} \qquad \frac{\Gamma \vdash \Delta, A \qquad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta', A - B}$$

À nouveau, on obtient les règles dérivées, cette fois ci différentes, qui définissent respectivement les négations intuitionnistes et faibles :

Règles dérivées d'intro. du $\neg$

$$\frac{\Gamma \vdash \Delta, A}{\Gamma, \neg A \vdash \Delta} \qquad \frac{\Gamma, A \vdash}{\Gamma \vdash \neg A}$$

Règles dérivées d'intro. du $\sim$

$$\frac{\vdash A, \Delta}{\sim A \vdash \Delta} \qquad \frac{\Gamma, A \vdash \Delta}{\Gamma \vdash \sim A, \Delta}$$

De même que dans le cadre classique, où on a montré qu'une seule négation, à équivalence près, vérifiait à la fois les deux règles d'introduction classique, on montre ici qu'il existe une unique négation faible, de même qu'il existe une unique négation intuitionniste. Supposons données deux négations $\sharp_g$ et $\sharp_d$, qui vérifient respectivement les règles d'introduction gauche et droite « contraintes ». Nous rappelons que les notations $\sharp_G$ et $\sharp_D$ introduites en sous-section 1.2 désignent des négations qui vérifient respectivement les règles d'introduction gauche et droite « non contraintes ». On montre alors que :

$$\cfrac{\cfrac{A \vdash A}{\vdash \sharp_D A, A}}{\sharp_g A \vdash \sharp_D A}$$

ce qui entraîne l'unicité d'une négation $\sharp_{g,D}$ (*i.e.* vérifiant la règle d'intro. gauche contrainte et la règle d'intro. droite non contrainte), autrement dit, l'unicité de la négation faible. Par dualité, $\sharp_G A \vdash \sharp_d A$, ce qui implique l'unicité de la négation intuitionniste.

Remarque. Au lieu de supprimer la négation classique $^\perp$, on peut contraindre ses règles d'introduction : contraindre la règle d'introduction droite rend $^\perp$ équivalente à la négation intuitionniste, contraindre la règle d'introduction gauche rend $^\perp$ équivalente à la négation faible.

2.2 Conservativité sur le calcul catégorique

Nous montrons dans cette sous-section que le calcul propositionnel LK^i (resp. symétrique) est conservatif sur le calcul propositionnel catégorique (resp. symétrique), et que le calcul du premier ordre SLK^i est conservatif sur le calcul catégorique symétrique du premier ordre.

Notation. Si $\Gamma \vdash \Delta$ est le séquent $\Gamma_1, \Gamma_2, \ldots, \Gamma_n \vdash \Delta_1, \Delta_2, \ldots, \Delta_m$, la notation $\Gamma^\wedge \vdash \Delta^\vee$ est une abréviation pour $(\ldots((\Gamma_1 \wedge \Gamma_2) \wedge \ldots \wedge \Gamma_n) \vdash (\ldots((\Delta_1 \vee \Delta_2) \vee \ldots \vee \Delta_m)$

Proposition 2.2.1 *Un séquent propositionnel $\Gamma \vdash \Delta$ est dérivable dans SLK^i ssi $\Gamma^\wedge \vdash \Delta^\vee$ est dérivable dans le calcul propositionnel catégorique symétrique.*

Preuve. Il est clair que $\Gamma \vdash \Delta$ est dérivable dans SLK^i ssi $\Gamma^\wedge \vdash \Delta^\vee$ est dérivable dans SLK^i (c'est la preuve usuelle dans LK puisque seules les règles structurelles et les règles d'introduction du $\wedge$ et du $\vee$ interviennent). Il est aussi facile de montrer que tous les axiomes et toutes les règles du calcul propositionnel catégorique symétrique sont dérivables dans SLK^i. On en déduit que si $\Gamma^\wedge \vdash \Delta^\vee$ est dérivable dans le calcul propositionnel symétrique catégorique alors $\Gamma \vdash \Delta$ est dérivable dans SLK^i.

Nous montrons maintenant la réciproque en détail. Pour cela, nous dérivons la traduction de tout axiome et toute règle de SLK^i (obtenue en remplaçant chaque séquent $\Gamma \vdash \Delta$ par le séquent $\Gamma^\wedge \vdash \Delta^\vee$) dans le calcul propositionnel catégorique symétrique. Pour simplifier les preuves, nous utiliserons les règles dérivées de commutativité et d'associativité du $\wedge$ et du $\vee$. Nous utiliserons également fréquemment les règles dérivées suivantes :

$$\frac{A \vdash C \Rightarrow D}{A \wedge C \vdash D} \qquad \frac{C - D \vdash A}{C \vdash A \vee D}$$

dont la première peut se dériver ainsi (l'autre s'obtient par dualité) :

$$\dfrac{\dfrac{\dfrac{A \wedge C \vdash A \qquad A \vdash C \Rightarrow D}{A \wedge C \vdash C \Rightarrow D} \qquad A \wedge C \vdash C}{A \wedge C \vdash (C \Rightarrow D) \wedge C} \qquad (C \Rightarrow D) \wedge C \vdash D}{A \wedge C \vdash D}$$

- Les axiomes d'identité sont inchangés par la traduction (et sont traduits par eux-mêmes).
- La traduction de la règle de coupure :

  $$\dfrac{\Gamma^\wedge \vdash \Delta^\vee \vee A \qquad \Gamma^\wedge \wedge A \vdash \Delta^\vee}{\Gamma^\wedge \vdash \Delta^\vee}$$

  peut se dériver ainsi :

  $$\dfrac{\dfrac{\dfrac{\Gamma^\wedge \vdash \Delta^\vee \vee A}{\Gamma^\wedge - \Delta^\vee \vdash A} \qquad \dfrac{\Gamma^\wedge \wedge A \vdash \Delta^\vee}{A \vdash \Gamma^\wedge \Rightarrow \Delta^\vee}}{\Gamma^\wedge - \Delta^\vee \vdash \Gamma^\wedge \Rightarrow \Delta^\vee}}{\Gamma^\wedge \vdash \Delta^\vee}$$

  où la dernière règle appliquée se dérive par exemple ainsi :

  $$\dfrac{\dfrac{A \vdash (A - B) \vee B \qquad \dfrac{A - B \vdash A \Rightarrow B \qquad \dfrac{B \wedge A \vdash B}{B \vdash A \Rightarrow B}}{(A - B) \vee B \vdash A \Rightarrow B}}{A \vdash A \Rightarrow B} \qquad A \vdash A}{A \vdash B}$$

- La traduction de la règle de contraction droite :

  $$\dfrac{\Gamma^\wedge \vdash \Delta^\vee \vee (A \vee A)}{\Gamma^\wedge \vdash \Delta^\vee \vee A}$$

  peut se dériver ainsi :

  $$\dfrac{\Gamma^\wedge \vdash \Delta^\vee \vee (A \vee A) \qquad \dfrac{\Delta^\vee \vdash \Delta^\vee \vee A \qquad \dfrac{A \vdash \Delta^\vee \vee A \qquad A \vdash \Delta^\vee \vee A}{A \vee A \vdash \Delta^\vee \vee A}}{\Delta^\vee \vee (A \vee A) \vdash \Delta^\vee \vee A}}{\Gamma^\wedge \vdash \Delta^\vee \vee A}$$

- La traduction de la règle d'affaiblissement droite :

  $$\dfrac{\Gamma^\wedge \vdash \Delta^\vee}{\Gamma^\wedge \vdash \Delta^\vee \vee A}$$

  peut se dériver ainsi :

  $$\dfrac{\Gamma^\wedge \vdash \Delta^\vee \qquad \Delta^\vee \vdash \Delta^\vee \vee A}{\Gamma^\wedge \vdash \Delta^\vee \vee A}$$

– La traduction de la règle d'échange droite :

$$\frac{\Gamma^{\wedge} \vdash (\ldots(((\Delta^{\vee} \vee A) \vee B) \vee \Delta_i) \vee \ldots \vee \Delta_m)}{\Gamma^{\wedge} \vdash (\ldots(((\Delta^{\vee} \vee B) \vee A) \vee \Delta_i) \vee \ldots \vee \Delta_m)}$$

peut se dériver ainsi :

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\Gamma^{\wedge} \vdash (\ldots(((\Delta^{\vee} \vee A) \vee B) \vee \Delta_i) \vee \ldots \vee \Delta_m)}{\Gamma^{\wedge} - \Delta_m - \ldots - \Delta_i \vdash (\Delta^{\vee} \vee A) \vee B}}{\Gamma^{\wedge} - \Delta_m - \ldots - \Delta_i \vdash \Delta^{\vee} \vee (A \vee B)}}{\Gamma^{\wedge} - \Delta_m - \ldots - \Delta_i \vdash \Delta^{\vee} \vee (B \vee A)}}{\Gamma^{\wedge} - \Delta_m - \ldots - \Delta_i \vdash (\Delta^{\vee} \vee B) \vee A}}{\Gamma^{\wedge} \vdash (\ldots(((\Delta^{\vee} \vee B) \vee A) \vee \Delta_i) \vee \ldots \vee \Delta_m)}$$

– La traduction de la règle du $\perp$:

$$\frac{\Gamma^{\wedge} \vdash \Delta^{\vee} \vee \perp}{\Gamma^{\wedge} \vdash \Delta^{\vee} \vee A}$$

peut se dériver ainsi :

$$\dfrac{\Gamma^{\wedge} \vdash \Delta^{\vee} \vee \perp \qquad \dfrac{\Delta^{\vee} \vdash \Delta^{\vee} \vee A \qquad \dfrac{\perp \vdash A \qquad A \vdash \Delta^{\vee} \vee A}{\perp \vdash \Delta^{\vee} \vee A}}{\Delta^{\vee} \vee \perp \vdash \Delta^{\vee} \vee A}}{\Gamma^{\wedge} \vdash \Delta^{\vee} \vee A}$$

– Les traductions des règles d'intro. gauche du $\wedge$:

$$\frac{\Gamma^{\wedge} \wedge A \vdash \Delta^{\vee}}{\Gamma^{\wedge} \wedge (A \wedge B) \vdash \Delta^{\vee}} \qquad \frac{\Gamma^{\wedge} \wedge B \vdash \Delta^{\vee}}{\Gamma^{\wedge} \wedge (A \wedge B) \vdash \Delta^{\vee}}$$

peuvent se dériver ainsi :

$$\dfrac{\dfrac{\dfrac{\Gamma^{\wedge} \wedge A \vdash \Delta^{\vee}}{A \vdash \Gamma^{\wedge} \Rightarrow \Delta^{\vee}}}{A \wedge B \vdash \Gamma^{\wedge} \Rightarrow \Delta^{\vee}}}{\Gamma^{\wedge} \wedge (A \wedge B) \vdash \Delta^{\vee}} \qquad \dfrac{\dfrac{\dfrac{\Gamma^{\wedge} \wedge B \vdash \Delta^{\vee}}{B \vdash \Gamma^{\wedge} \Rightarrow \Delta^{\vee}}}{A \wedge B \vdash \Gamma^{\wedge} \Rightarrow \Delta^{\vee}}}{\Gamma^{\wedge} \wedge (A \wedge B) \vdash \Delta^{\vee}}$$

– La traduction de la règle d'intro. droite du $\wedge$

$$\frac{\Gamma^{\wedge} \vdash \Delta^{\vee} \vee A \qquad \Gamma^{\wedge} \vdash \Delta^{\vee} \vee B}{\Gamma^{\wedge} \vdash \Delta^{\vee} \vee (A \wedge B)}$$

peut se dériver ainsi :

$$\dfrac{\dfrac{\dfrac{\Gamma^{\wedge} \vdash \Delta^{\vee} \vee A}{\Gamma^{\wedge} - \Delta^{\vee} \vdash A} \qquad \dfrac{\Gamma^{\wedge} \vdash \Delta^{\vee} \vee B}{\Gamma^{\wedge} - \Delta^{\vee} \vdash B}}{\Gamma^{\wedge} - \Delta^{\vee} \vdash A \wedge B}}{\Gamma^{\wedge} \vdash \Delta^{\vee} \vee (A \wedge B)}$$

– La traduction de la règle d'intro. gauche du $\Rightarrow$

$$\frac{\Gamma^{\wedge} \vdash A \vee \Delta^{\vee} \qquad \Gamma^{\wedge} \wedge B \vdash \Delta^{\vee}}{\Gamma^{\wedge} \wedge A \Rightarrow B \vdash \Delta^{\vee}}$$

peut se dériver à partir de la règle d'élimination droite de l'implication (nous utilisons désormais des règles qui cumulent les contextes puisque nous savons traduire la contraction et l'affaiblissement) :

$$\dfrac{\dfrac{A \Rightarrow B \vdash A \Rightarrow B \qquad \Gamma^{\wedge} \vdash A \vee \Delta^{\vee}}{\Gamma^{\wedge} \wedge (A \Rightarrow B) \vdash B \vee \Delta^{\vee}} \qquad \Gamma^{\wedge} \wedge B \vdash \Delta^{\vee}}{\Gamma^{\wedge} \wedge (A \Rightarrow B) \vdash \Delta^{\vee}}$$

La règle de coupure a déjà été dérivée précédemment, la règle d'élimination (droite) de l'implication :

$$\dfrac{C \vdash (A \Rightarrow B) \vee D \qquad C \vdash A \vee D}{C \vdash B \vee D}$$

peut se dériver ainsi :

$$\dfrac{\dfrac{\dfrac{\dfrac{C \vdash (A \Rightarrow B) \vee D}{C - D \vdash A \Rightarrow B} \qquad \dfrac{C \vdash A \vee D}{C - D \vdash A}}{C - D \vdash (A \Rightarrow B) \wedge A} \qquad (A \Rightarrow B) \wedge A \vdash B}{C - D \vdash B}}{C \vdash D \vee B}$$

– La règle d'intro. droite du $\Rightarrow$ se traduit par elle-même :

$$\dfrac{\Gamma^{\wedge} \wedge B \vdash A}{\Gamma^{\wedge} \vdash B \Rightarrow A}$$

Les dérivations des autres règles s'obtiennent par dualité. □

Remarque. Les contraintes sur SLK^i n'interviennent que pour la traduction des règles d'introduction droite de l'implication et d'introduction gauche de la soustraction, justement traduites par elles-mêmes. *Ces règles sont intraduisibles si elles ne sont pas contraintes* (puisqu'elles définissent la logique classique).

Corollaire 2.2.2 *Si* $\Gamma \vdash \Delta$ *est un séquent propositionnel dérivable dans* LK^i *alors* $\Gamma^{\wedge} \vdash \Delta^{\vee}$ *est dérivable dans le calcul propositionnel catégorique.*

Remarque. En particulier, cela signifie que l'on aurait pu se passer de la soustraction dans la traduction sauf pour traduire les règles de la soustraction elle-même (qui ne sont pas explicitées ci-dessus).

Proposition 2.2.3 *Si* $\Gamma \vdash \Delta$ *est un séquent du premier ordre dérivable dans* SLK^i *alors* $\Gamma^{\wedge} \vdash \Delta^{\vee}$ *est dérivable dans le calcul propositionnel catégorique symétrique (du premier ordre).*

Preuve. Les traductions des règles du $\forall$ (où x n'apparaît ni dans Γ ni dans Δ pour la règle d'introduction droite) :

$$\dfrac{\Gamma^{\wedge} \vdash A \vee \Delta^{\vee}}{\Gamma^{\wedge} \vdash \Delta^{\vee} \vee \forall x A} \qquad\qquad \dfrac{\Gamma^{\wedge} \wedge A[t/x] \vdash \Delta^{\vee}}{\Gamma^{\wedge} \wedge \forall x A \vdash \Delta^{\vee}}$$

peuvent se dériver ainsi :

$$\dfrac{\dfrac{\dfrac{\Gamma^\wedge \vdash A \vee \Delta^\vee}{\Gamma^\wedge - \Delta^\vee \vdash A}}{\Gamma^\wedge - \Delta^\vee \vdash \forall x A}}{\Gamma^\wedge \vdash \Delta^\vee \vee \forall x A} \qquad \dfrac{\dfrac{\dfrac{\Gamma^\wedge \wedge A[t/x] \vdash \Delta^\vee}{A[t/x] \vdash \Gamma^\wedge \Rightarrow \Delta^\vee}}{\forall x A \vdash \Gamma^\wedge \Rightarrow \Delta^\vee}}{\Gamma^\wedge \wedge \forall x A \vdash \Delta^\vee}$$

Les règles du $\exists$ s'obtiennent comme toujours par dualité. □

Corollaire 2.2.4 *Au premier ordre, le calcul* SLK^i *est conservatif sur le calcul* LK^i*.*

Preuve. En effet, SLK^i et LK^i permettent tout deux de prouver DIS (cf. proposition 1.2.1) mais sont conservatifs sur le calcul propositionnel catégorique symétrique du premier ordre. □

3 Traduction de la logique classique

Dans un espace bi-topologique $\mathcal{O}$ sur X, étant donné une valuation $\mathcal{V}$, il est possible de donner (indépendamment de son interprétation bi-topologique) l'interprétation classique d'une formule A sur $\mathcal{P}(X)$ vu comme une algèbre de Boole. Nous avons vu par ailleurs que l'interprétation de la négation donnait lieu à deux sémantiques bi-topologiques (l'intérieur et la couverture du complémentaire) qui encadrent l'interprétation de la négation classique (*i.e.* le complémentaire). Cette propriété se généralise aux autres connecteurs : l'intérieur et la couverture de la sémantique classique d'un connecteur sont toujours directement exprimables comme la sémantique intuitionniste de ce même connecteur ou de connecteurs dérivés. Par exemple, l'intérieur de la sémantique classique de l'implication est exactement la sémantique intuitionniste de l'implication, alors que la couverture de la sémantique de l'implication est exactement la sémantique intuitionniste de « l'implication faible » définie par $A \rightsquigarrow B \equiv {\sim}A \vee B$ (car $couv(A^c \cup B) = couv(A^c) \cup B$).

Cette constatation nous amène à la question suivante : peut-on construire récursivement, à partir d'une formule, A deux nouvelles formules A^- et A^+ dont les sémantiques bi-topologiques encadrent « le plus fidèlement possible » l'interprétation classique de A ? Nous répondons ici à cette question en définissant une traduction de la logique propositionnelle classique (soustractive) vers la logique propositionnelle intuitionniste soustractive puis en montrant que cette traduction s'étend directement aux preuves du calcul des séquents (LK) sans coupures.

3.1 Traduction des formules

On définit la traduction suivante des formules propositionnelles classiques vers les formules propositionnelles soustractives. La traduction notée $^-$ minore l'interprétation bi-topologique de la formule alors que la traduction notée $^+$ la majore. À nouveau la négation classique de A est notée $A^\perp$.

Minoration	**Majoration**
$A^- \equiv A$, si A est un atome	$A^+ \equiv A$, si A est un atome
$(A \wedge B)^- \equiv A^- \wedge B^-$	$(A \wedge B)^+ \equiv A^+ \wedge B^+$
$(A \vee B)^- \equiv A^- \vee B^-$	$(A \vee B)^+ \equiv A^+ \vee B^+$
$(A \Rightarrow B)^- \equiv A^+ \Rightarrow B^-$	$(A \Rightarrow B)^+ \equiv \sim A^- \vee B^+$
$(A - B)^- \equiv A^- \wedge \neg B^+$	$(A - B)^+ \equiv A^+ - B^-$
$(A^\perp)^- \equiv \neg A^+$	$(A^\perp)^+ \equiv \sim A^-$

Remarque. La traduction de la soustraction est donnée pour faire apparaître la symétrie entre la majoration et la minoration. Il est connu qu'en logique classique, toute formule propositionnelle est équivalente, par exemple, à une formule ne contenant que des conjonctions et des négations. Or de telles formules sont prouvables en logique classique si et seulement si elles sont prouvables en logique intuitionniste. Une traduction est donc d'autant plus intéressante qu'elle est « fidèle ». Selon ce critère, si l'on considère une formule ne contenant pas de soustraction, cette traduction ne fait que substituer les négations et implications classiques respectivement par les négations et implications faibles ou intuitionnistes selon leurs occurences positives ou négatives dans la formule de départ.

Théorème 3.1.1 *Si un séquent $A \vdash B$ est valide en logique classique, alors $A^- \vdash B^+$ est valide en logique intuitionniste soustractive.*

Corollaire 3.1.2 *Si une formule A est une tautologie classique, alors A^+ est une tautologie intuitionniste.*

Définition 3.1.3 *Étant donné un ensemble X et une valuation $\mathcal{V}$ sur $\mathcal{P}(X)$, on note $[\![A]\!]^K$ l'interprétation de A dans l'algèbre de Boole des parties de X. Plus précisément, pour toute formule propositionnelle A, (contenant éventuellement la soustraction) :*

– $[\![A]\!]^K = \mathcal{V}(A)$ *si A est un atome*

– $[\![A \wedge B]\!]^K = [\![A]\!]^K \cap [\![B]\!]^K$

– $[\![A \vee B]\!]^K = [\![A]\!]^K \cup [\![B]\!]^K$

– $[\![A \Rightarrow B]\!]^K = ([\![A]\!]^K)^c \cup [\![B]\!]^K$

– $[\![A - B]\!]^K = [\![A]\!]^K \cap ([\![B]\!]^K)^c$

– $[\![A^\perp]\!]^K = ([\![A]\!]^K)^c$

Preuve. (du théorème 3.1.1) Par récurrence sur la formule, on montre que pour tout espace bi-topologique $\mathcal{O}$ sur un ensemble X, toute valuation $\mathcal{V}$ intuitionniste sur $\mathcal{O}$:

$$[\![A^-]\!]_{\mathcal{O}} \subset [\![A]\!]^K \subset [\![A^+]\!]_{\mathcal{O}}$$

En effet :

– Si A est un atome alors $A^- = A = A^+$ et $[\![A]\!]^K = [\![A]\!]_{\mathcal{O}} = \mathcal{V}(A)$

- $[\![(A \wedge B)^-]\!]_\mathcal{O} = [\![A^- \wedge B^-]\!]_\mathcal{O} = [\![A^-]\!]_\mathcal{O} \cap [\![B^-]\!]_\mathcal{O} \subset [\![A]\!]^K \cap [\![B]\!]^K = [\![A \wedge B]\!]^K \subset [\![A^+]\!]_\mathcal{O} \cap [\![B^+]\!]_\mathcal{O} = [\![A^+ \wedge B^+]\!]_\mathcal{O} = [\![(A \wedge B)^+]\!]_\mathcal{O}$
- $[\![(A \Rightarrow B)^-]\!]_\mathcal{O} = [\![A^+ \Rightarrow B^-]\!]_\mathcal{O} = \mathit{int}([\![A^+]\!]^c \cup [\![B^-]\!]) \subset [\![A^+]\!]^c \cup [\![B^-]\!] \subset ([\![A]\!]^K)^c \cup [\![B]\!]^K = [\![A \Rightarrow B]\!]^K \subset ([\![A^-]\!]^c_\mathcal{O} \cup [\![B^+]\!]_\mathcal{O}) \subset \mathit{couv}([\![A^-]\!]^c_\mathcal{O} \cup [\![B^+]\!]_\mathcal{O}) = \mathit{couv}([\![A^-]\!]^c_\mathcal{O}) \cup [\![B^+]\!]_\mathcal{O} = [\![\sim A^- \vee B^+]\!]_\mathcal{O} = [\![(A \Rightarrow B)^+]\!]_\mathcal{O}$
- $[\![(A^\perp)^-]\!]_\mathcal{O} = [\![\neg A^+]\!]_\mathcal{O} = \mathit{int}([\![A^+]\!]^c) \subset ([\![A]\!]^K)^c = [\![A^\perp]\!]^K \subset [\![A^-]\!]^c_\mathcal{O} \subset \mathit{couv}([\![A^-]\!]^c_\mathcal{O}) = [\![\sim A^-]\!]_\mathcal{O}$

Les autres cas s'obtiennent par dualité. On en déduit alors que si pour toute valuation classique $[\![A]\!]^K \subset [\![B]\!]^K$ alors pour toute valuation intuitionniste $[\![A^-]\!]_\mathcal{O} \subset [\![A]\!]^K \subset [\![B]\!]^K \subset [\![B^+]\!]_\mathcal{O}$ et donc $A^- \vdash B^+$ est valide en logique intuitionniste. □

Remarque. Puisque $[\![A^-]\!]_\mathcal{O}$ et $[\![A^+]\!]_\mathcal{O}$ sont des ouverts, on a en fait montré que :

$$[\![A^-]\!]_\mathcal{O} \subset \mathit{int}([\![A]\!]^K) \quad \text{et} \quad \mathit{couv}([\![A]\!]^K) \subset [\![A^+]\!]_\mathcal{O}$$

Corollaire 3.1.4 *Si une formule A est une tautologie classique, alors A^+ est une tautologie intuitionniste.*

3.2 Traduction des preuves

Les résultats qui suivent ont déjà été prouvés en utilisant des arguments sémantiques. Nous nous intéressons ici à la traduction des preuves.

Lemme 3.2.1 *Dans* SLK^i*, il est possible de dériver les règles suivantes :*

$$\frac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash \sim A \vee B, \Delta} \qquad \frac{\Gamma, A \vdash B, \Delta}{\Gamma, A \wedge \neg B \vdash \Delta}$$

Preuve. Dérivons la première règle (l'autre étant la règle duale).

$$\dfrac{\dfrac{\Gamma, A \vdash \Delta, B}{\Gamma \vdash \sim A, B, \Delta}}{\Gamma \vdash \sim A \vee B, \Delta}$$

□

Lemme 3.2.2 *Pour toute formule A (contenant éventuellement la soustraction), le séquent $A^- \vdash A^+$ est dérivable dans* SLK^i.

Preuve. Par récurrence sur la formule A. Les seuls cas non triviaux sont ceux de l'implication et de la soustraction. Traitons le cas de l'implication, l'autre cas s'obtenant comme toujours par dualité. Supposons que $A^- \vdash_{\mathrm{SLK}^i} A^+$ et $B^- \vdash_{\mathrm{SLK}^i} B^+$ et montrons que $(A \Rightarrow B)^- \vdash_{\mathrm{SLK}^i} (A \Rightarrow B)^+$:

$$\dfrac{\dfrac{\dfrac{\begin{matrix}\vdots \\ A^- \vdash A^+\end{matrix} \quad \begin{matrix}\vdots \\ B^- \vdash B^+\end{matrix}}{A^-, A^+ \Rightarrow B^- \vdash B^+}}{A^+ \Rightarrow B^- \vdash \sim A^-, B^+}}{A^+ \Rightarrow B^- \vdash \sim A^- \vee B^+}$$

$\square$

Théorème 3.2.3 *Il est possible de traduire toute preuve dans SLK, sans coupure, du séquent $\Gamma \vdash \Delta$, en une preuve dans* SLK^i *(sans coupure) du séquent $\Gamma^- \vdash \Delta^+$ de la façon suivante : traduire chaque occurence d'axiome $A \vdash A$, par une preuve dans* SLK^i *de $A^- \vdash A^+$, chaque occurrence de la règle d'introduction gauche ou droite de la négation classique par les règles dérivées respectives :*

$$\frac{\Gamma \vdash A, \Delta}{\Gamma, \neg A \vdash \Delta} \qquad \frac{\Gamma, A \vdash \Delta}{\Gamma \vdash {\sim}A, \Delta}$$

et enfin chaque occurence de la règle d'introduction droite du $\Rightarrow$ ou d'introduction gauche du $-$ par les règles dérivées respectives :

$$\frac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash {\sim}A \vee B, \Delta} \qquad \frac{\Gamma, A \vdash B, \Delta}{\Gamma, A \wedge \neg B \vdash \Delta}$$

Preuve. La preuve se fait par récurrence sur la dérivation de $\Gamma \vdash_{\mathrm{SLK}} \Delta$. Un axiome $A \vdash A$ est, par construction, remplacé par une preuve de $A^- \vdash_{\mathrm{SLK}^i} A^+$ (cf. lemme 3.2.2). La traduction d'une instance de règle de SLK est soit déjà une instance d'une règle de SLK^i soit une instance de la règle dérivée donnée par la traduction. Par exemple, la traduction d'une instance de la règle d'introduction du $\Rightarrow$ nous donne :

$$\frac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \Rightarrow B, \Delta} \rightsquigarrow \frac{\Gamma^-, A^- \vdash B^+, \Delta^+}{\Gamma^- \vdash {\sim}A^- \vee B^+, \Delta^+}$$

or justement ${\sim}A^- \vee B^+ = (A \Rightarrow B)^+$. $\square$

Remarque. La restriction à des preuves sans coupure se justifie par le fait qu'une instance de la règle de coupure :

$$\frac{\Gamma \vdash A, \Delta \qquad \Gamma', A \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta'}$$

est traduite en l'instance suivante, qui n'est plus une instance de la règle de coupure :

$$\frac{\Gamma^- \vdash A^+, \Delta^+ \qquad \Gamma'^-, A^- \vdash \Delta'^+}{\Gamma^-, \Gamma'^- \vdash \Delta^+, \Delta'^+}$$

Cette règle est toutefois valide pour la raison suivante : remarquons tout d'abord que $\Gamma_i \vdash_{\mathrm{SLK}} \Gamma_i^-$ et $\Delta_j^+ \vdash_{\mathrm{SLK}} \Delta_j$, (puisque pour toute formule A, les formules A, A^-, A^+ sont équivalentes en logique classique, ces séquents sont donc dérivables dans SLK). Par conséquent, si $\Gamma^- \vdash \Delta^+$ est dérivable dans SLK^i alors $\Gamma \vdash \Delta$ est dérivable dans SLK. Supposons maintenant que $\Gamma^- \vdash A^+, \Delta^+$ et $\Gamma'^-, A^- \vdash \Delta'^+$ soient dérivables dans SLK^i, par la remarque qui précéde, $\Gamma \vdash A, \Delta$ et $\Gamma', A \vdash \Delta'$ sont dérivables dans SLK, et par application de la règle de coupure, $\Gamma, \Gamma' \vdash \Delta, \Delta'$ aussi. Par élimination des coupures, on obtient une preuve sans coupure de $\Gamma, \Gamma' \vdash \Delta, \Delta'$ et on en déduit alors que $\Gamma^-, \Gamma'^- \vdash \Delta^+, \Delta'^+$ est dérivable dans SLK^i.

3.3 L'implication faible

L'implication faible, notée $\rightsquigarrow$, est définie en posant $A \rightsquigarrow B = {\sim}A \vee B$. Réciproquement, la négation faible est bien entendue définissable à partir de l'implication faible en posant

$\sim A \equiv A \rightsquigarrow \perp$. les règles dérivées d'introduction gauche et droite de cette implication faible sont :

$$\frac{\Gamma, B \vdash \Delta \qquad \vdash A, \Delta'}{\Gamma, A \rightsquigarrow B \vdash \Delta, \Delta'} \qquad \frac{\Gamma, A \vdash \Delta, B}{\Gamma \vdash A \rightsquigarrow B, \Delta}$$

Nous avons déjà vu la dérivation de la règle d'introduction droite (cf. lemme 3.2.1), voici la dérivation de la règle d'introduction gauche (remarquez que l'introduction gauche de la négation faible nécessite l'absence d'hypothèse, ce qui justifie la contrainte dans la règle l'introduction gauche de l'implication faible) :

$$\frac{\Gamma, B \vdash \Delta \qquad \dfrac{\vdash A, \Delta'}{\sim A \vdash \Delta'}}{\Gamma, \sim A \vee B \vdash \Delta, \Delta'}$$

À nouveau, on peut vérifier que ces deux règles d'introduction définissent un unique connecteur à équivalence près. En effet, supposons que $\rightsquigarrow_1$ et $\rightsquigarrow_1$ soit deux connecteurs vérifiant les règles d'introduction droite et gauche de $\rightsquigarrow$:

$$\dfrac{\dfrac{\dfrac{\dfrac{B \vdash B}{B, A \vdash B}}{B \vdash A \rightsquigarrow_2 B} \qquad \dfrac{\dfrac{A \vdash A}{A \vdash B, A}}{\vdash A \rightsquigarrow_2 B, A}}{A \rightsquigarrow_1 B \vdash A \rightsquigarrow_2 B, A \rightsquigarrow_2 B}}{A \rightsquigarrow_1 B \vdash A \rightsquigarrow_2 B}$$

Comme cas particulier de la traduction précédente, on peut définir la traduction suivante sur les formules (sans soustraction) de calcul propositionnel classique vers le calcul propositionnel avec implication faible (dont on sait qu'il est conservatif sur la logique intuitionniste puisque la négation faible est conservative). À nouveau, cette traduction s'étend directement des preuves sans coupures de LK vers les preuves de du système LK^i plus les règles de l'implication faibles.

Minoration	**Majoration**
$A^- \equiv A$, si A est un atome	$A^+ \equiv A$, si A est un atome
$(A \wedge B)^- \equiv A^- \wedge B^-$	$(A \wedge B)^+ \equiv A^+ \wedge B^+$
$(A \vee B)^- \equiv A^- \vee B^-$	$(A \vee B)^+ \equiv A^+ \vee B^+$
$(A \Rightarrow B)^- \equiv A^+ \Rightarrow B^-$	$(A \Rightarrow B)^+ \equiv A^- \rightsquigarrow B^+$
$(A^\perp)^- \equiv \neg A^+$	$(A^\perp)^+ \equiv \sim A^-$

Exemples

– La traduction de l'axiome $A^{\perp\perp} \Rightarrow A$ est $(A^{\perp\perp})^- \rightsquigarrow A^+ \equiv \neg\sim A \rightsquigarrow A$

– La traduction de l'axiome de Peirce $((A \Rightarrow B) \Rightarrow A) \Rightarrow A$ est : $((A \Rightarrow B) \Rightarrow A)^- \rightsquigarrow A^+ \equiv ((A \Rightarrow B)^+ \Rightarrow A) \rightsquigarrow A \equiv ((A \rightsquigarrow B) \Rightarrow A) \rightsquigarrow A$

– La traduction de $A^{\perp\perp} \vdash A$ est $\neg\sim A \vdash A$

– La traduction de $((A \Rightarrow B) \Rightarrow A) \vdash A$ est $((A \rightsquigarrow B) \Rightarrow A) \vdash A$

3.4 Traduction du premier ordre

La traduction s'étend directement au premier ordre. Cependant, nous rappelons qu'au premier ordre, DIS est prouvable dans le calcul SLK^i (cf. la proposition 1.2.1). Par conséquent, la traduction présentée ici se fait de la logique classique du premier ordre vers la la logique intuitionniste soustractive du premier ordre (*i.e.* la logique intuitionniste + DIS).

Minoration	**Majoration**
$(\forall x A)^- \equiv \forall x A^-$	$(\forall x A)^+ \equiv \forall x A^+$
$(\exists x A)^- \equiv \exists x A^-$	$(\exists x A)^+ \equiv \exists x A^+$

Théorème 3.4.1 *Étant donnée une preuve sans coupure dans* SLK *au premier ordre de* $\Gamma \vdash \Delta$*, la preuve obtenue en appliquant la traduction décrite dans l'énoncé du théorème 3.1 est une preuve (sans coupure) dans* SLK^i *au premier ordre du séquent* $\Gamma^- \vdash \Delta^+$*.*

4 Liens de dépendances explicites

Dans cette section, nous définissons une contrainte intuitionniste plus faible que celle de LK^i. Nous verrons que cette contrainte s'exprime naturellement en déduction naturelle à plusieurs conclusions : nous nous placerons donc dans CND. Dans ce système, lors de la dérivation d'un séquent, les « dépendances » entre hypothèses et conclusions peuvent être facilement rendues explicites. Ceci permet alors de restreindre le déchargement d'une hypothèse afin que la preuve reste intuitionniste. Cette contrainte ne concerne pas la forme des séquents, mais la structure des dérivations. Nous verrons dans le chapitre 4 que cette contrainte est stable pour les règles de réduction de CND (alors que ce n'est pas le cas pour la contrainte de LK^i, cf. sous-section 4.4).

Nous définissons par ailleurs *une déduction naturelle classique symétrique* CND_{sym} en complètant par dualité les règles de CND (en ajoutant en particulier la soustraction). Puisque la contrainte intuitionniste sur CND est basée sur des *dépendances non orientées* par construction, elle s'étend alors directement à CND_{sym}.

4.1 La déduction naturelle classique CND

Nous présentons tout d'abord les règles structurelles, qui sont les mêmes que dans LK, puis les règles habituelles de connecteurs et des quantificateurs du premier ordre. Puisque la déduction naturelle classique vérifie de bonnes propriétés calculatoires au second ordre, nous donnerons aussi les règles concernant les quantificateurs du second ordre (nous renvoyons aux travaux de M. Parigot [47, 48] et de J.-L. Krivine [29] pour plus de détail). Il est connu qu'en logique intuitionniste (et bien sûr aussi en logique classique), les quantificateurs universels du second ordre et du premier ordre permettent de définir les autres connecteurs $(\exists, \exists^2, \top, \perp, \wedge, \vee)$ à partir de l'implication et de la quantification universelle. Nous les explicitons tout de même, puisque le choix des règles aura des conséquences dans la suite. Nous nous donnons aussi la règle de coupure bien que ce soit inhabituel en déduction naturelle. Sa présence sera justifiée dans le chapitre suivant.

Axiome

$$A \vdash A$$

Règle de coupure

$$\frac{\Gamma \vdash \Delta, A \qquad \Gamma', A \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta'}$$

Règles de contraction

$$\frac{\Gamma \vdash \Delta, A, A}{\Gamma \vdash \Delta, A} \qquad \frac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}$$

Règles d'affaiblisement

$$\frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta, A} \qquad \frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}$$

Règle d'élim. du $\perp$

$$\frac{\Gamma \vdash \Delta, \perp}{\Gamma \vdash \Delta, A}$$

Règle d'intro. du $\top$

$$\frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, \top}$$

Règle d'intro. et d'élim. du $\Rightarrow$

$$\frac{\Gamma, A \vdash \Delta, B}{\Gamma \vdash \Delta, A \Rightarrow B} \qquad \frac{\Gamma \vdash \Delta, A \Rightarrow B \qquad \Gamma' \vdash \Delta', A}{\Gamma, \Gamma' \vdash \Delta, \Delta', B}$$

Règles d'intro. et d'élim. du $\wedge$

$$\frac{\Gamma \vdash \Delta, A \qquad \Gamma' \vdash \Delta', B}{\Gamma, \Gamma' \vdash \Delta, \Delta', A \wedge B} \qquad \frac{\Gamma \vdash \Delta, A \wedge B}{\Gamma \vdash \Delta, A} \quad \frac{\Gamma \vdash \Delta, A \wedge B}{\Gamma \vdash \Delta, B}$$

Règles d'intro. et d'élim. du $\vee$

$$\frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, A \vee B} \quad \frac{\Gamma \vdash \Delta, B}{\Gamma \vdash \Delta, A \vee B} \qquad \frac{\Gamma \vdash \Delta, A \vee B \quad \Gamma' \vdash \Delta', (A \Rightarrow C) \quad \Gamma'' \vdash \Delta'', (B \Rightarrow C)}{\Gamma, \Gamma', \Gamma'' \vdash \Delta, \Delta', \Delta'', C}$$

$\star$ **Les règles des quantificateurs**

Les termes du premier ordre sont définis de manière standard à partir d'une signature. On suppose de plus donné un ensemble infini de variables de prédicats du second ordre pour chaque arité n.

Règle d'intro. (où x n'apparaît pas dans Γ, Δ) **et règle d'élim. du $\forall$**

$$\frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, \forall x A} \qquad \frac{\Gamma \vdash \Delta, \forall x A}{\Gamma \vdash \Delta, A[t/x]}$$

Règle d'intro. (où X n'apparaît pas dans Γ, Δ) **et règle d'élim. du $\forall^2$**

$$\frac{\Gamma \vdash \Delta, A}{\Gamma \vdash \Delta, \forall X A} \qquad \frac{\Gamma \vdash \Delta, \forall X A}{\Gamma \vdash \Delta, A[T/X]}$$

Règle d'intro. (où x n'apparaît pas dans Γ', Δ') **et règle d'élim. du** $\exists$

$$\frac{\Gamma \vdash \Delta, \exists x A \qquad \Gamma' \vdash \Delta', (A \Rightarrow B)}{\Gamma, \Gamma' \vdash \Delta, \Delta', B} \qquad\qquad \frac{\Gamma \vdash \Delta, A[t/x]}{\Gamma \vdash \Delta, \exists x A}$$

Règle d'intro. (où X n'apparaît pas dans Γ', Δ') **et règle d'élim. du** $\exists^2$

$$\frac{\Gamma \vdash \Delta, \exists X A \qquad \Gamma' \vdash \Delta', (A \Rightarrow B)}{\Gamma, \Gamma' \vdash \Delta, \Delta', B} \qquad\qquad \frac{\Gamma \vdash \Delta, A[T/X]}{\Gamma \vdash \Delta, \exists X A}$$

4.2 La déduction naturelle classique symétrique CND$_{sym}$

Nous avons vu que dans le calcul des séquents LK, les connecteurs et quantificateurs sont définis par des règles d'introduction gauches et droites. En appliquant la dualité, on obtenait donc à nouveau des règles d'introduction gauches et droites, qui définissaient bien sûr le connecteur dual. C'est ainsi que l'on a obtenu les règles d'introduction de la soustraction.

En déduction naturelle classique, il est aussi possible de prendre les règles duales de celles qui définissent un connecteur (resp. un quantificateur). Les règles ainsi obtenues définissent bien le connecteur (resp. le quantificateur) dual, mais ce ne sont pas des règles de déduction naturelle au sens habituel puisqu'il s'agit de *règles d'introduction et d'élimination gauches.* Les règles d'introduction gauches sont celles de LK, puisque les règles d'introduction droite sont les mêmes que celles de CND. Les règles d'élimination gauche sont celles de la déduction libre (FD) [46]. On peut récapituler ces différentes présentations dans le tableau ci-dessous :

Type de règles	intro. gauche	élim. droite
intro. droite	LK	CND
élim. gauche		FD

Nous appellerons CND$_{sym}$ le calcul dans lequel les règles (d'introduction et d'élimination gauches) de $\top$, $\vee$,$\exists$, $\exists^2$ sont duales respectivement des règles (d'introduction et d'élimination droites) de $\bot, \wedge, \forall, \forall^2$ (ces règles se substituent aux règles d'introduction et d'élimination droites de $\top$, $\vee$,$\exists$, $\exists^2$ de CND). De plus, naturellement, CND$_{sym}$ contient des règles qui définissent la soustraction, duales de celle de l'implication.

Règle d'élim. gauche du $\top$

$$\frac{\Gamma, \top \vdash \Delta}{\Gamma, A \vdash \Delta}$$

Règles d'intro. gauche et d'élim. gauche du $\vee$

$$\frac{\Gamma, A \vdash \Delta \qquad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \vee B \vdash \Delta, \Delta'} \qquad\qquad \frac{\Gamma, A \vee B \vdash \Delta}{\Gamma, A \vdash \Delta} \qquad \frac{\Gamma, A \vee B \vdash \Delta}{\Gamma, B \vdash \Delta}$$

Règle d'intro. gauche (où x n'apparaît pas dans Γ, Δ) **et règle d'élim. gauche du** $\exists$

$$\frac{\Gamma, A \vdash \Delta}{\Gamma, \exists x A \vdash \Delta} \qquad\qquad \frac{\Gamma, \exists x A \vdash \Delta}{\Gamma, A[t/x] \vdash \Delta}$$

Règle d'intro. gauche (où X n'apparaît pas dans Γ, Δ) **et règle d'élim. gauche du** $\exists^2$

$$\frac{\Gamma, A \vdash \Delta}{\Gamma, \exists X A \vdash \Delta} \qquad\qquad \frac{\Gamma, \exists X A \vdash \Delta}{\Gamma, A[T/X] \vdash \Delta}$$

⋆ **Règles de la soustraction**

Règle d'élim. gauche du $-$

$$\frac{\Gamma, A - B \vdash \Delta \qquad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \vdash \Delta, \Delta'}$$

Règle d'intro. gauche du $-$

$$\frac{\Gamma, A \vdash \Delta, B}{\Gamma, A - B \vdash \Delta}$$

Remarque. Comme cas particulier de la règle d'élimination gauche, en prenant comme séquent hypothèse de droite l'axiome $B \vdash B$, on obtient la règle dérivée :

$$\frac{\Gamma, A - B \vdash \Delta, B}{\Gamma, A \vdash \Delta, B}$$

Cette règle dérivée permet toutefois de retrouver la règle de départ de la façon suivante :

$$\frac{\dfrac{\dfrac{\Gamma, A - B \vdash \Delta}{\Gamma, A - B \vdash \Delta, B}}{\Gamma, A \vdash \Delta, B} \qquad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \vdash \Delta, \Delta'}$$

4.3 La déduction naturelle classique soustractive SND

Nous étendons ici la déduction naturelle classique à la soustraction. De même que pour la disjonction, un moyen de retrouver des règles d'élimination et d'introduction droite pour la soustraction consiste à prendre :

– d'une part, la règle duale de l'introduction gauche de l'implication dans LK, qui est bien une règle d'introduction droite pour la soustraction :

$$\frac{\Gamma, A \vdash \Delta \qquad \Gamma' \vdash \Delta', B}{\Gamma, \Gamma', B \Rightarrow A \vdash \Delta, \Delta'} \qquad\qquad \frac{\Gamma \vdash \Delta, A \qquad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta', A - B}$$

– d'autre part, la règle duale de l'élimination gauche de l'implication en déduction libre (plus précisément la règle du système FD', cf.[46] p. 364), qui est bien une règle d'élimination droite de la soustraction :

$$\frac{\Gamma, A \Rightarrow B \vdash \Delta \qquad \Gamma', A \vdash \Delta', B}{\Gamma, \Gamma' \vdash \Delta, \Delta'} \qquad\qquad \frac{\Gamma \vdash \Delta, A - B \qquad \Gamma', A \vdash \Delta', B}{\Gamma, \Gamma' \vdash \Delta, \Delta'}$$

Définition 4.3.1 *On appelle déduction naturelle soustractive, noté* SND*, le système de la déduction naturelle classique CND plus les règles d'introduction et d'élimination de la soustraction suivantes:*

$$\frac{\Gamma \vdash \Delta, A \qquad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma' \vdash \Delta, \Delta', A - B} \qquad\qquad \frac{\Gamma \vdash \Delta, A - B \qquad \Gamma', A \vdash \Delta', B}{\Gamma, \Gamma' \vdash \Delta, \Delta'}$$

Remarque. Les règles de SND sont dérivables à partir de celles de CND_{sym}. En effet :

– Dérivation de la règle d'élimination droite

$$\dfrac{\Gamma \vdash \Delta, A - B \qquad \dfrac{\Gamma', A \vdash \Delta', B}{\Gamma', A - B \vdash \Delta'}}{\Gamma, \Gamma' \vdash \Delta, \Delta'}$$

– Dérivation de la règle d'introduction droite

$$\dfrac{\Gamma \vdash \Delta, A \qquad \dfrac{A - B \vdash A - B \qquad \Gamma', B \vdash \Delta'}{\Gamma', A \vdash \Delta', A - B}}{\Gamma, \Gamma' \vdash \Delta, \Delta', A - B}$$

Réciproquement, les règles de CND_{sym} sont dérivables à partir de celles de SND. En effet :

– Dérivation de la règle d'introduction gauche

$$\dfrac{A - B \vdash A - B \qquad \Gamma, A \vdash \Delta, B}{\Gamma, A - B \vdash \Delta}$$

– Dérivation de la règle d'élimination gauche

$$\dfrac{\dfrac{A \vdash A \qquad \Gamma', B \vdash \Delta'}{\Gamma' \vdash \Delta', A - B} \qquad \Gamma, A - B \vdash \Delta}{\Gamma, \Gamma', A \vdash \Delta, \Delta'}$$

4.4 Calcul des séquents LK et déduction naturelle classique CND

Les liens entre le calcul des séquents LK et la déduction naturelle classique, ici présentée sous forme de calcul des séquents ([48]), sont très étroits. Dans les deux systèmes, les séquents manipulés ont de multiples hypothèses et de multiples conclusions. De plus, puisque la règle de coupure est ajoutée au système CND, la traduction d'un système vers l'autre est directe. Nous donnons ici une traduction pour les règles de l'implication.

⋆ De CND vers LK

Nous traduisons la règle d'élimination droite de CND :

$$\dfrac{\Gamma \vdash A \Rightarrow B, \Delta \qquad \Gamma' \vdash A, \Delta'}{\Gamma, \Gamma' \vdash B, \Delta, \Delta'}$$

en utilisant la règle d'introduction gauche de LK :

$$\dfrac{\Gamma \vdash A \Rightarrow B, \Delta \qquad \dfrac{\Gamma' \vdash A, \Delta' \qquad B \vdash B}{\Gamma', A \Rightarrow B \vdash B, \Delta'}}{\Gamma, \Gamma' \vdash B, \Delta, \Delta'}$$

Remarque. La β-réduction (vue comme une étape du processus de normalisation des preuves) qui correspond à la transformation de preuve suivante :

$$\dfrac{\dfrac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \Rightarrow B, \Delta} \qquad \Gamma' \vdash A, \Delta'}{\Gamma, \Gamma' \vdash B, \Delta, \Delta'} \quad \rightsquigarrow \quad \dfrac{\Gamma, A \vdash B, \Delta \qquad \Gamma' \vdash A, \Delta'}{\Gamma, \Gamma' \vdash B, \Delta, \Delta'}$$

est simplement traduite en « remontant » une occurrence de la règle de coupure :

$$\dfrac{\dfrac{\Gamma, A \vdash B, \Delta}{\Gamma \vdash A \Rightarrow B, \Delta} \qquad \dfrac{\Gamma' \vdash A, \Delta' \qquad B \vdash B}{\Gamma', A \Rightarrow B \vdash B, \Delta'}}{\Gamma, \Gamma' \vdash B, \Delta, \Delta'} \quad \rightsquigarrow \quad \dfrac{\Gamma, A \vdash B, \Delta \qquad \Gamma' \vdash A, \Delta'}{\Gamma, \Gamma' \vdash B, \Delta, \Delta'}$$

$\star$ De LK vers CND

Nous traduisons la règle d'introduction gauche de LK :

$$\dfrac{\Gamma \vdash A, \Delta \qquad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \Rightarrow B \vdash \Delta, \Delta'}$$

en utilisant la règle d'élimination droite de CND :

$$\dfrac{\dfrac{A \Rightarrow B \vdash A \Rightarrow B \qquad \Gamma \vdash A, \Delta}{\Gamma, A \Rightarrow B \vdash B, \Delta} \qquad \Gamma', B \vdash \Delta'}{\Gamma, \Gamma', A \Rightarrow B \vdash \Delta, \Delta'}$$

$\star$ À propos de la contrainte de LK^i

Nous pouvons vérifier que la contrainte intuitionniste de LK^i, (qui s'applique aussi à CND) consistant à restreindre à une conclusion de déchargement d'une hypothèse n'est pas stable par réduction des preuves. Plus précisément, cette contrainte n'est pas stable par substitution. En effet, en déduction naturelle, puisqu'il n'y a aucune règle d'introduction gauche, l'élimination de la coupure se ramène à une simple substitution (cf. chapitre 4 pour plus de détails). L'exemple suivant, où la coupure est « déplacée » d'une règle à droite, le montre :

$$\dfrac{\Gamma \vdash \Delta, C \qquad \dfrac{C, \Gamma', A \vdash B}{C, \Gamma' \vdash A \Rightarrow B}}{\Gamma, \Gamma' \vdash \Delta, A \Rightarrow B} \quad \rightsquigarrow \quad \dfrac{\dfrac{\Gamma \vdash \Delta, C \qquad C, \Gamma', A \vdash B}{\Gamma, \Gamma', A \vdash \Delta, B}}{\Gamma, \Gamma' \vdash \Delta, A \Rightarrow B}$$

4.5 La restriction intuitionniste CND^i de CND

Nous décorons les séquents par des « liens de dépendance non orientés » entre hypothèses et conclusions. Pour cela, nous nommons les *occurrences* d'hypothèses $x, y, z \ldots$ et les *occurrences* de conclusion $\alpha, \beta, \gamma \ldots$ de tout séquent. Un nom d'hypothèse, ou de conclusion n'apparaît jamais deux fois dans le même séquent.

Des liens de dépendances sur un séquent $\Gamma_1^{x_1}, \ldots, \Gamma_n^{x_n} \vdash \Delta_1^{\alpha_1}, \ldots, \Delta_m^{\alpha_m}$ forment donc une partie de $\{x_1, \ldots, x_n\} \times \{\alpha_1, \ldots, \alpha_m\}$ ou encore un ensemble de couples (x_i, α_j) où $1 \leq i \leq n$ et $1 \leq j \leq m$. Cette relation entre (occurrences d') hypothèses et (occurrences de) conclusions peut donc être indifféremment représentée soit en décorant chaque conclusion par l'ensemble des hypothèses auxquelles elle est liée (on dira aussi « dont elle dépend ») soit en décorant chaque hypothèse par l'ensemble des conclusions auxquelles elle est liée (on dira aussi « qui en dépendent »).

Remarquez que pour la première représentation, les noms des conclusions n'ont pas besoin d'être explicités, de même que dans la seconde représentation les noms des hypothèses n'ont pas besoin d'être explicités. Le passage d'une représentation à l'autre est direct.

Exemple. Considérons le séquent $A^x, B^y, C^z \vdash D^\alpha, E^\beta, F^\gamma, G^\delta$ décoré par les liens de dépendances $\{(x,\beta),(x,\delta),(z,\alpha),(z,\beta),(z,\delta)\}$. Ce séquent pourra être représenté en décorant les conclusions :

$$A^x, B^y, C^z \vdash \{z\} : D,\ \{x,z\} : E,\ \{\} : F,\ \{x,z\} : G$$

ou en décorant les hypothèses :

$$\{\beta,\delta\}{:}A,\ \{\} : B,\ \{\alpha,\beta,\delta\} : C \vdash D^\alpha, E^\beta, F^\gamma, G^\delta$$

Nous utiliserons pour commencer la représentation à droite (*i.e.* chaque conclusion est décorée par l'ensemble des hypothèses dont elle dépend). La décoration des séquents est définie par récurrence sur la dérivation.

$\star$ Liens de dépendances explicites

Axiome

$$A^x \vdash \{x\} : A$$

Règles d'affaiblissement

$$\frac{\Gamma \vdash \Delta}{\Gamma, A^x \vdash \Delta} \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta,\ \{\} : A}$$

Règles de contraction

$$\frac{\Gamma, A^x, A^y \vdash S_1 : \Delta_1, \ldots,\ S_n : \Delta_n}{\Gamma, A^z \vdash S_1[z/x, z/y] : \Delta_1, \ldots,\ S_n[z/x, z/y] : \Delta_n} \qquad \frac{\Gamma \vdash \Delta,\ U : A,\ V : A}{\Gamma \vdash \Delta,\ U \cup V : A}$$

Règle d'introduction du $\Rightarrow$

$$\frac{\Gamma, A^x \vdash S_1 : \Delta_1, \ldots,\ S_n : \Delta_n,\ V : B}{\Gamma \vdash S_1 \setminus \{x\} : \Delta_1, \ldots,\ S_n \setminus \{x\} : \Delta_n,\ V \setminus \{x\} : (A \Rightarrow B)}$$

Règle d'élimination du $\Rightarrow$

$$\frac{\Gamma \vdash \Delta,\ U : (A \Rightarrow B) \qquad \Gamma' \vdash \Delta',\ V : A}{\Gamma, \Gamma' \vdash \Delta, \Delta',\ U \cup V : B}$$

Règle d'introduction du $\wedge$

$$\frac{\Gamma \vdash \Delta,\ U : A \qquad \Gamma' \vdash \Delta',\ V : B}{\Gamma, \Gamma' \vdash \Delta, \Delta',\ U \cup V : A \wedge B}$$

Règle d'élimination du $\wedge$

$$\frac{\Gamma \vdash \Delta,\ U : A \wedge B}{\Gamma \vdash \Delta,\ U : A} \qquad \frac{\Gamma \vdash \Delta,\ V : A \wedge B}{\Gamma \vdash \Delta,\ V : B}$$

Règle d'introduction du $\vee$

$$\frac{\Gamma \vdash \Delta,\ U : A}{\Gamma \vdash \Delta,\ U : A \vee B} \qquad \frac{\Gamma \vdash \Delta,\ V : B}{\Gamma \vdash \Delta,\ V : A \vee B}$$

Règle d'élimination du $\vee$

$$\frac{\Gamma \vdash \Delta,\ U : A \vee B \qquad \Gamma' \vdash \Delta',\ V : (A \Rightarrow C) \qquad \Gamma'' \vdash \Delta'',\ W : (B \Rightarrow C)}{\Gamma, \Gamma', \Gamma'' \vdash \Delta, \Delta', \Delta'',\ U \cup V \cup W : C}$$

Règle de coupure

$$\frac{\Gamma \vdash S_1 : \Delta_1, \ldots,\ S_n : \Delta_n,\ S : A \qquad \Gamma', A^x \vdash S'_1 : \Delta'_1, \ldots,\ S'_p : \Delta'_p}{\Gamma, \Gamma' \vdash S_1 : \Delta_1, \ldots,\ S_n : \Delta_n,\ S'_1[S/x] : \Delta'_1, \ldots,\ S'_p[S/x] : \Delta'_p}$$

où la notation $U[V/x]$ signifie « remplacer la variable x, si elle apparaît dans U, par les variables de V qui n'apparaissent pas déjà dans U ».

Remarque. On peut vérifier, pour les règles qui respectent une symétrie droite/gauche (contraction, affaiblissement, échange) ainsi que pour la règle de coupure (laquelle est sa propre règle symétrique) que les décorations à gauche, symétriques des décoration droites données, définissent bien les mêmes liens de dépendances. Par exemple, la règle de contraction gauche aurait pu être décorée ainsi (où les conclusions sont nommées cette fois ci) :

$$\frac{\Gamma,\ U : A,\ V : A \vdash \Delta}{\Gamma,\ U \cup V : A \vdash \Delta}$$

Les règles d'introduction et d'élimination des quantificateurs universels du premier et du second ordre laissent les liens de dépendances inchangés :

Règle d'intro. (droite) (si x n'apparaît pas dans Γ, Δ) **et règle d'élim. (droite) du** $\forall$

$$\frac{\Gamma \vdash \Delta,\ S : A}{\Gamma \vdash \Delta,\ S : \forall x A} \qquad\qquad \frac{\Gamma \vdash \Delta,\ S : \forall x A}{\Gamma \vdash \Delta,\ S : A[t/x]}$$

Règle d'intro. (droite) (si x n'apparaît pas dans Γ, Δ) **et règle d'élim. (droite) du** $\forall^2$

$$\frac{\Gamma \vdash \Delta,\ S : A}{\Gamma \vdash \Delta,\ S : \forall X A} \qquad\qquad \frac{\Gamma \vdash \Delta,\ S : \forall X A}{\Gamma \vdash \Delta,\ S : A[T/X]}$$

Définition 4.5.1 *On dit qu'une instance de la règle de déchargement d'une hypothèse (introduction du $\Rightarrow$) respecte la* ***contrainte intuitionniste*** *si x n'apparaît dans aucun S_i. La règle devient alors :*

$$\frac{\Gamma, A^x \vdash S_1 : \Delta_1, \ldots,\ S_n : \Delta_n,\ V : B}{\Gamma \vdash S_1 : \Delta_1, \ldots,\ S_n : \Delta_n,\ V \setminus \{x\} : (A \Rightarrow B)} \quad \textit{où } x \notin S_1 \cup \ldots \cup S_n$$

Autrement dit, pour rester intuitionniste, *une conclusion ne peut être déchargée d'une hypothèse que si aucune autre conclusion n'en dépend.* Cette contrainte se généralise à une preuve :

Définition 4.5.2 *Une preuve de* CND *à liens de dépendances explicites est dite* ***intuitionniste*** *si tout déchargement d'hypothèse apparaissant dans la preuve respecte la contrainte intuitionniste.*

Notation. On appelle CND^i la restriction de CND aux preuves intuitionnistes.

Théorème 4.5.3 *Tout séquent propositionnel* $\Gamma \vdash A$ *dérivable dans* CND^i *est prouvable en déduction naturelle intuitionniste (i.e. dans le système* NJ de Gentzen*).*

Preuve. Pour prouver ce théorème, nous allons montrer que pour tout séquent $\Gamma \vdash \Delta$ dérivable dans CND^i, *au moins une conclusion* Δ_j est dérivable sous les hypothèses Γ dans la déduction naturelle intuitionniste NJ. Plus précisément, si Σ est le sous-ensemble de Γ auxquelles Δ_j est liée par les liens de dépendances explicites qui décorent la preuve dans CND de $\Gamma \vdash \Delta$, alors $\Sigma \vdash \Delta_j$ est dérivable dans NJ.

Nous allons montrer de plus que toute conclusion Δ_j qui ne provient pas, directement ou indirectement, d'un affaiblissement vérifie cette propriété. Afin d'expliciter ces conclusions qui proviennent d'un affaiblissement, nous allons donc ajouter une nouvelle décoration possible pour les conclusions : le *symbole* ∞. Les règles d'affaiblissement et de contraction droites deviennent alors :

$$\frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta,\ \infty : B} \qquad \frac{\Gamma \vdash \Delta,\ U : B,\ V : B}{\Gamma \vdash \Delta,\ W : B}$$

où W est défini ainsi :

– $W = \infty$ si $U = \infty$ et $V = \infty$

– $W = V$ si $U = \infty$ et $V \neq \infty$

– $W = U$ si $U \neq \infty$ et $V = \infty$

– $W = V$ *ou* $W = U$ si $U \neq \infty$ et $V \neq \infty$

Dans le dernier cas, les deux décorations conviennent : le processus d'extraction est donc intrinsèquement non déterministe.

Pour propager cette information dans les preuves, nous étendons les opérations ensemblistes utilisées pour définir la décoration des preuves à ce nouveau symbole de la façon suivante :

– $U \cup \infty = \infty \cup U = \infty$

– $\infty \backslash \{x\} = \infty$

– $\infty[U/x] = \infty$

– $U[\infty/x] = \infty$, si x apparaît dans U et $U[\infty/x] = U$ sinon

Si Γ est l'ensemble d'hypothèses nommées $\Gamma^{x_1}, \dots, \Gamma^{x_n}$ et S est un sous-ensemble de $\{x_1, \dots, x_n\}$, on note Γ^S le sous-ensemble d'hypothèses de Γ dont le nom figure dans S. On vérifie alors par récurrence sur la preuve, pour tout séquent dérivé $\Gamma \vdash \Delta$, que d'une part, il existe au moins une conclusion qui n'est pas décorée par ∞, et d'autre part, pour toute conclusion Δ_j décorée par l'ensemble S (éventuellement vide) de noms d'hypothèses, $\Gamma^S \vdash_{\mathrm{NJ}} \Delta_j$.

Le seul cas délicat est l'introduction de l'implication. Par définition, une occurrence de la règle de déchargement d'hypothèse :

$$\frac{\Gamma, A^x \vdash V : B,\ S_1 : \Delta_1, \dots,\ S_n : \Delta_n}{\Gamma \vdash V \setminus \{x\} : (A \Rightarrow B),\ S_1 \setminus \{x\} : \Delta_1, \dots,\ S_n \setminus \{x\} : \Delta_n}$$

respecte la contrainte intuitionniste si, pour toute conclusion Δ_j décorée par S_j on a soit $S_j = \infty$ soit S_j est un ensemble de noms d'hypothèses tel que $x \notin S_j$. Dans ce dernier cas, par hypothèse de récurrence, on a $\Gamma^{S_j} \vdash_{\mathrm{NJ}} \Delta_j$ et par conséquent, $\Gamma^{S_j \setminus \{x\}} \vdash_{\mathrm{NJ}} \Delta_j$ puisque $S_j \setminus \{x\} = S_j$. Maintenant, si $V \neq \infty$, alors par hypothèse de récurrence, $\Gamma^V \vdash_{\mathrm{NJ}} B$ d'où $\Gamma^{V \setminus \{x\}} \vdash_{\mathrm{NJ}} A \Rightarrow B$. Si $V = \infty$ alors il existe au moins une conclusion Δ_j qui ne soit pas décorée par ∞ et donc telle que $\Gamma^{S_j} \vdash_{\mathrm{NJ}} \Delta_j$. □

Remarques

1. La preuve ci-dessus suggère de supprimer la règle d'affaiblissement droite, pour ne dériver que les conclusions qui sont effectivement prouvées. Pourtant, si l'on supprime cette règle, on retrouve un calcul où *tous les séquents* ont une unique conclusion (en effet aucune autre règle n'introduit de conclusion multiples). Il est néanmoins possible, pour obtenir plusieurs conclusions prouvées, de remplacer la règle d'affaiblissement par la règle suivante :

$$\frac{\Gamma \vdash S_1 : \Delta_1, \ldots,\ S_n : \Delta_n \qquad \Gamma' \vdash S'_1 : \Delta'_1, \ldots,\ S'_m : \Delta'_m}{\Gamma, \Gamma' \vdash S_1 : \Delta_1, \ldots,\ S_n : \Delta_n,\ S'_1 : \Delta'_1, \ldots,\ S'_m : \Delta'_m}$$

   Cette règle est dérivable à partir de la règle d'affaiblissement de la manière suivante, où A une formule quelconque :

$$\frac{\dfrac{\dfrac{\Gamma \vdash S_1 : \Delta_1, \ldots,\ S_n : \Delta_n}{\Gamma \vdash S_1 : \Delta_1, \ldots,\ S_n : \Delta_n,\ \{\} : A \Rightarrow \Delta'_j} \qquad \dfrac{\Gamma' \vdash \Delta'}{\Gamma' \vdash \Delta',\ \{\} : A}}{\Gamma, \Gamma' \vdash S_1 : \Delta_1, \ldots,\ S_n : \Delta_n,\ S'_1 : \Delta'_1, \ldots,\ S'_m : \Delta'_m,\ \{\} : \Delta'_j}}{\Gamma, \Gamma' \vdash S_1 : \Delta_1, \ldots,\ S_n : \Delta_n,\ S'_1 : \Delta'_1, \ldots,\ S'_m : \Delta'_m}$$

2. La règle d'introduction gauche du $\vee$, (qui est une règle de CND_{sym}, duale de la règle d'introduction droite du $\wedge$), qui est équivalente à celle de CND en logique classique, et ou les décorations sont représentées sur les hypothèses :

$$\frac{\Gamma,\ U : A \vdash \Delta \qquad \Gamma',\ V : B \vdash \Delta'}{\Gamma,\ U \cup V : A \vee B \vdash \Delta, \Delta'}$$

   ne respecte pas la seconde condition du théorème : il suffit de considérer l'exemple suivant,

$$\frac{\{\alpha\} : A \vdash A^{\alpha} \qquad \{\beta\} : B \vdash B^{\beta}}{\{\alpha, \beta\} : A \vee B \vdash A^{\alpha}, B^{\beta}}$$

   où clairement aucun des deux séquents $A \vee B \vdash A$ et $A \vee B \vdash B$ n'est dérivable dans NJ. Nous allons voir que ces règles sont toujours intuitionnistes, et que seule la propriété (très forte) consistant à pouvoir extraire une dérivation d'au moins une des conclusions (sous les mêmes hypothèses) dans NJ n'est plus vérifiée.

4.6 La restriction intuitionniste SND^i de SND

⋆ Décoration des règles de la soustraction

Comme nous l'avons déjà souligné, les liens de dépendances sont non-orientés. La relation qu'ils traduisent peut donc indifféremment être représentée du côté des hypothèses ou du côté des conclusions. Les décorations des règles de l'implication se symétrisent donc directement en des décorations pour les règles de la la soustraction. Il suffit pour cela de passer à la représentation des liens de dépendances qui décorent les hypothèses (nous utiliserons les lettres grecques α,β,γ,... pour nommer les conclusions).

Règle d'introduction gauche du $-$

$$\frac{S_1 : \Gamma_1, \ldots,\ S_n : \Gamma_n,\ V : A \vdash B^\alpha, \Delta}{S_1 \setminus \{\alpha\} : \Gamma_1, \ldots,\ S_n \setminus \{\alpha\} : \Gamma_n,\ V \setminus \{\alpha\} : A - B \vdash \Delta}$$

Règle d'élimination gauche du $-$

$$\frac{\Gamma,\ U : A - B \vdash \Delta \qquad \Gamma',\ V : A \vdash \Delta'}{\Gamma, \Gamma',\ U \cup V : B \vdash \Delta, \Delta'}$$

Il est alors possible de déduire les décorations des règles d'introduction et d'élimination droite de la soutraction, à partir de leur dérivation donnée en sous-section 4.2. On utilise une décoration des hypothèses, la règle de coupure se décore alors ainsi :

$$\frac{S_1 : \Gamma_1, \ldots,\ S_n : \Gamma_n \vdash \Delta,\ A^\alpha \qquad S'_1 : \Gamma'_1, \ldots,\ S'_p : \Gamma'_p,\ S' : A \vdash \Delta'}{S_1[S'/\alpha] : \Gamma_1, \ldots,\ S_n[S'/\alpha] : \Gamma_n,\ S'_1 : \Gamma'_1, \ldots,\ S'_p : \Gamma'_p \vdash \Delta, \Delta'}$$

Dérivation de la règle d'introduction droite :

$$\frac{S_1 : \Gamma_1, \ldots,\ S_n : \Gamma_n \vdash \Delta,\ A^\beta \qquad \dfrac{\{\alpha\} : A - B \vdash (A-B)^\alpha \quad S'_1 : \Gamma'_1, \ldots,\ S'_p : \Gamma'_p,\ S' : B \vdash \Delta'}{S'_1 : \Gamma'_1, \ldots,\ S'_p : \Gamma'_p,\ S' \cup \{\alpha\} : A \vdash \Delta', (A-B)^\alpha}}{S_1[\ S' \cup \{\alpha\}/\beta] : \Gamma_1, \ldots,\ S_n[\ S' \cup \{\alpha\}/\beta] : \Gamma_n,\ S'_1 : \Gamma'_1, \ldots,\ S'_p : \Gamma'_p \vdash \Delta, \Delta', (A-B)^\alpha}$$

Dérivation de la règle d'élimination droite :

$$\frac{S_1 : \Gamma_1, \ldots,\ S_n : \Gamma_n \vdash \Delta,\ A - B^\alpha \qquad \dfrac{S'_1 : \Gamma'_1, \ldots,\ S'_p : \Gamma'_p,\ U : A \vdash \Delta', B^\beta}{S'_1\setminus\{\beta\} : \Gamma'_1, \ldots,\ S'_p\setminus\{\beta\} : \Gamma'_p,\ U\setminus\{\beta\} : A - B \vdash \Delta'}}{S_1[U\setminus\{\beta\}/\alpha] : \Gamma_1, \ldots,\ S_n[U\setminus\{\beta\}/\alpha] : \Gamma_n,\ S'_1\setminus\{\beta\} : \Gamma'_1, \ldots,\ S'_p\setminus\{\beta\} : \Gamma'_p \vdash \Delta, \Delta'}$$

On obtient donc les deux règles décorées suivantes :

Règle d'introduction droite du $-$

$$\frac{S_1 : \Gamma_1, \ldots,\ S_n : \Gamma_n \vdash \Delta,\ A^\beta \qquad S'_1 : \Gamma'_1, \ldots,\ S'_p : \Gamma'_p,\ S' : B \vdash \Delta'}{S_1[\ S' \cup \{\alpha\}/\beta] : \Gamma_1, \ldots,\ S_n[\ S' \cup \{\alpha\}/\beta] : \Gamma_n,\ S'_1 : \Gamma'_1, \ldots,\ S'_p : \Gamma'_p \vdash \Delta, \Delta', (A-B)^\alpha}$$

Règle d'élimination droite du $-$

$$\frac{S_1 : \Gamma_1, \ldots,\ S_n : \Gamma_n \vdash \Delta,\ A - B^\alpha \qquad S'_1 : \Gamma'_1, \ldots,\ S'_p : \Gamma'_p,\ U : A \vdash \Delta', B^\beta}{S_1[U\setminus\{\beta\}/\alpha] : \Gamma_1, \ldots,\ S_n[U\setminus\{\beta\}/\alpha] : \Gamma_n,\ S'_1\setminus\{\beta\} : \Gamma'_1, \ldots,\ S'_p\setminus\{\beta\} : \Gamma'_p \vdash \Delta, \Delta'}$$

⋆ **Contrainte intuitionniste pour la soustraction**

Nous avons vu en section 4 que les règles ci-dessus non contraintes permettent de dériver $A \wedge \sim A \vdash \bot$ et donc transformer la négation faible en une négation classique. Il est donc nécessaire, pour rester conservatif sur la logique intutionniste, de contraindre aussi les règles de la soustraction. La contrainte s'obtient par dualite :

Définition 4.6.1 *On dit qu'une instance de la règle de déchargement d'une conclusion (introduction du $-$) respecte la* ***contrainte intuitionniste*** *si β n'apparaît dans aucun S_i. La règle devient alors :*

$$\frac{S_1 : \Gamma_1, \ldots, S_n : \Gamma_n, V : A \vdash B^\beta, \Delta}{S_1 : \Gamma_1, \ldots, S_n : \Gamma_n, V \setminus \{\beta\} : A - B \vdash \Delta} \quad \textit{où } \beta \notin S_1 \cup \ldots \cup S_n$$

Définition 4.6.2 *Une preuve de* CND_{sym} *à liens de dépendances explicites est dite* ***intuitionniste*** *si tout déchargement d'hypothèse ou de conclusion apparaissant dans la preuve respecte la contrainte intuitionniste.*

Remarque. Il est alors possible de dériver la contrainte intuitionniste pour la règle d'élimination de la soustraction de SND, à partir de sa dérivation qui utilise une instance de la règle d'introduction gauche. On obtient la définition suivante :

Définition 4.6.3 *On dit qu'une instance de la règle d'élimination (droite) de la soustraction respecte la* ***contrainte intuitionniste*** *si β n'apparaît dans aucun S'_i. La règle devient alors :*

$$\frac{S_1 : \Gamma_1, \ldots,\ S_n : \Gamma_n \vdash \Delta,\ A - B^\alpha \qquad S'_1 : \Gamma'_1, \ldots,\ S'_p : \Gamma'_p,\ U : A \vdash \Delta', B^\beta}{S_1[U \backslash \{\beta\}/\alpha] : \Gamma_1, \ldots,\ S_n[U \backslash \{\beta\}/\alpha] : \Gamma_n,\ S'_1 \backslash \{\beta\} : \Gamma'_1, \ldots,\ S'_p \backslash \{\beta\} : \Gamma'_p \vdash \Delta, \Delta'}$$

où $\beta \notin S'_1 \cup \ldots \cup S'_n$.

Définition 4.6.4 *Une preuve de* SND *à liens de dépendances explicites est dite* ***intuitionniste*** *si toute occurrence de la règle d'introduction de l'implication et toute occurence de la règle d'élimination de la soustraction apparaissant dans la preuve respecte la contrainte intuitionniste.*

Notation. On appelle CND^i_{sym} la restriction de CND_{sym} aux preuves intuitionnistes.

La dernière section de ce chapitre sera consacrée à la preuve du théorème suivant :

Théorème 4.6.5 *Tout séquent $\Gamma \vdash \Delta$ dérivable dans* CND^i_{sym} *est valide en logique intuitionniste soustractive (i.e. le séquent $\Gamma^\wedge \vdash \Delta^\vee$ est dérivable dans le calcul catégorique symétrique).*

Corollaire 4.6.6 *Tout séquent dérivable dans* SND^i *est valide en logique intuitionniste soustractive.*

Preuve. On observe que, par construction, la dérivation des règles de SND^i dans CND_{sym} sont en fait des dérivations de CND^i_{sym}. □

5 Conservativité de CND^i_{sym}

Nous montrons ici que CND^i_{sym} est conservatif sur la logique intuitionniste soustractive (propositionnelle, du premier et du second ordre). La preuve se fait en montrant que toute dérivation d'un séquent dans CND^i_{sym} peut être transformée en une dérivation ne contenant aucun déchargement d'hypothèse ou de conclusion, mais contenant des axiomes dérivables dans le calcul catégorique symétrique. Une telle dérivation est clairement intuitionniste puisque les seules règles « classiques » ont été supprimées.

La partie délicate de la preuve consiste donc à montrer qu'une dérivation de CND^i_{sym} peut être transformée en une dérivation ne contenant aucun déchargement d'hypothèse ou de conclusion. Pour cela, on montre que le déchargement d'hypothèse « commute » avec toutes les autres règles. Par dualité, on aura donc aussi montré que le déchargement de conclusion commute avec toutes les autres règles. Ces preuves de commutation utiliseront éventuellement des séquents dérivables dans le calcul catégorique symétrique. Nous montrerons donc en premier que le déchargement d'hypothèse ou de conclusion appliqué directement à un séquent dérivable dans le calcul catégorique symétrique ne pose pas de problème.

Nous noterons $hyp(\Delta)$ l'ensemble des (occurrences d') hypothèses liées à au moins une des (occurrences de) conclusions de Δ et nous utiliserons la règle contrainte de déchargement d'une hypothèse généralisée suivante :

$$\frac{\Gamma \vdash \Delta}{\Gamma\backslash\{H\} \vdash \Delta\backslash S, H \Rightarrow S^{\vee}} \quad \text{où } H \notin hyp(\Delta\backslash S)$$

où H n'apparaît pas nécéssairement dans Γ, et les occurences d'hypothèses de S n'apparaissent pas nécessairement toutes (et éventuellement aucune) dans Γ.

Sa règle duale qui généralise le déchargement d'une conclusion (où $concl(\Gamma)$ désigne l'ensemble des occurrences de conclusions liées à au moins une hypothèse de Γ) s'écrit :

$$\frac{\Gamma \vdash \Delta}{\Gamma\backslash S, S^{\wedge} - C \vdash \Delta\backslash\{C\}} \quad \text{où } C \notin concl(\Gamma\backslash S)$$

où C n'apparaît pas nécéssairement dans Δ, et les occurences de conclusions de S n'apparaissent pas nécessairement toutes (et éventuellemnet aucune) dans Δ.

Remarque. Ces règles sont bien sûr dérivable dans CND^i_{sym}. Pour le voir, rappelons que les règles suivantes sont dérivables dans CND^i_{sym} :

$$\frac{\Gamma,\ U_1 : \Gamma_1, \ldots,\ U_n : \Gamma_n \vdash \Delta}{\Gamma,\ U_1 \cup \ldots \cup U_n : \Gamma_1 \wedge \ldots \wedge \Gamma_n \vdash \Delta} \qquad \frac{\Gamma \vdash \Delta,\ U_1 : \Delta_1, \ldots,\ U_n : \Delta_n}{\Gamma \vdash \Delta,\ U_1 \cup \ldots \cup U_n : \Delta_1 \vee \ldots \vee \Delta_n}$$

En effet, il suffit de plusieurs introduction du connecteur $\wedge$ à gauche (resp. $\vee$ à droite) suivie de plusieurs contractions gauche (resp. droite). Les règles généralisées de déchargement d'hypothèses (resp de conclusions) s'obtiennent alors en plusieurs affaiblissements à droite et à gauche, suivie de la règle généralisée d'introduction du $\vee$ (resp du $\wedge$) ci-dessus puis du déchargement habituel de l'hypothèse (resp. de la conclusion).

Nous aurons aussi besoin du lemme suivant :

Lemme 5.0.7 *Tout séquent $A \vdash B$ dérivable dans le calcul catégorique symétrique est dérivable dans CND^i_{sym}, où A et B sont liés.*

Preuve. Tous les axiomes et toutes les règles du calcul catégorique symétrique sont clairement dérivables dans CND^i_{sym}. Le lien, s'il n'est pas présent, peut être ajouté simplement par :

$$\dfrac{\dfrac{A^z \vdash \{\} : B}{\vdash \{\} : A \Rightarrow B} \qquad A^x \vdash \{x\} : A}{A^x \vdash \{x\} : B}$$

□

Lemme 5.0.8 *Pour tout séquent $\Gamma \vdash \Delta$ dérivable dans CND_{sym} à partir de séquents dérivables dans le calcul catégorique sans utiliser ni la règle de déchargement d'une hypothèse, ni la règle de déchargement d'une conclusion, le séquent $\Gamma^\wedge \vdash \Delta^\vee$ est dérivable dans le calcul catégorique symétrique.*

Preuve. Les seules règles posant problème ont été écartées, les autres se traduisent comme dans la preuve de la proposition 2.2. □

Théorème 5.0.9 *Le système CND_{sym} est conservatif sur la logique intuituionniste soustractive.*

Preuve. La preuve se fait en deux étapes :

1. Dans la sous-section « commutation des règles » ci-dessous nous montrons que toute occurrence de la règle généralisée d'introduction de l'implication (et par dualité d'introduction gauche de la soustraction) peut être remplacée par une ou plusieurs occurrences de ces règles sur des dérivations strictement plus petites en nombre de règles. Dans la sous-section « axiomes » ci-dessous, nous montrons que ces occurrences de règles peuvent être supprimées lorsqu'elles sont appliquées à un axiome généralisé.

2. Le lemme 5.0.8 ci-dessus nous dit que la preuve obtenue en supprimant toutes les occurrences de la règle généralisée d'introduction de l'implication (et par dualité d'introduction gauche de la soustraction) est valide en logique intuituionniste soustractive.

□

⋆ Axiomes

Proposition 5.0.10 *Dans CND^i_{sym}, l'ensemble des séquents appartenant à l'une des trois catégories suivantes :*

- $\Gamma, A \vdash \Delta, B$ où $A \vdash B$ est dérivable dans le calcul catégorique symétrique, et l'unique lien qui décore ce séquent relie A et B.

- $\Gamma, Y \vdash \Delta$ où $Y \vdash \perp$ est dérivable dans le calcul catégorique symétrique,

- $\Gamma \vdash \Delta, X$ où $\top \vdash X$ est dérivable dans le calcul catégorique symétrique.

est stable par la règle généralisée de déchargement d'une hypothèse (resp. d'une conclusion).

Preuve. Considérons la règle généralisée de déchargement d'une hypothèse pour les trois catégories de séquents :

1. Le séquent hypothèse est de la forme $\Gamma, A \vdash \Delta, B$ où $A \vdash B$ est dérivable dans le calcul catégorique symétrique, et l'unique lien qui décore ce séquent relie A et B.

   – Premier cas, $H \neq A$ et $B \notin S$.

   $$\frac{\Gamma, A \vdash \Delta, B}{\Gamma\backslash\{H\}, A \vdash \Delta\backslash S, B, H \Rightarrow S^{\vee}}$$

   Le séquent conclusion est bien de la première forme.

   – Deuxième cas, $H \neq A$ et H est déchargé sur $S \cup \{B\}$

   $$\frac{\Gamma, A \vdash \Delta, B}{\Gamma\backslash\{H\}, A \vdash \Delta\backslash S, H \Rightarrow (S^{\vee} \vee B)}$$

   Le séquent conclusion est bien de la troisième forme puisque dans dans le calcul catégorique symétrique, si $A \vdash B$ est dérivable et $B \in S$ alors $A \vdash H \Rightarrow (S^{\vee} \vee B)$ est aussi dérivable.

   – Troisième cas, $H = A$ et A est déchargé sur $S \cup \{B\}$

   $$\frac{\Gamma, A \vdash \Delta, B}{\Gamma \vdash \Delta\backslash S, A \Rightarrow (S^{\vee} \vee B)}$$

   Le séquent conclusion est bien de la troisième forme puisque dans dans le calcul catégorique symétrique, si $A \vdash B$ est dérivable alors $\top \vdash A \Rightarrow (S^{\vee} \vee B)$ est aussi dérivable.

   Le cas $H = A$ et $B \notin S$, c'est-à-dire où A est déchargé sur une autre conclusion que B viole la contrainte intuitionniste et n'est donc pas à considérer.

2. Le séquent hypothèse est de la forme $\Gamma, Y \vdash \Delta$ où $Y \vdash \bot$ est dérivable dans le calcul catégorique symétrique.

   – Premier cas, $H \neq Y$ et $H \notin hyp(\Delta\backslash S)$

   $$\frac{\Gamma, Y \vdash \Delta}{\Gamma\backslash\{H\}, Y \vdash \Delta\backslash S, H \Rightarrow S^{\vee}}$$

   Le séquent conclusion est toujours de la deuxième forme.

   – Deuxième cas, $H = Y$ et $Y \notin hyp(\Delta\backslash S)$

   $$\frac{\Gamma, Y \vdash \Delta}{\Gamma \vdash \Delta\backslash S, Y \Rightarrow S^{\vee}}$$

   Puisque $Y \vdash \bot$ est dérivable dans le calcul catégorique symétrique, on en déduit que $Y \vdash S^{\vee}$ et donc $\top \vdash Y \Rightarrow S^{\vee}$ sont aussi dérivables dans le calcul catégorique symétrique, et le séquent conclusion est donc de la troisième forme.

3. Le séquent hypothèse est de la forme $\Gamma \vdash \Delta, X$ où $\top \vdash X$ est dérivable dans le calcul catégorique symétrique.

   – Premier cas, $X \notin S$

   $$\frac{\Gamma \vdash \Delta, X}{\Gamma \backslash \{H\} \vdash \Delta \backslash S, X, H \Rightarrow S^{\vee}}$$

   Le séquent conclusion est toujours de la troisième forme.

   – Deuxième cas, H est déchargé sur $S \cup \{X\}$

   $$\frac{\Gamma \vdash \Delta, X}{\Gamma \backslash \{H\} \vdash \Delta \backslash S, H \Rightarrow (S^{\vee} \vee X)}$$

   Puisque $\top \vdash X$ est dérivable dans le calcul catégorique symétrique, on en déduit que $H \vdash S^{\vee} \vee X$ et donc $\top \vdash H \Rightarrow (S^{\vee} \vee X)$ sont aussi dérivables dans le calcul catégorique symétrique, et le séquent conclusion est donc toujours de la troisième forme.

La clôture pour la règle de déchargement d'une conclusion s'obtient par dualité. □

⋆ Commutation des règles

Affaiblissement gauche

– Premier cas, $H \neq A$:

$$\frac{\dfrac{\Gamma \vdash^{1} \Delta}{\Gamma, A \vdash^{2} \Delta}}{\Gamma \backslash \{H\}, A \vdash^{3} \Delta \backslash S, A \Rightarrow S^{\vee}}$$

où $H \notin hyp_2(\Delta \backslash S)$ et donc $H \notin hyp_1(\Delta \backslash S)$

On remplace par :

$$\frac{\dfrac{\Gamma \vdash \Delta}{\Gamma \backslash \{H\} \vdash \Delta \backslash S, A \Rightarrow S^{\vee}}}{\Gamma \backslash \{H\}, A \vdash \Delta \backslash S, A \Rightarrow S^{\vee}}$$

– Deuxième cas, $H = A$:

$$\frac{\dfrac{\Gamma \vdash^{1} \Delta}{\Gamma, A \vdash^{2} \Delta}}{\Gamma \vdash^{3} \Delta \backslash S, A \Rightarrow S^{\vee}}$$

où $A \notin hyp_2(\Delta \backslash S)$ et donc $A \notin hyp_1(\Delta \backslash S)$

On remplace par

$$\frac{\Gamma \vdash \Delta}{\Gamma \vdash \Delta \backslash S, A \Rightarrow S^{\vee}}$$

Contraction gauche

- Premier cas, $H = A^z$:

$$\dfrac{\dfrac{\Gamma, A^x, A^y \vdash^1 \Delta}{\Gamma, A^z \vdash^2 \Delta}}{\Gamma \vdash^3 \Delta \backslash S, A \Rightarrow S^\vee}$$

  où $A^z \notin hyp_2(\Delta\backslash S)$ et donc $A^x \notin hyp_1(\Delta\backslash S)$ et $A^y \notin hyp_1(\Delta\backslash\{B^S\})$

  On remplace par :

$$\dfrac{\dfrac{\Gamma, A^x, A^y \vdash \Delta}{\Gamma, A^z \vdash \Delta\backslash S, A \Rightarrow S^\vee}}{\Gamma \vdash \Delta\backslash S, A \Rightarrow (A \Rightarrow S^\vee)}$$

  puis on coupe avec le séquent $A \Rightarrow (A \Rightarrow S^\vee) \vdash A \Rightarrow S^\vee$ dérivable dans le calcul catégorique symétrique.

- Deuxième cas, $H \neq A^z$:

$$\dfrac{\dfrac{\Gamma, A^x, A^y \vdash^1 \Delta}{\Gamma, A^z \vdash^2 \Delta}}{\Gamma\backslash\{H\}, A^z \vdash^3 \Delta\backslash S, H \Rightarrow S^\vee}$$

  où $H \notin hyp_2(\Delta\backslash S)$ et donc $H \notin hyp_1(\Delta\backslash S)$

  On remplace par :

$$\dfrac{\dfrac{\Gamma, A^x, A^y, H \vdash \Delta}{\Gamma\backslash\{H\}, A^x, A^y \vdash \Delta\backslash S, H \Rightarrow S^\vee}}{\Gamma\backslash\{H\}, A^z \vdash \Delta\backslash S, H \Rightarrow S^\vee}$$

Contraction droite

- Premier cas, $B \notin S$:

$$\dfrac{\dfrac{\Gamma \vdash^1 \Delta, B^\alpha, B^\beta}{\Gamma \vdash^2 \Delta, B^\gamma}}{\Gamma\backslash\{H\} \vdash^3 \Delta\backslash S, H \Rightarrow S^\vee, B^\gamma}$$

  où $H \notin hyp_2(\Delta\backslash S, B^\gamma)$ et donc $H \notin hyp_1(\Delta\backslash S, B^\alpha, B^\beta)$

  On remplace par :

$$\dfrac{\dfrac{\Gamma \vdash \Delta, B, B}{\Gamma\backslash\{H\} \vdash \Delta\backslash S, H \Rightarrow S^\vee, B, B}}{\Gamma\backslash\{H\} \vdash \Delta\backslash S, H \Rightarrow S^\vee, B}$$

- Deuxième cas, H est déchargé sur $S \cup \{B\}$:

$$\dfrac{\dfrac{\Gamma \vdash^1 \Delta, B^\alpha, B^\beta}{\Gamma \vdash^2 \Delta, B^\gamma}}{\Gamma\backslash\{H\} \vdash^3 \Delta\backslash S, H \Rightarrow (S^\vee \vee B)}$$

  où $H \notin hyp_2(\Delta\backslash S, B^\gamma)$ et donc $H \notin hyp_1(\Delta\backslash S, B^\alpha, B^\beta)$

  On remplace par :

$$\dfrac{\Gamma \vdash \Delta, B, B}{\Gamma\backslash\{H\} \vdash \Delta\backslash S, H \Rightarrow (S^\vee \vee B \vee B)}$$

  puis on coupe avec le séquent $H \Rightarrow (S^\vee \vee B \vee B) \vdash H \Rightarrow (S^\vee \vee B)$ dérivable dans le calcul catégorique symétrique.

Affaiblissement droite

– Premier cas, $B \notin S$:

$$\dfrac{\dfrac{\Gamma \vdash^1 \Delta}{\Gamma \vdash^2 \Delta, B}}{\Gamma\backslash\{H\} \vdash^3 \Delta\backslash S, H \Rightarrow S^\vee, B}$$

où $H \notin hyp_2(\Delta\backslash S, B)$ et donc $H \notin hyp_1(\Delta\backslash S)$

On remplace par :

$$\dfrac{\dfrac{\Gamma \vdash \Delta}{\Gamma\backslash\{H\} \vdash \Delta\backslash S, H \Rightarrow S^\vee}}{\Gamma\backslash\{H\} \vdash \Delta\backslash S, H \Rightarrow S^\vee, B}$$

– Deuxième cas, H est déchargé sur $S \cup \{B\}$:

$$\dfrac{\dfrac{\Gamma \vdash^1 \Delta}{\Gamma \vdash^2 \Delta, B^\alpha}}{\Gamma\backslash\{H\} \vdash^3 \Delta\backslash S, H \Rightarrow (S^\vee \vee B)}$$

où $H \notin hyp_2(\Delta\backslash S, B^\alpha)$ et donc $H \notin hyp_1(\Delta\backslash S)$

On remplace par :

$$\dfrac{\Gamma \vdash \Delta}{\Gamma\backslash\{H\} \vdash \Delta\backslash S, H \Rightarrow S^\vee}$$

puis on coupe avec le séquent $H \Rightarrow S^\vee \vdash H \Rightarrow (S^\vee \vee B)$ dérivable dans le calcul catégorique symétrique.

Élimination droite de la implication

– Premier cas, $B \notin S$:

$$\dfrac{\dfrac{\Gamma' \vdash^1 \Delta', A \Rightarrow B \qquad \Gamma'' \vdash^4 \Delta'', A}{\Gamma', \Gamma'' \vdash^2 \Delta', \Delta'', B}}{(\Gamma', \Gamma'')\backslash\{H\} \vdash^3 (\Delta', \Delta'')\backslash S, B, H \Rightarrow S^\vee}$$

où $H \notin hyp_2(B, (\Delta', \Delta'')\backslash S)$ et donc, en posant $S' = S \cap \Gamma'$ et $S'' = S \cap \Gamma''$, on a $H \notin hyp_1(A \Rightarrow B, \Delta\backslash S')$ et $H \notin hyp_4(A, \Delta\backslash S')$

On remplace par :

$$\dfrac{\dfrac{\dfrac{\Gamma' \vdash \Delta', A \Rightarrow B}{\Gamma'\backslash\{H\} \vdash \Delta'\backslash S', H \Rightarrow S'^\vee, A \Rightarrow B} \qquad \dfrac{\Gamma'' \vdash \Delta'', A}{\Gamma''\backslash\{H\} \vdash \Delta''\backslash S'', H \Rightarrow S''^\vee, A}}{(\Gamma', \Gamma'')\backslash\{H\} \vdash (\Delta', \Delta'')\backslash S, H \Rightarrow S'^\vee, H \Rightarrow S''^\vee, B}}{(\Gamma', \Gamma'')\backslash\{H\} \vdash (\Delta', \Delta'')\backslash S, (H \Rightarrow S'^\vee) \vee (H \Rightarrow S''^\vee), B}$$

puis on coupe avec le séquent $(H \Rightarrow S'^\vee) \vee (H \Rightarrow S''^\vee) \vdash H \Rightarrow (S'^\vee \vee S''^\vee)$ dérivable dans le calcul catégorique symétrique.

– Deuxième cas, H est déchargé sur $S \cup \{B\}$: :

$$\dfrac{\dfrac{\Gamma' \vdash^1 \Delta', A \Rightarrow B \qquad \Gamma'' \vdash^4 \Delta'', A}{\Gamma', \Gamma'' \vdash^2 \Delta', \Delta'', B}}{(\Gamma', \Gamma'')\backslash\{H\} \vdash^3 (\Delta', \Delta'')\backslash S, H \Rightarrow (S^\vee \vee B)}$$

où $H \notin hyp_2((\Delta', \Delta'')\backslash S)$ et donc, en posant $S' = S \cap \Gamma'$ et $S'' = S \cap \Gamma''$, on a $H \notin hyp_1(\Delta\backslash S')$ et $H \notin hyp_4(\Delta\backslash S')$.

On remplace par :

$$\dfrac{\dfrac{\Gamma' \vdash \Delta', A \Rightarrow B}{\Gamma'\backslash\{H\} \vdash \Delta'\backslash S', H \Rightarrow (S'^{\vee} \vee (A \Rightarrow B))} \qquad \dfrac{\Gamma'' \vdash \Delta'', A}{\Delta''\backslash S', H \Rightarrow (S''^{\vee} \vee A)}}{(\Gamma', \Gamma'')\backslash\{H\} \vdash (\Delta', \Delta'')\backslash S, (H \Rightarrow (S'^{\vee} \vee (A \Rightarrow B))) \wedge (H \Rightarrow (S''^{\vee} \vee A))}$$

puis on coupe avec le séquent $(H \Rightarrow (S'^{\vee} \vee (A \Rightarrow B))) \wedge (H \Rightarrow (S''^{\vee} \vee A)) \vdash H \Rightarrow (S^{\vee} \vee B)$ dérivable dans le calcul catégorique symétrique.

Introduction droite de la conjonction

– Premier cas, $A \wedge B \notin S$:

$$\dfrac{\dfrac{\Gamma' \vdash^1 \Delta', A \qquad \Gamma'' \vdash^4 \Delta'', B}{\Gamma', \Gamma'' \vdash^2 \Delta', \Delta'', A \wedge B}}{(\Gamma', \Gamma'')\backslash\{H\} \vdash^3 (\Delta', \Delta'')\backslash S, A \wedge B, H \Rightarrow S^{\vee}}$$

où $H \notin hyp_2(A, B, (\Delta', \Delta'')\backslash S)$ et donc, en posant $S' = S \cap \Gamma'$ et $S'' = S \cap \Gamma''$, on a $H \notin hyp_1(A, \Delta\backslash S')$ et $H \notin hyp_4(B, \Delta\backslash S')$

On remplace par :

$$\dfrac{\dfrac{\dfrac{\Gamma' \vdash \Delta', A}{\Gamma'\backslash\{H\} \vdash \Delta'\backslash S', H \Rightarrow S'^{\vee}, A} \qquad \dfrac{\Gamma'' \vdash \Delta'', B}{\Gamma''\backslash\{H\} \vdash \Delta''\backslash S'', H \Rightarrow S''^{\vee}, B}}{(\Gamma', \Gamma'')\backslash\{H\} \vdash (\Delta', \Delta'')\backslash S, H \Rightarrow S'^{\vee}, H \Rightarrow S''^{\vee}, A \wedge B}}{(\Gamma', \Gamma'')\backslash\{H\} \vdash (\Delta', \Delta'')\backslash S, (H \Rightarrow S'^{\vee}) \vee (H \Rightarrow S''^{\vee}), A \wedge B}$$

puis on coupe avec le séquent $(H \Rightarrow S'^{\vee}) \vee (H \Rightarrow S''^{\vee}) \vdash H \Rightarrow (S'^{\vee} \vee S''^{\vee})$ dérivable dans le calcul catégorique symétrique.

– Deuxième cas, H est déchargé sur $S \cup \{A \wedge B\}$:

$$\dfrac{\dfrac{\Gamma' \vdash^1 \Delta', A \qquad \Gamma'' \vdash^4 \Delta'', B}{\Gamma', \Gamma'' \vdash^2 \Delta', \Delta'', A \wedge B}}{(\Gamma', \Gamma'')\backslash\{H\} \vdash^3 (\Delta', \Delta'')\backslash S, H \Rightarrow (S^{\vee} \vee (A \wedge B))}$$

où $H \notin hyp_2((\Delta', \Delta'')\backslash S)$ et donc, en posant $S' = S \cap \Gamma'$ et $S'' = S \cap \Gamma''$, on a $H \notin hyp_1(\Delta\backslash S')$ et $H \notin hyp_4(\Delta\backslash S')$.

On remplace par :

$$\dfrac{\dfrac{\Gamma' \vdash \Delta', A}{\Gamma'\backslash\{H\} \vdash \Delta'\backslash S', H \Rightarrow (S'^{\vee} \vee A)} \qquad \dfrac{\Gamma'' \vdash \Delta'', B}{\Delta''\backslash S', H \Rightarrow (S''^{\vee} \vee B)}}{(\Gamma', \Gamma'')\backslash\{H\} \vdash (\Delta', \Delta'')\backslash S, (H \Rightarrow (S'^{\vee} \vee A)) \wedge (H \Rightarrow (S''^{\vee} \vee B))}$$

puis on coupe avec le séquent $(H \Rightarrow (S'^{\vee} \vee A)) \wedge (H \Rightarrow (S''^{\vee} \vee B)) \vdash H \Rightarrow (S^{\vee} \vee (A \wedge B))$ dérivable dans le calcul catégorique symétrique.

Élimination droite de la conjonction On ne traite que le cas de première « projection », l'autre s'obtenant de manière analogue.

– Premier cas, $A \notin S$:

$$\dfrac{\dfrac{\Gamma \vdash^1 \Delta, A \wedge B}{\Gamma \vdash^2 \Delta, A}}{\Gamma \backslash \{H\} \vdash^3 \Delta \backslash S, H \Rightarrow S^\vee, A}$$

  où $H \notin hyp_2(A, \Delta \backslash S)$ et donc $H \notin hyp_1(A \wedge B, \Delta \backslash S)$

  On remplace par :

$$\dfrac{\dfrac{\Gamma \vdash \Delta, A \wedge B}{\Gamma \backslash \{H\} \vdash \Delta \backslash S, H \Rightarrow S^\vee, A \wedge B}}{\Gamma \backslash \{H\} \vdash \Delta \backslash S, H \Rightarrow S^\vee, A}$$

– Deuxième cas, H est déchargé sur $S \cup \{A\}$:

$$\dfrac{\dfrac{\Gamma \vdash^1 \Delta, A \wedge B}{\Gamma \vdash^2 \Delta, A}}{\Gamma \backslash \{H\} \vdash^3 \Delta \backslash S, H \Rightarrow (S^\vee \vee A)}$$

  où $H \notin hyp_2(\Delta \backslash S)$ et donc $H \notin hyp_1(\Delta \backslash S)$ et $A \notin hyp_4(\Delta \backslash S)$

  On remplace par :

$$\dfrac{\Gamma \vdash \Delta, A \wedge B}{\Gamma \backslash \{H\} \vdash \Delta \backslash S, H \Rightarrow (S^\vee \vee (A \wedge B))}$$

  puis on coupe avec le séquent $H \Rightarrow (S^\vee \vee (A \wedge B)) \vdash H \Rightarrow (S^\vee \vee A)$ dérivable dans le calcul catégorique symétrique.

Élimination gauche de la soustraction

– Premier cas, $H \neq A$:

$$\dfrac{\dfrac{\Gamma', A - B \vdash^1 \Delta' \qquad \Gamma'', B \vdash^4 \Delta''}{\Gamma', \Gamma'', A \vdash^2 \Delta', \Delta''}}{(\Gamma', \Gamma'') \backslash \{H\}, A \vdash^3 (\Delta', \Delta'') \backslash S, H \Rightarrow S^\vee}$$

  où $H \notin hyp_2((\Delta', \Delta'') \backslash S)$ et donc, en posant $S' = S \cap \Gamma'$ et $S'' = S \cap \Gamma''$, on a $H \notin hyp_1(\Delta \backslash S')$ et $H \notin hyp_4(\Delta \backslash S')$

  On remplace par :

$$\dfrac{\dfrac{\dfrac{\Gamma', A - B \vdash \Delta'}{\Gamma' \backslash \{H\}, A - B \vdash \Delta' \backslash S', H \Rightarrow S'^\vee} \qquad \dfrac{\Gamma'', B \vdash \Delta''}{\Gamma'' \backslash \{H\}, B \vdash \Delta'', H \Rightarrow S''^\vee}}{(\Gamma', \Gamma'') \backslash \{H\}, A \vdash (\Delta', \Delta'') \backslash S, H \Rightarrow S'^\vee, H \Rightarrow S''^\vee}}{(\Gamma', \Gamma'') \backslash \{H\}, A \vdash (\Delta', \Delta'') \backslash S, (H \Rightarrow S'^\vee) \vee (H \Rightarrow S''^\vee)}$$

  puis on coupe avec le séquent $(H \Rightarrow S'^\vee) \vee (H \Rightarrow S''^\vee) \vdash H \Rightarrow (S'^\vee \vee S''^\vee)$ dérivable dans le calcul catégorique symétrique.

– Deuxième cas, $H = A$:

$$\cfrac{\cfrac{\Gamma', A - B \vdash^1 \Delta' \qquad \Gamma'', B \vdash^4 \Delta''}{\Gamma', \Gamma'', A \vdash^2 \Delta', \Delta''}}{\Gamma', \Gamma'' \vdash^3 (\Delta', \Delta'')\backslash S, A \Rightarrow S^\vee}$$

où $A \notin hyp_2((\Delta', \Delta'')\backslash S)$ et donc, en posant $S' = S \cap \Gamma'$ et $S'' = S \cap \Gamma''$, on a $A - B \notin hyp_1(\Delta\backslash S')$ et $B \notin hyp_4(\Delta\backslash S')$.

On remplace par :

$$\cfrac{\cfrac{\Gamma', A - B \vdash \Delta'}{\Gamma'\backslash\{H\} \vdash \Delta'\backslash S', (A - B) \Rightarrow S'^\vee} \qquad \cfrac{\Gamma'', B \vdash \Delta''}{\Delta''\backslash S', B \Rightarrow S''^\vee}}{(\Gamma', \Gamma'')\backslash\{H\} \vdash (\Delta', \Delta'')\backslash S, ((A - B) \Rightarrow S'^\vee) \wedge (B \Rightarrow S''^\vee)}$$

puis on coupe avec le séquent $((A - B) \Rightarrow S'^\vee) \wedge (B \Rightarrow S''^\vee) \vdash (A \Rightarrow (S'^\vee \vee S''^\vee)$ dérivable dans le calcul catégorique symétrique.

Introduction gauche de la soustraction

– Premier cas, $H \neq A - B$:

$$\cfrac{\cfrac{\Gamma, A \vdash^1 \Delta, B}{\Gamma, A - B \vdash^2 \Delta}}{\Gamma\backslash\{H\}, A - B \vdash^3 \Delta\backslash S, H \Rightarrow S^\vee}$$

où $H \notin hyp_2(\Delta\backslash S)$ et $H \notin hyp_1(B)$ donc $H \notin hyp_1(\Delta\backslash S, B)$

On remplace par :

$$\cfrac{\cfrac{\Gamma, A \vdash \Delta, B}{\Gamma\backslash\{H\}, A \vdash \Delta\backslash S, H \Rightarrow S^\vee, B}}{\Gamma\backslash\{H\}, A - B \vdash \Delta\backslash S, H \Rightarrow S^\vee}$$

– Deuxième cas, $H = A - B$:

$$\cfrac{\cfrac{\Gamma, A \vdash^1 \Delta, B}{\Gamma, A - B \vdash^2 \Delta}}{\Gamma \vdash^3 \Delta\backslash S, (A - B) \Rightarrow S^\vee}$$

où $H \notin hyp_2(\Delta\backslash S)$ et $H \notin hyp_1(B)$ donc $H \notin hyp_1(\Delta\backslash S, B)$

On remplace par :

$$\cfrac{\Gamma, A \vdash \Delta, B}{\Gamma \vdash \Delta\backslash S, A \Rightarrow (S^\vee \vee B)}$$

puis on coupe avec le séquent $A \Rightarrow (S^\vee \vee B) \vdash (A - B) \Rightarrow S^\vee$ dérivable dans le calcul catégorique symétrique.

Introduction gauche de la disjonction

– Premier cas, $H \neq A \vee B$:

$$\dfrac{\dfrac{\Gamma', A \vdash^1 \Delta' \qquad \Gamma'', B \vdash^4 \Delta''}{\Gamma', \Gamma'', A \vee B \vdash^2 \Delta', \Delta''}}{(\Gamma', \Gamma'')\backslash\{H\}, A \vee B \vdash^3 (\Delta', \Delta'')\backslash S, H \Rightarrow S^{\vee}}$$

où $H \notin hyp_2((\Delta', \Delta'')\backslash S)$ et donc, en posant $S' = S \cap \Gamma'$ et $S'' = S \cap \Gamma''$, on a $H \notin hyp_1(\Delta\backslash S')$ et $H \notin hyp_4(\Delta\backslash S')$

On remplace par :

$$\dfrac{\dfrac{\dfrac{\Gamma', A \vdash \Delta'}{\Gamma'\backslash\{H\}, A \vdash \Delta'\backslash S', H \Rightarrow S'^{\vee}} \qquad \dfrac{\Gamma'', B \vdash \Delta''}{\Gamma''\backslash\{H\}, B \vdash \Delta'', H \Rightarrow S''^{\vee}}}{(\Gamma', \Gamma'')\backslash\{H\}, A \vee B \vdash (\Delta', \Delta'')\backslash S, H \Rightarrow S'^{\vee}, H \Rightarrow S''^{\vee}}}{(\Gamma', \Gamma'')\backslash\{H\}, A \vee B \vdash (\Delta', \Delta'')\backslash S, (H \Rightarrow S'^{\vee}) \vee (H \Rightarrow S''^{\vee})}$$

puis on coupe avec le séquent $(H \Rightarrow S'^{\vee}) \vee (H \Rightarrow S''^{\vee}) \vdash H \Rightarrow (S'^{\vee} \vee S''^{\vee})$ dérivable dans le calcul catégorique symétrique.

– Deuxième cas, $H = A \vee B$:

$$\dfrac{\dfrac{\Gamma', A \vdash^1 \Delta' \qquad \Gamma'', B \vdash^4 \Delta''}{\Gamma', \Gamma'', A \vee B \vdash^2 \Delta', \Delta''}}{\Gamma', \Gamma'' \vdash^3 (\Delta', \Delta'')\backslash S, A \vee B \Rightarrow S^{\vee}}$$

où $A \vee B \notin hyp_2((\Delta', \Delta'')\backslash S)$ et donc, en posant $S' = S \cap \Gamma'$ et $S'' = S \cap \Gamma''$, on a $A \notin hyp_1(\Delta\backslash S')$ et $B \notin hyp_4(\Delta\backslash S')$.

On remplace par :

$$\dfrac{\dfrac{\Gamma', A \vdash \Delta'}{\Gamma'\backslash\{H\} \vdash \Delta'\backslash S', A \Rightarrow S'^{\vee}} \qquad \dfrac{\Gamma'', B \vdash \Delta''}{\Delta''\backslash S', B \Rightarrow S''^{\vee}}}{(\Gamma', \Gamma'')\backslash\{H\} \vdash (\Delta', \Delta'')\backslash S, (A \Rightarrow S'^{\vee}) \wedge (B \Rightarrow S''^{\vee})}$$

puis on coupe avec le séquent $(A \Rightarrow S'^{\vee}) \wedge (B \Rightarrow S''^{\vee}) \vdash (A \vee B) \Rightarrow (S'^{\vee} \vee S''^{\vee})$ dérivable dans le calcul catégorique symétrique.

Élimination gauche de la disjonction On ne traite que le cas de première « injection », l'autre s'obtenant de manière analogue.

– Premier cas, $H \neq A$:

$$\dfrac{\dfrac{\Gamma, A \vee B \vdash^1 \Delta}{\Gamma, A \vdash^2 \Delta}}{\Gamma\backslash\{H\}, A \vdash^3 \Delta\backslash S, H \Rightarrow S^{\vee}}$$

où $H \notin hyp_2(\Delta\backslash S)$ et donc $H \notin hyp_1(\Delta\backslash S)$

On remplace par :

$$\dfrac{\dfrac{\Gamma, A \vee B \vdash \Delta}{\Gamma\backslash\{H\}, A \vee B \vdash \Delta\backslash S, H \Rightarrow S^{\vee}}}{\Gamma\backslash\{H\}, A \vdash \Delta\backslash S, H \Rightarrow S^{\vee}}$$

– Deuxième cas, $H = A$:

$$\dfrac{\dfrac{\Gamma, A \vee B \vdash^1 \Delta}{\Gamma, A \vdash^2 \Delta}}{\Gamma \backslash \{H\} \vdash^3 \Delta \backslash S, A \Rightarrow S^\vee}$$

où $H \notin hyp_2(\Delta \backslash S)$ et donc $H \notin hyp_1(\Delta \backslash S)$ et $A \notin hyp_4(\Delta \backslash S)$

On remplace par :

$$\dfrac{\Gamma, A \vee B \vdash \Delta}{\Gamma \backslash \{H\} \vdash \Delta \backslash S, (A \vee B) \Rightarrow S^\vee}$$

puis on coupe avec le séquent $(A \vee B) \Rightarrow S^\vee \vdash A \Rightarrow S^\vee$ dérivable dans le calcul catégorique symétrique.

Règle de coupure

$$\dfrac{\dfrac{\Gamma' \vdash^1 A, \Delta' \qquad \Gamma'', A \vdash^4 \Delta''}{\Gamma', \Gamma'' \vdash^2 \Delta', \Delta''}}{(\Gamma', \Gamma'') \backslash \{H\} \vdash^3 (\Delta', \Delta'') \backslash S, H \Rightarrow S^\vee}$$

où $H \notin hyp_2((\Delta', \Delta'') \backslash S)$ et donc, en posant $S' = S \cap \Gamma'$ et $S'' = S \cap \Gamma''$, on a

– Premier cas, $H \notin \Gamma'$

On remplace par :

$$\dfrac{\dfrac{\dfrac{\Gamma' \vdash \Delta', A}{\Gamma' \vdash \Delta' \backslash S', H \Rightarrow S'^\vee, A} \qquad \dfrac{\Gamma'', A \vdash \Delta''}{\Gamma'' \backslash \{H\}, A \vdash \Delta'' \backslash S', H \Rightarrow S''^\vee}}{(\Gamma', \Gamma'') \backslash \{H\} \vdash (\Delta', \Delta'') \backslash S, H \Rightarrow S'^\vee, H \Rightarrow S''^\vee}}{(\Gamma', \Gamma'') \backslash \{H\} \vdash (\Delta', \Delta'') \backslash S, (H \Rightarrow S'^\vee) \vee (H \Rightarrow S''^\vee)}$$

puis on coupe avec le séquent $(H \Rightarrow S'^\vee) \vee (H \Rightarrow S''^\vee) \vdash H \Rightarrow (S'^\vee \vee S''^\vee)$ dérivable dans le calcul catégorique symétrique.

– Deuxième cas, $H \notin \Gamma''$ et $H \notin hyp_1(A)$ d'où $H \notin hyp_1(A, \Delta' \backslash S')$

On remplace par :

$$\dfrac{\dfrac{\dfrac{\Gamma' \vdash \Delta', A}{\Gamma' \backslash \{H\} \vdash \Delta' \backslash S', H \Rightarrow S'^\vee, A} \qquad \dfrac{\Gamma'', A \vdash \Delta''}{\Gamma'', A \vdash \Delta'' \backslash S', H \Rightarrow S''^\vee}}{(\Gamma', \Gamma'') \backslash \{H\} \vdash (\Delta', \Delta'') \backslash S, H \Rightarrow S'^\vee, H \Rightarrow S''^\vee}}{(\Gamma', \Gamma'') \backslash \{H\} \vdash (\Delta', \Delta'') \backslash S, (H \Rightarrow S'^\vee) \vee (H \Rightarrow S''^\vee)}$$

puis on coupe avec le séquent $(H \Rightarrow S'^\vee) \vee (H \Rightarrow S''^\vee) \vdash H \Rightarrow (S'^\vee \vee S''^\vee)$ dérivable dans le calcul catégorique symétrique.

– Troisième cas, $H \in \Gamma'$ et $H \in hyp_1(A)$ et comme $H \notin hyp_1(\Delta'' \backslash S'')$ on a $A \notin hyp_1(\Delta'' \backslash S')$.

On remplace par :

$$\dfrac{\dfrac{\Gamma' \vdash \Delta', A}{\Gamma' \backslash \{H\} \vdash \Delta' \backslash S', H \Rightarrow (S'^\vee \vee A)} \qquad \dfrac{\Gamma'', A \vdash \Delta''}{\Gamma'' \vdash \Delta' \backslash S', A \Rightarrow S''^\vee}}{(\Gamma', \Gamma'') \backslash \{H\} \vdash (\Delta', \Delta'') \backslash S, (H \Rightarrow (S'^\vee \vee A)) \wedge (A \Rightarrow S''^\vee)}$$

puis on coupe avec le séquent $(H \Rightarrow (S'^\vee \vee A)) \wedge (A \Rightarrow S''^\vee) \vdash H \Rightarrow (S'^\vee \vee S''^\vee)$ dérivable dans le calcul catégorique symétrique.

Introduction droite du $\forall$ (où x n'apparaît pas dans Γ, Δ)

– Premier cas, $\forall x A \notin S$:

$$\dfrac{\dfrac{\Gamma \vdash^1 \Delta, A}{\Gamma \vdash^2 \Delta, \forall x A}}{\Gamma \backslash \{H\} \vdash^3 \Delta \backslash S, H \Rightarrow S^\vee, \forall x A}$$

où $H \notin hyp_2(\Delta \backslash S, \forall x A)$ et donc $H \notin hyp_1(\Delta \backslash S, A)$

On remplace par :

$$\dfrac{\dfrac{\Gamma \vdash^1 \Delta, A}{\Gamma \backslash \{H\} \vdash^3 \Delta \backslash S, H \Rightarrow S^\vee, A}}{\Gamma \backslash \{H\} \vdash^3 \Delta \backslash S, H \Rightarrow S^\vee, \forall x A}$$

– Deuxième cas, H est déchargé sur $S \cup \{\forall x A\}$:

$$\dfrac{\dfrac{\Gamma \vdash^1 \Delta, A}{\Gamma \vdash^2 \Delta, \forall x A}}{\Gamma \backslash \{H\} \vdash^3 \Delta \backslash S, H \Rightarrow (S^\vee \vee \forall x A)}$$

où $H \notin hyp_2(\Delta \backslash S)$ et donc $H \notin hyp_1(\Delta \backslash S)$ et $A \notin hyp_4(\Delta \backslash S)$

On remplace par :

$$\dfrac{\dfrac{\Gamma \vdash \Delta, A}{\Gamma \backslash \{H\} \vdash \Delta \backslash S, H \Rightarrow (S^\vee \vee A)}}{\Gamma \backslash \{H\} \vdash \Delta \backslash S, \forall x (H \Rightarrow (S^\vee \vee A))}$$

puis on coupe avec le séquent $\forall x (H \Rightarrow (S^\vee \vee A)) \vdash H \Rightarrow (S^\vee \vee \forall x A)$ dérivable dans le calcul catégorique symétrique, car x n'apparaît pas dans $H, S^\vee$ (la preuve utilise DIS).

Élimination droite du $\forall$

– Premier cas, $A \notin S$:

$$\dfrac{\dfrac{\Gamma \vdash^1 \Delta, \forall x A}{\Gamma \vdash^2 \Delta, A[t/x]}}{\Gamma \backslash \{H\} \vdash^3 \Delta \backslash S, H \Rightarrow S^\vee, A}$$

où $H \notin hyp_2(A[t/x], \Delta \backslash S)$ et donc $H \notin hyp_1(\forall x A, \Delta \backslash S)$

On remplace par :

$$\dfrac{\dfrac{\Gamma \vdash^1 \Delta, \forall x A}{\Gamma \backslash \{H\} \vdash^3 \Delta \backslash S, H \Rightarrow S^\vee, \forall x A}}{\Gamma \backslash \{H\} \vdash^3 \Delta \backslash S, H \Rightarrow S^\vee, A[t/x]}$$

– Deuxième cas, H est déchargé sur $S \cup \{\forall x A\}$:

$$\dfrac{\dfrac{\Gamma \vdash^1 \Delta, \forall x A}{\Gamma \vdash^2 \Delta, A[t/x]}}{\Gamma \backslash \{H\} \vdash^3 \Delta \backslash S, H \Rightarrow (S^\vee \vee A[t/x])}$$

où $H \notin hyp_2(\Delta \backslash S)$ et donc $H \notin hyp_1(\Delta \backslash S)$ et $A \notin hyp_4(\Delta \backslash S)$

On remplace par :

$$\dfrac{\Gamma \vdash \Delta, \forall x A}{\Gamma \backslash \{H\} \vdash \Delta \backslash S, H \Rightarrow (S^\vee \vee A)}$$

puis on coupe avec le séquent $H \Rightarrow (S^\vee \vee \forall x A) \vdash H \Rightarrow (S^\vee \vee A[t/x])$ dérivable dans le calcul catégorique symétrique.

Introduction gauche du $\exists$ (où x n'apparaît pas dans Γ, Δ)

– Premier cas, $H \neq \exists x A$:

$$\dfrac{\dfrac{\Gamma, A \vdash^1 \Delta}{\Gamma, \exists x A \vdash^2 \Delta}}{\Gamma \backslash \{H\} \vdash^3 \Delta \backslash S, H \Rightarrow S^\vee}$$

où $H \notin hyp_2(\Delta \backslash S)$ et donc $H \notin hyp_1(\Delta \backslash S)$

On remplace par :

$$\dfrac{\dfrac{\Gamma, A \vdash \Delta, A}{\Gamma \backslash \{H\}, A \vdash \Delta \backslash S, H \Rightarrow S^\vee}}{\Gamma \backslash \{H\}, \exists x A \vdash \Delta \backslash S, H \Rightarrow S^\vee}$$

– Deuxième cas, $H = \exists x A$:

$$\dfrac{\dfrac{\Gamma, A \vdash^1 \Delta}{\Gamma, \exists x A \vdash^2 \Delta}}{\Gamma \vdash^3 \Delta \backslash S, \exists x A \Rightarrow S^\vee}$$

où $\exists x A \notin hyp_2(\Delta \backslash S)$ et donc $A \notin hyp_1(\Delta \backslash S)$

On remplace par :

$$\dfrac{\dfrac{\Gamma \vdash \Delta, A}{\Gamma \backslash \{H\} \vdash \Delta \backslash S, A \Rightarrow S^\vee}}{\Gamma \backslash \{H\} \vdash \Delta \backslash S, \forall x (A \Rightarrow S^\vee)}$$

puis on coupe avec le séquent $\forall x (A \Rightarrow S^\vee) \vdash \exists x A \Rightarrow S^\vee$ dérivable dans le calcul catégorique symétrique, car x n'apparaît pas dans $S^\vee$.

Élimination gauche du $\exists$

– Premier cas, $H \neq \exists x A$:

$$\dfrac{\dfrac{\Gamma, \exists x A \vdash^1 \Delta}{\Gamma, A[t/x] \vdash^2 \Delta}}{\Gamma \backslash \{H\}, A[t/x] \vdash^3 \Delta \backslash S, H \Rightarrow S^\vee}$$

où $H \notin hyp_2(\Delta \backslash S)$ et donc $H \notin hyp_1(\Delta \backslash S)$

On remplace par :

$$\dfrac{\dfrac{\Gamma, \exists x A \vdash \Delta}{\Gamma \backslash \{H\}, \exists x A \vdash \Delta \backslash S, H \Rightarrow S^\vee}}{\Gamma \backslash \{H\}, A[t/x] \vdash \Delta \backslash S, H \Rightarrow S^\vee}$$

– Deuxième cas, $H = \exists x A$:

$$\dfrac{\dfrac{\Gamma, \exists x A \vdash^1 \Delta}{\Gamma, A[t/x] \vdash^2 \Delta}}{\Gamma \vdash^3 \Delta \backslash S, A[t/x] \Rightarrow S^\vee}$$

où $A[t/x] \notin hyp_2(\Delta \backslash S)$ et donc $\exists x A \notin hyp_1(\Delta \backslash S)$

On remplace par :

$$\dfrac{\Gamma, \exists x A \vdash \Delta}{\Gamma \backslash \{H\} \vdash \Delta \backslash S, \exists x A \Rightarrow S^\vee}$$

puis on coupe avec le séquent $\exists x A \Rightarrow S^\vee \vdash A[t/x] \Rightarrow S^\vee$ dérivable dans le calcul catégorique symétrique.

Chapitre 4

Le $\lambda\mu$-calcul intuitionniste

Dans ce chapitre, nous utilisons le $\lambda\mu$-calcul de M. PARIGOT pour définir un λ-calcul étendu par des opérateurs de contrôle **catch** et **throw**, appelé $\lambda_{\mathbf{ct}}$-calcul, qui soit confluent, à la différence du $\lambda_{\mathbf{c/t}}$-calcul non-déterministe de H. NAKANO. Nous définissons alors une contrainte sur les $\lambda\mu$-termes purs (resp. sur les $\lambda_{\mathbf{ct}}$-termes purs), stable pour les règles de réduction du $\lambda\mu$-calcul (resp. du $\lambda_{\mathbf{ct}}$-calcul) et telle que tout $\lambda\mu$-terme contraint (resp. $\lambda_{\mathbf{ct}}$-terme) typable soit typé par un théorème intuitionniste. Plus précisément, pour tout $\lambda\mu$-terme (resp. $\lambda_{\mathbf{ct}}$-terme) *typé*, la contrainte définie dans ce chapitre correspond exactement à la contrainte intuitionniste sur le jugement de typage du terme, vu comme une preuve de CND^i (cf. chapitre 3). Nous étendons alors ce $\lambda\mu$-calcul (resp. ce $\lambda_{\mathbf{ct}}$-calcul) à l'interprétation calculatoire de la soustraction, dans les cadres classique puis intuitionniste. La soustraction apparaît alors comme un candidat naturel pour typer une instruction proche du **try with** du langage ML.

1 Introduction

Nous introduisons tout d'abord les opérateurs de contrôle à partir de la machine à environnement de J.-L. KRIVINE [32]. Pour cette machine, le $\lambda_{\mathbf{ct}}$-calcul étendu par les opérateurs de contrôle **catch** et **throw** et le $\lambda\mu$-calcul sont clairement équivalents. Nous étendons cette équivalence des machines en des règles de calcul pour les opérateurs **catch** et **throw** (le calcul obtenu est appelé $\lambda_{\mathbf{ct}}$-calcul). Nous dérivons alors la confluence du $\lambda_{\mathbf{ct}}$-calcul (alors que le $\lambda_{\mathbf{c/t}}$-calcul de H. NAKANO [44] n'est pas confluent) à partir de la confluence de $\lambda\mu$-calcul. Nous donnerons pour cela deux traductions permettant de passer facilement d'un calcul à l'autre.

Nous définissons ensuite une notion de « variable locale piégée » par un **throw** et nous montrons que les $\lambda_{\mathbf{ct}}$-termes qui ne « piègent » pas de variables, lorsqu'ils sont typables, sont le squelette d'une preuve (intuitionniste) de CND^i. On retrouve alors, dans le cadre typé, une contrainte semblable à celle de H. NAKANO [45] (reprise par M. SATO [59]).

Nous terminons ce chapitre en proposant un « sens calculatoire » pour la soustraction. Dans le cadre classique tout d'abord, où la soustraction est définissable par $A - B \equiv A \wedge \neg B$, nous obtenons l'interprétation prévisible de ce connecteur comme constructeur couplant une valeur et une continuation. Les instructions dérivées correspondant aux règles d'introduction et d'élimination de ce connecteur peuvent être interprétées comme un mécanisme de traitement des exceptions de haut niveau semblable au **try with** du langage ML. À nouveau, il est possible de donner une restriction intuitionniste de ces règles, qui s'exprime aussi comme l'interdiction de capturer des variables locales, mais cette fois ci lors du traitement des exceptions.

2 Préliminaires

L'origine des opérateurs de contrôle remonte aux machines à environnement utilisées pour l'implantation des langages fonctionnels [58]. Alors que ces machines effectuent généralement une réduction « par valeur » (*i.e.* elles évaluent d'abord l'argument d'une application), nous présentons ici la « machine de Krivine », qui effectue une réduction « par nom » (*i.e.* elle réduit toujours le radical de tête du λ-terme). Ces machines ont toutefois un point commun : elles effectuent une réduction faible (*i.e.* elles n'évaluent pas sous une abstraction).

2.1 La machine de Krivine

Comme dans toute machine à environnement, la structure de base manipulée est la clôture. Les types de structures utilisés dans la machine de Krivine sont les suivants :

- *clôture = couple ⟨terme, environnement⟩*
- *environnement = liste de couples (variable, clôture)*
- *pile = liste de clôtures*

Notation. On note la liste vide « () » et le constructeur qui ajoute un élément en tête de liste « : : ». Les chevrons « $\langle \dots \rangle$ » seront réservés aux couples désignant une clôture.

La machine de Krivine peut alors être présentée comme un système de réécriture typé, à la manière de [32]. Chaque membre de chaque règle est un couple formé d'une clôture et d'une pile. On note $E(x)$ la clôture associée à une variable x définie dans l'environnement E, les variables c et P désignent respectivement une clôture et une pile :

- $(\langle x, E\rangle, P) \rightsquigarrow (E(x), P)$
- $(\langle (u\ v), E\rangle, P) \rightsquigarrow (\langle u, E\rangle, \langle v, E\rangle :: P)$
- $(\langle \lambda x.t, E\rangle, c :: P) \rightsquigarrow (\langle t, (x, c) :: E\rangle, P)$

Remarques

- Toute clôture $\langle t, E\rangle$ représente un unique λ-terme : il suffit d'itérer le processus de substitution, notons ce terme $E^*(t)$. Par conséquent, à tout moment, un état de la machine $(\langle t, E\rangle, (\langle t_1, E_1\rangle, \dots, \langle t_n, E_n\rangle))$ représente l'unique λ-terme $(E^*(t)\ E_1^*(t_1) \dots E_n^*(t_n))$ avec le parenthèsage habituel de l'application $(t\ t_1 \dots t_n) = (\dots(t\ t_1)\dots t_n)$.
- Pour faire exécuter par cette machine un terme clos t, il suffit de commencer les étapes de réductions sur le couple $(\langle t, ()\rangle, ())$. Puisque le terme de départ est clos, la recherche d'une variable dans un environnement n'échoue jamais. À tout moment, l'état de la machine représente un unique terme.

– Ce système de réécriture est déterministe, puisqu'il est défini par filtrage sur le λ-terme. Le seul cas où aucune règle n'est applicable apparaît lorsque le λ-terme est une abstraction et la pile est vide : la machine s'arrête alors, et le résultat de l'exécution est une clôture $\langle t, E\rangle$ qui dénote le λ-terme $E^*(t)$.

– La machine ne s'arrête néanmoins pas necéssairement : puisque elle exécute des λ-termes purs, il suffit d'exécuter le point fixe bien connu $\delta\delta$ où $\delta = \lambda x.(x\ x)$.

– Comme il est usuel dans les machines à environnement, cette machine effectue une réduction faible (*i.e.* aucune réduction n'est effectuée sous une abstraction). Il n'y a par conséquent pas de risque de capture de variable, il n'est donc pas nécessaire de supposer les noms de variables liées distincts : il suffit que $E(x)$ soit par définition la clôture associée à la première occurence (*i.e.* la dernière ajoutée) de x dans E.

2.2 Machine de Krivine et opérateurs de contrôle

Le principe des opérateurs de contrôle consiste à accéder, au cours du calcul, à la pile. Cette pile, qui est aussi appelée « contexte d'évaluation », représente la « continuation » du calcul. Dans notre cas, en raison de la réduction de tête, cette continuation est juste une série d'applications. En effet, lorsque l'état de la pile est $(\langle t, E\rangle, (\langle t_1, E_n\rangle, \dots, \langle t_n, E_n\rangle))$, on est en train d'évaluer le terme $E^*(t)$ et la continuation (*i.e.* la suite du calcul à effectuer lorsque cette évaluation de $E^*(t)$ sera terminée) consistera à appliquer le résultat à $E_1^*(t_1)\dots E_n^*(t_n)$.

⋆ L'opérateur $\mathcal{C}$

L'idée de l'opérateur $\mathcal{C}$ de Krivine (à l'origine dû à M. Felleisen [16] dont l'opérateur $\mathcal{C}$ est une variante théorique du **call/cc** de Scheme, qui lui-même est issu de l'opérateur J de J. Reynolds, cf. [52] pour un historique plus détaillé) est basée sur la remarque suivante : la continuation est une fonction représentable par un λ-terme.

Plus précisément, une pile (*i.e.* un contexte) $(\langle t_1, E_1\rangle, \dots, \langle t_n, E_n\rangle)$ peut être représentée par le λ-terme : $\lambda x.(x\ E_1^*(t_1)\dots E_n^*(t_n))$. En effet, faire exécuter par la machine de Krivine $\langle(\lambda x.(x\ E_1^*(t_1)\dots E_n^*(t_n))\ t), ()\rangle, ())$, revient bien à faire exécuter $(\langle t, ()\rangle\ (\langle t_1, E_1\rangle, \dots, \langle t_n, E_n\rangle))$ (cf. les remarques de la sous-section 2.1).

Il est donc possible d'ajouter un opérateur qui donne au programmeur l'accès à la continuation courante sous la forme d'une fonction : c'est l'opérateur $\mathcal{C}$. Remarquons enfin que la conversion de la pile en λ-terme n'a pas besoin d'être réellement effectuée : il suffit que le comportement soit le même à l'exécution. L'intérêt de cette approche (qui est d'ailleurs sa raison historique cf. [52]) est de permettre d'utilisation des variables du λ-calcul pour manipuler les continuations.

Le λ-calcul est donc simplement étendu par un nouveau constructeur unaire $\mathcal{C}$. Les types de structures utilisés dans la définition de la machine sont légèrement modifiés (les trois types sont définis par récurrence simultanée, le symbole | signifie « ou ») :

– *clôture = couple ⟨terme, environnement⟩ | pile*

– *environnement = liste de couples (variable, clôture)*

– *pile = liste de clôtures*

Les règles de réduction sont étendues pour traiter les deux nouveaux cas possibles, le cas où le $\lambda\mathcal{C}$-terme de la clôture est l'opérateur de contrôle et le cas où la clôture est une pile (le symbole @@ dénote la concaténation des listes) :

– $(\langle Ct, E\rangle, P) \rightsquigarrow (\langle t, E\rangle, P :: ())$

– $(P, c :: P') \rightsquigarrow (c, P@@P')$

Remarques

– La première règle capture la continuation courante (*i.e.* la pile) et la remplace par une nouvelle pile, ne contenant qu'un unique élément : la pile capturée. Elle représente alors la fonction-continuation décrite ci-dessus, simplement la conversion en λ-terme n'a pas été effectuée. La deuxième règle reproduit (modulo les substitutions) ce qu'aurait donné la réduction de la fonction-continuation si elle avait été convertie en λ-terme. Remarquons que cette pile a nécéssairement transité par les variables λ-calcul pour passer de la position d'élément de la pile à la position d'argument de la seconde règle.

– Cet opérateur C permet de définir un autre opérateur unaire $\mathcal{A}$ (pour « abort ») qui oublie la continuation courante, ce qui a pour effet de terminer le calcul sur l'évaluation de son argument dans le contexte vide. On pose :

$$\mathcal{A}(t) \equiv C\lambda k.t \qquad \textit{où } k \notin FV(t)$$

L'exécution de $\mathcal{A}(t)$ transforme la pile courante en l'unique élément de la nouvelle pile, qui est ensuite dépilé pour être ajouté, sous le nom k à l'environnement de t, où elle ne servira plus puisque k n'apparaît pas dans t.

– Les règles ci-dessus se traduisent directement en une règle de réduction de tête du $\lambda\mathcal{C}$-calcul de J.-L Krivine [30] :

$$(Ct\ t_1 \dots t_n) \rightsquigarrow (t\ \lambda x.(x\ t_1 \dots t_n))$$

La continuation courante « appliquer à $t_1, \dots, t_n$ » est transformée en un λ-terme $\lambda x.(x\, t_1 \dots t_n)$. La nouvelle continuation courante devient alors uniquement « appliquer à $\lambda x.(x\, t_1 \dots t_n)$ ». Néanmoins, on trouve plutôt dans la littérature l'opérateur $\mathcal{C}$ original de M. Felleisen [16] (l'opérateur $\mathcal{C}$ décrit précédemment figure alors sous l'appellation $\mathcal{F}$) dont la règle de réduction de tête est :

$$(Ct\ t_1 \dots t_n) \rightsquigarrow (t\ \lambda x.\mathcal{A}(x\ t_1 \dots t_n))$$

Cette règle s'exprime alors dans la machine de Krivine par la réduction suivante, qui se substitue à la seconde donnée précédemment :

$$(P, c :: P') \rightsquigarrow (c, P)$$

Autrement dit, l'évaluation d'une continuation capturée détruit la continuation courante.

– L'opérateur $\mathcal{K}$ (qui est le **call/cc** de Scheme) a le comportement décrit par la règle ci-dessus, mais à la différence de l'opérateur $\mathcal{C}$ de J.-L Krivine ou de M. Felleisen, lors de la capture de la continuation (première règle), la continuation courante est conservée :

$$(\langle \mathcal{K}t, E\rangle, P) \rightsquigarrow (\langle t, E\rangle, P :: P)$$

L'opérateur $\mathcal{K}$ (dont on supposera l'argument η-expansé) peut donc être défini à partir de l'opérateur $\mathcal{C}$ de M. Felleisen par $\mathcal{K}\lambda k.t = \mathcal{C}\lambda k.(k\ t)$.

Exemple. Nous reprenons ici un exemple classique de la littérature (cf. [16] par exemple) : il s'agit de calculer le produit des entiers qui décorent les nœuds d'un arbre binaire (où, par définition récursive, un arbre est soit vide, soit composé d'un entier et deux sous-arbres). Naturellement, on souhaite arrêter le calcul dès qu'un 0 est rencontré. On supposera définies les primitives **vide**? qui détermine si un arbre est vide, **num**, **fg** et **fd** qui extraient d'un arbre respectivement l'entier qui décore sa racine, son sous-arbre gauche et son sous-arbre droit. Nous utiliserons la syntaxe **where rec** du langage Caml pour la définition d'une fonction locale par récurrence. Il est connu que toutes ces primitives (y compris le point fixe permettant la définition par récurrence) peuvent être exprimée par des λ-termes purs.

$$\begin{array}{ll}
\mathbf{let}\ f = & \lambda t.\mathbf{call/cc}\ \lambda k.(f'\ t) \\
& \mathbf{where\ rec}\ f' = \\
& \quad\quad \lambda t'.(\mathbf{if}\ (\mathbf{vide}?\ t') \\
& \quad\quad\quad\quad 1 \\
& \quad\quad\quad\quad (\mathbf{if}\ (\mathbf{num}\ t' = 0) \\
& \quad\quad\quad\quad\quad\quad (k\ 0) \\
& \quad\quad\quad\quad\quad\quad (\mathbf{num}\ t') * (f'\ (\mathbf{fg}\ t')) * (f'\ (\mathbf{fd}\ t'))))
\end{array}$$

L'interprétation de ce programme est la suivante : dès que la fonction f reçoit un arbre en argument, elle sauvegarde la continuation courante sous le nom k, puis elle commence l'évaluation récursive du produit des étiquettes. Si l'entier 0 est rencontré, $(k\ 0)$ est exécuté, ce qui revient à abandonner tous les calculs en cours puis à restaurer la continuation sauvegardée sous le nom k, c'est-à-dire celle présente à l'appel de la fonction et renvoie donc 0.

⋆ Les opérateurs « catch » et « throw »

Ces opérateurs de contrôle sont antérieurs à ceux décrits précédemment. Nous utiliserons la terminologie **catch** et **throw** issue du langage Lisp, où les liaisons de variables sont dynamiques, dans un cadre où les *liaisons sont toutes statiques* (pour suivre [44, 45, 59]) bien que des variantes statiques de ces opérateurs appellées **block** et **return-from** existent (à nouveau nous renvoyons à [52]).

À la différence des opérateurs décrits précédemment nous utiliserons ici des « espaces de noms » distincts pour les variables habituelles du λ-calcul et les variables dénotant des continuations (que nous appellerons désormais aussi variables d'exception). Ceci va se traduire par deux types d'environnements : les environnements de clôtures et les environnements de piles. Nous récapitulons ici les types de structures utilisés :

– *clôture = couple ⟨terme, (env-clôture, env-piles)⟩*

– *env-clôtures = liste de couples (λ-variable, clôture)*

– *env-piles = liste de couples (exc-variable, pile)*

– *pile = liste de clôtures*

Le langage du λ-calcul est étendu par deux nouveaux constructeurs, notés **catch** et **throw**, qui prennent tous deux comme arguments une variable d'exception α et un terme t (on notera **catch** α t et **throw** α t). La variable d'exception argument d'un **catch** *est liée par ce* **catch**, la variable d'exception argument d'un **throw** *doit toujours être liée* par un **catch** englobant. Les variables d'exception seront notées $\alpha, \beta, \gamma, \ldots$

Les règles de la machine sont les suivantes, les variables EC et EP et E désignent respectivement un environnement de clôtures, un environnement de piles, et un environnement complet :

1. $(\langle x, (EC, EP)\rangle, P) \rightsquigarrow (EC(x), P)$
2. $(\langle (u\ v), E\rangle, P) \rightsquigarrow (\langle u, E\rangle, \langle v, E\rangle :: P)$
3. $(\langle \lambda x.t, (EC, EP)\rangle, c :: P) \rightsquigarrow (\langle t, ((x, c) :: EC, EP)\rangle, P)$
4. $(\langle \mathbf{catch}\ \alpha\ t, (EC, EP)\rangle, P) \rightsquigarrow (\langle t, (EC, (\alpha, P) :: EP\rangle, P)$
5. $(\langle \mathbf{throw}\ \alpha\ t, (EC, EP)\rangle, P) \rightsquigarrow (\langle t, (EC, EP)\rangle, EP(\alpha))$

Remarque. De même que pour les λ-variables, si le terme est clos pour les variables d'exception, la recherche $EP(\alpha)$ dans la règle du **throw** n'échoue jamais. Le sens des primitives apparaît clairement dans les règles de la machine : l'instruction **catch** α t « sauvegarde » le contexte courant sous le nom α puis exécute t, l'intruction **throw** α t « oublie » la continuation courante et « restaure » le contexte sauvegardé précédemment sous le nom α puis exécute t. Remarquons que l'on ne sauvegarde et restaure que le contexte et non l'environnement. Le sens intuitif est sans doute plus clair que celui des opérateurs de contrôle vus jusqu'ici : l'instruction **catch** permet de marquer une position dans le programme, position à laquelle on pourra revenir directement grâce au **throw**. On peut donc voir l'opérateur **throw** comme une version de haut niveau du « goto » de certains langages impératifs.

Exemple. Voici la traduction de l'exemple précédent qui utilise les opérateurs **catch** et **throw**.

$$
\begin{array}{ll}
\mathbf{let}\ f = & \lambda t.\mathbf{catch}\ \alpha\ (f'\ t) \\
& \quad \mathbf{where\ rec}\ f' = \\
& \qquad \lambda t'.(\mathbf{if}\ (\mathbf{vide}?\ t') \\
& \qquad\qquad 1 \\
& \qquad\qquad (\mathbf{if}\ (\mathbf{num}\ t' = 0) \\
& \qquad\qquad\quad (\mathbf{throw}\ \alpha\ 0) \\
& \qquad\qquad\quad (\mathbf{num}\ t') * (f'\ (\mathbf{fg}\ t')) * (f'\ (\mathbf{fd}\ t'))))
\end{array}
$$

L'appel récursif dans la définition de f' a clairement une action sur la pile (par la règle 2). Le **throw** permet d'oublier tout ce qui a été empilé (*i.e.* ce qui reste à faire).

Remarque. La traduction des opérateurs **catch** et **throw** vers le **call/cc** ne pose pas de difficulté : il suffit de remplacer toute occurrence de **catch** α t par **call/cc** $\lambda k_\alpha.t'$ et toute occurrence **throw** α t simplement par $(k_\alpha\ t')$, où t' est obtenu en appliquant récursivement la traduction. Réciproquement, pour traduire le **call/cc** à partir des opérateurs **catch** et **throw**, définissons l'opérateur K suivant :

$$Ky \equiv \textbf{catch}\ \alpha\ (y\ (\lambda x.\textbf{throw}\ \alpha\ x))$$

Considérons maintenant un terme de la forme $K\lambda k.t \equiv \textbf{catch}\ \alpha\ (\lambda k.t\ (\lambda x.\textbf{throw}\ \alpha\ x))$. Ce terme se comporte de la façon suivante : il sauvegarde tout d'abord la continuation courante sous le nom α, puis il associe à k le terme $\lambda x.\textbf{throw}\ \alpha\ x$, et enfin il évalue t. Dans l'évaluation de t, si l'on rencontre un sous-terme de la forme $(k\ u)$, on exécute alors **throw** $\alpha\ u$, ce qui revient à oublier la continuation courant et à restaurer la continuation sauvegardée lors de l'exécution de K pour évaluer u. Le comportement de l'opérateur K est donc exactement celui du **call/cc**.

$\star$ Le $\lambda\mu$-calcul

Les structures définies pour la « machine **catch** et **throw** » permettent aussi d'exécuter des $\lambda\mu$-termes [47]. Nous rappelons que ces termes sont (comme pour les opérateurs **catch** et **throw**) construits à partir de nouveaux noms de variables $\alpha, \beta, \gamma, \ldots$ (ici appelées μ-variables), d'un constructeur μ qui lie une μ-variable dans un terme, et d'un constructeur [ ] qui « nomme » un terme par une μ-variable, et qui suit toujours directement un μ : ces constructeurs respectent toujours la forme $\mu\alpha[\beta]t$. La règle de calcul associée est la suivante :

– $(\langle \mu\alpha[\beta]t, (EC, EP)\rangle, P) \rightsquigarrow (\langle t, (EC, (\alpha, P) :: EP)\rangle, ((\alpha, P) :: EP)(\beta))$

Le sens de l'instruction $\mu\alpha[\beta]t$ est bien celui présenté dans [47] : sauvegarder le contexte courant sous le nom α puis restaurer le contexte de nom β et enfin exécuter t. Remarquons que l'on prend soin de sauvegarder le contexte courant *avant* de restaurer le contexte β (c'est pourquoi on écrit $((\alpha, P) :: EP)(\beta)$ et non simplement $EP(\beta)$), ceci bien sûr pour traiter le cas où $\alpha = \beta$.

Remarque. Dans la machine définie par la structure ci-dessus, les opérateurs du $\lambda\mu$-calcul peuvent simuler les opérateurs **catch** et **throw** et réciproquement.

– Il est clair que l'opérateur $\mu\alpha[\beta]$ permet (dans cette machine) de simuler les opérateurs **catch** et **throw** : l'instruction $\mu\alpha[\alpha]t$ sauvegarde la pile courante sous le nom α, puis restaure *cette même pile*, avant d'exécuter t. Finalement, elle a fait exactement ce que fait l'instruction **catch** $\alpha\ t$. Maintenant, considérons l'instruction $\mu\alpha[\beta]t$ où α n'apparaît pas dans t, cette instruction sauvegarde la pile courante sous le nom α, puis restaure la pile de nom β avant d'exécuter t. Puisque α n'apparaît dans t, on est certain de ne jamais utiliser cette pile sauvegardée : on aurait pu l'oublier. C'est exactement ce que fait l'instruction **throw** $\beta\ t$.

– Réciproquement, considérons l'instruction **catch** α **throw** $\beta\ t$ de la « machine **catch** et **throw** », cette instruction sauvegarde la pile courante sous le nom α, puis restaure la pile de nom β avant d'exécuter t : c'est exactement ce que fait l'instruction $\mu\alpha[\beta]t$.

3 Opérateurs de « catch » et « throw » et $\lambda\mu$-calcul

Dans cette section, nous utilisons les simulations des opérateurs **catch** et **throw** en $\lambda\mu$-calcul décrites ci-dessus pour définir un système de règles de réduction confluentes pour ces opérateurs : nous appellerons $\lambda_{\mathbf{ct}}$-calcul le calcul ainsi obtenu. Nous rappelons tout d'abord les règles de réduction du $\lambda\mu$-calcul qui définissent un système confluent [47].

Règles de réduction du $\lambda\mu$-calcul

- $(\lambda x.u\ v) \rightsquigarrow u\{v/x\}$
- $(\mu\alpha.u\ v) \rightsquigarrow \mu\alpha.u\{[\alpha](t\ v)/[\alpha]t\}$
- $[\beta]\mu\alpha.t \rightsquigarrow t[\beta/\alpha]$
- $\mu\alpha[\alpha]t \rightsquigarrow t$ si α n'apparaît pas libre dans t.

La notation $u\{v/x\}$ désigne la substitution habituelle (sans capture de variable, donc avec renommages éventuels des variables liées) de la variable x par v dans u. La notation $u\{[\alpha](t\ v)/[\alpha]t\}$ signifie « remplacer dans le terme u toutes les occurrences de sous-termes de la forme $[\alpha]t$ par $[\alpha](t\ v)$ », à nouveau, ce remplacement s'effectue sans capture de variable.

3.1 Règles dérivées

Nous avons vu (cf. sous-section 2.1) que les opérateurs **catch** $\alpha\ t$ et **throw** $\alpha\ t$ peuvent se définir respectivement par $\mu\alpha[\alpha]t$ et $\mu\beta[\alpha]t$ où β est une μ-variable différente de α et qui n'apparaît pas dans t (nous noterons souvent « _ » une telle variable). Il est alors naturel de considérer le sous-langage des $\lambda\mu$-termes ne contenant que les « macros » définissant **catch** et **throw** (autrement dit, où toute occurence d'un sous-terme $\mu\alpha[\beta]t$ est soit de la forme $\mu\alpha[\alpha]t$ soit de la forme $\mu\_[\beta]t$). Nous appellerons $\lambda\mu\mathbf{ct}$-termes les $\lambda\mu$-termes de ce sous-langage. Malheureusement l'ensemble $\lambda\mu\mathbf{ct}$-termes n'est pas stable par réduction, en raison de la règle suivante :

$$\mathbf{catch}\ \alpha\ \mathbf{throw}\ \beta\ t = \mu\alpha[\alpha]\mu\_[\beta]t \rightsquigarrow \mu\alpha[\beta]t\{\alpha/\_\} = \mu\alpha[\beta]t$$

On peut néanmoins montrer que les instances de règles qui ne font pas sortir du sous-langage des $\lambda\mu\mathbf{ct}$-termes définissent un calcul confluent. On obtient ces règles en énumérant toutes les instances de radicaux de $\lambda\mu\mathbf{ct}$-termes.

- L'ensemble des $\lambda\mu\mathbf{ct}$-termes est clairement clos par substitution. On obtient donc la β-réduction comme règle dérivée (puisque le terme réduit est toujours un $\lambda\mu\mathbf{ct}$-terme) :

$$(\lambda x.u\ v) \rightsquigarrow u\{v/x\}$$

- Tout radical de la forme $\mu\alpha[\alpha]t$ dans un $\lambda\mu\mathbf{ct}$-terme est de la forme **catch** $\alpha\ t$. La règle $\mu\alpha[\alpha]t \rightsquigarrow t$, si α n'apparaît pas libre dans t, nous donne donc une unique règle dérivée (puisque le terme réduit est toujours un $\lambda\mu\mathbf{ct}$-terme) :

$$\mathbf{catch}\ \alpha\ t = \mu\alpha[\alpha]t \rightsquigarrow t$$

– Tout radical de la forme $(\mu\alpha.w\ v)$ (*i.e.* de la forme $(\mu\alpha[\beta]u\ v)$) dans un $\lambda\mu$**ct**-terme est soit de la forme (**catch** $\alpha\ u)\ v$ soit de la forme (**throw** $\alpha\ u)\ v$. La règle $(\mu\alpha.w\ v) \rightsquigarrow \mu\alpha.u\{[\alpha](t\ \ v)/[\alpha]t\}$ nous donne deux règles dérivées (puisque les termes réduits sont toujours des $\lambda\mu$**ct**-termes) :

$$\begin{aligned}((\mathbf{catch}\ \alpha\ u)\ v) = (\mu\alpha[\alpha].u\ v) &\rightsquigarrow \mu\alpha([\alpha].u)\{\mu_\_[\alpha](t\ v)/\mu_\_[\alpha]t\} \\ &= \mathbf{catch}\ \alpha\ (u\{\mathbf{throw}\ \alpha\ (t\ v)/\mathbf{throw}\ \alpha\ t\}\ v) \\ ((\mathbf{throw}\ \alpha\ u)\ v) = (\mu\delta[\alpha].u\ v) &\rightsquigarrow \mu\delta([\alpha].u)\{[\delta](t\ v)/[\delta]t\} = \mathbf{throw}\ \alpha\ u\end{aligned}$$

puisque, par définition de **throw**, la variable δ est différente de α et n'apparaît pas libre dans u.

– Tout radical de la forme $[\alpha]\mu\beta.w$ (*i.e.* de la forme $\mu\gamma[\alpha]\mu\beta[\delta]t$) dans un $\lambda\mu$**ct**-terme est d'une des quatre formes suivantes : **catch** α **catch** $\beta\ t$, **throw** α **throw** $\beta\ t$, **throw** α **catch** $\beta\ t$, **catch** α **throw** $\beta\ t$. La règle $[\alpha]\mu\beta.w \rightsquigarrow w\{\alpha/\beta\}$ nous donne quatre cas :

$$\begin{aligned}\mathbf{catch}\ \alpha\ \mathbf{catch}\ \beta\ t &= \mu\alpha[\alpha]\mu\beta[\beta]t \rightsquigarrow \mu\alpha[\alpha]t\{\alpha/\beta\} = \mathbf{catch}\ \alpha\ t\{\alpha/\beta\} \\ \mathbf{throw}\ \alpha\ \mathbf{throw}\ \beta\ t &= \mu_\_[\alpha]\mu_\_[\beta]t \rightsquigarrow \mu_\_[\beta]t\{\alpha/\_\} = \mathbf{throw}\ \beta\ t \\ \mathbf{throw}\ \alpha\ \mathbf{catch}\ \beta\ t &= \mu_\_[\alpha]\mu\beta[\beta]t \rightsquigarrow \mu_\_[\alpha]t\{\alpha/\beta\} = \mathbf{throw}\ \alpha\ t\{\alpha/\beta\} \\ \mathbf{catch}\ \alpha\ \mathbf{throw}\ \beta\ t &= \mu\alpha[\alpha]\mu_\_[\beta]t \rightsquigarrow \mu\alpha[\beta]t\{\alpha/\_\} = \mu\alpha[\beta]t\end{aligned}$$

Les trois premiers cas nous donnent bien des règles dérivées, mais comme nous l'avons vu, la dernière règle nous fait sortir du sous-langage des $\lambda\mu$**ct**-termes sauf dans le cas particulier $\alpha = \beta$ où l'on obtient :

$$\mathbf{catch}\ \alpha\ \mathbf{throw}\ \alpha\ t = \mu\alpha[\alpha]\mu_\_[\alpha]t \rightsquigarrow \mu\alpha[\alpha]t = \mathbf{catch}\ \alpha\ t$$

Le cas où α n'apparaît pas dans t nous donne une instance particulière de la règle **catch** $\alpha\ t \rightsquigarrow t$ que nous avons déjà obtenue :

$$\mathbf{catch}\ \alpha\ \mathbf{throw}\ \beta\ t = \mu\alpha[\alpha]\mu_\_[\beta]t \rightsquigarrow \mu\alpha[\beta]t = \mathbf{throw}\ \beta\ t$$

Par contre, la règle **catch** α **throw** $\beta\ t \rightsquigarrow \mu\alpha[\beta]t$ montre que les deux « macros » **catch** et **throw** suffisent à exprimer tous les $\lambda\mu$-termes (on retrouve la traduction réciproque donnée dans la cadre de la machine à environnement). D'autre part, nous allons voir que les règles dérivées définissent déjà un calcul confluent. Nous les récapitulons ici :

Définition 3.1.1 *Nous appellerons $\lambda_{\mathbf{ct}}$-calcul le λ-calcul muni des opérateurs* **catch** *et* **throw** *et défini par la β-réduction plus les 7 règles ci-dessous :*

1. **catch** $\alpha\ t \rightsquigarrow t$ si α n'apparaît pas libre dans t.
2. $((\mathbf{catch}\ \alpha\ u)\ v) \rightsquigarrow \mathbf{catch}\ \alpha\ (u\{\mathbf{throw}\ \alpha\ (t\ v)/\mathbf{throw}\ \alpha\ t\}\ v)$
3. $((\mathbf{throw}\ \alpha\ u)\ v) \rightsquigarrow \mathbf{throw}\ \alpha\ u$
4. **catch** α **catch** $\beta\ t \rightsquigarrow$ **catch** $\alpha\ t\{\alpha/\beta\}$
5. **throw** α **throw** $\beta\ t \rightsquigarrow$ **throw** $\beta\ t$
6. **throw** α **catch** $\beta\ t \rightsquigarrow$ **throw** $\alpha\ t\{\alpha/\beta\}$

7. **catch** α **throw** α $t \leadsto$ **catch** α t

Remarque. On continuera à appeler les variables d'exception $\alpha, \beta, \gamma, \ldots$ « μ-variables ». D'autre part, en $\lambda_{\mathbf{ct}}$-calcul, comme en $\lambda\mu$-calcul, les termes sont toujours considérés à renommage des λ-variables et μ-variables liées près.

3.2 $\lambda_{\mathbf{ct}}$-calcul et $\lambda\mu$-calcul

La construction du $\lambda_{\mathbf{ct}}$-calcul laisse espérer l'existence de deux traductions particulièrement « simples », que nous appellerons Φ et Ψ, permettant respectivement de traduire le $\lambda\mu$-calcul en $\lambda_{\mathbf{ct}}$-calcul et réciproquement, qui soient des morphismes pour la réduction, c'est-à-dire que pour tout $\lambda\mu$-terme t, si $t \leadsto^{\star}_{\lambda\mu} t'$ alors $\Phi(t) \leadsto^{\star}_{\mathbf{ct}} \Phi(t')$ et réciproquement, pour tout $\lambda_{\mathbf{ct}}$-terme, si $t \leadsto^{\star}_{\mathbf{ct}} t'$ alors $\Psi(t) \leadsto^{\star}_{\lambda\mu} \Psi(t')$. Ce sont ces traductions que nous définissons ici.

Remarque. Nous aurons besoin de la notion (usuelle en λ-calcul) de contexte (*i.e.* de terme à « trou »), dans le cadre du $\lambda\mu$-calcul et du $\lambda_{\mathbf{ct}}$-calcul. Ce sont des termes contenant une *unique* occurrence d'un symbole particulier noté « $\bullet$ » (qui est traité comme un symbole de constante dans la définition récursive des termes). Étant donné un contexte, noté $C\{\bullet\}$, la notation $C\{t\}$, désigne alors le terme obtenu en substituant sans renommage (*i.e.* avec capture éventuelle de variables) le symbole $\bullet$ par le terme t dans $C\{\bullet\}$. On obtient bien sûr ainsi à nouveau un $\lambda\mu$-terme (resp. de $\lambda_{\mathbf{ct}}$-terme).

Traduction du $\lambda\mu$-calcul vers le $\lambda_{\mathbf{ct}}$-calcul : remplacer toute occurrence de $\mu\alpha[\alpha]$ par **catch** α, toute occurrence de $\mu\_[\alpha]$ par **throw** α, puis finalement toute occurrence de $\mu\alpha[\beta]$ par **catch** α **throw** β. En voici une définition formelle :

Définition 3.2.1 *On définit la traduction Φ du $\lambda\mu$-calcul vers le $\lambda_{\mathbf{ct}}$-calcul par récurrence sur les $\lambda\mu$-termes :*

– $\Phi(x) = x$, *si x est une λ-variable,*

– $\Phi((u\ v)) = (\Phi(u)\ \Phi(v))$

– $\Phi(\lambda x.t) = \lambda x.\Phi(t)$

– $\Phi(\mu\alpha[\beta]t) = \begin{cases} \textbf{catch}\ \alpha\ \Phi(t) & \textit{si } \alpha = \beta \\ \textbf{throw}\ \beta\ \Phi(t) & \textit{si } \alpha \neq \beta \textit{ et } \alpha \textit{ n'est pas libre dans } t \\ \textbf{catch}\ \alpha\ \textbf{throw}\ \beta\ \Phi(t) & \textit{sinon} \end{cases}$

Remarque. Les λ-variables et μ-variables libres sont les mêmes dans t et dans $\Phi(t)$.

Nous aurons besoin du lemme de substitution suivant :

Lemme 3.2.2 *Pour tout $\lambda\mu$-terme u, v et toute variable x libre dans u :*

$$\Phi(u[v/x]) = \Phi(u)\{\Phi(v)/x\}$$

Preuve. Par récurrence sur le terme u. □

Lemme 3.2.3 *Pour toute règle $u \leadsto v$ du $\lambda\mu$-calcul on a $\Phi(u) \leadsto^{\star}_{\mathbf{ct}} \Phi(v)$.*

Preuve. Considérons chaque règle du $\lambda\mu$-calcul :

- Cas de la β-réduction $(\lambda x.u\ v) \leadsto_\beta u\{v/x\}$:

$$\Phi((\lambda x.u\ v)) = (\lambda x\Phi(u)\ \Phi(v)) \leadsto_\beta \Phi(u)\{\Phi(v)/x\} = \Phi(u\{v/x\})$$

  par le lemme de substitution.

- Cas de la règle $(\mu\alpha[\beta]u\ v) \leadsto \mu\alpha([\beta]u)\{[\alpha](t\ v)/[\alpha]t\}$

  1. Si $\alpha = \beta$ alors $\mu\alpha([\beta]u)\{[\alpha](t\ v)/[\alpha]t\}= \mu\alpha[\alpha](u\{[\alpha](t\ v)/[\alpha]t\}\ v)$,

$$\begin{array}{rcl} \Phi(\mu\alpha[\alpha]u\ v) & = & (\textbf{catch}\ \alpha\ \Phi(u)\ \Phi(v)) \\ & \leadsto_2 & \textbf{catch}\ \alpha\ (\Phi(u)\{\textbf{throw}\ \alpha\ (t\ \Phi(v))/\textbf{throw}\ \alpha\ t\}\ \Phi(v)) \\ & = & \Phi(\mu\alpha[\alpha](u\{[\alpha](t\ v)/[\alpha]t\}\ v)) \end{array}$$

     puisque toute occurrence d'un sous-terme $\mu\beta[\alpha]t$ (où nécéssairement $\beta \neq \alpha$) se traduit, par définition de Φ, soit en **catch** β **throw** $\alpha\ \Phi(t)$ soit en **throw** $\alpha\ \Phi(t)$.
  2. Si $\alpha \neq \beta$ et α n'apparaît pas dans $[\beta]u$, $\mu\alpha([\beta]u)\{[\alpha](t\ v)/[\alpha]t\}) = \mu\alpha[\beta]u$,

$$\begin{array}{rcl} \Phi(\mu\alpha[\beta]u\ v) & = & ((\textbf{throw}\ \alpha\ \Phi(u))\ \Phi(v)) \\ & \leadsto_3 & \textbf{throw}\ \alpha\ \Phi(u) \\ & = & \Phi(\mu\alpha[\beta]u) \end{array}$$

  3. Si $\alpha \neq \beta$ et α apparaît dans $[\beta]u$, alors $\mu\alpha([\beta]u)\{[\alpha](t\ v)/[\alpha]t\}= \mu\alpha[\beta]u\{[\alpha](t\ v)/[\alpha]t\}$,

$$\begin{array}{rcl} \Phi(\mu\alpha[\beta]u\ v) & = & ((\textbf{catch}\ \alpha\ \textbf{throw}\ \beta\ \Phi(u))\ \Phi(v)) \\ & \leadsto_2 & \textbf{catch}\ \alpha\ ((\textbf{throw}\ \beta\ \Phi(u))\{\textbf{throw}\ \alpha\ (t\ \Phi(v))/\textbf{throw}\ \alpha\ t\}\ \Phi(v)) \\ & = & \textbf{catch}\ \alpha\ (\textbf{throw}\ \beta\ \Phi(u)\{\textbf{throw}\ \alpha\ (t\ \Phi(v))/\textbf{throw}\ \alpha\ t\}\ \Phi(v)) \\ & \leadsto_3 & \textbf{catch}\ \alpha\ \textbf{throw}\ \beta\ \Phi(u)\{\textbf{throw}\ \alpha\ (t\ \Phi(v))/\textbf{throw}\ \alpha\} \\ & = & \Phi(\mu\alpha[\beta]u\{[\alpha](t\ v)/[\alpha]t\}) \end{array}$$

     à nouveau, puisque toute occurrence d'un sous-terme $\mu\beta[\alpha]t$ (où nécéssairement $\beta \neq \alpha$) se traduit, par définition de Φ, soit en **catch** β **throw** $\alpha\ \Phi(t)$ soit en **throw** $\alpha\ \Phi(t)$.

- Cas de la règle $[\beta]\mu\alpha.t \leadsto t\{\beta/\alpha\}$, c'est-à-dire $\mu\gamma[\beta]\mu\alpha[\delta]t \leadsto \mu\gamma([\delta]t)\{\beta/\alpha\}$

  1. Si $\gamma = \beta$ et $\alpha = \delta$ alors $\mu\gamma([\delta]t)\{\beta/\alpha\} = \mu\beta[\beta]t\{\beta/\alpha\}$

$$\begin{array}{rcl} \Phi(\mu\beta[\beta]\mu\alpha[\alpha]t) & = & \textbf{catch}\ \beta\ \textbf{catch}\ \alpha\ \Phi(t) \\ & \leadsto_4 & \textbf{catch}\ \beta\ \Phi(t)\{\beta/\alpha\} \\ & = & \Phi(\mu\beta[\beta]t\{\beta/\alpha\}) \end{array}$$

  2. Si $\gamma = \beta$ et α n'apparaît pas dans $[\delta]t$ alors $\mu\gamma([\delta]t)\{\beta/\alpha\} = \mu\beta[\delta]t$
     (a) Si $\beta = \delta$ alors $\mu\beta[\delta]t = \mu\beta[\beta]t$

$$\begin{array}{rcl} \Phi(\mu\beta[\beta]\mu\alpha[\beta]t) & = & \textbf{catch}\ \beta\ \textbf{throw}\ \beta\ \Phi(t) \\ & \leadsto_7 & \textbf{catch}\ \beta\ \Phi(t) \\ & = & \Phi(\mu\beta[\beta]t) \end{array}$$

(b) Si $\beta \neq \delta$ et β n'apparaît pas dans t

$$\begin{array}{rcl} \Phi(\mu\beta[\beta]\mu\alpha[\delta]t) & = & \textbf{catch } \beta \textbf{ throw } \delta\ \Phi(t) \\ & \rightsquigarrow_1 & \textbf{throw } \delta\ \Phi(t) \\ & = & \Phi(\mu\beta[\delta]t) \end{array}$$

(c) Si $\beta \neq \delta$ et β apparaît dans t

$$\begin{array}{rcl} \Phi(\mu\beta[\beta]\mu\alpha[\delta]t) & = & \textbf{catch } \beta \textbf{ throw } \delta\ \Phi(t) \\ & = & \Phi(\mu\beta[\delta]t) \end{array}$$

3. Si $\gamma = \beta$ et α apparaît dans $[\delta]t$ alors $\mu\gamma([\delta]t)\{\beta/\alpha\} = \mu\beta([\delta]t)\{\beta/\alpha\}$
   (a) Si $\alpha = \delta$, le cas a déjà été traité.
   (b) Si $\alpha \neq \delta$ alors $\mu\beta([\delta]t)\{\beta/\alpha\}= \mu\beta[\delta]t\{\beta/\alpha\}$
      – Si $\beta = \delta$ alors $\mu\beta[\delta]t\{\beta/\alpha\}=\mu\beta[\beta]t\{\beta/\alpha\}$

$$\begin{array}{rcl} \Phi(\mu\beta[\beta]\mu\alpha[\beta]t) & = & \textbf{catch } \beta \textbf{ catch } \alpha \textbf{ throw } \beta\ \Phi(t) \\ & \rightsquigarrow_4 & \textbf{catch } \beta\ (\textbf{throw } \beta\ \Phi(t))\{\beta/\alpha\} \\ & = & \textbf{catch } \beta \textbf{ throw } \beta\ \Phi(t)\{\beta/\alpha\} \\ & \rightsquigarrow_7 & \textbf{catch } \beta\ \Phi(t)\{\beta/\alpha\} \\ & = & \Phi(\mu\beta[\beta]t\{\beta/\alpha\}) \end{array}$$

      – Si $\beta \neq \delta$ alors

$$\begin{array}{rcl} \Phi(\mu\beta[\beta]\mu\alpha[\delta]t) & = & \textbf{catch } \beta \textbf{ catch } \alpha \textbf{ throw } \delta\ \Phi(t) \\ & \rightsquigarrow_4 & \textbf{catch } \beta\ (\textbf{throw } \delta\ \Phi(t))\{\beta/\alpha\} \\ & = & \textbf{catch } \beta \textbf{ throw } \delta\ \Phi(t)\{\beta/\alpha\} \\ & = & \Phi(\mu\beta[\delta]t\{\beta/\alpha\}) \end{array}$$

4. Si $\gamma \neq \beta$ et γ n'apparaît pas dans $\mu\alpha[\delta]t$ (alors en particulier $\gamma \neq \delta$)
   (a) Si $\alpha = \delta$ alors $\mu\gamma([\delta]t)\{\beta/\alpha\} = \mu\gamma[\beta]t\{\beta/\alpha\}$

$$\begin{array}{rcl} \Phi(\mu\gamma[\beta]\mu\alpha[\alpha]t) & = & \textbf{throw } \beta \textbf{ catch } \alpha\ \Phi(t) \\ & \rightsquigarrow_6 & \textbf{throw } \beta\ \Phi(t)\{\beta/\alpha\} \\ & = & \Phi(\mu\gamma[\beta]t\{\beta/\alpha\}) \end{array}$$

   car $\gamma \neq \beta$ pour la dernière équation.
   (b) Si $\alpha \neq \delta$ et α n'apparaît pas dans t alors $\mu\gamma([\delta]t)\{\beta/\alpha\} = \mu\gamma[\delta]t$

$$\begin{array}{rcl} \Phi(\mu\gamma[\beta]\mu\alpha[\delta]t) & = & \textbf{throw } \beta \textbf{ throw } \delta\ \Phi(t) \\ & \rightsquigarrow_5 & \textbf{throw } \delta\ \Phi(t) \\ & = & \Phi(\mu\gamma[\delta]t) \end{array}$$

   car $\gamma \neq \delta$ pour la dernière équation.
   (c) Si $\alpha \neq \delta$ et α apparaît dans t alors $\mu\gamma([\delta]t)\{\beta/\alpha\} = \mu\gamma[\delta]t\{\beta/\alpha\}$

$$\begin{array}{rcl} \Phi(\mu\gamma[\beta]\mu\alpha[\delta]t) & = & \textbf{throw } \beta \textbf{ catch } \alpha \textbf{ throw } \delta\ \Phi(t) \\ & \rightsquigarrow_6 & \textbf{throw } \beta\ (\textbf{throw } \delta\ \Phi(t))\{\beta/\alpha\} \\ & = & \textbf{throw } \beta \textbf{ throw } \delta\ \Phi(t)\{\beta/\alpha\} \\ & \rightsquigarrow_5 & \textbf{throw } \delta\ \Phi(t)\{\beta/\alpha\} \\ & = & \Phi(\mu\gamma[\delta]t\{\beta/\alpha\}) \end{array}$$

5. Si $\gamma \neq \beta$ et γ apparaît dans $\mu\alpha[\delta]t$
   (a) Si $\alpha = \delta$ alors $\mu\gamma([\delta]t)\{\beta/\alpha\} = \mu\gamma[\beta]t\{\beta/\alpha\}$

$$\begin{array}{rcl} \Phi(\mu\gamma[\beta]\mu\alpha[\alpha]t) & = & \mathbf{catch}\ \gamma\ \mathbf{throw}\ \beta\ \mathbf{catch}\ \alpha\ \Phi(t) \\ & \leadsto_6 & \mathbf{catch}\ \gamma\ \mathbf{throw}\ \beta\ \Phi(t)\{\beta/\alpha\} \\ & = & \Phi(\mu\gamma[\beta]t\{\beta/\alpha\}) \end{array}$$

   (b) Si $\alpha \neq \delta$ et α n'apparaît pas dans t alors $\mu\gamma([\delta]t)\{\beta/\alpha\} = \mu\gamma[\delta]t$
   - Si $\gamma = \delta$ alors $\mu\gamma[\delta]t = \mu\gamma[\gamma]t$

$$\begin{array}{rcl} \Phi(\mu\gamma[\beta]\mu\alpha[\gamma]t) & = & \mathbf{catch}\ \gamma\ \mathbf{throw}\ \beta\ \mathbf{throw}\ \gamma\ \Phi(t) \\ & \leadsto_5 & \mathbf{catch}\ \gamma\ \mathbf{throw}\ \gamma\ \Phi(t) \\ & \leadsto_7 & \mathbf{catch}\ \gamma\ \Phi(t) \\ & = & \Phi(\mu\gamma[\gamma]t) \end{array}$$

   - Si $\gamma \neq \delta$

$$\begin{array}{rcl} \Phi(\mu\gamma[\beta]\mu\alpha[\delta]t) & = & \mathbf{catch}\ \gamma\ \mathbf{throw}\ \beta\ \mathbf{throw}\ \delta\ \Phi(t) \\ & \leadsto_5 & \mathbf{catch}\ \gamma\ \mathbf{throw}\ \delta\ \Phi(t) \\ & = & \Phi(\mu\gamma[\delta]t) \end{array}$$

   (c) Si $\alpha \neq \delta$ et α apparaît dans t alors $\mu\gamma([\delta]t)\{\beta/\alpha\} = \mu\gamma[\delta]t\{\beta/\alpha\}$
   - Si $\gamma = \delta$ alors $\mu\gamma[\delta]t\{\beta/\alpha\} = \mu\gamma[\gamma]t\{\beta/\alpha\}$

$$\begin{array}{rcl} \Phi(\mu\gamma[\beta]\mu\alpha[\gamma]t) & = & \mathbf{catch}\ \gamma\ \mathbf{throw}\ \beta\ \mathbf{catch}\ \alpha\ \mathbf{throw}\ \gamma\ \Phi(t) \\ & \leadsto_6 & \mathbf{catch}\ \gamma\ \mathbf{throw}\ \beta\ (\mathbf{throw}\ \gamma\ \Phi(t))\{\beta/\alpha\} \\ & = & \mathbf{catch}\ \gamma\ \mathbf{throw}\ \beta\ \mathbf{throw}\ \gamma\ \Phi(t)\{\beta/\alpha\} \\ & \leadsto_5 & \mathbf{catch}\ \gamma\ \mathbf{throw}\ \gamma\ \Phi(t)\{\beta/\alpha\} \\ & \leadsto_7 & \mathbf{catch}\ \gamma\ \Phi(t)\{\beta/\alpha\} \\ & = & \Phi(\mu\gamma[\gamma]t\{\beta/\alpha\}) \end{array}$$

   - Si $\gamma \neq \delta$ alors $\mu\gamma[\gamma]t\{\beta/\alpha\}$

$$\begin{array}{rcl} \Phi(\mu\gamma[\beta]\mu\alpha[\delta]t) & = & \mathbf{catch}\ \gamma\ \mathbf{throw}\ \beta\ \mathbf{catch}\ \alpha\ \mathbf{throw}\ \delta\ \Phi(t) \\ & \leadsto_6 & \mathbf{catch}\ \gamma\ \mathbf{throw}\ \beta\ (\mathbf{throw}\ \delta\ \Phi(t))\{\beta/\alpha\} \\ & = & \mathbf{catch}\ \gamma\ \mathbf{throw}\ \beta\ \mathbf{throw}\ \delta\ \Phi(t)\{\beta/\alpha\} \\ & \leadsto_5 & \mathbf{catch}\ \gamma\ \mathbf{throw}\ \delta\ \Phi(t)\{\beta/\alpha\} \\ & = & \Phi(\mu\gamma[\delta]t\{\beta/\alpha\}) \end{array}$$

- Cas de la règle $\mu\alpha[\alpha]t \leadsto t$ si α n'apparaît pas libre dans t.

$$\Phi(\mu\alpha[\alpha]t) = \mathbf{catch}\ \alpha\ \Phi(t) \leadsto_1 \Phi(t)$$

□

Nous aurons aussi besoin du lemme suivant :

Lemme 3.2.4 *Pour tout contexte $C\{\bullet\}$ et tous $\lambda\mu$-termes u, v, si $\Phi(u) \leadsto^{\star}_{\mathbf{ct}} \Phi(v)$ alors :*

$$\Phi(C\{u\}) \leadsto^{\star}_{\mathbf{ct}} \Phi(C\{v\})$$

Preuve. Par récurrence sur le contexte :

- Si le contexte est $\bullet$, $\Phi(C\{u\}) = \Phi(u) \rightsquigarrow^{\star}_{\mathbf{ct}} \Phi(v) = \Phi(C\{v\})$.
- Si le contexte est une application ou une abstraction, le pas de récurrence est direct.
- Si le contexte est de la forme $\mu\alpha[\beta]C\{\bullet\}$ alors, par définition de la traduction Φ, d'une part,

$$\Phi(\mu\alpha[\beta]C\{u\}) = \begin{cases} \mathbf{catch}\ \alpha\ \Phi(C\{u\}) & \textit{si } \alpha = \beta \\ \mathbf{throw}\ \beta\ \Phi(C\{u\}) & \textit{si } \alpha \neq \beta \textit{ et } \alpha \textit{n'est pas libre dans } C\{u\} \\ \mathbf{catch}\ \alpha\ \mathbf{throw}\ \beta\ \Phi(C\{u\}) & \textit{sinon} \end{cases}$$

et d'autre part,

$$\Phi(\mu\alpha[\beta]C\{v\}) = \begin{cases} \mathbf{catch}\ \alpha\ \Phi(C\{v\}) & \textit{si } \alpha = \beta \\ \mathbf{throw}\ \beta\ \Phi(C\{v\}) & \textit{si } \alpha \neq \beta \textit{ et } \alpha \textit{n'est pas libre dans } C\{v\} \\ \mathbf{catch}\ \alpha\ \mathbf{throw}\ \beta\ \Phi(C\{v\}) & \textit{sinon} \end{cases}$$

Par hypothèse de récurrence $\Phi(C\{u\}) \rightsquigarrow^{\star}_{\mathbf{ct}} \Phi(C\{v\})$, et nous raisonnons alors par cas :

1. Si $\alpha = \beta$, en appliquant l'hypothèse de récurrence :

$$\Phi(\mu\alpha[\beta]C\{u\}) = \mathbf{catch}\ \alpha\ \Phi(C\{u\}) \rightsquigarrow^{\star}_{\mathbf{ct}} \mathbf{catch}\ \alpha\ \Phi(C\{v\}) = \Phi(\mu\alpha[\beta]C\{v\})$$

2. Si $\alpha \neq \beta$ et α n'apparaît pas libre dans $C\{u\}$, alors α ne peut pas apparaître libre dans $C\{v\}$ car aucune réduction du $\lambda\mu$-calcul n'introduit de μ-variable libre, d'où en appliquant l'hypothèse de récurrence :

$$\Phi(\mu\alpha[\beta]C\{u\}) = \mathbf{throw}\ \beta\ \Phi(C\{u\}) \rightsquigarrow^{\star}_{\mathbf{ct}} \mathbf{throw}\ \beta\ \Phi(C\{v\}) = \Phi(\mu\alpha[\beta]C\{v\})$$

3. Si $\alpha \neq \beta$ et α apparaît libre dans $C\{u\}$, et α n'apparaît libre dans $C\{v\}$ alors en appliquant l'hypothèse de récurrence puis la règle (1) :

$$\begin{array}{rcl} \Phi(\mu\alpha[\beta]C\{u\}) & = & \mathbf{catch}\ \alpha\ \mathbf{throw}\ \beta\ \Phi(C\{u\}) \\ & \rightsquigarrow^{\star}_{\mathbf{ct}} & \mathbf{catch}\ \alpha\ \mathbf{throw}\ \beta\ \Phi(C\{v\}) \\ & \rightsquigarrow^{1}_{\mathbf{ct}} & \mathbf{throw}\ \beta\ \Phi(C\{v\}) \\ & = & \Phi(\mu\alpha[\beta]C\{v\}) \end{array}$$

4. Si $\alpha \neq \beta$ et α apparaît libre dans $C\{u\}$ et dans $C\{v\}$ en appliquant l'hypothèse de récurrence :

$$\begin{array}{l} \Phi(\mu\alpha[\beta]C\{u\}) = \mathbf{catch}\ \alpha\ \mathbf{throw}\ \beta\ \Phi(C\{u\}) \\ \qquad \rightsquigarrow^{\star}_{\mathbf{ct}} \mathbf{catch}\ \alpha\ \mathbf{throw}\ \beta\ \Phi(C\{v\}) = \Phi(\mu\alpha[\beta]C\{v\}) \end{array}$$

□

Proposition 3.2.5 *Pour tous $\lambda\mu$-termes u, v, si $u \rightsquigarrow_{\lambda\mu} v$ alors $\Phi(u) \rightsquigarrow^{\star}_{\mathbf{ct}} \Phi(v)$.*

Preuve. Supposons que u soit de la forme $C\{t\}$ où C est un contexte et t le radical à réduire, par une règle dont l'instance est $t \rightsquigarrow_{\lambda\mu} t'$. Par le lemme 3.2.2, on sait que $\Phi(t) \rightsquigarrow^{\star}_{\mathbf{ct}} \Phi(t')$, puis par le lemme 3.2.4, $\Phi(u) = \Phi(C\{t\}) \rightsquigarrow^{\star}_{\mathbf{ct}} \Phi(C\{t'\}) = \Phi(v)$. □

Traduction du $\lambda_{\mathbf{ct}}$-calcul vers le $\lambda\mu$-calcul : remplacer toute occurrence de **throw** α par $\mu\_[\alpha]$ et enfin toute occurence de **catch** α par $\mu\alpha[\alpha]$. En voici une définition formelle :

Définition 3.2.6 *On définit la traduction Ψ du $\lambda\mu$-calcul vers le $\lambda_{\mathbf{ct}}$-calcul par récurrence sur les $\lambda\mu$-termes :*

- *$\Psi(x) = x$, si x est une λ-variable,*
- *$\Psi((u\ v)) = (\Psi(u)\ \Psi(v))$*
- *$\Psi(\lambda x.t) = \lambda x.\Psi(t)$*
- *$\Psi(\mathbf{catch}\ \alpha\ t) = \mu\alpha[\alpha]\Psi(t)$*
- *$\Psi(\mathbf{throw}\ \alpha\ t) = \mu\delta[\alpha]\Psi(t)$ où δ est une μ-variable n'apparaissant pas dans $\Psi(t)$.*

Remarque. Pour tout $\lambda_{\mathbf{ct}}$-terme t, $\Psi(t)$ est un $\lambda\mu\mathbf{ct}$-terme.

Proposition 3.2.7 *Pour tous termes t, t' du $\lambda_{\mathbf{ct}}$-calcul, si $t \rightsquigarrow_{\mathbf{ct}} t'$ alors $\Psi(t) \rightsquigarrow^{\star}_{\lambda\mu} \Psi(t')$.*

Preuve. Par construction des règles du $\lambda_{\mathbf{ct}}$-calcul. En effet, celles-ci correspondent toutes à des règles du $\lambda\mu$-calcul portant sur des $\lambda\mu$-termes de la forme $\mu\alpha[\beta]t$ pour lesquels $\alpha = \beta$ ou $\alpha \neq \beta$ et α n'apparaît pas dans t. □

Lemme 3.2.8 $\Phi \circ \Psi = \mathrm{Id}$.

Preuve. Montrons que $\Phi(\Psi(t)) = t$ par récurrence sur le $\lambda_{\mathbf{ct}}$-termes t :

- $\Phi(\Psi(x)) = \Phi(x) = x$, si x est une λ-variable,
- $\Phi(\Psi((u\ v))) = \Phi((\Psi(u)\ \Psi(v)) = (\Phi(\Psi(u))\ \Phi(\Psi(v))) = (u\ v)$
- $\Phi(\Psi(\lambda x.t)) = \Phi(\lambda x.\Psi(t)) = \lambda x.\Phi(\Psi(t)) = \lambda x.t$
- $\Phi(\Psi(\mathbf{catch}\ \alpha\ t)) = \Phi(\mu\alpha[\alpha]\Psi(t)) = \mathbf{catch}\ \alpha\ \Phi(\Psi(t)) = \mathbf{catch}\ \alpha\ t$
- $\begin{array}{rcll} \Phi(\Psi(\mathbf{throw}\ \alpha\ t)) & = & \Phi(\mu\delta[\alpha]\Psi(t)) & \text{où } \delta \text{ est n'apparaît pas dans } \Psi(\mathrm{t}) \\ & = & \mathbf{throw}\ \alpha\ \Phi(\Psi(t)) & \text{car } \delta \text{ est n'apparaît pas dans } \Psi(\mathrm{t}) \\ & = & \mathbf{throw}\ \alpha\ t & \end{array}$

□

Remarque. La traduction Φ est donc surjective, la traduction Ψ injective. Par ailleurs, Ψ n'est clairement pas surjective puisque toute image d'un $\lambda_{\mathbf{ct}}$-terme par Ψ est un $\lambda\mu\mathbf{ct}$-terme. D'autre part, Φ n'est pas injective puisque :

$$\Phi(\mu\alpha[\alpha]\mu\_[\beta]t) = \mathbf{catch}\ \alpha\ \mathbf{throw}\ \beta\ \Phi(t) = \Phi(\mu\alpha[\beta]t)$$

Par contre, il est clair que pour tout $\lambda\mu\mathbf{ct}$-terme t on a $\Psi(\Phi(t)) = t$.

3.3 Confluence du $\lambda_{\mathbf{ct}}$-calcul

Nous allons maintenant montrer que le $\lambda_{\mathbf{ct}}$-calcul est confluent. La preuve pourrait sans doute se faire directement en montrant que les règles commutent deux à deux, mais nous montrons ici que ce résultat découle de la confluence du $\lambda\mu$-calcul qui nous permet alors d'obtenir le diagramme suivant :

$$
\begin{tikzcd}
 & & t \arrow[dll, "\rightsquigarrow^{\star}_{\mathbf{ct}}"'] \arrow[d, "\Psi"'] \arrow[drr, "\rightsquigarrow^{\star}_{\mathbf{ct}}"] & & \\
u \arrow[d, "\Psi"'] & & \Psi(t) \arrow[dll, "\rightsquigarrow^{\star}_{\lambda\mu}"] \arrow[drr, "\rightsquigarrow^{\star}_{\lambda\mu}"'] & & v \arrow[d, "\Psi"] \\
\Psi(u) \arrow[d, "\Phi"'] \arrow[drr, "\rightsquigarrow^{\star}_{\lambda\mu}"] & & & & \Psi(v) \arrow[d, "\Phi"] \arrow[dll, "\rightsquigarrow^{\star}_{\lambda\mu}"'] \\
\Phi(\Psi(u)) \arrow[drr, "\rightsquigarrow^{\star}_{\mathbf{ct}}"'] & & w \arrow[d, "\Phi"'] & & \Phi(\Psi(v)) \arrow[dll, "\rightsquigarrow^{\star}_{\mathbf{ct}}"] \\
 & & \Phi(w) & &
\end{tikzcd}
$$

On a montré que pour tout $\lambda_{\mathbf{ct}}$-terme t, $t = \Phi(\Psi(t))$, (la propriété $t \rightsquigarrow^{\star}_{\mathbf{ct}} \Phi(\Psi(t))$ aurait en fait suffit) et on obtient ainsi la confluence en raison du diagramme suivant :

$$
\begin{tikzcd}
 & t \arrow[dl, "\rightsquigarrow^{\star}_{\mathbf{ct}}"'] \arrow[dr, "\rightsquigarrow^{\star}_{\mathbf{ct}}"] & \\
\Phi(\Psi(u)) = u \arrow[dr, "\rightsquigarrow^{\star}_{\mathbf{ct}}"'] & & v = \Phi(\Psi(v)) \arrow[dl, "\rightsquigarrow^{\star}_{\mathbf{ct}}"] \\
 & \Phi(w) &
\end{tikzcd}
$$

Théorème 3.3.1 *Les règles du $\lambda_{\mathbf{ct}}$-calcul sont confluentes.*

Preuve. Considérons un $\lambda_{\mathbf{ct}}$-terme t tel que $t \rightsquigarrow^{\star}_{\mathbf{ct}} u$ et $t \rightsquigarrow^{\star}_{\mathbf{ct}} v$. Par la proposition 3.2.7, $\Psi(t) \rightsquigarrow^{\star}_{\lambda\mu} \Psi(u)$ et $\Psi(t) \rightsquigarrow^{\star}_{\lambda\mu} \Psi(v)$. Puisque le $\lambda\mu$-calcul est confluent, on sait qu'il existe un $\lambda\mu$-terme z tel que $\Psi(u) \rightsquigarrow^{\star}_{\lambda\mu} z$ et $\Psi(v) \rightsquigarrow^{\star}_{\lambda\mu} z$. Par la proposition 3.2.5, $\Phi(\Psi(u)) \rightsquigarrow^{\star}_{\mathbf{ct}} \Phi(z)$ et $\Phi(\Psi(v)) \rightsquigarrow^{\star}_{\mathbf{ct}} \Phi(z)$. Or, $\Phi(\Psi(u)) = u$ et $\Phi(\Psi(u)) = v$ par le lemme ci-dessus, d'où $u \rightsquigarrow^{\star}_{\mathbf{ct}} \Phi(z)$ et $v \rightsquigarrow^{\star}_{\mathbf{ct}} \Phi(z)$. $\square$

3.4 Variables locales « piégées »

Dans cette sous-section nous commençons par introduire une notion de variable locale « piégée » pour aboutir à une définition de $\lambda\mu$-terme (resp. $\lambda_{\mathbf{ct}}$-terme) ne piégeant aucune variable que nous appellerons « intuitionniste » (cette terminologie sera justifiée dans la section suivante). Nous montrons alors que l'ensemble des $\lambda\mu$-termes intuitionnistes est stable par réduction.

⋆ Présentation informelle

Rappelons tout d'abord le terme K, présenté en sous-section 2.1, qui simule le **call/cc** en $\lambda_{\mathbf{ct}}$-calcul :

$$\lambda y.\mathbf{catch}\ \gamma\ (y\ \lambda x.\mathbf{throw}\ \gamma\ x)$$

Ce terme possède la particularité suivante : il « piège » une variable locale. En effet, la variable x a une occurence dans un **throw** alors que le **catch** correspondant est extérieur à la déclaration de x. La situation n'est pourtant pas toujours aussi claire. Par exemple, dans le schéma de programme suivant :

$$\ldots\mathbf{catch}\ \beta\ (\lambda x.(\mathbf{throw}\ \beta\ (\mathbf{catch}\ \gamma\ \ldots(\mathbf{throw}\ \gamma\ x))))$$

la variable x n'est plus déclarée entre le **throw** γ et le **catch** γ mais entre le **throw** β et le **catch** β. La variable x a donc toujours été piégée. Alors que dans l'exemple suivant :

$$\ldots\mathbf{catch}\ \beta\ (\lambda x.(\mathbf{catch}\ \gamma\ \ldots(\mathbf{throw}\ \beta\ \ldots(\mathbf{throw}\ \gamma\ x))))$$

la variable x est aussi déclarée entre le **throw** β et le **catch** β, mais elle n'est pas piégée par le **throw** β car le **catch** γ se trouve sous la portée de x.

Plus généralement, si l'on considère qu'un $\lambda_{\mathbf{ct}}$-terme est un arbre, il existe une branche allant de toute occurrence d'une λ-variable vers la racine. L'algorithme permettant de savoir si une occurrence de λ-variable est piégée par un **throw** consiste donc à « remonter » le long de cette branche en effectuant un « bond » de chaque **throw** au **catch** correspondant. Si lors de ce parcours, on « bondit » par dessus la déclaration de la variable, on sait que cette occurrence a été piégée. Une variable est alors dite piégée si au moins une de ses occurrences est piégée. En soulignant par une flèche les zones parcourues, on retrouve que la variable x a été piégée dans les premiers exemples :

$$\underleftarrow{\lambda y.\mathbf{catch}\ \gamma}\ (y\ \lambda x.\mathbf{throw}\ \underleftarrow{\gamma\ x})$$

$$\underleftarrow{\ldots\mathbf{catch}\ \beta}\ (\lambda x.(\mathbf{throw}\ \underleftarrow{\beta\ (\mathbf{catch}\ \gamma}\ \ldots(\mathbf{throw}\ \underleftarrow{\gamma\ x}))))$$

alors qu'elle ne l'a pas été dans le dernier :

$$\underleftarrow{\ldots\mathbf{catch}\ \beta\ (\lambda x.(\mathbf{catch}\ \gamma}\ \ldots(\mathbf{throw}\ \beta\ \ldots(\mathbf{throw}\ \underleftarrow{\gamma\ x}))))$$

À nouveau, si l'on se concentre sur la branche partant d'une occurrence d'une variable libre et allant jusqu'à la racine du terme, la suite des « bonds » peut se représenter comme une suite de **catch** α_i ... **throw** α_i (où les points de suspension désignent la zone non soulignée dans les exemples. Deux cas peuvent donc se produire lorsqu'on explicite cette suite pour une occurrence donnée d'une λ-variable libre x, selon que l'on s'arrête sur un **throw** (d'une μ-variable α_i libre) ou non :

1. $\underleftarrow{\ldots\mathbf{catch}\ \alpha_i}\ \ldots\mathbf{throw}\ \underleftarrow{\alpha_i\ \ldots\mathbf{catch}\ \alpha_2}\ \ldots\mathbf{throw}\ \underleftarrow{\alpha_2\ \ldots\mathbf{catch}\ \alpha_1}\ \ldots\mathbf{throw}\ \underleftarrow{\alpha_1\ \ldots x}$

2. $\ldots\mathbf{throw}\ \underleftarrow{\alpha_i\ \ldots\mathbf{catch}\ \alpha_2}\ \ldots\mathbf{throw}\ \underleftarrow{\alpha_2\ \ldots\mathbf{catch}\ \alpha_1}\ \ldots\mathbf{throw}\ \underleftarrow{\alpha_1\ \ldots x}$

où dans les deux cas, **throw** α_1 est le premier **throw** rencontré en partant de x, et il n'y a aucun throw entre **throw** α_{j+1} et **catch** α_j pour tout $1 \leq j < i$. De plus, dans le cas (1) il n'y a aucun **throw** à gauche de **catch** α_i (mais peut-être des **catch**) et dans le cas (2), il n'y a pas de **catch** α_i à gauche de **throw** α_i (mais peut-être d'autres **catch** ou **throw**).

Pour résumer, étant donné une occurrence d'une λ-variable x libre dans un $\lambda_{\mathbf{ct}}$-terme, l'information qui nous intéresse pour déterminer si cette occurrence est piégée est, s'il existe, le dernier **throw** β sur lequel on s'est arrêté en remontant la branche (deuxième schéma ci-dessus) et on dira alors que cette occurrence est « libre sous β dans t », dans le cas contraire (premier schéma ci-dessus) on dira que cette occurrence de x est « libre à la racine dans t ».

⋆ Définitions

Nous allons maintenant donner une définition plus formelle, qui nous servira dans les preuves par récurrence, de cette notion de « piège », dans le cadre du $\lambda\mu$-calcul, puis du $\lambda_{\mathbf{ct}}$-calcul. Les notions de λ-variables et μ-variable libres dans un $\lambda\mu$-terme sont définies de manière habituelle. La définition suivante partitionne l'ensemble des occurrences de λ-variables libres dans un $\lambda\mu$-terme t en celles qui sont « libres à la racine » de t et celle qui sont « libres sous α » où α est une μ-variable libre de t. L'intuition de cette définition est la suivante : l'opération $[\,]$ est l'opération de « nommage » des termes du $\lambda\mu$-calcul, nous étendons ce nommage aux occurrences de variables libres dans un $\lambda\mu$-terme. Ne sont nommées que les occurrences qui n'ont pas déjà un nom : les occurrences de variables libres non nommées dans t, sont nommées α dans $[\alpha]t$. L'opération inverse, $\mu\alpha$ « affranchit » les occurrences de variables libres nommées : les occurrences de variables libres nommées α dans t, ne sont plus nommées dans $\mu\alpha.t$.

Définition 3.4.1 *Étant donné un $\lambda\mu$-terme t, l'ensemble des occurrences de λ-variables libres à la racine dans t et l'ensemble des occurrences de λ-variables libres sous α dans t, où α est une μ-variable libre de t, sont définis par récurrence simultanée sur t de la manière suivante :*

- *Si $t = x$, la seule occurrence de la λ-variable x est libre à la racine de t.*
- *Si $t = \lambda x.u$, une occurrence d'une λ-variable libre de t est libre à la racine dans t si libre à la racine dans u. Pour toute μ-variable α libre dans u, une occurrence d'une λ-variable libre de t est libre sous α dans t si elle est libre sous α dans u.*
- *Si $t = (u\ v)$, une occurrence d'une λ-variable libre de t est libre à la racine dans t si elle est libre à la racine dans u ou dans v. Pour toute μ-variable α libre dans t, une occurrence d'une λ-variable libre de t est libre sous α dans t si elle est libre sous α dans u ou dans v.*
- *Si $t = [\alpha]u$, aucune occurrence d'une λ-variable libre de t n'est libre à la racine dans t. Pour toute μ-variable β, différente de α, libre dans t, une occurrence d'une λ-variable libre de t est libre sous β dans t si elle est libre sous β dans u. Une occurrence d'une λ-variable libre de t est libre sous α dans t si elle est libre sous α ou libre à la racine dans u.*
- *Si $t = \mu\alpha.u$, une occurrence d'une λ-variable libre de t est libre à la racine dans t si elle est libre sous α dans u. Pour toute μ-variable β libre dans t, une occurrence d'une λ-variable libre de t est libre sous β dans t si elle est libre sous β dans u.*

Remarque. Dans le cas particulier d'un $\lambda\mu\mathbf{ct}$-terme, la définition précédente nous donne :

– Si $t = \mu\alpha[\alpha]u$, une occurrence d'une λ-variable libre de t est libre à la racine dans t, si elle est libre sous α dans $[\alpha]u$, c'est-à-dire, si elle libre à la racine ou libre sous α dans u. Pour toute μ-variable β, libre dans t, une occurrence d'une λ-variable libre de t est libre sous β dans t si elle est libre sous β dans $[\alpha]u$, c'est-à-dire si elle est libre sous β dans u.

– Si $t = \mu\delta[\alpha]u$, où δ n'apparaît pas dans $[\alpha]u$, aucune occurrence d'une λ-variable libre de t n'est libre à la racine dans t, puique δ n'apparaît pas dans $[\alpha]u$. Pour toute μ-variable β, différente de α, libre dans t, une occurrence d'une λ-variable libre de t est libre sous β dans t si elle est libre sous β dans $[\alpha]u$ et donc libre sous β dans u. Une occurrence d'une λ-variable libre de t est libre sous α dans t si elle est libre sous α dans $[\alpha]u$, c'est-à-dire libre à la racine dans u ou libre sous α dans u.

On obtient donc la définition dérivée suivante :

Définition 3.4.2 *Étant donné un $\lambda_{\mathbf{ct}}$-terme t, l'ensemble des λ-variables libres à la racine dans t et l'ensemble des λ-variables libres sous α dans t, où α est une μ-variable libre de t, sont définis par récurrence simultanée sur t de la même manière que dans la définition 3.4 pour les variables, l'abstraction et l'application, et de la manière suivante pour les opérateurs* **catch** *et* **throw** *:*

– *Si $t =$ **catch** $\alpha\ u$, une occurrence d'une λ-variable libre de t est libre à la racine dans t si elle est libre à la racine ou libre sous α dans* u. *Pour toute μ-variable β, libre dans t, une occurrence d'une λ-variable libre de t est libre sous β dans t si elle est libre sous β dans u.*

– *Si $t =$ **throw** $\alpha\ u$, aucune occurrence d'une λ-variable libre de t n'est libre à la racine dans t. Pour toute μ-variable β, différente de α, libre dans t, une occurrence d'une λ-variable libre de t est libre sous β dans t si elle est libre sous β dans u. Une occurrence d'une λ-variable libre de t est libre sous α dans t si elle est libre à la racine dans u ou libre sous α dans u.*

Remarques

– On peut vérifier par récurrence qu'une occurrence d'une λ-variable libre d'un terme t est soit libre à la racine, soit libre sous α dans t pour une μ-variable α libre de t. Autrement dit cette définition partitionne bien les λ-variables libres.

– On vérifie de même qu'une λ-variable x est donc libre sous α, si pour une occurrence de x, en remontant de cette occurrence vers la racine et en « bondissant » de chaque **throw** β au **catch** β correspondant, on est obligé de s'arrêter sur une occurence de **throw** α (où α est une μ-variable libre). La λ-variable x est libre à la racine si le même procédé nous permet d'atteindre la racine du terme.

Définition 3.4.3 *Une λ-variable x est dite libre sous α dans t, pour une μ-variable α libre de t, (resp. libre à la racine) si elle possède au moins une occurrence libre sous α dans t (resp. libre à la racine).*

Remarque. Une λ-variable peut donc à la fois être libre sous α dans t, pour une μ-variable α libre de t et libre à la racine dans t.

Définition 3.4.4 *Un $\lambda\mu$-terme (resp. un $\lambda_{\mathbf{ct}}$-terme) t est dit **intuitionniste** ssi pour tout sous-terme de t de la forme $\lambda x.u$, pour toute μ-variable α libre dans u, x n'est pas libre sous α dans u. Dans le cas contraire, on dit que la variable x est **piégée** par α dans t.*

Remarque. Un $\lambda\mu$-terme clos (resp. un $\lambda_{\mathbf{ct}}$-terme) est donc intuitionniste si pour aucune déclaration de λ-variable, et pour aucune occurence de cette variable, en remontant de cette occurence vers la racine du terme comme décrit dans la remarque précédente, on ne « bondit » par dessus la déclaration de cette variable.

$\star$ Stabilité par réduction

Lemme 3.4.5 *Le sous-ensemble des $\lambda\mu$-termes intuitionnistes est stable par substitution (i.e. si u et v sont intuitionnistes alors $u\{v/x\}$ est aussi intuitionniste).*

Preuve. Posons $w = u\{v/x\}$ et supposons que pour un sous-terme de w de la forme $\lambda y.s$, y apparaisse libre sous α, pour une μ-variable α libre dans s. Deux cas peuvent se présenter : soit le lieur λy provient de v, soit il provient de u. Dans le premier cas, cela signifie que y apparaissait déjà libre sous α dans v, ce qui est impossible car v est intuitionniste. Dans le second cas, y ne peut apparaître libre dans v (sinon il y aurait capture de variable), donc y devait apparaître libre sous α dans u, ce qui est impossible car u est intuitionniste. $\square$

Théorème 3.4.6 *Le sous-ensemble des $\lambda\mu$-termes intuitionnistes est stable pour les règles de réduction du $\lambda\mu$-calcul.*

Preuve. Montrons, pour chacune des règles de réduction que si t est un $\lambda\mu$-terme intuitionniste et $t \rightsquigarrow t'$ alors t' est aussi intuitionniste. Supposons que y soit piégée par α dans t', (*i.e.* supposons qu'il existe un sous-terme $\lambda y.s$ de t' tel que y soit libre sous α dans s, où α est une μ-variable libre de s). Appelons r le radical de t que l'on réduit.

- $r = (\lambda x.u\ v)$ et $r' = u\{v/x\}$. Trois cas peuvent se présenter : soit r' est un sous-terme de s, soit $\lambda y.s$ est un sous-terme de r', soit r' et $\lambda y.s$ sont disjoints. Dans le premier cas, puisque u est intuitionniste, x ne peut pas être libre sous α dans u. Par conséquent, si y est libre sous α dans r', alors y est déjà libre sous α dans u ou dans v, et t n'est pas intuitionniste, d'où la contadiction. Dans les deux autre cas, la contradiction provient du lemme 3.4.5.
- $r = (\mu\alpha.u\ v)$ et $r' = \mu\alpha.u\{[\alpha](w\ v)/[\alpha]w\}$. Le seul point à remarquer est qu'une variable libre de v, devient libre sous α dans $u\{[\alpha](w\ v)/[\alpha]w\}$, mais n'est pas piégé par α dans r'.
- $r = [\beta]\mu\alpha.u$ et $r' = u\{\beta/\alpha\}$. Il suffit de voir que si y est libre sous β dans r', alors y est soit déjà libre sous β dans u (et donc dans r) soit libre à la racine dans $\mu\alpha.r$ et donc libre sous β dans $[\beta]\mu\alpha.u = r$
- $r = \mu\alpha[\alpha]u$ et $r' = u$ où α n'apparaît pas libre dans u. En effet, car y ne peut être piégé par β dans r' car sinon il le serait déjà dans $\mu\alpha[\alpha]u = r$.

$\square$

4 Logique classique et opérateurs de contrôle

Les opérateurs de contrôle permettent de donner un sens calculatoire à la logique classique. Cette constatation est due à l'origine à T. G. GRIFFIN [23]. Cette investigation à été poursuivie ensuite par C. R. MURTHY [40, 41, 42], F. BARBANERA et S. BERARDI [2, 3], N. J. REHOF et M. H. SØRENSEN [57], P. DE GROOTE [10],...

Nous utilisons à nouveau, dans cette section, le $\lambda\mu$-calcul de M. PARIGOT [47, 48], qui vérifie la normalisation forte au second ordre [49], pour dériver les règles de typage et le même résultat de normalisation pour le $\lambda_{\mathbf{ct}}$-calcul. Nous montrons ensuite que, dans le cas d'un $\lambda\mu$-terme (resp $\lambda_{\mathbf{ct}}$-terme) typable, la contrainte intuitionniste définie dans la section précédente correspond exactement à la contrainte intuitionniste de CND^i (chapitre 3, section 4) appliquée à la dérivation du jugement de typage. On retrouve alors une contrainte semblable à celle de H. NAKANO [44, 45].

4.1 Présentation

Avant de donner les règles de typage formelles associées au $\lambda\mu$-calcul (resp. au $\lambda_{\mathbf{ct}}$-calcul), nous allons tenter d'expliquer informellement comment les opérateurs de contrôle peuvent donner un sens calculatoire à la logique classique. Pour cela, nous choisisons la formulation du tiers-exclu sous la forme de la règle utilisée dans le chapitre 2 :

$$\frac{\begin{array}{cc} [f : A \to \bot] & [x : A] \\ \vdots & \vdots \\ u : C & v : C \end{array}}{C}$$

Supposons pour simplifier que u et v dénotent des preuves intuitionnistes : on peut donc considérer, par l'isomorphisme de Curry-Howard, que u et v sont des λ-termes, qui contiennent comme variables libres (nous supposerons que ce sont les seules) respectivement f de type $A \to \bot$ et x de type A. Il est donc possible d'évaluer u, par exemple, par la machine de Krivine décrite en section 2.1. La machine s'arrête bien entendu sur une erreur si elle doit évaluer la variable libre f (puisqu'elle n'est pas définie dans l'environnement).

Si l'on considère maintenant la preuve dans son ensemble, l'évaluation du tiers-exclu peut se faire de la manière suivante : *tenter d'évaluer u, et si cette évaluation « échoue » en cherchant à évaluer $(f\ a)$, alors évaluer $(v\ a)$.*

Cette présentation synthétique met en valeur plusieurs aspects : tout d'abord, le choix par défaut (on tente d'évaluer u, autrement dit, on suppose $A \to \bot$), qui sera peut-être remis en cause ultérieurement ; le cas échéant, la « récupération » de l'argument a de f (qui est une preuve de A) et l'abandon de l'évaluation en cours (ce n'est plus u qu'il faut évaluer mais $(v\ a)$, puisque A est vrai) et enfin la reprise de l'évaluation dans l'état où on a commencé à évaluer u.

Le rôle des opérateurs de contrôle apparaît alors clairement : ils permettent de réaliser cet abandon puis cette reprise de l'évaluation. En effet, il suffit de restaurer le contexte d'évaluation qui était présent lorsqu'on a commencé à évaluer u; contexte que l'on aura pris soin de sauvegarder à ce moment-là.

4.2 Typage du $\lambda\mu$-calcul

Nous présentons ici les règles de typage du $\lambda\mu$-calcul de [47]. Les règles de typage sont les règles de CND à la différence qu'une des conclusions de chaque séquent est non nommée (nous la séparerons des autres conclusions par un point-virgule). Chaque *séquent* est décoré par un $\lambda\mu$-terme : la syntaxe d'un séquent décoré sera donc $t : \Gamma \vdash \Delta; A$. Le séparateur « : » (resp. « ; ») sera omis lorsque Γ (resp. Δ) est vide. À nouveau nous donnons les règles de la conjonction et de la disjonction bien qu'elles soient dérivables au second ordre.

Axiome

$$x : A^x \vdash A$$

Règles de coupure

$$\frac{u : \Gamma \vdash \Delta; A \qquad t : \Gamma, A^x \vdash \Delta; B}{\mathbf{let}\ x = u\ \mathbf{in}\ t : \Gamma \vdash \Delta; B}$$

Remarque. La règle de calcul associée au **let** ... **in** est bien entendu la substitution :

$$\mathbf{let}\ x = u\ \mathbf{in}\ t \rightsquigarrow t\{u/x\}$$

Règle d'affaiblissement gauche

$$\frac{t : \Gamma \vdash \Delta; B}{t : \Gamma, A^x \vdash \Delta; B}$$

Règle de contraction gauche

$$\frac{t : \Gamma, A^x, A^y \vdash \Delta; B}{t\{z/x, z/y\} : \Gamma, A^z \vdash \Delta; B}$$

Règles du $\rightarrow$

$$\frac{t : \Gamma, A^x \vdash \Delta; B}{\lambda x.t : \Gamma \vdash \Delta; A \rightarrow B} \qquad \frac{u : \Gamma \vdash \Delta; A \rightarrow B \qquad v : \Gamma \vdash \Delta; A}{(u\ v) : \Gamma \vdash \Delta; B}$$

Remarque. Il est connu que les connecteurs $\wedge, \vee$ sont définissables en λ-calcul et typables au second ordre, en utilisant les définitions au second ordre de ces connecteurs. On obtient alors les règles de typage et de calcul dérivées suivantes :

– **Règles du $\wedge$**

$$\frac{u : \Gamma \vdash \Delta; A \qquad v : \Gamma \vdash \Delta; B}{\langle u, v\rangle : \Gamma \vdash \Delta; A \wedge B} \qquad \frac{t : \Gamma \vdash \Delta; A \wedge B}{\mathbf{fst}(t) : \Gamma \vdash \Delta; A} \qquad \frac{t : \Gamma \vdash \Delta; A \wedge B}{\mathbf{snd}(t) : \Gamma \vdash \Delta; B}$$

Règles de calcul :

$$\mathbf{fst}(\langle u, v\rangle) \rightsquigarrow u \qquad \mathbf{snd}(\langle u, v\rangle) \rightsquigarrow v$$

– **Règles du** $\vee$

$$\frac{t : \Gamma \vdash \Delta; A}{\mathbf{inl}(t) : \Gamma \vdash \Delta; A \vee B} \qquad \frac{t : \Gamma \vdash \Delta; B}{\mathbf{inr}(t) : \Gamma \vdash \Delta; A \vee B}$$

$$\frac{t : \Gamma \vdash \Delta; A \vee B \qquad u : \Gamma \vdash \Delta; (A \to C) \qquad v : \Gamma \vdash \Delta; (B \to C)}{\mathbf{cases}\ t\ u\ v : \Gamma \vdash \Delta; C}$$

Règles de calcul :

$$\mathbf{cases}\ \mathbf{inl}(x)\ u\ v \rightsquigarrow (u\ x) \qquad \mathbf{cases}\ \mathbf{inr}(y)\ u\ v \rightsquigarrow (v\ y)$$

$\star$ Règles des quantificateurs

Comme dans le système AF_2 de J.-L. Krivine [31] les règles d'introduction et d'élimination des quantificateurs ne sont pas explicitée par les $\lambda\mu$-termes qui décorent les règles. les règles des connecteurs $\exists, \exists^2$ peuvent d'obtenir soit par dualité, en à partir de leur définition au second ordre.

Règles du $\forall$

$$\frac{u : \Gamma \vdash \Delta; A}{u : \Gamma \vdash \Delta; \forall x A} \qquad \frac{u : \Gamma \vdash \Delta; \forall x A}{u : \Gamma \vdash \Delta; A\{t/x\}}$$

Règles du $\forall^2$

$$\frac{u : \Gamma \vdash \Delta; A}{u : \Gamma \vdash \Delta; \forall X A} \qquad \frac{u : \Gamma \vdash \Delta; \forall X A}{u : \Gamma \vdash \Delta; A\{T/X\}}$$

$\star$ Règles de nommage

Ces règles sont « les règles » du $\lambda\mu$-calcul, puisqu'elles permettent de gérer les conclusions multiples. Remarquons qu'un terme « nommé » (*i.e.* de la forme $[\alpha]t$) décore un séquent ne contenant aucune formule non nommée, elle est donc nécéssairement suivie d'une règle μ. Cette contrainte correspond à la contraint syntaxique des $\lambda\mu$-termes où toute occurrence du constructeur μ est nécéssairement de la forme $\mu\alpha[\beta]$.

$$\frac{t : \Gamma \vdash \Delta; A}{[\alpha]t : \Gamma \vdash \Delta, A^\alpha;} \qquad \frac{t : \Gamma \vdash \Delta, A^\alpha;}{\mu\alpha.t : \Gamma \vdash \Delta; A}$$

Remarque. Dans la règle μ, la formule A^α peut ne pas figurer dans le séquent hypothèse : dans ce cas le sens logique de cette règle est l'affaiblissement à droite. Dans la règle [ ], la formule A^α peut déjà figurer dans le Δ du séquent hypothèse : dans ce cas, le sens logique de cette règle est la contraction à droite. Nous allons maintenant isoler ces cas particuliers pour obtenir les règles dérivées d'affaiblissement et de contraction à droite.

– Règle d'affaiblissement droite (où B^β n'apparaît pas dans Δ)

$$\frac{\dfrac{t : \Gamma \vdash \Delta; A}{[\alpha]t : \Gamma \vdash \Delta, A^\alpha;}}{\mu\beta[\alpha]t : \Gamma \vdash \Delta, A^\alpha; B}$$

– Règle de contraction droite

$$\dfrac{\dfrac{t : \Gamma \vdash \Delta, A^\alpha; t : A}{[\alpha]t : \Gamma \vdash \Delta, A^\alpha;}}{\mu\alpha[\alpha]t : \Gamma \vdash \Delta; A}$$

On retrouve donc les deux « macros » du $\lambda\mu$**ct**-calcul « **catch** » et « **throw** ». Nous pouvons par conséquent en déduire des règles de typage dérivées pour ces opérateurs.

⋆ **Typage des opérateurs « catch » et « throw »**

Règle de contraction droite

$$\frac{t : \Gamma \vdash \Delta, A^\alpha; A}{\mathbf{catch}\ \alpha\ t : \Gamma \vdash \Delta; A}$$

Règle d'affaiblissement droite

$$\frac{t : \Gamma \vdash \Delta; A}{\mathbf{throw}\ \alpha\ t : \Gamma \vdash \Delta, A^\alpha; B}$$

Proposition 4.2.1 *Un $\lambda_{\mathbf{ct}}$-terme t est donc typable de type $\Gamma \vdash \Delta; A$ si et seulement si le $\lambda\mu$**ct**-calcul associé $\Psi(t)$ est typable de type $\Gamma \vdash \Delta; A$.*

Exemple. Le terme $K \equiv \lambda y.\mathbf{catch}\ \alpha\ (y\ \lambda x.\mathbf{throw}\ \alpha\ x)$, de même que le $\lambda\mu$**ct**-terme qu'il représente [47], peut se typer par l'axiome de Peirce :

$$\dfrac{\dfrac{\dfrac{y : ((A \to B) \to A)^y \vdash (A \to B) \to A \qquad \dfrac{\dfrac{x : A^x \vdash A}{\mathbf{throw}\ \alpha\ x : A^x \vdash A^\alpha; B}}{\lambda x.\mathbf{throw}\ \alpha\ x \vdash A^\alpha; A \to B}}{(y\ \lambda x.\mathbf{throw}\ \alpha\ x) : ((A \to B) \to A)^y \vdash A^\alpha; A}}{\mathbf{catch}\ \alpha\ (y\ \lambda x.\mathbf{throw}\ \alpha\ x) : ((A \to B) \to A)^y \vdash A}}{\lambda y.\mathbf{catch}\ \alpha\ (y\ \lambda x.\mathbf{throw}\ \alpha\ x) \vdash ((A \to B) \to A) \to A}$$

⋆ **Préservation du typage**

Le $\lambda\mu$-calcul vérifie la propriété de préservation du typage (dans les cadres propositionnel, du premier et du second ordre) par réduction (« subject reduction property »), c'est-à-dire que si un $\lambda\mu$-terme t est typable de type $\Gamma \vdash \Delta; A$ et $t \rightsquigarrow_{\lambda\mu} t'$ alors t' est aussi typable de type $\Gamma \vdash \Delta; A$. Cette propriété s'étend directement au $\lambda_{\mathbf{ct}}$-calcul :

Proposition 4.2.2 *Étant donné un $\lambda_{\mathbf{ct}}$-terme t, si t est typable de type $\Gamma \vdash \Delta; A$ et $t \rightsquigarrow_{\mathbf{ct}} t'$ alors t' est aussi typable de type $\Gamma \vdash \Delta; A$.*

Preuve. Par la proposition 4.2.1, si t est typable de type $\Gamma \vdash \Delta; A$, alors le $\lambda\mu$**ct**-terme $\Psi(t)$ est aussi typable de type $\Gamma \vdash \Delta; A$. On sait aussi que $\Psi(t) \rightsquigarrow_{\lambda\mu} \Psi(t')$, et puisque le $\lambda\mu$-calcul préserve le typage par réduction, $\Psi(t')$ est typable de type $\Gamma \vdash \Delta; A$, et t' l'est donc aussi, à nouveau par la proposition 4.2.1. □

⋆ **Normalisation forte au second ordre**

Le $\lambda\mu$-calcul vérifie la normalisation forte au second ordre, c'est-à-dire que si un $\lambda\mu$-terme t est typable d'un type $\Gamma \vdash \Delta; A$ alors il n'existe pas de suite infinie de réductions partant de t. Cette propriété s'étend directement au $\lambda_{\mathbf{ct}}$-calcul :

Proposition 4.2.3 *Étant donné un $\lambda_{\mathbf{ct}}$-terme t, si t est typable de type $\Gamma \vdash \Delta; A$ alors il n'existe pas de suite infinie de réductions partant de t.*

Preuve. Si t est typable de type $\Gamma \vdash \Delta; A$, alors le $\lambda\mu\mathbf{ct}$-terme $\Psi(t)$ est aussi typable de type $\Gamma \vdash \Delta; A$. S'il existait une suite infinie de réductions $t_1 \leadsto_{\mathbf{ct}} t_2 \ldots \leadsto_{\mathbf{ct}} t_n \ldots$ alors il existerait aussi une suite infinie de réductions $\Psi(t_1) \leadsto_{\lambda\mu} \Psi(t_2) \ldots \leadsto_{\lambda\mu} \Psi(t_n) \ldots$, ce qui contredit la normalisation forte du $\lambda\mu$-calcul. □

5 Contrainte intuitionniste de CND^i et $\lambda\mu$-calcul

Nous avons maintenant défini deux notions, la notion de liens entre une hypothèse x et une conclusion α d'un séquent dérivé dans CND et la notion de variable libre sous α dans une $\lambda\mu$-terme. Or, nous venons de voir comment les $\lambda\mu$-termes (resp. les $\lambda_{\mathbf{ct}}$-termes) permettent de décorer des dérivations de CND. Nous allons maintenant montrer que dans le cas de $\lambda\mu$-termes (resp. les $\lambda_{\mathbf{ct}}$-termes) typables, les deux notions de liens coïncident. Comme corollaire, nous obtenons que les preuves de CND qui respectent la contrainte intuitionniste (*i.e.* les preuves de CND^i) sont exactement celles décorées par des $\lambda\mu$-termes intuitionnistes.

Remarque. Puisque dans la version de CND où les preuves sont décorées par des $\lambda\mu$-termes, une des conclusions du séquent n'est pas nommée, nous allons lui donner systématiquement le nom de « racine ».

Théorème 5.0.4 *Étant donné la dérivation d'un séquent dans* CND*, décorée par les liens de dépendances et les $\lambda\mu$-termes, le jugement conclusion $t : \Gamma_1^{x_1}, \ldots, \Gamma_n^{x_n} \vdash \Delta_1^{\alpha_1}, \ldots, \Delta_p^{\alpha_p}; A$, possède un lien entre Γ_i et A ssi x_i est libre à la racine dans t et un lien entre Γ_i et Δ_j ssi x_i est libre sous α_j dans t.*

Preuve. Par récurrence sur la dérivation du séquent. □

Corollaire 5.0.5 *Un séquent dérivé dans* CND *est en fait dérivé dans* CND^i *si et seulement si le $\lambda\mu$-terme qui décore le jugement conclusion de la preuve est intuitionniste.*

5.1 Décoration de CND^i par des λ-termes

Nous avons vu dans le chapitre 3 qu'il est possible d'extraire d'une preuve $\Gamma \vdash \Delta$ dans CND^i, une preuve dans NJ de $\Gamma \vdash \Delta_j$ pour au moins une conclusion Δ_j. Cette extraction va maintenant s'exprimer comme la possibilité de décorer au moins une conclusion par un λ-terme. De plus, toute conclusion qui ne provient pas directement ou indirectement d'un affaiblissement (qui sont elles décorée à nouveau par ∞, comme dans la preuve du théorème 5 du chapitre 3) peut être décorée par un λ-terme. Nous rappelons que cette extraction peut être non déterministe lors d'une contraction à droite où les deux conclusions contractées sont décorées par des λ-termes.

Exemples

1. Les $\lambda\mu$-termes représentant les entiers classiques sont en fait des $\lambda\mu$-termes intuitionnistes. Nous reprenons un exemple de [48] qui est une dérivation de $N(s0)$ où N est le codage du type des entiers au second ordre habituel :

$$N(n) \equiv \forall X(X0 \to (\forall y(Xy \to Xsy) \to Xn))$$

Ce premier exemple montre comment les parties de la preuve qui proviennent d'un affaiblissement disparaissent (nous utilisons une présentation des preuves sous forme de dérivation de formules, et non de séquent, pour faciliter la lecture) :

Décoration par un $\lambda_{\mathbf{ct}}$-terme :

$$\dfrac{\dfrac{\dfrac{\dfrac{f : \forall y(Xy \to Xsy)}{f : (X0 \to Xs0)} \quad \dfrac{\dfrac{\dfrac{f : \forall y(Xy \to Xsy)}{f : (X0 \to Xs0)} \quad x : X0}{(f\ x) : Xs0}}{\mathbf{throw}\ \alpha\ (f\ x) : Xs0^{\alpha}; X0}}{(f\ \mathbf{throw}\ \alpha\ (f\ x)) : Xs0^{\alpha}; Xs0}}{\mathbf{catch}\ \alpha\ (f\ \mathbf{throw}\ \alpha\ (f\ x)) : Xs0}}{\lambda x.\lambda f.\mathbf{catch}\ \alpha\ (f\ \mathbf{throw}\ \alpha\ (f\ x)) : \forall X(X0 \to (\forall y(Xy \to Xsy) \to Xs0))}$$

Décoration par des λ-termes :

$$\dfrac{\dfrac{\dfrac{\dfrac{f : \forall y(Xy \to Xsy)}{f : (X0 \to Xs0)} \quad \dfrac{\dfrac{\dfrac{f : \forall y(Xy \to Xsy)}{f : (X0 \to Xs0)} \quad x : X0}{(f\ x) : Xs0}}{(f\ x) : Xs0, \infty : X0}}{Xs0, (f\ x) : Xs0, \infty : Xs0}}{(f\ x) : Xs0}}{\lambda x.\lambda f.(f\ x) : \forall X(X0 \to (\forall y(Xy \to Xsy) \to Xs0))}$$

2. Ce deuxième exemple met en lumière l'importance du premier ordre dans le typage. En effet, dans la preuve suivante le non-déterminisme de l'extraction est sans conséquence :

Décoration par un λ_{ct}-terme :

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{f : \forall y(Xy \to Xsy)}{f : (X0 \to Xs0)} \quad x : X0}{(f\ x) : Xs0}}{\mathbf{throw}\ \beta\ (f\ x) : Xs0^{\beta}; (X0 \to Xs0)} \quad \dfrac{\dfrac{\dfrac{f : \forall y(Xy \to Xsy)}{f : (X0 \to Xs0)} \quad x : X0}{(f\ x) : Xs0}}{\mathbf{throw}\ \alpha\ (f\ x) : Xs0^{\alpha}; X0}}{((\mathbf{throw}\ \beta\ (f\ x))\ (\mathbf{throw}\ \alpha\ (f\ x))) : Xs0^{\alpha}, Xs0^{\beta}; Xs0}}{\mathbf{catch}\ \beta\ \mathbf{catch}\ \alpha\ ((\mathbf{throw}\ \beta\ (f\ x))\ (\mathbf{throw}\ \alpha\ (f\ x))) : Xs0}}{\lambda x.\lambda f.\mathbf{catch}\ \beta\ \mathbf{catch}\ \alpha\ ((\mathbf{throw}\ \beta\ (f\ x))\ (\mathbf{throw}\ \alpha\ (f\ x))) : \forall X(X0 \to (\forall y(Xy \to Xsy) \to Xs0))}$$

Décoration par des λ-termes :

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{f : \forall y(Xy \to Xsy)}{f : (X0 \to Xs0)} \quad x : X0}{(f\ x) : Xs0}}{(f\ x) : Xs0, \infty : (X0 \to Xs0)} \quad \dfrac{\dfrac{\dfrac{f : \forall y(Xy \to Xsy)}{f : (X0 \to Xs0)} \quad x : X0}{(f\ x) : Xs0}}{(f\ x) : Xs0, \infty : X0}}{(f\ x) : Xs0, (f\ x) : Xs0, \infty : Xs0}}{(f\ x) : Xs0}}{\lambda x.\lambda f.(f\ x) : \forall X(X0 \to (\forall y(Xy \to Xsy) \to Xs0))}$$

3. Voici une variante de la preuve précédente typée cette fois-ci au second ordre propositionnel (système F). Cette fois-ci un choix apparaît clairement lors de l'extraction.

Décoration par un λ_{ct}-terme :

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{f:(X\to X)\quad \dfrac{f:X\to X\quad x:X}{(f\ x):X}}{(f\ (f\ x)):X}}{\mathbf{throw}\ \alpha\ (f\ (f\ x)):X^{\alpha};(X\to X)}\qquad \dfrac{\dfrac{f:(X\to X)\quad x:X}{(f\ x):X}}{\mathbf{throw}\ \alpha\ (f\ x):X^{\alpha};X}}{((\mathbf{throw}\ \alpha\ (f\ (f\ x)))\ (\mathbf{throw}\ \alpha\ (f\ x))):X^{\alpha},X^{\alpha};X}}{\mathbf{catch}\ \alpha\ ((\mathbf{throw}\ \alpha\ (f\ (f\ x)))\ (\mathbf{throw}\ \alpha\ (f\ x))):X}}{\lambda x.\lambda f.\mathbf{catch}\ \alpha\ ((\mathbf{throw}\ \alpha\ (f\ (f\ x)))\ (\mathbf{throw}\ \alpha\ (f\ x))):\forall X(X\to(\forall y(Xy\to Xsy)\to X))}$$

Première décoration par des λ-termes :

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{f:(X\to X)\quad \dfrac{f:X\to X\quad x:X}{(f\ x):X}}{(f\ (f\ x)):X}}{(f\ (f\ x)):X,\infty:(X\to X)}\qquad \dfrac{\dfrac{f:(X\to X)\quad x:X}{(f\ x):X}}{(f\ x):X,\infty:X}}{(f\ (f\ x)):X,(f\ x):X,\infty:X}}{(f\ (f\ x)):X}}{\lambda x.\lambda f.(f\ (f\ x)):\forall X(X\to(\forall y(Xy\to Xsy)\to X))}$$

Deuxième décoration par des λ-termes :

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{f:(X\to X)\quad \dfrac{f:X\to X\quad x:X}{(f\ x):X}}{(f\ (f\ x)):X}}{(f\ (f\ x)):X,\infty:(X\to X)}\qquad \dfrac{\dfrac{f:(X\to X)\quad x:X}{(f\ x):X}}{(f\ x):X,\infty:X}}{(f\ (f\ x)):X,(f\ x):X,\infty:X}}{(f\ x):X}}{\lambda x.\lambda f.(f\ x):\forall X(X\to(\forall y(Xy\to Xsy)\to X))}$$

⋆ Extraction de λ-termes à partir de $\lambda_{\mathbf{ct}}$-termes intuitionnistes

Étant donné un $\lambda_{\mathbf{ct}}$-terme typable dans CND^i, nous avons vu qu'il est possible d'en extraire un λ-terme typable en déduction naturelle intuitionniste NJ. Le principe de l'extraction consiste à répéter l'opération suivante : « choisir un **throw** α t le plus interne (*i.e.* tel que t ne contiennent pas de **throw**) et substituer t au **catch** α u correspondant » jusqu'à ce qu'il ne reste plus de **throw**, puis à supprimer tout **catch** restant. La contrainte intuitionniste nous garantit que ce procédé ne « libère » pas de λ-variable liée. Le fait de choisir un **throw** le plus interne nous garantit qu'on ne « libère » pas non plus de μ-variable. Enfin, le typage est préservé puisque pour tout **catch** α u, et pour tout sous-terme de u de la forme **throw** α t, le terme t est du même type que u.

Ce processus d'extraction peut aussi être défini à partir de règles de réduction, qui sont celles de H. Nakano [44]. Il faut pour cela se donner toutes les règles permettant à un

throw de « remonter » vers le **catch** correspondant. Dans les règles du $\lambda_{\mathbf{ct}}$-calcul, plusieurs constructeurs peuvent « bloquer » un **throw** :

– l'application, lorsque le **throw** est argument : (t **throw** α u). Il suffit de se donner la règle suivante :

  8. (t **throw** α u) $\rightsquigarrow$ **throw** α u

  Cette règle mène clairement à un calcul non-confluent. Il suffit de considérer l'exemple suivant dû à H. Nakano [44] :

$$M \equiv \textbf{catch}\ \alpha\ ((\lambda x.\lambda y.1\ (\textbf{throw}\ \alpha\ 2)\ (\textbf{throw}\ \alpha\ 3))$$

  On a alors 3 formes normales possibles :

$$M \rightsquigarrow_\beta \textbf{catch}\ \alpha\ ((\lambda y.1\ (\textbf{throw}\ \alpha\ 2)) \rightsquigarrow_\beta \textbf{catch}\ \alpha\ 1 \rightsquigarrow_1 1$$
$$M \rightsquigarrow_8 \textbf{catch}\ \alpha\ (\textbf{throw}\ \alpha\ 3) \rightsquigarrow_7 \textbf{catch}\ \alpha\ 3 \rightsquigarrow_1 3$$
$$M \rightsquigarrow_8 \textbf{catch}\ \alpha\ ((\textbf{throw}\ \alpha\ 2)\ (\textbf{throw}\ \alpha\ 3)) \rightsquigarrow_3 \textbf{catch}\ \alpha\ \textbf{throw}\ \alpha\ 2 \rightsquigarrow_7 \textbf{catch}\ \alpha\ 2 \rightsquigarrow_1 2$$

– l'abstraction : $\lambda x.$**throw** α t, il suffit de se donner la règle suivante :

  9. $\lambda x.$**throw** α $t \rightsquigarrow$ **throw** α t

Cette règle est valide (*i.e.* elle ne libère pas de variable) uniquement en raison de la contrainte intuitionniste. En effet, x ne peut pas apparaître libre dans t, car sinon x serait soit libre sous β dans t, pour une μ-variable β libre dans t, soit libre à la racine dans t et donc libre sous α dans **throw** α t.

Naturellement, les règles 8 et 9 préservent le typage puisque le **throw** est une instruction qui peut prendre n'importe quel type (sa règle de typage est la règle d'affaiblissement). Enfin, le dernier constructeur qui peut s'opposer à la « montée » d'un **throw** α est un **catch** β où $\beta \neq \alpha$. Ce dernier cas n'empêche toutefois pas tous les **catch** et **throw** d'un terme de disparaitre par réduction, comme le montre la proposition suivante :

Proposition 5.1.1 *Dans le* $\lambda_{\mathbf{ct}}$*-calcul intuitionniste, muni des règles* $1, 3, 5, 7, 8, 9$*, un terme* μ*-clos en forme normale ne contient plus aucun* **throw** *ni aucun* **catch**.

Preuve. Soit t un $\lambda_{\mathbf{ct}}$-terme en forme normale (où aucune des règles 1,3,5,7,8,9 ne soit applicable). Supposons que t contiennent au moins un **catch** et considérons alors un sous-terme de t de la forme **catch** α v tel que v ne contiennent pas de **catch**. Par conséquent, v contient donc nécéssairement un sous-terme de la forme **throw** α u (sinon le sous-terme **catch** α v serait un radical 1). Considérons alors le constructeur immédiatement au-dessus de **throw** α u. Ce ne peut être le **catch** α, (ce serait un radical 7) ni un **catch** β (où $\beta \neq \alpha$) par hypothèse. Mais, ce constructeur ne peut être non plus ni une application (ce serait un radical 3 ou 9), ni une abstraction (ce serait un radical 8), ni un **throw** (ce serait un radical 5), d'où la contradiction. □

Remarque. Par ailleurs, il est facile de voir que la règle 9 commute avec les règles 1-7 du $\lambda_{\mathbf{ct}}$-calcul, on obtient alors un $\lambda_{\mathbf{ct}}$-calcul *intuitionniste confluent*.

6 La soustraction

Pour donner un sens calculatoire à la soustraction, nous revenons au cadre classique, où ce connecteur est définissable. Les primitives ainsi obtenues dépendent de la formule choisie pour définir la soustraction et de la manière dont les règles d'introduction et d'élimination du connecteur sont dérivées. En effet, l'équivalence logique entre les différentes formulations ne se traduit pas nécessairement en un isomorphisme du point de vue calculatoire. L'interprétation de la soustraction proposée dans cette section est guidée par la volonté de typer un mécanisme de traitement des exceptions comme le **try with** du langage ML. En effet, dans ce langage les exceptions sont déclarées globalement et n'apparaissent donc pas dans le typage des programmes. Une solution à ce problème à été proposée par J. C. Guzmán et À. Suárez [25]. Nous nous placons ici dans un cadre différent puisque, à nouveau, toutes les liaisons de variables considérées ici sont statiques, alors que le **try with** de ML effectue des liaisons dynamiques.

6.1 $\lambda\mu$-calcul et opérateur $\mathcal{C}$

Pour dériver les règles de calcul associées à la soustraction, définie en logique classique par $A - B \equiv A \wedge \neg B$, nous utiliserons *provisoirement* le « typage des continuations par la négation intuitionniste » comme dans [30]. L'opérateur de contrôle $\mathcal{C}$ est alors typé par $\neg\neg A \to A$. Toutefois, nous avons vu que cet opérateur permet de définir l'opérateur « abort » par $\mathcal{A}(t) = \mathcal{C}\lambda k.t$ où k est donc typé par $\neg A$ et t par $\bot$. Nous avons vu aussi que le sens calculatoire de l'opérateur $\mathcal{A}$ est l'arrêt brutal du calcul en cours (*i.e.*, l'abandon de la continuation courante) pour ne donner comme résultat que la valeur de son argument. Un programme ne sera donc bien typé que si le terme contenant cet opérateur est lui même de type $\bot$ (on peut montrer que sous cette condition, la réduction préserve le typage [30]).

Remarque. Un moyen de contourner cette contrainte en $\lambda_{\mathbf{ct}}$-calcul (resp. en $\lambda\mu$-calcul), consiste à supposer que tout $\lambda_{\mathbf{ct}}$-terme commence par un **catch** ϕ : il est alors possible de traduire l'opérateur $\mathcal{A}$ simplement par **throw** ϕ. Il n'est alors plus nécessaire de supposer que tout terme t « exécutable » soit de type $\bot$, la seule contrainte reste bien sûr que pour toute occurrence **throw** ϕ u le sous-terme u soit de même type que t. Il faut pour cela remplacer toute occurrence de $\bot$ par O, où O est le type du terme exécutable t. En particulier, la négation doit alors être définie comme $A \to O$.

Il suffit alors de dériver $\neg\neg A \to A$ dans CND pour obtenir un terme du $\lambda\mu$-calcul correspondant à l'opérateur $\mathcal{C}$ (cf. [47]).

$$\dfrac{\dfrac{y : \neg\neg A^y \vdash \neg\neg A \qquad \dfrac{x : A^x \vdash A}{\lambda x.\mu\delta[\alpha]x \vdash A^\alpha ; \neg A}}{\mu\alpha[\phi](y\ \lambda x.\mu\delta.[\alpha]x) : \neg\neg A^y \vdash \bot^\phi ; A}}{\lambda y.\mu\alpha[\phi](y\ \lambda x.\mu\delta.[\alpha]x) \vdash \bot^\phi ; \neg\neg A \to A}$$

On pose donc $C \equiv \lambda y.\mu\alpha[\phi](y\ \lambda x.\mu\delta.[\alpha]x)$. La conclusion $\bot^\phi$ sera supposée présente dans tous les contextes et ne sera donc plus explicitée. La preuve (resp. le $\lambda\mu$-terme) terminera toujours par une contraction de $\bot^\phi$ (resp. un **catch** ϕ).

Nous rappelons que les opérateurs **catch** et **throw** sont alors définissables à partir de l'opérateur $\mathcal{C}$ et en posant **catch** $\alpha\ t \equiv \mathcal{C}\lambda\alpha.(\alpha\ t)$ (qui peut aussi s'écrire $\mathcal{K}\lambda\alpha.t$) et **throw** $\alpha\ t$

par $\mathcal{C}\lambda\delta.(\alpha\ t)$ où δ n'apparaît pas libre dans t (qui peut aussi s'écrire $\mathcal{A}(\alpha\ t)$). Les termes sont maintenant typés par des séquents de la forme $\Gamma_1, \ldots, \Gamma_n, \neg\Delta_1, \ldots, \neg\Delta_p \vdash A$. Par exemple, les règles de typage des opérateurs **catch** et **throw** s'écrivent ainsi :

$$\frac{t : \Gamma; \neg A^\alpha \vdash A}{\mathbf{catch}\ \alpha\ t : \Gamma \vdash A} \qquad \frac{t : \Gamma;\ \vdash A}{\mathbf{throw}\ \alpha\ t : \Gamma; \neg A^\alpha \vdash B}$$

6.2 Traitement des exceptions et soustraction

Nous avons maintenant tous les éléments permettant de donner un sens « calculatoire » à la soustraction. Tout d'abord, nous savons que ce connecteur est définissable en logique classique, en posant $A - B \equiv A \wedge \neg B$ par exemple. Nous pouvons donc dériver ses règles d'introduction et d'élimination (gauches), ainsi que les règles de calcul associées. Nous verrons alors que ce connecteur permet de typer une généralisation du mécanisme **try**... **with** de traitement des exceptions du langage ML où les noms d'exception disparaissent du traitement au profit de leur types. Cette généralisation est l'analogue, dans le λ-calcul typé, de l'abstraction fonctionnelle comparée au **let**... **in**... : dans le premier cas, l'application d'un terme à un autre est autorisée dès que le terme en position de fonction est d'un type $A \to B$ et que le terme en position d'argument est de type A, dans le second cas, on a besoin de connaître le nom de la variable à substituer. De même que l'abstraction permet de « différer » la substitution, l'instruction typée par la règle d'introduction de la soustraction permet de différer la capture d'une exception (effectuée par le **catch**).

Remarque. Nous utiliserons les règles d'introduction et d'élimination (gauche) de la conjonction suivantes :

$$\frac{t : \Gamma, A^x, B^y \vdash C}{\mathbf{match}\ z\ \mathbf{with}\ \langle x, y\rangle \mapsto t : \Gamma, (A \wedge B)^z \vdash C}$$

$$\frac{t : \Gamma, (A \wedge B)^z \vdash C}{\mathbf{let}\ z = \langle x, y\rangle\ \mathbf{in}\ t : \Gamma, A^x, B^y \vdash C}$$

et la règle de calcul dérivée est :

$$\mathbf{match}\ \langle u, v\rangle\ \mathbf{with}\ \langle x, y\rangle \mapsto t \rightsquigarrow t\{u/x, v/y\}$$

La règle d'introduction gauche est habituelle, la règle d'élimination gauche se dérive à partir la règle d'introduction droite et de la règle de coupure :

$$\frac{t : \Gamma, (A \wedge B)^z \vdash C \qquad \dfrac{x : A^x \vdash A \qquad y : B^y \vdash B}{\langle x, y\rangle : A^x, B^y \vdash A \wedge B}}{\mathbf{let}\ z = \langle x, y\rangle\ \mathbf{in}\ t : \Gamma, A^x, B^y \vdash C}$$

⋆ Décoration des règles de la soustraction

Nous pouvons maintenant décorer les règles d'introduction et d'élimination gauche de la soustraction, que nous rappelons ici :

$$\frac{\Gamma, A \vdash B, \Delta}{\Gamma, A - B \vdash \Delta} \qquad \frac{\Gamma, A - B \vdash \Delta \qquad \Gamma, B \vdash \Delta}{\Gamma, A \vdash \Delta}$$

Ces règles se traduisent ainsi, dans le calcul où les exceptions sont typées par des hypothèses négatives :

$$\frac{\Gamma, A; \neg B, \Delta \vdash C}{\Gamma, A - B; \Delta \vdash C} \qquad \frac{\Gamma, A - B; \Delta \vdash C \qquad \Gamma, B; \Delta \vdash C}{\Gamma, A; \Delta \vdash C}$$

Nous posons alors $A - B = A \wedge \neg B$. La règle d'introduction gauche de la soustraction se dérive directement, à partir de la règle d'introduction gauche de la conjonction :

$$\frac{t : \Gamma, A^x, \neg B^\alpha \vdash C}{\textbf{match } y \textbf{ with } \langle x, \alpha \rangle \mapsto t : \Gamma, (A \wedge \neg B)^y \vdash C}$$

Pour décorer la règle d'élimination gauche, nous allons tout d'abord décorer une instance de cette règle dont nous avons montré qu'elle suffit à retrouver la règle générale (sur laquelle nous reviendrons plus tard) :

$$\frac{\Gamma, A - B; \Delta \vdash B}{\Gamma, A; \Delta \vdash B}$$

Cette règle se dérive aussi simplement :

$$\frac{\dfrac{t : \Gamma, (A \wedge \neg B)^y \vdash B}{\textbf{let } y = \langle x, \alpha \rangle \textbf{ in } t : \Gamma, A^x, \neg B^\alpha \vdash B}}{\textbf{catch } \alpha \ (\textbf{let } y = \langle x, \alpha \rangle \textbf{ in } t) : \Gamma, A^x \vdash B}$$

et la règle de calcul dérivée, qui correspond à la réduction de la preuve formée d'une règle d'introduction gauche suivie directement d'une règle d'élimination gauche :

$$\textbf{catch } \alpha \ (\textbf{match } \langle x, \alpha \rangle \textbf{ with } \langle x', \alpha' \rangle \mapsto t) \rightsquigarrow \textbf{catch } \alpha \ t[x/x', \alpha/\alpha']$$

⋆ Interprétation

La règle d'introduction gauche de la soustraction lie une μ-variable α en la *couplant* avec une λ-variable x. Alors que l'opérateur **catch**, qui est le lieur habituel des μ-variables, associe à une μ-variable la continuation courante, il s'agit ici d'une simple « déclaration » de la μ-variable α. Il est possible, pour reprendre une syntaxe plus proche de ML par exemple, de coupler cette déclaration avec la déclaration de la λ-variable x en une instruction de la forme :

$$\textbf{exception } \alpha \textbf{ fun } x \mapsto t \equiv \lambda y.\textbf{match } y \textbf{ with } \langle x, \alpha \rangle \mapsto t$$

Cette instruction se type alors ainsi (en revenant au typage par de CND) :

$$\frac{t : \Gamma, A^x \vdash \Delta, B^\alpha; C}{\textbf{exception } \alpha \textbf{ fun } x \mapsto t : \Gamma \vdash \Delta; (A - B) \to C}$$

La règle d'élimination gauche de la soustraction capture la continuation courante et la couple avec la λ-variable libre x (qui peut être ensuite substituée par un terme). Ce couple formé d'un terme et de la continuation courante est alors substitué à la variable libre y de type $A - B$ dans t. À nouveau, il est possible de se donner une syntaxe plus proche de ML en posant :

$$\textbf{try } t \ a \equiv \textbf{catch } \alpha \ (t \ \langle a, \alpha \rangle)$$

dont la règle de typage est :

$$\frac{t : \Gamma \vdash \Delta; (A - B) \to B \qquad a : \Gamma \vdash \Delta; A}{\Gamma \vdash \textbf{try } t \ a : \Gamma \vdash \Delta; B}$$

Pour résumer, l'opérateur **try** évalue $(t\ a)$ et si un **throw** de type B est exécuté la valeur finale est celle retournée par le **throw**. Autrement dit, le « traitement d'exception » se borne à capturer le résultat transmis par le **throw**. Ceci se voit dans la règle de calcul, si le terme t dans **try** $(t\ a)$ est déjà de la forme **exception** α **fun** $x \mapsto u$, l'intruction **try** se réduit à un simple **catch**.

$$\mathbf{try}\ (\mathbf{exception}\ \alpha\ \mathbf{fun}\ x \mapsto t)\ a \rightsquigarrow \mathbf{catch}\ \alpha\ t\{a/x\}$$

Naturellement il est possible d'envisager des traitements d'exception qui ne font pas que transmettre la valeur du **throw** comme l'instruction **try with** de ML. Cette instruction se dérive, en décorant la règle générale d'élimination gauche de la soustraction :

$$\dfrac{\dfrac{\dfrac{t : \Gamma, (A \wedge \neg B)^y; \Delta \vdash C}{\mathbf{throw}\ \gamma\ t : \Gamma, (A \wedge \neg B)^y; \Delta, \neg C^\gamma \vdash B}}{\mathbf{catch}\ \alpha\ (\mathbf{let}\ y = \langle x, \alpha\rangle\ \mathbf{in}\ t) : \Gamma, A^x; \Delta, \neg C^\gamma \vdash B} \qquad h : \Gamma', B^z; \Delta' \vdash C}{\dfrac{\mathbf{let}\ z = (\mathbf{catch}\ \alpha\ (\mathbf{let}\ y = \langle x, \alpha\rangle\ \mathbf{in}\ t))\ \mathbf{in}\ h : \Gamma, A^x; \Delta, \neg C^\gamma \vdash C}{\mathbf{catch}\ \gamma\ (\mathbf{let}\ z = (\mathbf{catch}\ \alpha\ (\mathbf{let}\ y = \langle x, \alpha\rangle\ \mathbf{in}\ t))\ \mathbf{in}\ h) : \Gamma, A^x; \Delta \vdash C}}$$

Cette instruction peut aussi s'exprimer plus simplement à partir du **try** :

$$\mathbf{try}\ t\ a\ \mathbf{with}\ h \equiv \mathbf{catch}\ \gamma\ (h\ \mathbf{try}\ (\mathbf{throw}\ \gamma\ t)\ a)$$

On obtient alors la règle de typage dérivée :

$$\dfrac{t : \Gamma \vdash \Delta; (A - B) \to C \qquad a : \Gamma \vdash \Delta; A \qquad h : \Gamma \vdash \Delta; B \to C}{\Gamma \vdash \mathbf{try}\ t\ a\ \mathbf{with}\ h : \Gamma \vdash \Delta; C}$$

Autrement dit, pour appliquer une fonction de type $(A - B) \to C$ sur un argument de type A, il faut de plus un traitement d'exception (« handler ») de type $B \to C$. On souhaiterait naturellement une règle de calcul comme :

$$\mathbf{try}\ (\mathbf{exception}\ \alpha\ \mathbf{fun}\ x \mapsto t)\ a\ \mathbf{with}\ h \rightsquigarrow \mathbf{try}\ (\mathbf{exception}\ \alpha\ t\{\mathbf{throw}\ \alpha\ (h\ u)/\mathbf{throw}\ \alpha\ u\})\ a$$

C'est-à-dire :

$$\mathbf{try}\ (\mathbf{exception}\ \alpha\ \mathbf{fun}\ x \mapsto t)\ a\ \mathbf{with}\ h \rightsquigarrow \mathbf{catch}\ \alpha\ t\{a/x, \mathbf{throw}\ \alpha\ (h\ u)/\mathbf{throw}\ \alpha\ u\}$$

Cette règle de réduction traduit bien l'application du « handler » aux arguments des **throw**. Pour obtenir cette règle, il faudrait toutefois une nouvelle règle du λ**ct**-calcul :

$$(f\ \mathbf{catch}\ \alpha\ t) \rightsquigarrow \mathbf{catch}\ \alpha\ (f\ t\{\mathbf{throw}\ \alpha\ (f\ u)/\mathbf{throw}\ \alpha\ u\})$$

⋆ La contrainte intuitionniste

Nous rappelons la contrainte intuitionniste définie au chapitre 3 sur le déchargement d'une conclusion (duale de la contrainte sur le déchargement d'une hypothèse) :

$$\dfrac{U_1 : \Gamma_1, \ldots, U_n : \Gamma_n, U : A \vdash \Delta, B^\alpha; C}{U_1 : \Gamma_1, \ldots, U_n : \Gamma_n, U : A - B \vdash \Delta; C} \text{ où } \alpha \notin U_1 \cup \ldots \cup U_n$$

Nous avons aussi montré que l'existence d'un lien entre une hypothèse x et une conclusion α s'exprime directement comme « x est libre sous α » dans le λ**ct**-terme qui décore le séquent. On obtient ainsi la contrainte à imposer à la déclaration d'une exception pour rester intuitionniste. Dans la règle suivante, en plus de la contrainte due à l'introduction de l'implication « x n'apparaît pas libre sous une μ-variable de Δ dans t », on a la contrainte supplémentaire, due à l'introduction de la soustraction « x est la seule λ-variable pouvant apparaître libre sous α dans t » :

$$\frac{t : \Gamma, A^x \vdash \Delta, B^\alpha ; C}{\textbf{exception } \alpha \textbf{ fun } x \mapsto t : \Gamma \vdash \Delta ; (A - B) \to C}$$

Chapitre 5

La réalisabilité modifiée

Dans ce chapitre, nous rappelons certains résultats généraux concernant les techniques de réalisabilité et d'extraction de programmes. Nous nous concentrons alors plus spécifiquement sur la réalisabilité modifiée de G. Kreisel. Nous dégageons la notion de théorie réalisée, théorie dans laquelle justement l'extraction de programme est possible, et nous montrons qu'en particulier l'arithmétique avec fonctionnelles de types finis HA^ω est une théorie réalisée. Les programmes extraits sont « certifiés corrects » dans toute théorie réalisée, modulo les schémas d'axiomes du choix et d'indépendance des prémisses. Nous proposons alors une théorie des types, possédant une règle d'élimination forte du $\exists$ dans le style de la théorie des types de P. Martin-Löf, dans laquelle ces deux schémas sont dérivables et qui « internalise » donc la réalisabilité modifiée. Nous terminons ce chapitre par une sémantique de ce système, basée sur l'interprétation de la réalisabilité dans les modèles usuels du calcul des prédicats en logique classique. On obtient alors une notion de « vérité uniforme » comparable à la sémantique de H. Läuchli [35].

1 Introduction

L'objectif de ce chapitre n'est pas de présenter un panorama des différentes réalisabilités existantes (nous donnerons plutôt les références nécessaires) ; ce n'est pas non plus une introduction au domaine (nous renvoyons pour cela le lecteur à l'introduction de C. Paulin [51]). Pour une étude plus approfondie et des références supplémentaires nous renvoyons le lecteur à la thèse de C. Paulin [50].

Nous nous concentrons ici sur la différence entre la réalisabilité, vue comme une formalisation de la sémantique de Brouwer-Heyting-Kolmogoroff, et la prouvabilité, vue à la lumière de l'isomorphisme de Curry-Howard.

Nous rappelons tout d'abord deux grandes classes de réalisabilité issues de la sémantique de Brouwer-Heyting-Kolmogoroff (dite sémantique BHK), que sont respectivement la réalisabilité récursive de S. C. Kleene et la réalisabilité modifiée de G. Kreisel (abrégées en r-réalisabilité et mr-réalisabilité). Alors que la première s'appuie sur une classe de fonctions partielles (généralement la classe des fonctions semi-récursives), la seconde se restreint à une classe de fonction totales. En ce sens, la réalisabilité modifiée est plus proche de la prouvabilité puisque tout programme extrait d'une preuve (en s'appuyant sur l'isomorphisme de Curry-Howard) dénote une fonction totale.

La réalisabilité modifiée diffère néanmoins de la prouvabilité par deux axiomes (réalisables mais non prouvables), l'*axiome du choix* (AC) :

$$\forall x \exists y B(y) \Rightarrow \exists f \forall x B(f(x))$$

et l'*axiome d'indépendance des prémisses* (IP), où H est une formule de Harrop (cf. définition 2.2.1) :

$$(H \Rightarrow \exists y B(y)) \Rightarrow \exists y (H \Rightarrow B(y))$$

Il est connu qu'une règle d'élimination forte du $\exists$ dans le style de la théorie des types de P. Martin-Löf (notée ML, [36, 38]) permet de prouver l'axiome du choix. Par contre, l'axiome d'indépendance des prémisses n'est pas prouvable dans ML (qui est conservative sur HA, cf. [5] p. 323). Ceci s'explique par le fait que, bien que les termes de preuves dans ML dénotent des fonctions totales (ce sont en fait les fonctionnelles du Système T de Gödel), cette information n'est pas explicite dans le système formel.

Afin de caractériser la prouvabilité dans ML, M. D. G. Swaen [60, 61] est amené à étendre HA^{ω} par une « application conditionnelle », ce qui a malheureusement pour effet, de même que dans ML, d'oublier que les termes dénotent des fonctions totales. On peut par conséquent considérer, paradoxalement, que ML internalise la réalisabilité récursive.

Nous aurons ici une démarche complémentaire : nous allons définir une théorie des types munie d'une règle exprimant que « tout terme réalisant une formule est typable ». Il sera alors possible de prouver l'axiome d'indépendance des prémisses et de faire coïncider, cette fois-ci, prouvabilité et réalisabilité modifiée. Cette théorie, qui est plus « traditionnelle » que ML dans le sens où les formules et les types (respectivement les preuves et les programmes) ne sont plus identifiés, sera définie en section 6.

Remarque. Le terme « réalisabilité » est parfois utilisé dans des contextes légèrement différents, mais il désigne toujours un prédicat entre des fonctions, généralement calculables et représentées par un programme (un λ-terme, leur code de Gödel,...) et des formules. Il est important de remarquer que la classe des fonctions (l'ensemble de programmes...) préexiste. C'est la différence essentielle entre la réalisabilité et l'isomorphisme de Curry-Howard. Dans le premier cas, on prouve qu'une fonction donnée (par un programme) réalise une formule donnée (*i.e.* une spécification). Dans le second cas, c'est de la preuve de la formule qu'est extrait le programme.

Dans un premier contexte, qui est à l'origine celui de S. C. Kleene [28], « t réalise A » est une formule définie par récurrence sur A. Cette formule peut être vue comme une formalisation de la sémantique de BHK. La réalisabilité modifiée de G. Kreisel entre dans ce cadre.

Le deuxième contexte est celui de la théorie des types. Le prédicat de réalisabilité prend alors la forme d'un « jugement de typage ». Plus précisément, il s'agit du « typage *à la* Curry » [4] (le « typage *à la* Church », lorsque les types sont des formules, n'est qu'une autre façon de nommer l'isomorphisme de Curry-Howard). Dans ce cas, on prouve que « t réalise A » en dérivant le jugement de typage $t : A$.

Dans un dernier contexte, qui est celui des preuves de normalisation (que nous n'aborderons pas ici) le terme « réalisabilité » est parfois utilisé comme synonyme de « réductibilité » [22]. Nous préférons ici restreindre le champ de la réalisabilité à l'interaction entre programmes et spécifications, si possible indépendamment des propriétés calculatoires du langage de programmation. Ce point de vue sera concrétisé par la notion de théorie réalisée (cf. 4).

2 Préliminaires

La définition qui suit reprend la sémantique bien connue de BHK exprimée en terme de réalisabilité.

⋆ **Sémantique de Brouwer-Heyting-Kolmogoroff**

- une *formule atomique* est réalisée si et seulement si elle est valide ;
- une *conjonction* $A \wedge B$ est réalisée par un *couple* (p, q) si et seulement si A est réalisée par p et B est réalisée par q ;
- une *implication* $A \Rightarrow B$ est réalisée par une *fonction* f si et seulement si pour tout p réalisant A, $f(p)$ réalise B ;
- une quantification *existentielle* $\exists x A$ est réalisée par un *couple* (a, p), où a est un élément du domaine de quantification, si et seulement si p réalise $A[a/x]$;
- une quantification *universelle* $\forall x A$ est réalisée par une *fonction* f si et seulement si pour tout élément a du domaine de quantification, $f(a)$ réalise $A[a/x]$.

Remarque. L'ensemble des éléments susceptibles de réaliser une formule atomique n'est pas spécifié. N'importe quel ensemble non vide peut convenir, bien qu'un singleton semble plus adéquat. Cela ne signifie pas que les formules atomiques soient « sans contenu calculatoire » mais simplement qu'il n'est pas traité par la réalisabilité (sauf dans [35]). La raison en est qu'à l'origine, la réalisabilité a été introduite dans le cadre de théorie arithmétiques, où les formules atomiques sont décidables.

Un point important à remarquer est qu'une fonction qui réalise une implication $A \Rightarrow B$ doit être définie en tout point qui réalise A mais pas nécessairement ailleurs. On peut donc choisir, ou non, de se restreindre à des fonctions totales. C'est ce choix justement qui est à l'origine de la différence entre la r-réalisabilité et la mr-réalisabilité. En effet, toute fonction semi-récursive n'est pas nécessairement prolongeable en une fonction récursive totale. On obtiendra donc deux notions de réalisabilité réellement différentes.

Pour les définitions formelles des langages et des théories dans lesquels ces réalisabilités ont été étudiées, nous renvoyons aux travaux sur la réalisabilité récursive de S. C. Kleene [28] (dans la théorie HA, où les fonctions sont codées par des entiers), de S. Feferman [15] (dans la théorie APP, où les fonctions sont représentées par des λ-termes avec point fixes) et M. J. Beeson [5] p. 148 (dans la théorie EON, où les fonctions sont représentées par des λ-termes non typés) et les travaux de A. S. Troelstra sur la réalisabilité modifiée de G. Kreisel [64] p. 220 (dans la théorie HA, où les fonctions sont codées par des entiers) et p. 213 (dans la théorie HA$^\omega$, où les fonctions sont codées par des λ-termes typés).

2.1 Réalisabilité récursive

Voici tout d'abord quelques rappels sur la réalisabilité récursive. On se placera dans HA bien les mêmes résultats soient vérifiés dans EON. On note $(f\ a) \downarrow$ pour signifier que la fonction f est définie en a (ce prédicat de terminaison est primitif dans EON et défini par codage dans HA à partir du prédicat T de Kleene). Dans la définition suivante, $x\ r\ A$ est une *formule* de HA où x est une nouvelle variable n'apparaissant pas dans A.

– $x\ r\ A \equiv A$, si A est atomique

– $x\ r\ A \wedge B \equiv \pi(x)\ r\ A \wedge \pi'(x)\ r\ B$

– $x\ r\ A \Rightarrow B \equiv \forall y(y\ r\ A \Rightarrow ((x\ y) \downarrow \wedge (x\ y)\ r\ B))$

– $x\ r\ \forall y B(y) \equiv \forall y((x\ y) \downarrow \wedge (x\ y)\ r\ B)$

– $x\ r\ \exists y B(y) \equiv \pi'(x)\ r\ B(\pi(x))$

Remarque. La notation $x\ r\ A$ (ou $x\ mr\ A$) est justifiée par la propriété suivante (généralement appelé « lemme de substitution ») : $(x\ r\ A)[t/y] = x\ r\ A[t/y]$ où y est une variable libre de A, qui se prouve facilement par récurrence sur A.

Définition 2.1.1 *Une formule φ est dite r-réalisée (resp. r-réalisable) dans* HA *ssi il existe un terme t ayant les mêmes variables libres que φ telle* HA $\vdash t\ r\ \varphi$ *(resp.* HA $\vdash \exists x(x\ r\ \varphi)$*). Le terme t est alors appelé une réalisation de φ.*

2.2 Caractérisation de la réalisabilité récursive

Nous ne donnons ici que l'idée des preuves (pour plus de détails, cf. [64], par exemple). Nous donnerons les preuves détaillées dans le cadre de la réalisabilité modifiée. L'objectif de cette sous-section est de rappeler pourquoi l'axiome d'indépendance des prémisses n'est pas r-réalisable.

Définition 2.2.1 *Les* ***formules de Harrop*** *sont définies récursivement de la façon suivante :*

– *les formules atomiques sont de Harrop,*

– *si A et B sont de Harrop alors $A \wedge B$ est de Harrop,*

– *si B est de Harrop et A est* ***quelconque*** *alors $A \Rightarrow B$ est de Harrop,*

– *si A est de Harrop alors $\forall x A$ est de Harrop.*

Définition 2.2.2 *On appelle* ECT *(pour « Extended Church Thesis ») le schéma d'axiomes suivant :*

$$\forall x(H \Rightarrow \exists y B(y)) \Rightarrow \exists f \forall x(H \Rightarrow (f\ x) \wedge B(f\ x))$$

où H est une formule de Harrop.

Lemme 2.2.3 *Le schéma d'axiomes* ECT *est r-réalisé.*

Remarque. La restriction sur H aux formules de Harrop dans ECT est justifiée par la raison intuitive suivante : une preuve intuitionniste de $A \Rightarrow B$ est un procédé effectif (un programme, une fonction calculable...) qui transforme une preuve de A en une preuve de B. Donc dans l'hypothèse $H \Rightarrow \exists y.B$, la valeur de y dépend a priori de la preuve de H. Pour que l'axiome soit réalisé, il faut supposer que H soit une formule dont les preuves éventuelles n'auront pas de « contenu calculatoire » (cf. sous-section $\star$), ce qui est le cas des formules de Harrop. Par conséquent, si H est une formule de Harrop, une réalisation de $\forall x(H \Rightarrow \exists y B(y))$ est une fonction *partielle* qui est définie en tout x où H est réalisé, et qui associe alors à ce x une réalisation de $\exists y B(y)$.

Lemme 2.2.4 (validité) *Toute formule prouvable dans* HA *est r-réalisée dans* HA.

Preuve. Par récurrence sur la preuve. □

Remarque. Ce résultat s'étend en fait à toute théorie où les axiomes sont r-réalisés. Nous reviendrons en détail sur cette question dans le cadre de la réalisabilité modifiée.

Notation. Nous notons $\vdash_I$ la prouvabilité dans le calcul des prédicats intuitionniste.

Lemme 2.2.5 *Pour toute formule* φ, *on a* $\mathrm{ECT} \vdash_I \varphi \Leftrightarrow \exists x(x\ r\ \varphi)$

Preuve. Par récurrence sur la *formule* φ. □

Proposition 2.2.6 (caractérisation) *Pour tout formule* φ, *il existe un terme* t *tel que* $\mathrm{HA} \vdash t\ r\ \varphi$ *si et seulement si* $\mathrm{HA} + \mathrm{ECT} \vdash \varphi$.

Preuve. Soit t_{ECT} un terme (clos) qui réalise ECT (cf. lemme 2.2.3). Si $\mathrm{HA} + \mathrm{ECT} \vdash \varphi$ on a $\mathrm{HA} \vdash (\mathrm{ECT} \Rightarrow \varphi)$ et donc il existe un terme clos u tel que $\mathrm{HA} \vdash u\ r\ (\mathrm{ECT} \Rightarrow \varphi)$. On en déduit que $\mathrm{HA} \vdash (u\ t_{\mathrm{ECT}})\ r\ \varphi$. Réciproquement, si $\mathrm{HA} \vdash t\ r\ \varphi$, alors $\mathrm{HA} \vdash \exists x(x\ r\ \varphi)$ et $\mathrm{HA} + \mathrm{ECT} \vdash \varphi$. □

$\star$ Le schéma d'indépendance des prémisses n'est pas r-réalisable

Le schéma ECT est la généralisation du schéma d'axiomes CT suivant (prendre $H = \top$) :

$$\forall x \exists y B(y) \Rightarrow \exists f \forall x((f\ x) \downarrow \wedge B(f\ x))$$

Considérons maintenant le schéma d'axiomes d'*indépendance de prémisses* IP suivant (où H est une formule de Harrop) :

$$(H \Rightarrow \exists y B(y)) \Rightarrow \exists y(H \Rightarrow B(y))$$

Il est clair que $\mathrm{CT} + \mathrm{IP} \Rightarrow \mathrm{ECT}$. En effet, on peut dériver la preuve suivante dans le calcul des prédicats intuitionniste sous les hypothèses CT et IP :

$$\begin{array}{rcll}
\forall x(H \Rightarrow \exists y B(y)) & \Rightarrow & \forall x \exists y(H \Rightarrow B(y)) & (\mathrm{IP}) \\
& \Rightarrow & \exists f \forall x((f\ x) \downarrow \wedge H \Rightarrow B(f\ x)) & (\mathrm{CT}) \\
& \Rightarrow & \exists f \forall x(H \Rightarrow ((f\ x) \downarrow \wedge B(f\ x))) &
\end{array}$$

Remarquons que la dernière conclusion ci-dessus est strictement plus faible que la précédente. Par contre, $\mathrm{ECT} \not\Rightarrow \mathrm{IP}$ puisque le schéma d'axiomes IP n'est pas r-réalisé dans HA. Considérons la clôture suivante du schéma IP :

$$\forall x(H \Rightarrow \exists y B(y)) \Rightarrow \forall x \exists y(H \Rightarrow B(y))$$

Si IP était r-réalisé dans dans HA, on aurait un moyen effectif (la réalisation de IP) de transformer une réalisation de $\forall x(H \Rightarrow \exists y B(y))$, qui est une fonction partielle, en une réalisation de $\forall x \exists y(H \Rightarrow B(y))$, qui est une fonction totale. Or, toute fonction semi-récursive ne peut pas nécessairement être prolongée en une fonction totale récursive. Il suffit de diagonaliser le prédicat de Kleene pour obtenir un contre-exemple. Nous renvoyons à [51] p. 171 pour la preuve formelle.

3 La réalisabilité modifiée

La réalisabilité modifiée de G. Kreisel est habituellement définie dans le cadre de l'arithmétique avec fonctionnelles de types finis (HA^ω). Nous la définissons ici dans un calcul des prédicats « ω-sorté » et nous montrons que HA^ω est une théorie « réalisée ». À la différence de la réalisabilité récursive, les formules sont ici réalisées par des fonctions totales. Un moyen de se restreindre à des fonctions totales consiste à les représenter par des λ-termes (simplement) typés.

3.1 Le calcul des prédicats ω-sorté avec égalité IQC^ω

Un calcul des prédicats ω-sorté est un calcul des prédicats dont le langage des termes contient le λ-calcul simplement typé. Il est clair que tout calcul des prédicats (éventuellement multi-sorté) peut être considéré comme ω-sorté. L'égalité est utilisée pour définir la relation habituelle de convertibilité entre λ-termes.

$\star$ **Langage**

Le langage d'un calcul des prédicats ω-sorté est donné par :

– Un ensemble Σ de types de base. On note Σ^ω la clôture de $\Sigma \cup \{I_1\}$ par les constructeurs de types $\times$, et $\rightarrow$ (I_1 représente le type singleton).

– Un ensemble de symboles de fonctions et de prédicats typés par des éléments de Σ^ω. Cet ensemble contient en particulier un symbole d'égalité $=_\sigma$ pour chaque type σ de Σ^ω.

L'ensemble des termes de chaque type est défini récursivement à partir des symboles de fonctions, de la constante $\mathbf{e} : I_1$, en utilisant les règles suivantes (présentées dans le style de la déduction naturelle).

$$\frac{x:\sigma \quad y:\tau}{\langle x,y\rangle : \sigma\times\tau} \qquad \frac{z:\sigma\times\tau}{\pi(z):\sigma} \quad \frac{z:\sigma\times\tau}{\pi'(z):\tau} \qquad \frac{\begin{matrix}[x:\sigma]\\ \vdots \\ t:\tau\end{matrix}}{\lambda x:\sigma.t:\sigma\rightarrow\tau} \qquad \frac{f:\sigma\rightarrow\tau \quad a:\sigma}{f\ a:\tau}$$

$\star$ **Formules**

L'ensemble des formules est défini récursivement :

– les formules atomiques sont des formules ;

– Si A et B sont des formules alors $A \wedge B$, $A \vee B$, $A \Rightarrow B$ sont aussi des formules ;

– Si x est une variable de type σ et A une formule, alors $\exists x : \sigma.A$ et $\forall x : \sigma.A$ sont aussi des formules (dans lesquelles la variable x est liée).

⋆ **Règles de l'égalité**

– L'axiome de réflexivité :

$$x =_\sigma x$$

– La règle de substitution :

$$\frac{u =_\sigma v \quad \phi[v/x]}{\phi[u/x]}$$

On peut alors déduire les axiomes de Leibniz et les règles de passage au contexte.

– Les équations de conversion habituelles du λ-calcul simplement typé avec type produit :

$$\begin{gathered}(\lambda x : \sigma.t)\ u = t[u/x] \\ \pi(\langle x, y\rangle) = x \\ \pi'(\langle x, y\rangle) = y\end{gathered}$$

⋆ **Règles des connecteurs et quantificateurs**

$$\frac{A \qquad B}{A \wedge B} \qquad \frac{A \wedge B}{A} \qquad \frac{A \wedge B}{B} \qquad \frac{\begin{matrix}[A] \\ \vdots \\ B\end{matrix}}{A \Rightarrow B} \qquad \frac{A \Rightarrow B \qquad A}{B}$$

$$\frac{t : \sigma \qquad B[t/x]}{\exists x : \sigma.B} \qquad \frac{\exists x : \sigma.B \qquad \begin{matrix}[B] \\ \vdots \\ C\end{matrix}}{C} \qquad \frac{\begin{matrix}[x : \sigma] \\ \vdots \\ t \vdash B\end{matrix}}{\forall x : \sigma.B} \qquad \frac{\forall x : \sigma.B \qquad a : \sigma}{B[a/x]}$$

Notation. Nous appelons IQC$^\omega$ ce système (la prouvabilité y sera simplement notée $\vdash_I$).

3.2 Exemple : la théorie HA$^\omega$

Le langage du système HA$^\omega$ contient un type de base N, le symbole 0 de type N, le symbole S de type $N \to N$ et une infinité de symboles $\mathbf{rec}_\sigma$ de type $(\sigma \times (N \to \sigma \to \sigma)) \to N \to \sigma$ pour chaque type σ. La théorie HA^ω contient les axiomes égalitaires qui définissent ces constantes,

$$\begin{array}{rcl}(\mathbf{rec}_\sigma\ h\ 0) & =_\sigma & \pi(h) \\ (\mathbf{rec}_\sigma\ h\ Sn) & =_\sigma & (\pi'(h)\ n\ (\mathbf{rec}_\sigma\ h\ n))\end{array}$$

et le schéma de récurrence $Rec(\phi)$,

$$\phi(0) \wedge \forall n : N.(\phi(n) \Rightarrow \phi(Sn)) \Rightarrow \forall n : N.\phi(n)$$

Remarque. En supposant $\perp$ défini par $0 = 1$, la règle du faux est dérivable par récurrence (cf. [65] vol. II, p. 592, dans le cadre de ML^i). Si la négation $\neg A$ est définie par $A \Rightarrow \perp$ alors le quatrième axiome de Péano ($\neg 0 =_N 1$) devient trivial. De plus, on peut définir la disjonction par :

$$A \vee B \equiv \exists n : N((n = 0 \Rightarrow A) \wedge (\neg(n = 0) \Rightarrow B))$$

3.3 Définition de la réalisabilité modifiée

⋆ Contenu calculatoire d'une formule

Nous commençons donc par associer à toute formule un type, qui dénotera l'espace où trouver une fonction (totale) susceptible de réaliser cette formule. La présence des types singleton, produit cartésien et de la flèche permet de donner une définition simple du type des réalisations.

- $A^\star \equiv I_1$, si A est atomique,
- $(A \Rightarrow B)^\star \equiv A^\star \to B^\star$
- $(A \wedge B)^\star \equiv A^\star \times B^\star$
- $(\forall x : \sigma.A)^\star \equiv \sigma \to A^\star$
- $(\exists x : \sigma.A)^\star \equiv \sigma \times A^\star$

Remarque. Si l'on interprète le type singleton par le singleton (au sens ensembliste), le type produit par le produit cartésien ensembliste et la flèche par « l'espace des fonction », on vérifie facilement que le contenu calculatoire d'une formule sans quantificateurs existentiel est toujours interprété par le singleton.

⋆ Définition

L'interprétation d'une formule A est une formule $f\ mr\ A$ où f est une nouvelle variable libre de type $A^\star$. Nous donnons la définition habituelle (cf. [64] p .218, dans le cadre de HA^ω, par exemple). On rappelle que $f\ mr\ A$ est une *formule* où f est une nouvelle variable n'apparaissant pas dans A.

1. $f\ mr\ A \equiv A$, si A est atomique
2. $f\ mr\ A \Rightarrow B \equiv \forall g : A^\star.(g\ mr\ A \Rightarrow (f\ g)\ mr\ B)$
3. $f\ mr\ A \wedge B \equiv \pi(f)\ mr\ A \wedge \pi'(f)\ mr\ B$
4. $f\ mr\ \forall x : \sigma.B \equiv \forall x : \sigma.(f\ x\ mr\ B)$
5. $f\ mr\ \exists x : \sigma.B \equiv \pi'(f)\ mr\ B[\pi(f)/x]$

Lemme 3.3.1 *L'interprétation $f\ mr\ A$ d'une formule A est une formule de Harrop.*

⋆ **Validité**

Il est bien connu que toute formule prouvable est prouvablement réalisable et qu'en utilisant l'isomorphisme de Curry-Howard on peut extraire une réalisation. Nous en donnerons une preuve détaillée en section 6.

Proposition 3.3.2 *Si A est une formule prouvable dans* IQC^ω *sous les hypothèses $H_1, \ldots, H_n$, alors il existe un λ-terme $t(x_1, \ldots, x_n)$, où $x_1, \ldots, x_n$ sont des variables de type $H_1^\star, \ldots, H_n^\star$, tel que t mr A soit prouvable sous les hypothèses x_1 mr $H_1, \ldots, x_n$ mr H_n.*

Preuve. Cf. corollaire 6.1.2. □

4 Théorie réalisée et extraction de programmes

Définition 4.0.3 *Une formule A est dite **réalisée** dans une théorie Γ s'il existe un terme clos $t : A^\star$ tel que $\Gamma \vdash_I t \; mr \; A$; elle est dite **réalisable** si $\Gamma \vdash_I \exists x : A^\star.(x \; mr \; A)$.*

Définition 4.0.4 *Une théorie Γ est **réalisée** (resp. réalisable) si et seulement si tous ses axiomes sont réalisés (resp. réalisables) dans Γ.*

Théorème 4.0.5 (validité) *Dans une théorie réalisée Γ, toute formule A prouvable est réalisée.*

Preuve. Par la proposition 3.3.2, si $H_1, \ldots, H_n$ sont les axiomes de Γ qui apparaissent dans la preuve de A, il existe un terme $t(x_1, \ldots, x_n)$, où $x_1, \ldots, x_n$ sont des variables de type $H_1^\star, \ldots, H_n^\star$, tel que $t \; mr \; A$ soit prouvable sous les hypothèses $x_1 \; mr \; H_1, \ldots, x_n \; mr \; H_n$. Or, puisque Γ est une théorie réalisée, il existe $t_1, \ldots, t_n$ de types respectifs $H_1^\star, \ldots, H_n^\star$ tels que $\Gamma \vdash_I t_i \; mr \; H_i$. On en déduit alors que $\Gamma \vdash t[t_1/x_1, \ldots t_n/x_n] \; mr \; A$. □

Remarque. La notion de théorie réalisée permet de dissocier la possibilité d'extraire un terme à partir d'une preuve des propriétés calculatoires du langage des termes. Par exemple, nous allons montrer que la théorie HA^ω est réalisée, mais ce résultat est indépendant des propriétés du λ-calcul avec récurseur.

4.1 Exemple : la théorie HA^ω

Pour montrer que la théorie HA^ω est réalisée, il suffit de montrer que le schéma de récurrence est réalisée (en effet, tous les autres axiomes sont équationnels, et les formules de réalisabilité associées coïncident donc avec ces axiomes). Évidemment, le schéma de récurrence est réalisée par le récurseur ; prouvons donc que $\mathrm{HA}^\omega \vdash \mathbf{rec}_{\varphi^*} \; mr \; Rec(\varphi)$. Pour cela, considérons l'instance du schéma de récurrence $Rec((\mathbf{rec}_{\varphi^*} \; h \; n) \; mr \; \varphi(n))$:

$$\begin{aligned}\mathrm{HA}^\omega \quad \vdash (\mathbf{rec}_{\varphi^*} \; h \; 0) \; mr \; \varphi(0) \wedge \forall n : N.((\mathbf{rec}_{\varphi^*} \; h \; n) \; mr \; \varphi(n) \Rightarrow (\mathbf{rec}_{\varphi^*} \; h \; Sn) \; mr \; \varphi(Sn)) \\ \Rightarrow \forall n : N(\mathbf{rec}_{\varphi^*} \; h \; n) \; mr \; \varphi(n)\end{aligned}$$

Or, $\mathrm{HA}^\omega \vdash (\mathbf{rec}_{\varphi^*} \; h \; 0) =_{\varphi^*} \pi(h)$ et $\mathrm{HA}^\omega \vdash (\mathbf{rec}_{\varphi^*} \; h \; Sn) =_{\varphi^*} (\pi'(h) \; n \; (\mathbf{rec}_{\varphi^*} \; h \; n))$, d'où :

$$\begin{aligned}\mathrm{HA}^\omega \quad \vdash \pi(h) \; mr \; \varphi(0) \wedge \forall n : N.((\mathbf{rec}_{\varphi^*} \; h \; n) \; mr \; \varphi(n) \Rightarrow \pi'(h) \; n \; (\mathbf{rec}_{\varphi^*} \; h \; n) \; mr \; \varphi(Sn)) \\ \Rightarrow \forall n : N.(\mathbf{rec}_{\varphi^*} \; h \; n) \; mr \; \varphi(n)\end{aligned}$$

On peut alors généraliser la seconde partie de l'hypothèse en remplaçant :

$$(\mathbf{rec}_{\varphi^*}\ h\ n)\ mr\ \varphi(n) \Rightarrow (\pi'(h)\ n\ (\mathbf{rec}_{\varphi^*}\ h\ n))\ mr\ \varphi(Sn)$$

par :

$$\forall j : \varphi^*(j\ mr\ \varphi(n) \Rightarrow (\pi'(h)\ n\ j)\ mr\ \varphi(Sn))$$

et obtenir ainsi le théorème :

$$\begin{aligned}\mathrm{HA}^\omega\ \ &\vdash \pi(h)\ mr\ \varphi(0) \wedge \forall n : N.\forall j : \varphi^*.(j\ mr\ \varphi(n) \Rightarrow (\pi'(h)\ n\ j)\ mr\ \varphi(Sn))\\ &\qquad\qquad \Rightarrow \forall n : N.(\mathbf{rec}_{\varphi^*}\ h\ n)\ mr\ \varphi(n)\end{aligned}$$

On en déduit alors en utilisant les points (2), (4) puis (3) de la définition de la mr-réalisabilité :

$$\begin{aligned}\mathrm{HA}^\omega\ \ &\vdash \pi(h)\ mr\ \varphi(0) \wedge \forall n : N.(\pi'(h)\ n\ mr\ \varphi(n) \Rightarrow \varphi(Sn)) \Rightarrow \forall n : N.(\mathbf{rec}_{\varphi^*}\ h\ n)\ mr\ \varphi(n)\\ &\equiv \pi(h)\ mr\ \varphi(0) \wedge \pi'(h)\ mr\ \forall n : N.(\varphi(n) \Rightarrow \varphi(Sn)) \Rightarrow (\mathbf{rec}_{\varphi^*}\ h)\ mr\ \forall n^N \varphi(n)\\ &\equiv h\ mr\ (\varphi(0) \wedge \forall n : N.(\varphi(n) \Rightarrow \varphi(Sn))) \Rightarrow (\mathbf{rec}_{\varphi^*}\ h)\ mr\ \forall n^N \varphi(n)\end{aligned}$$

D'où,

$$\begin{aligned}\mathrm{HA}^\omega\ \ &\vdash \forall h : \varphi^\star \times (N \to \varphi^\star \to \varphi^\star).(h\ mr\ (\varphi(0) \wedge \forall n : N.(\varphi(n) \Rightarrow \varphi(Sn)))\\ &\qquad\qquad \Rightarrow (\mathbf{rec}_{\varphi^*}\ h)\ mr\ \forall n^N \varphi(n))\end{aligned}$$

Et, par conséquent en utilisant le point (4),

$$\begin{aligned}\mathrm{HA}^\omega\ \ &\vdash \mathbf{rec}_{\varphi^*}\ mr\ \varphi(0) \wedge \forall n : N.(\varphi(n) \Rightarrow \varphi(Sn)) \Rightarrow\ \forall n : N.\varphi(n)\\ &\equiv \mathbf{rec}_{\varphi^*}\ mr\ Rec(\varphi)\end{aligned}$$

5 Correction de l'extraction

On montre ici que dans toute théorie réalisée Γ le programme extrait d'une preuve est « correct » (*i.e.* il réalise sa spécification). De plus, la preuve de correction est dérivable dans $\Gamma + \mathrm{AC} + \mathrm{IP}$. Nous montrons tout d'abord que les axiomes AC et IP sont réalisés.

5.1 Le schéma d'axiomes du choix AC

Proposition 5.1.1 *Le schéma d'axiomes du choix :*

$$\forall x : \sigma.\exists y : \tau.A(x,y) \Rightarrow \exists f : \sigma \to \tau.\forall x : \sigma.A(x, f\ x)$$

est réalisé par le λ-terme $t_{\mathrm{AC}}^{\sigma,\tau} \equiv \lambda p : (\forall x : \sigma.\exists y : \tau.A)^\star.\langle \lambda x : \sigma.\pi(p\ x), \lambda x : \sigma.\pi'(p\ x)\rangle$.

Preuve. En effet, par définition :

$$\begin{aligned}&t_{\mathrm{AC}}^{\sigma,\tau}\ mr\ [\forall x : \sigma.\exists y : \tau.A(x,y) \Rightarrow \exists f : \sigma \to \tau.\forall x : \sigma.A(x, f\ x)]\\ &\equiv \forall p : (\forall x : \sigma.\exists y : \tau.A)^\star.[p\ mr\ \forall x : \sigma.\exists y : \tau.A(x,y)] \Rightarrow [(t_{\mathrm{AC}}^{\sigma,\tau}\ p)\ mr\ \exists f : \sigma \to \tau.\forall x : \sigma.A(x, f\ x)]\\ &\equiv \forall p : (\forall x : \sigma.\exists y : \tau.A)^\star.[p\ mr\ \forall x : \sigma.\exists y : \tau.A(x,y)] \Rightarrow [\pi'(t_{\mathrm{AC}}^{\sigma,\tau}\ p)\ mr\ \forall x : \sigma.A(x, (\pi(t_{\mathrm{AC}}^{\sigma,\tau}\ p)\ x))]\end{aligned}$$

De plus,

$$p\ mr\ \forall x : \sigma.\exists y : \tau.A(x,y) \equiv \forall x : \sigma.(p\ x)\ mr\ \exists y : \tau.A(x,y) \equiv \forall x : \sigma.\pi'(p\ x)\ mr\ A(x, \pi(p\ x))$$

et aussi,

$$\pi'(t_{\mathrm{AC}}^{\sigma,\tau}\ p)\ mr\ \forall x : \sigma.A(x, (\pi(t_{\mathrm{AC}}^{\sigma,\tau}\ p)\ x)) \equiv \forall x : \sigma.(\pi'(t_{\mathrm{AC}}^{\sigma,\tau}\ p)\ x)\ mr\ A(x, (\pi(t_{\mathrm{AC}}^{\sigma,\tau}\ p)\ x))$$

Or on peut dériver les égalités suivantes $(\pi'(t_{\mathrm{AC}}^{\sigma,\tau}\ p)\ x) = \pi'(p\ x)$ et $(\pi(t_{\mathrm{AC}}^{\sigma,\tau}\ p)\ x) = \pi(p\ x)$, et en appliquant la règle de substitution on arrive à une tautologie de la forme $\forall p(\varphi \Rightarrow \varphi)$. $\square$

5.2 L'axiome d'indépendance des prémisses

Pour montrer que le schéma d'indépendance des prémisses est réalisé, nous aurons besoin de la notion de formule pré-réalisée.

⋆ Formules pré-réalisées

Une formule pré-réalisée est une formule dont on connaît une « réalisation potentielle ». Lorsqu'une formule pré-réalisée est réalisée (éventuellement sous des hypothèses), elle l'est en particulier par sa pré-réalisation.

Définition 5.2.1 *Une formule A est dite* ***pré-réalisée*** *s'il existe un terme de type $A^\star$ ayant les mêmes variables libres que A, noté $pr(A)$ et appelé une pré-réalisation de A, tel que pour tout terme t de type $A^\star$:*

$$\vdash_I t \ mr \ A \Rightarrow pr(A) \ mr \ A$$

Lemme 5.2.2 *Les formules de Harrop sont pré-réalisées.*

Preuve. Par récurrence sur la définition des formules de Harrop : □

- Une formule atomique A est pré-réalisée par $pr(A) = \mathbf{e}$ (où $\mathbf{e}$ est la constante de type I_1 de IQC$^\omega$) :

  $$\vdash_I t \ mr \ A \Rightarrow \mathbf{e} \ mr \ A$$

  puisque $t \ mr \ A \equiv \mathbf{e} \ mr \ A \equiv A$.

- Une conjonction $A \wedge B$, où A et B sont des formules de Harrop, est pré-réalisée par $\langle pr(A), pr(B)\rangle$:

  $$\vdash_I t \ mr \ A \wedge B \Rightarrow \langle pr(A), pr(B)\rangle \ mr \ A \wedge B$$

  En effet, car :

  $$\begin{array}{rcl} \vdash_I \ \ t \ mr \ A \wedge B & \equiv & (\pi(t) \ mr \ A) \wedge (\pi'(t) \ mr \ B) \\ & \Rightarrow & (pr(A) \ mr \ A) \wedge (pr(B) \ mr \ B) \end{array}$$

  or $\langle pr(A), pr(B)\rangle \ mr \ A \wedge B \equiv (\pi\langle pr(A), pr(B)\rangle \ mr \ A) \wedge (\pi'\langle pr(A), pr(B)\rangle \ mr \ B)$ et les équations $\pi\langle pr(A), pr(B)\rangle = pr(A)$ et $\pi'\langle pr(A), pr(B)\rangle = pr(B)$ nous donnent le résultat.

- L'implication $A \Rightarrow B$, où B est une formule de Harrop, est pré-réalisée par $\lambda x : A^\star.pr(B)$:

  $$\vdash_I t \ mr \ (A \Rightarrow B) \Rightarrow \lambda x : A^\star.pr(B) \ mr \ (A \Rightarrow B)$$

  En effet, car :

  $$\begin{array}{rcl} \vdash_I \ \ t \ mr \ A \Rightarrow B & \equiv & \forall y : A^\star.(y \ mr \ A \Rightarrow (t \ y) \ mr \ B) \\ & \Rightarrow & \forall y : A^\star.(y \ mr \ A \Rightarrow pr(B) \ mr \ B) \end{array}$$

  Or $\lambda x : A^\star.pr(B) \ mr \ A \Rightarrow B \equiv \forall y : A^\star.(y \ mr \ A \Rightarrow (\lambda x : A^\star.pr(B) \ y) \ mr \ B)$ et l'équation $(\lambda x : A^\star.pr(B) \ y) = pr(B)$ nous donne le résultat.

– La quantification universelle $\forall x:\sigma.A$, où A est une formule de Harrop, est pré-réalisée par $\lambda x:\sigma.pr(A)$:

$$\vdash_I t\ mr\ \forall x:\sigma.A \Rightarrow \lambda x:\sigma.pr(A)\ mr\ \forall x:\sigma.A$$

En effet, car :

$$\begin{array}{rcl} \vdash_I\ \ t\ mr\ \forall x:\sigma.A & \equiv & \forall x:\sigma.(t\ x)\ mr\ A \\ & \Rightarrow & \forall x:\sigma.pr(A)\ mr\ A \end{array}$$

Or $(\lambda x:\sigma.pr(A))\ mr\ \forall x:\sigma.A \equiv \forall x:\sigma.(\lambda x:\sigma.pr(A)\ x)\ mr\ A)$ et l'équation $(\lambda x:\sigma.pr(A)\ x) = pr(A)$ nous donne le résultat.

$\star$ **Le schéma d'indépendance des prémisses** IP

Proposition 5.2.3 *Le schéma d'indépendance des prémisses (où H est une formule de Harrop) :*

$$(H \Rightarrow \exists y:\sigma.B(y)) \Rightarrow \exists y:\sigma.(H \Rightarrow B(y))$$

est réalisé par : $t^{\sigma}_{\mathrm{IP}} \equiv \lambda f:(H \Rightarrow \exists y:\sigma.B(y))^{\star}.\langle \pi(f\ pr(H)), \lambda x:H^{\star}.\pi'(f\ pr(H))\rangle$.

Preuve. En effet, par définition,

$$\begin{array}{l} t^{\sigma}_{\mathrm{IP}}\ mr\ (H \Rightarrow \exists y B(y)) \Rightarrow \exists y(H \Rightarrow B(y)) \\ \equiv \forall f:(H \Rightarrow \exists y:\sigma.B(y))^{\star}.(f\ mr\ (H \Rightarrow \exists y B(y)) \Rightarrow (t_{\mathrm{IP}}\ f)\ mr\ \exists y(H \Rightarrow B(y)) \end{array}$$

De plus,

$$\begin{array}{rcl} f\ mr\ (H \Rightarrow \exists y B(y)) & \equiv & \forall x:H^{\star}.(x\ mr\ H) \Rightarrow ((f\ x)\ mr\ (\exists y B(y)) \\ & \equiv & \forall x:H^{\star}.(x\ mr\ H) \Rightarrow (\pi'(f\ x)\ mr\ (B(\pi(f\ x))) \end{array}$$

Et comme $\vdash_I (x\ mr\ H) \Rightarrow (pr(H)\ mr\ H)$ (lemme 5.2.2) on en déduit :

$$\begin{array}{rcl} f\ mr\ (H \Rightarrow \exists y B(y)) & \vdash_I & \forall x:H^{\star}.(x\ mr\ H) \Rightarrow \pi'(f\ pr(H))\ mr\ (B(\pi(f\ pr(H)))) \\ & \vdash_I & \lambda x:H^{\star}.\pi'(f\ pr(H))\ mr\ (H \Rightarrow B(\pi(f\ pr(H)))) \end{array}$$

Par ailleurs,

$$(t^{\alpha}_{\mathrm{IP}}\ f)\ mr\ \exists y(H \Rightarrow B(y)) \equiv \pi'(t^{\sigma}_{\mathrm{IP}}\ f)\ mr\ (H \Rightarrow B(\pi(t^{\sigma}_{\mathrm{IP}}\ f)))$$

Or les équations $\pi(t^{\sigma}_{\mathrm{IP}}\ f) = \pi(f\ pr(H))$ et $\pi'(t^{\sigma}_{\mathrm{IP}}\ f) = \lambda x:H^{\star}.\pi'(f\ pr(H))$ sont dérivables, et en appliquant la règle de substitution, on obtient une tautologie de la forme $\forall f(\varphi \Rightarrow \varphi)$. □

Lemme 5.2.4 *Pour toute formule A, on a* $\mathrm{AC}+\mathrm{IP} \vdash_I \exists f:A^{\star}.f\ mr\ A \Leftrightarrow A$

Preuve. Par récurrence sur la formule :

– $\exists f:I_1.(f\ mr\ A) \Leftrightarrow A$, si A est atomique (par définition, $f\ mr\ A \equiv A$)

– $\exists f : (A \Rightarrow B)^\star.(f\ mr\ A \Rightarrow B)$

$\Leftrightarrow \exists f : (A \Rightarrow B)^\star.\forall g : A^\star.(g\ mr\ A) \Rightarrow ((f\ g)\ mr\ B)$

$\Leftrightarrow \forall g : A^\star.\exists h : B^\star.((g\ mr\ A) \Rightarrow (h\ mr\ B))$ en appliquant AC pour le sens $\Leftarrow$

$\Leftrightarrow \forall g : A^\star.((g\ mr\ A) \Rightarrow \exists h : B^\star.(h\ mr\ B))$ en appliquant IP pour le sens $\Leftarrow$, car toute formule u $mr\ \varphi$ est de Harrop

$\Leftrightarrow (\exists g : A^\star.(g\ mr\ A)) \Rightarrow (\exists h : B^\star.(h\ mr\ B))$ car g n'apparaît pas dans le second membre de l'implication

$\Leftrightarrow (A \Rightarrow B)$

– $\exists f : (A \wedge B)^\star.(f\ mr\ A \wedge B))$

$\Leftrightarrow \exists f : (A \wedge B)^\star.((\pi(f)\ mr\ A) \wedge (\pi'(f)\ mr\ B))$

$\Leftrightarrow \exists f_1 : A^\star.(f_1\ mr\ A) \wedge \exists f_2 : B^\star.(f_2\ mr\ B)$

$\Leftrightarrow A \wedge B$

– $\exists f : (\forall x : \sigma.B)^\star.(f\ mr\ \forall x^\sigma.B)$

$\Leftrightarrow \exists f : (\forall x : \sigma.B)^\star.\forall x : \sigma.((f\ x)\ mr\ B)$

$\Leftrightarrow \forall x : \sigma.\exists d : B^\star.(d\ mr\ B)$ en appliquant AC pour le sens $\Leftarrow$

$\Leftrightarrow \forall x : \sigma.B$

– $\exists f : (\exists x : \sigma.B)^\star.(f\ mr\ \exists x : \sigma.B)$

$\Leftrightarrow \exists f : (\exists x : \sigma.B)^\star.(\pi'(f)\ mr\ B[\pi(f)/x])$

$\Leftrightarrow \exists e : \sigma.\exists d : (B[e/x])^\star.(d\ mr\ B[e/x])$

$\Leftrightarrow \exists e : \sigma.B[e/x]$

$\Leftrightarrow \exists x : \sigma.B$

□

Proposition 5.2.5 *Si* Γ *est une théorie réalisée alors pour toute formule* φ, $\Gamma \vdash_I \exists t : \varphi^\star.t\ mr\ \varphi$ *si et seulement si* $\Gamma + \mathrm{AC} + \mathrm{IP} \vdash_I \varphi$.

Preuve. Si $\Gamma + \mathrm{AC} + \mathrm{IP} \vdash \varphi$ on a $\Gamma \vdash (\mathrm{AC} \Rightarrow \mathrm{IP} \Rightarrow \varphi)$ et donc il existe un terme clos u tel que $\Gamma \vdash u\ mr\ (\mathrm{AC} \Rightarrow \mathrm{IP} \Rightarrow \varphi)$. On en déduit que $\Gamma \vdash (u\ t_{\mathrm{AC}}\ t_{\mathrm{IP}})\ mr\ \varphi$. Réciproquement, le lemme précédent montre que si $\Gamma \vdash_I \exists t : \varphi^\star.t\ mr\ \varphi$ alors $\Gamma + \mathrm{AC} + \mathrm{IP} \vdash \varphi$. □

Corollaire 5.2.6 *Si* Γ *est une théorie réalisée alors la propriété d'existence est vérifiée modulo* $\mathrm{AC} + \mathrm{IP}$, *c'est-à-dire* $\Gamma \vdash \exists x : \sigma.A$ *implique l'existence d'un terme* t *de type* σ *tel que* $\Gamma + \mathrm{AC} + \mathrm{IP} \vdash A[t/x]$.

Preuve. $\Gamma \vdash \exists x : \sigma.A$ implique l'existence d'un terme clos $t : (\exists x : \sigma.A)^\star$ tel que $\Gamma \vdash t\ mr\ \exists x : \sigma.A$, et par définition, $\Gamma \vdash \pi'(t)\ mr\ A(\pi(t))$, et $\Gamma + \mathrm{AC} + \mathrm{IP} \vdash A(\pi(t))$ par la proposition. □

6 Internalisation de la réalisabilité modifiée

Dans cette section, nous définissons une théorie des types dans laquelle toute formule *mr*-réalisable est prouvable. Ce système contient une règle d'élimination forte du $\exists$ dans le style de la théorie des types de P. Martin-Löf, qui permet de dériver l'axiome du choix.

6.1 Le système formel MR

Les termes, les types et les formules sont ceux de IQC$^\omega$.
Les jugements de ce système sont de trois formes :

1. les jugements de typage, notés $t : \sigma$, où σ est un type et t est un terme de type σ.
2. les jugements de prouvabilité (*i.e.* de réalisabilité), notés $t \vdash A$, où A une formule et t un terme de type $A^\star$.
3. les jugements d'égalité, notés $u =_\sigma v$, où u et v sont deux termes de type σ.

Les hypothèses sont de la forme $x \vdash A$ où x est une variable de type $A^\star$. De plus, une variable qui décore une hypothèse ne doit apparaître libre dans aucune hypothèse.
Les règles de typage sont celles de IQC$^\omega$ (cf. sous-section 3.1).

Règles des connecteurs

$$\frac{u \vdash A \qquad v \vdash B}{\langle u, v\rangle \vdash A \wedge B} \qquad\qquad \frac{t \vdash A \wedge B}{\pi(t) \vdash A} \quad \frac{t \vdash A \wedge B}{\pi'(t) \vdash B}$$

$$\frac{\begin{array}{c}[x \vdash A] \\ \vdots \\ t \vdash B\end{array}}{\lambda x : A^\star . t \vdash A \Rightarrow B} \qquad\qquad \frac{f \vdash A \Rightarrow B \qquad a \vdash A}{(f\ a) \vdash B}$$

où x n'apparaît pas libre dans B dans la règle d'introduction de l'implication.

Règles des quantificateurs

$$\frac{t : \sigma \qquad u \vdash B[t/x]}{\langle t, u\rangle \vdash \exists x : \sigma . B} \qquad\qquad \frac{t \vdash \exists x : \sigma . B}{\pi\,(t) : \sigma} \quad \frac{t \vdash \exists x : \sigma . B}{\pi'(t) \vdash B[\pi\,(t)/x]}$$

$$\frac{\begin{array}{c}[x : \sigma] \\ \vdots \\ t \vdash B\end{array}}{\lambda x : \sigma . t \vdash \forall x : \sigma . B} \qquad\qquad \frac{f \vdash \forall x : \sigma . B \qquad a : \sigma}{(f\ a) \vdash B[a/x]}$$

Règles de l'égalité

– L'axiome de réflexivité :

$$x =_\sigma x$$

– La règle de substitution :

$$\frac{u =_\sigma v \qquad t \vdash \phi[v/x]}{t \vdash \phi[u/x]}$$

– Les équations de conversion habituelles du λ-calcul simplement typé avec type produit :

$$\begin{aligned}(\lambda x : \sigma.t)\ u &= t[u/x]\\ \pi\langle x, y\rangle &= x\\ \pi'\langle x, y\rangle &= y\end{aligned}$$

Remarque. La règle d'élimination forte du $\exists$ est strictement plus forte que la règle habituelle donnée dans IQC^ω, puisqu'elle permet de prouver l'axiome du choix (cf. sous-section 6.2). La règle habituelle peut se dériver ainsi (où, bien sûr, x n'apparaît pas dans C) :

$$\frac{\dfrac{t \vdash \exists x : \sigma.B}{\pi'(t) \vdash [\pi(t)/x]} \qquad \dfrac{\begin{matrix}[x \vdash B]\\ \vdots\\ u \vdash C\end{matrix}}{\lambda x : B^\star.u : B \Rightarrow C}}{(\lambda x : B^\star.u\ \pi'(t)) : C}$$

Proposition 6.1.1 *Si $t \vdash A$ est dérivable dans* MR *sous les hypothèses $x_1 \vdash \Gamma_1, \ldots, x_n \vdash \Gamma_n$, alors t mr A est prouvable dans* IQC^ω *sous les hypothèses x_1 mr $\Gamma_1, \ldots, x_n$ mr Γ_n.*

Preuve. Par récurrence sur la dérivation de $t : A$. Si $\Gamma = x_1 : \Gamma_1, \ldots, x_n : \Gamma_n$ est un contexte d'hypothèses de MR, on note Γ^{mr} l'ensemble d'hypothèses de IQC^ω composé de $x_1\ mr\ \Gamma_1, \ldots, x_n\ mr\ \Gamma_n$.

– L'axiome : si $x_i \vdash \Gamma_I$ est une hypothèse de Γ, alors $x_i\ mr\ \Gamma_i$ est une hypothèse de Γ^{mr}.

– La règle d'introduction de la conjonction sous les hypothèses Γ,

$$\frac{u \vdash A \qquad v \vdash B}{\langle u, v\rangle \vdash A \wedge B}$$

Par hypothèse de récurrence, $u\ mr\ A$ et $v\ mr\ B$ sont dérivables dans IQC^ω sous les hypothèses Γ^{mr}. Par conséquent, $(u\ mr\ A)\wedge(v\ mr\ B)$ et donc $(\pi(\langle u, v\rangle)\ mr\ A)\wedge(\pi'(\langle u, v\rangle)\ mr\ B)$ sont dérivables dans IQC^ω et cette dernière formule est par définition $\langle u, v\rangle\ mr\ A \wedge B$.

– La règle d'élimination de la conjonction sous les hypothèses Γ, (nous ne traitons que la première projection),

$$\frac{t \vdash A \wedge B}{\pi(t) \vdash A}$$

Par hypothèse de récurrence, $t\ mr\ A \wedge B$ est dérivable dans IQC^ω sous les hypothèses Γ^{mr}. Par définition de la réalisabilité modifiée (point 3), cette formule est exactement $(\pi(t)\ mr\ A)\wedge(\pi'(t)\ mr\ B)$ et par conséquent $\pi(t)\ mr\ A$ est aussi dérivable dans IQC^ω.

– La règle d'introduction de l'implication sous les hypothèses Γ,

$$\dfrac{\begin{array}{c}[x \vdash A]\\ \vdots \\ t \vdash B\end{array}}{\lambda x : A^{\star}.t \vdash A \Rightarrow B}$$

Si $t \vdash B$ est dérivable sous les hypothèses $\Gamma, x \vdash A$, par hypothèse de récurrence, $t\ mr\ B$ est dérivable dans IQC$^{\omega}$ sous les hypothèses $\Gamma^{mr}, x\ mr\ A$. Par conséquent, $(x\ mr\ A) \Rightarrow (t\ mr\ B)$ et donc $(x\ mr\ A) \Rightarrow (((\lambda x : A^{\star}.t)\ x)\ mr\ B)$ sont dérivables dans IQC$^{\omega}$ sous les hypothèses Γ^{mr}. Finalement, puisque x n'apparaît pas dans Γ par hypothèse (nous rappelons que dans MR, une variable décorant une hypothèse ne doit apparaître libre dans aucune hypothèse), x n'apparaît pas non plus dans Γ^{mr}. Donc $\forall x : A^{\star}.(x\ mr\ A \Rightarrow ((\lambda x : A^{\star}.t)\ x)\ mr\ B)$ qui est par définition $\lambda x : A^{\star}.t\ mr\ (A \Rightarrow B)$ est aussi dérivable dans IQC$^{\omega}$ sous les hypothèses Γ^{mr}.

– La règle d'élimination de l'implication sous les hypothèses Γ,

$$\dfrac{f \vdash A \Rightarrow B \qquad a \vdash A}{(f\ a) \vdash B}$$

Par hypothèse de récurrence, $f\ mr\ (A \Rightarrow B)$ et $a\ mr\ A$ sont dérivables dans IQC$^{\omega}$ sous les hypothèses Γ^{mr}. Par définition, $f\ mr\ (A \Rightarrow B) \equiv \forall x : A^{\star}.(x\ mr\ A \Rightarrow (f\ x)\ mr\ B)$ d'où $a\ mr\ A \Rightarrow (f\ a)\ mr\ B$ et par conséquent $(f\ a)\ mr\ B$ sont aussi dérivables.

– La règle d'introduction du quantificateur existentiel sous les hypothèses Γ,

$$\dfrac{t : \sigma \qquad u \vdash B[t/x]}{\langle t, u\rangle \vdash \exists x : \sigma.B}$$

Par hypothèse de récurrence, $u\ mr\ B[t/x]$ est dérivable dans IQC$^{\omega}$ sous les hypothèses Γ^{mr}. Par conséquent $\pi'\langle t, u\rangle\ mr\ B[\pi\langle t, u\rangle/x]$ est aussi dérivable, or cette formule est justement $\langle t, u\rangle\ mr\ \exists x : \sigma.B$.

– La règle d'élimination du quantificateur existentiel sous les hypothèses Γ,

$$\dfrac{t \vdash \exists x : \sigma.B}{\pi'(t) \vdash B[\pi\,(t)/x]}$$

Par hypothèse de récurrence, $t\ mr\ \exists x : \sigma.B$ est dérivable dans IQC$^{\omega}$ sous les hypothèses Γ^{mr}. Par le point (5) de la définition de la réalisabilité modifiée, cette formule est justement $\pi'(t)\ mr\ B[\pi\,(t)/x]$.

– La règle d'introduction du quantificateur universel sous les hypothèses Γ,

$$\dfrac{\begin{array}{c}[x : \sigma]\\ \vdots \\ t \vdash B\end{array}}{\lambda x : \sigma.t \vdash \forall x : \sigma.B}$$

Par hypothèse de récurrence, $t\ mr\ B$ est dérivable dans IQC$^{\omega}$ sous les hypothèses Γ^{mr}. Par conséquent, $\forall x : \sigma.(t\ mr\ B)$ et donc $\forall x : \sigma.(\lambda x : \sigma.t\ x)\ mr\ B$ sont aussi dérivables. Or cette dernière formule est par définition justement $\lambda x : \sigma.t\ mr\ \forall x : \sigma.B$.

– La règle d'élimination du quantificateur universel sous les hypothèses Γ,

$$\frac{f \vdash \forall x : \sigma.B \qquad a : \sigma}{(f\ a) \vdash B[a/x]}$$

Par hypothèse de récurrence, $f\ mr\ \forall x : \sigma.B$ est dérivable dans IQC^ω sous les hypothèses Γ^{mr}. Par le point (4) de la définition de la réalisabilité modifiée, cette formule est exactement $\forall x : \sigma.((f\ x)\ mr\ B)$ et donc en particulier $(f\ a)\ mr\ B[a/x]$ est aussi dérivable.

– La règle de la substitution sous les hypothèses Γ :

$$\frac{u =_\sigma v \qquad t \vdash \phi[v/x]}{t \vdash \phi[u/x]}$$

Par hypothèse de récurrence, $t\ mr\ \phi[v/x]$ est dérivable dans IQC^ω sous les hypothèses Γ^{mr}. Par conséquent, $t\ mr\ \phi[u/x]$ est aussi dérivable.

– Les axiomes égalitaires sont les mêmes dans IQC^ω et MR.

$\square$

Corollaire 6.1.2 *Si A est une formule prouvable dans* IQC^ω *sous les hypothèses $H_1, \dots, H_n$, alors il existe un λ-terme $t(x_1, \dots, x_n)$, où $x_1, \dots, x_n$ sont des variables de type $H_1^\star, \dots, H_n^\star$, tel que $t\ mr\ A$ soit prouvable sous les hypothèses $x_1\ mr\ H_1, \dots, x_n\ mr\ H_n$.*

Preuve. Le calcul des prédicats IQC^ω est plongeable dans MR puisque nous avons vu que la règle d'élimination habituelle du $\exists$ (qui est la seule qui diffère entre les deux systèmes) est dérivable dans MR. Une preuve de IQC^ω plongée dans MR est alors décorée par un λ-terme qui vérifie la propriété du corollaire (par le théorème précédent). $\square$

6.2 Preuve du schéma d'axiomes du choix

La présence de la règle d'élimination forte du $\exists$ permet de prouver l'axiome du choix. Remarquons que le λ-terme qui décore la conclusion de cette preuve est exactement $t_{\mathrm{AC}}^{\sigma,\tau}$. Cette preuve est aussi la traduction de la preuve de l'axiome du choix dans ML.

Proposition 6.2.1 *Dans* MR*, on peut dériver $t_{\mathrm{AC}}^{\sigma,\tau} \vdash AC$*

Preuve.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[p \vdash \forall x : \sigma.\exists y : \tau.A]^1 \quad [x : \sigma]^2}{(p\ x) \vdash \exists y : \tau.A}}{\pi(p\ x) : \tau}}{\lambda x : \sigma.\pi(p\ x) : \sigma \to \tau}\,(2) \qquad \dfrac{(\lambda x : \sigma.\pi(p\ x)\ x) = \pi(p\ x) \quad \dfrac{\dfrac{\dfrac{[p \vdash \forall x : \sigma.\exists y : \tau.A]^1 \quad [x : \sigma]^3}{(p\ x) \vdash \exists y : \tau.A}}{\pi'(p\ x) \vdash A[\pi(p\ x)/y]}}{\lambda x : \sigma.\pi'(p\ x) \vdash \forall x : \sigma.A[\pi(p\ x)/y]}\,(3)}{\lambda x : \sigma.\pi'(p\ x) \vdash \forall x : \sigma.A[(\lambda x : \sigma.\pi(p\ x)\ x)/y]}}{\langle \lambda x : \sigma.\pi(p\ x), \lambda x : \sigma.\pi'(p\ x) \rangle \vdash \exists f : \sigma \to \tau.\forall x : \sigma.A[(f\ x)/y]}}{\lambda p : (\forall x : \sigma.\exists y : \tau.A)^\star.\langle \lambda x : \sigma.\pi(p\ x), \lambda x : \sigma.\pi'(p\ x) \rangle \vdash \forall x : \sigma.\exists y : \tau.A \Rightarrow \exists f : \sigma \to \tau.\forall x : \sigma.A[(f\ x)/y]}\,(1)$$

$\square$

6.3 Prouver le schéma d'indépendance des prémisses

Pour prouver le schéma d'indépendance des prémisses, nous nous donnons tout d'abord une règle qui exprime que nous n'allons pas considérer le « contenu calculatoire » des formules atomiques, de même que dans la réalisabilité. Pour une discussion sur le sens de cette règle, cf. [11], p. 268. Remarquons tout de même qu'ici les formules atomiques ne sont pas nécessairement décidables.

Règle des formules atomiques $\mathrm{R}_{\mathbf{e}}$

$$\frac{t \vdash A}{\mathbf{e} \vdash A} \quad \text{où } A \text{ est atomique}$$

Notation. On appelle $\mathrm{MR}_{\mathbf{e}}$ le système $\mathrm{MR} + \mathrm{R}_{\mathbf{e}}$.

Proposition 6.3.1 *La proposition* 6.1.1 *est encore vérifiée dans* $\mathrm{MR}_{\mathbf{e}}$.

Preuve. Il suffit d'étendre la preuve par récurrence à cette nouvelle règle. En effet, par hypothèse de récurrence, $t\ mr\ A$ est dérivable dans IQC^{ω} sous les hypothèses Γ^{mr}. Or, par définition, $t\ mr\ A \equiv A \equiv \mathbf{e}\ mr\ A$ d'où $\mathbf{e}\ mr\ A$ est aussi dérivable. □

⋆ Notion de formule « pré-prouvée »

La règle des formules atomiques $\mathrm{R}_{\mathbf{e}}$ dit que ces formules sont « pré-prouvées » (au sens de pré-réalisées). De même que pour la réalisabilité, cette propriété s'étend aux formules de Harrop.

Définition 6.3.2 *Une formule A est dite pré-prouvée s'il existe un λ-terme $pr(A)$ ayant les mêmes variables libres que t tel que pour tout t, si $t \vdash A$ est dérivable dans* $\mathrm{MR}_{\mathbf{e}}$ *sous les hypothèses Γ alors $pr(A) \vdash A$ l'est aussi.*

Proposition 6.3.3 *Les formules de Harrop sont pré-prouvées.*

Preuve. On montre par récurrence sur la formule A que pour tout t, si $t \vdash A$ est dérivable sous les hypothèses Γ alors $pr(A) \vdash A$ l'est aussi, où $pr(A)$ a été défini dans la preuve du lemme 5.2.2. On note Π_A cette dérivation.

– Si A est atomique, si $t \vdash A$ est dérivable sous les hypothèses Γ alors on complète la dérivation en appliquant la règle des formules atomiques :

$$\frac{t \vdash A}{\mathbf{e} \vdash A}$$

– Si $t \vdash A \wedge B$ est dérivable sous les hypothèses Γ alors on construit la dérivation suivante :

$$\frac{\begin{array}{c} \dfrac{t \vdash A \wedge B}{\pi(t) \vdash A} \\ \cdots \\ \Pi_A \\ \cdots \\ pr(A) \vdash A \end{array} \qquad \begin{array}{c} \dfrac{t \vdash A \wedge B}{\pi'(t) \vdash B} \\ \cdots \\ \Pi_B \\ \cdots \\ pr(B) \vdash B \end{array}}{\langle pr(A),\ pr(B)\rangle \vdash A \wedge B}$$

– Si $t \vdash A \Rightarrow B$ est dérivable sous les hypothèses Γ alors on construit la dérivation suivante :

$$\dfrac{\dfrac{t \vdash A \Rightarrow B \qquad [x \vdash A]}{(t\ x) \vdash B} \quad \begin{matrix}\cdots \\ \Pi_B \\ \cdots\end{matrix} \quad pr(B) \vdash B}{\lambda x : A^\star.\ pr(B) \vdash A \Rightarrow B}$$

– Si $t \vdash \forall x : \sigma.B$ est dérivable sous les hypothèses Γ alors on construit la dérivation suivante :

$$\dfrac{\dfrac{t \vdash \forall x : \sigma.B \qquad [x : \sigma]}{(t\ x) \vdash B} \quad \begin{matrix}\cdots \\ \Pi_B \\ \cdots\end{matrix} \quad pr(B) \vdash B}{\lambda x : \sigma.\ pr(B) \vdash \forall x : \sigma.B}$$

□

Remarque. Le schéma d'indépendance des prémisses n'est toujours pas dérivable dans MR, même en présence de la règle des formules atomiques. En effet, dans le cas contraire, une preuve de IP dans ce système pourrait être simplement traduite en une preuve de IP dans la théorie de types de P. Martin-Löf [37], dont on sait qu'elle est conservative sur HA (cf. [5], par exemple). En réalité, pour pouvoir typer t_{IP} (qui *mr*-réalise IP) dans notre système, il manque seulement une règle exprimant qu'un λ-terme qui décore une preuve d'une formule A est typable de type $\mathrm{A}^\star$ (et dénote donc une fonction totale). Cette propriété, qui est évidente au niveau du méta-langage par construction de $\mathrm{MR}_{\mathbf{e}}$, doit être rendue explicite dans le système pour pouvoir être exploitée dans les preuves. On se donne donc la règle suivante :

Règle de typage de l'extraction $\mathrm{R}^\star$

$$\frac{t \vdash A}{t : A^\star}$$

Notation. On appelle $\mathrm{MR}^\star_{\mathbf{e}}$ le système $\mathrm{MR} + \mathrm{R}_{\mathbf{e}} + \mathrm{R}^\star$.

Proposition 6.3.4 *La proposition* 6.1.1 *est encore vérifiée dans* $\mathrm{MR}^\star_{\mathbf{e}}$.

Preuve. La règle $\mathrm{R}^\star$ internalise dans $\mathrm{MR}_{\mathbf{e}}$ une propriété vérifiée au niveau du méta-langage, et la réalisabilité modifiée dans IQC^ω est justement définie à ce niveau. □

Remarque. Il suffit en fait de se donner la règle $\mathrm{R}^\star$ pour les hypothèses. En effet, pour toute règle de MR, la règle obtenue en remplaçant tout jugement de prouvabilité de la forme $t \vdash F$ par le jugement de typage $t : F^\star$ est une instance d'une règle de typage de MR. Par exemple, la transformée de la règle :

$$\dfrac{\begin{array}{c}[x \vdash A]\\ \vdots \\ t \vdash B\end{array}}{\lambda x : A^\star . t \vdash A \Rightarrow B}$$

est la règle :

$$\dfrac{\begin{array}{c}[x : A^\star]\\ \vdots \\ t : B^\star\end{array}}{\lambda x : A^\star . t : (A \Rightarrow B)^\star}$$

qui est bien une instance de la règle d'introduction de $\to$ puisque $(A \Rightarrow B)^\star = A^\star \to B^\star$ par définition de *.

$\star$ Preuve du schéma d'indépendance des prémisses IP

Nous sommes maintenant en mesure de donner la preuve du schéma d'indépendance des prémisses IP dans $\mathrm{MR}^\star_{\mathbf{e}}$. La conclusion qui décore cette preuve est bien sûr t^σ_{IP}. Remarquons aussi l'utilisation de la règle $\mathrm{R}^\star$ pour dériver le type de f, qui traduit que f dénote une fonction totale définie en particulier au point $pr(H)$.

Proposition 6.3.5 *Dans* $\mathrm{MR}^\star_{\mathbf{e}}$*, on peut dériver* $t^\sigma_{\mathrm{IP}} \vdash \mathrm{IP}$.

Preuve.

$$\dfrac{\dfrac{\dfrac{\dfrac{\vdots}{pr(H) : H^\star} \quad \dfrac{[f \vdash H \Rightarrow \exists y : \sigma.B]^2}{f : (H \Rightarrow \exists y : \sigma.B)^\star}}{(f\ pr(H)) : (\exists y : \sigma.B)^\star}}{\pi(f\ pr(H)) : \sigma} \quad \dfrac{\dfrac{\dfrac{\begin{array}{c}[x \vdash H]^1\\ \cdots \\ \Pi_H \\ \cdots \\ pr(H) \vdash H\end{array} \quad [f \vdash H \Rightarrow \exists y : \sigma.B]^2}{(f\ pr(H)) \vdash \exists y : \sigma.B}}{\pi'(f\ pr(H)) \vdash B[\pi(f\ pr(H))/y]}}{\lambda x : H^\star . \pi'(f\ pr(H)) \vdash H \Rightarrow B[\pi(f\ pr(H))/y]}\,(1)}{\dfrac{\langle \pi(f\ pr(H)), \lambda x : H^\star . \pi'(f\ pr(H))\rangle \vdash \exists y : \sigma.(H \Rightarrow B)}{\lambda f : (H \Rightarrow \exists y : \sigma.B)^\star . \langle \pi(f\ pr(H)), \lambda x : H^\star . \pi'(f\ pr(H))\rangle \vdash (H \Rightarrow \exists y : \sigma.B) \Rightarrow \exists y : \sigma.(H \Rightarrow B)}\,(2)}$$

□

Nous pouvons maintenant conclure cette section par le résultat annoncé, à savoir qu'une formules est prouvable dans $\mathrm{MR}^\star_{\mathbf{e}}$ si et seulement si elle est *mr*-réalisable dans IQC^ω.

Théorème 6.3.6 *Pour toute formule φ, il existe un terme t de type $\varphi^\star$ tel que $t \vdash \varphi$ soit dérivable dans* $\mathrm{MR}^\star_{\mathbf{e}}$ *si et seulement si* $\mathrm{IQC}^\omega \vdash \exists x : \varphi^\star.(x\ mr\ \varphi)$.

Preuve. Par la proposition 6.3.4, si $t \vdash \varphi$ est dérivable dans $\mathrm{MR}^\star_{\mathbf{e}}$ alors $\mathrm{IQC}^\omega \vdash t\ mr\ \varphi$ d'où $\mathrm{IQC}^\omega \vdash \exists t : \varphi^\star.(t\ mr\ \varphi)$. Réciproquement, si $\mathrm{IQC}^\omega \vdash t\ mr\ \varphi$ alors $\mathrm{IQC}^\omega + \mathrm{IP} + \mathrm{AC} \vdash \varphi$ et, par conséquent, il existe un terme t de type $\varphi^\star$ tel que $t \vdash \varphi$ soit dérivable dans $\mathrm{MR}^\star_{\mathbf{e}}$. □

7 Sémantique « naïve »

Le théorème de complétude (et de validité) de la logique classique du premier ordre s'exprime habituellement ainsi : « une formule est prouvable dans une théorie si et seulement si elle est vraie dans tous les modèles de cette théorie ».

En logique intuitionniste, la prouvabilité d'une formule implique l'existence d'un programme (exprimé dans un λ-calcul qui étend le langage des termes) qui réalise cette formule. Ce programme peut être vu comme apportant l'information nécessaire à la vérification de la formule dans un modèle classique.

Comme cette information est de nature syntaxique elle peut être utilisée dans tous les modèles. Nous définissons ici formellement la notion de « vérité uniforme » afin de formaliser cette intuition que certaines formules peuvent être vérifiée dans tous les modèles de la même manière : en particulier, si la formule est existentielle, elle est toujours vérifiée pour le même témoin (représenté par un λ-terme).

Notation. Les formules et parfois les types du calcul seront soulignés pour éviter toute confusion avec le méta-langage. De plus, les noms de prédicats du méta-langage commencent par une majuscule.

7.1 Vérité classique usuelle

⋆ Calcul des prédicats multi-sorté

Nous rappelons la définition de valeur de vérité d'une formule d'un calcul des prédicats multi-sorté dans un modèle classique $\mathcal{M}$ (défini par un domaine M_σ pour chaque type σ et d'une interprétation $\mathcal{I}$ des symboles de fonctions et de prédicats typés). Le prédicat du méta-langage *Eval* qui représente la valeur de vérité d'une formule A peut être définie récursivement de la façon suivante : (nous supposons donné une valuation v qui associe à toute variable libre de type σ de A un élément de M_σ, et nous notons $[\![t]\!]_v$ l'extension usuelle de v aux termes à partir de l'interprétation des symboles de fonction par $\mathcal{I}$).

- $Eval(v, \underline{P(t_1, \ldots, t_n)}) \equiv \mathcal{I}(\underline{P})([\![\underline{t_1}]\!]_v, \ldots, [\![\underline{t_n}]\!]_v)$
- $Eval(v, \underline{A \Rightarrow B}) \equiv Eval(v, \underline{A}) \Rightarrow Eval(v, \underline{B})$
- $Eval(v, \underline{A \wedge B}) \equiv Eval(v, \underline{A}) \wedge Eval(v, \underline{B})$
- $Eval(v, \underline{\forall x : \sigma.B}) \equiv \forall e \in M_\sigma\ Eval(\{v; x = e\}, \underline{B})$
- $Eval(v, \underline{\exists x : \sigma.B}) \equiv \exists e \in M_\sigma\ Eval(\{v; x = e\}, \underline{B})$

⋆ Calcul des prédicats ω-sorté

Les termes, les types et les formules sont ceux de IQC^ω.

Un modèle *plein* de ce calcul est obtenu en étendant récursivement la définition de la sous-section précédente :

- Interprétation d'un type σ : le singleton ensembliste est noté $\{*\}$, et les symboles $\times$ et $\rightarrow$ de droite désignent respectivement le produit cartésien ensembliste et l'espace des fonctions totales,
  - $M_{I_1} \equiv \{*\}$

  - $M_{\sigma\to\tau} \equiv M_\sigma \to M_\tau$
  - $M_{\sigma\times\tau} \equiv M_\sigma \times M_\tau$

- Interprétation d'un terme t : on suppose donné une valuation v qui à tout variable libre de type σ de t associe un élément de M_σ et on définit $[\![t]\!]_v$ par récurrence sur t, (les notations utilisées à droite appartiennent au méta-langage)

  - $[\![x]\!]_v \equiv v(x)$, si x est une variable (typée),
  - $[\![f]\!]_v \equiv \mathcal{I}(f)$, si f est une constante (typée),
  - $[\![\langle u, v\rangle]\!]_v \equiv \langle [\![u]\!]_v, [\![v]\!]_v\rangle$
  - $[\![\pi(t)]\!]_v \equiv \pi([\![t]\!]_v)$, $[\![\pi(t)]\!]_v \equiv \pi([\![t]\!]_v)$
  - $[\![\lambda x : \sigma.t]\!]_v \equiv \lambda a \in M_\sigma \; ([\![t]\!]_{v,x=a})$
  - $[\![(t\ u)]\!]_v \equiv ([\![t]\!]_v\ [\![u]\!]_v)$

- Interprétation des formules : le prédicat *Eval* est défini comme dans la sous-section précédente.

Remarque. Étant donné un calcul des prédicats multi-sorté, le calcul des prédicats ω-sorté engendré par ce calcul est une extension conservative en logique classique. En effet, une formule du calcul multi-sorté vraie dans tous les modèles pleins du calcul ω-sorté engendré, est vraie dans les modèles du calcul multi-sorté de départ, elles y est donc prouvable (par le théorème de complétude dans les théories du premier ordre). Autrement dit, tout modèle du calcul multi-sorté peut s'étendre en un modèle plein du calcul ω-sorté engendré.

7.2 Notion d'information

Considérons une formule de la forme $\underline{\forall x : \sigma.\exists y : \tau.A(x, y)}$ où $\underline{A}$ est atomique. Si cette formule est vérifiée *classiquement dans un modèle donné* $\mathcal{M}$, on peut extraire de cette vérification (modulo l'axiome du choix) une fonction de M_σ dans M_τ, qui à tout élément x associe un élément y vérifiant $\underline{A(x, y)}$.

Étant donné un modèle donné $\mathcal{M}$, on peut définir par récurrence, pour une formule quelconque $\underline{A}$, l'espace $\underline{A}^\star_M$ où se situe l'« information » contenue dans une vérification de $\underline{A}$:

- $\underline{A}^\star_M \equiv \{*\}$, si $\underline{A}$ est atomique,
- $(\underline{A \Rightarrow B})^\star_M \equiv \underline{A}^\star_M \to \underline{B}^\star_M$
- $(\underline{A \wedge B})^\star_M \equiv \underline{A}^\star_M \times \underline{B}^\star_M$
- $(\underline{\forall x : \sigma.B})^\star_M \equiv M_\sigma \to \underline{B}^\star_M$
- $(\underline{\exists x : \sigma.B})^\star_M \equiv M_\sigma \times \underline{B}^\star_M$

Remarque. L'espace fonctionnel $\underline{A}^\star_M$ est exactement $M_{\underline{A}^\star}$.

7.3 Réalisabilité sémantique classique

On ajoute au prédicat d'évaluation un argument qui représente l'information qui doit servir lors de la vérification. Ce prédicat définit en quelque sorte une « réalisabilité sémantique » dans un modèle classique fixé $\mathcal{M}$:

- $Real(v, f, \underline{P(t_1, \dots, t_n)}) \equiv \mathcal{I}(\underline{P})([\![\underline{t_1}]\!]_v, \dots, [\![\underline{t_n}]\!]_v)$
- $Real(v, f, \underline{A \wedge B})) \equiv Real(v, \pi(f), \underline{A}) \wedge Real(v, \pi'(f), \underline{B})$
- $Real(v, f, \underline{A \Rightarrow B}) \equiv \forall g \in \underline{A}^\star_M \; Real(v, g, \underline{A}) \Rightarrow Real(v, f(g), \underline{B})$
- $Real(v, f, \underline{\forall x : \sigma . B}) \equiv \; \forall e \in M_\sigma \; Real(\{v; x = e\}, f(e), \underline{B})$
- $Real(v, f, \underline{\exists x : \sigma . B}) \equiv Real(\{v; x = \pi(f)\}, \pi'(f), \underline{B})$

La terminologie « réalisabilité sémantique » est justifiée par la proposition suivante, qui exprime que le prédicat *Real* ci-dessus est exactement la sémantique classique de l'interprétation par la réalisabilité modifiée :

Proposition 7.3.1 *Étant donné une formule $\underline{A}$ et un modèle $\mathcal{M}$, pour tout terme t de type $\underline{A}^\star$ et toute valuation v des variables libres de $\underline{t}$ et de $\underline{A}$, $Real(v, [\![t]\!]_v, \underline{A})$ est équivalent à $Eval(v, \underline{t \; mr \; A})$.*

Preuve. Par une simple récurrence sur la formule A. □

Corollaire 7.3.2 *Étant donné une formule $\underline{A}$, un modèle $\mathcal{M}$ et une valuation v des variables libres de $\underline{A}$, alors $\exists f \in \underline{A}^\star_M \;\; Real(v, f, \underline{A})$ si et seulement si $Eval(v, \underline{\exists x : A^\star . (x \; mr \; A)})$.*

Preuve. Par définition, $Eval(v, \underline{\exists x : A^\star . (x \; mr \; A)})$ est équivalent à $\exists f \in \underline{A}^\star_M \; Eval(\{v, x = f\}, \underline{x \; mr \; A})$, et par la proposition précédente est aussi équivalent à $\exists f \in \underline{A}^\star_M \; Real(\{v, x = f\}, [\![x]\!]_v, \underline{A})$, ce qui peut enfin s'écrire $\exists f \in \underline{A}^\star_M \; Real(v, f, \underline{A})$ puisque x n'apparaît pas dans $\underline{A}$. □

On peut montrer par ailleurs que, dans un modèle classique fixé, la réalisabilité sémantique correspond bien à la vérité classique :

Proposition 7.3.3 *En supposant le modèle $\mathcal{M}$ fixé, étant donné une formule $\underline{A}$ et une valuation v, $\exists f \in \underline{A}^\star_M \;\; Real(v, f, \underline{A})$ si et seulement si $Eval(v, \underline{A})$.*

Preuve. On a montré que $\mathrm{IQC}^\omega + \mathrm{AC} + \mathrm{IP} \vdash \underline{\exists x : A^\star . (x \; mr \; A) \Leftrightarrow A}$. Or IP est valide en logique classique, AC est valide dans les modèles (puisqu'ils sont supposés pleins), et par le corollaire 7.3.2, $\exists f \in \underline{A}^\star_M \;\; Real(v, f, \underline{A})$ si et seulement si $Eval(v, \underline{\exists x : A^\star . (x \; mr \; A)})$. □

Corollaire 7.3.4 *Étant donné un calcul des prédicats multi-sorté, une formule $\underline{A}$ est prouvable en **logique classique** si et seulement si, dans tout modèle $\mathcal{M}$, pour toute valuation v, il existe une fonction $f \in \underline{A}^\star_M$ telle que $Real(v, f, \underline{A})$.*

Preuve. Par la proposition 7.3.3 et la complétude au premier ordre (multi-sorté). □

7.4 Vérité uniforme

Nous utilisons maintenant la réalisabilité sémantique pour définir une sémantique naïve de $\mathrm{MR}^\star_{\mathbf{e}}$. Nous n'avons pas de résultat de complétude pour cette sémantique, pour un calcul multi-sorté (du premier ordre) puisque la réalisabilité modifiée n'interprète pas le contenu calculatoire des formules atomiques. Pour une sémantique complète basée sur la réalisabilité, nous renvoyons à [35].

Définition 7.4.1 *Étant donné un calcul des prédicats ω-sorté, une formule $\underline{A}$ est dite* ***vérifiée uniformément*** *s'il existe un terme $\underline{t}$ de type $\underline{A}^\star$ tel que dans tout modèle $\mathcal{M}$, pour toute valuation v des variables libres de $\underline{t}$ et de $\underline{A}$, on a $Real(v, [\![\underline{t}]\!]_v, \underline{A})$.*

Proposition 7.4.2 *Toute formule $\underline{A}$ prouvable dans* $\mathrm{MR}^\star_{\mathbf{e}}$ *est vérifiée uniformément.*

Preuve. Par la proposition 6.3.4, si $t \vdash A$ est dérivable dans $\mathrm{MR}^\star_{\mathbf{e}}$, alors $t\ mr\ A$ est prouvable dans IQC^ω, par la proposition 7.3.1, dans tout modèle $\mathcal{M}$, pour toute valuation v, $Real(v, [\![t]\!]_v, \underline{A})$. $\square$

Table des matières

Bibliographie

[1] Asperti, A., and Longo, G. ***Categories, Types and Structures***. MIT Press, 1991.

[2] Barbanera, F., and Berardi, S. A Symmetric Lambda Calculus for "Classical" Program Extraction. In ***Theoretical Aspects of Computer Software***, vol. 542 of ***LNCS***. Springer-Verlag, 1994, pp. 495–515.

[3] Barbanera, F., and Berardi, S. Extracting constructive content from classical logic via control-like reductions. vol. 662 of ***LNCS***. Springer-Verlag, 1994, pp. 47–59.

[4] Barendrect, H. P. Lambda calculi with types. In ***Handbook of Logic in Computer Science***, vol. 2. Clarendon Press, Oxford, 1992.

[5] Beeson, M. J. ***Foundations of Constructive Mathematics***. Springer-Verlag, Berlin, 1973.

[6] Birkhoff, G. ***Lattice Theory***, 3rd ed. American Mat. Soc. New-York, 1978.

[7] Császár, Á. ***Fondement de la topologie générale***. Gauthier-Villars, Paris, 1960.

[8] Curry, H. B. ***Foundations of mathematical logic***. McGraw Hill, 1963.

[9] Danos, V., Joinet, J.-B., and Schellinx, H. LKQ and LKT : Sequent calculi for second order logic based upon dual linear decompositions of classical implication. In ***Advances in Linear Logic*** (1995), J.-Y. Girard, Y. Lafont, and L. Regnier, Eds., no. 222 in London Mathematical Society Lecture Note series, Cambridge University Press, pp. 211–224.

[10] de Groote, P. A simple calculus of exception handling. In ***Second International Conference on Typed Lambda Calculi and Applications*** (Edinburgh, United Kingdom, 1995), LNCS, pp. 201–215.

[11] Diller, J., and Troelstra, A. S. Realizability and intuitionistic logic. ***Synthese***, 60 (1984), 253–282.

[12] Dougherty, D. J., and Subrahmanyam, R. Equality between functionals in the presence of coproducts. In ***Proc. 10th Annual IEEE Symp. on Logic in Computer Science*** (1995), pp. 282–291.

[13] Dragalin, A. G. Mathematical intuitionism–introduction to proof theory. In ***Translations of Mathematical Monographs***, vol. 67 of ***American Matematical Society***. Providence, Rhode Island, 1988.

[14] Dyckhoff, R. Contraction-free sequent calculi for intuitionistic logic. ***The Journal of Symbolic Logic 57***, 3 (1992), 795–807.

[15] Feferman, S. Constructive theories of functions and classes. In ***Logic Colloquium '78: Proceedings of the Logic Colloquium at Mons*** (Amsterdam, 1979), M. Boffa, D. van Dalen, and K. McAloon, Eds., North-Holland.

[16] Felleisen, M., Friedman, D. P., Kohlbecker, E., and Duba, B. F. A syntactic theory of sequential control. ***Theoretical Computer Science 52***, 3 (1987), 205–237.

[17] Filinski, A. Declarative Continuations: An Investigation of Duality in Programming Language Semantics. In ***Category Theory and Comp. Sci.***, vol. 389 of ***LNCS***. Springer-Verlag, 1989, pp. 224–249.

[18] Filinski, A. Declarative continuations and categorical duality. Tech. Rep. 89/11, DIKU, Computer Science Department, University of Copenhagen, 1989. Masters Thesis.

[19] Girard, J.-Y. Linear logic. ***Theoretical Computer Science 50*** (1987), 1–101.

[20] Girard, J.-Y. A new constructive logic: Classical logic. ***Mathematical Structures in Computer Science 1*** (1991), 255–296.

[21] Girard, J.-Y. On the unity of logic. ***Annals of Pure and Applied Logic 59*** (1993), 201–217.

[22] Girard, J.-Y., Lafont, Y., and Taylor, P. ***Proofs and Types***. Cambridge Tracts in Theoretical Computer Science 7. Cambridge University Press, 1988.

[23] Griffin, T. G. A formulæ-as-type notion of control. In ***Conference Record of the Seventeenth Annual ACM Symposium on Principles of Programming Langages*** (1990), pp. 47–58.

[24] Gurevich, Y. Intuitionistic logic with strong negation. ***Studia Logica 36*** (1977), 49–59.

[25] Guzmán, J. C., and Suárez, A. An extended type system for exceptions. In ***Record of the 1994 ACM Conference on ML and its Aplications*** (1994), INRIA, pp. 127–135.

[26] Herbelin, H. ***Séquents qu'on calcule : de l'interprétation du calcul des séquents comme lambda-termes et commme de stratégies gagnantes***. PhD thesis, Université Denis Diderot - Paris 7, 1995.

[27] Huet, G. Cartesian closed categories and lambda-calculus. In ***Combinators and functionnal programming languages***, P. L. Curien, G. Cousineau, and B. Robinet, Eds., vol. 242. Springer-Verlag, 1986, pp. 123–135.

[28] Kleene, S. C. Realizability: a retrospective survey. In ***Proceedings of the Cambridge Summer School in Mathematical Logic*** (Berlin, 1973), vol. 37 of ***Lecture Notes in Mathematics***, Springer-Verlag, pp. 95–112.

[29] Krivine, J.-L. ***Lambda-calcul, types et modèles***. Masson, Paris, 1990. English translation, Ellis Horwood, 1993.

[30] KRIVINE, J.-L. Classical logic, storage operators and second order λ-calculus. ***Ann. of Pure and Appl. Logic 68*** (1994), 53–78.

[31] KRIVINE, J.-L., AND PARIGOT, M. Programming with proofs. ***J. Inf. Process. Cybern. EIK 26***, 3 (1990), 149–167.

[32] LAFONT, Y., BEUS, B., AND STREICHER, T. From continuation semantics to abstract machines. 1994.

[33] LAMBEK, J., AND SCOTT, P. J. ***An introduction to higher order categorical logic***. Studies in Advanced Mathematics 7. Cambridge University Press, Amsterdam, 1986.

[34] LANE, S. M. ***Categories for the working mathematicians***. Springer, 1971.

[35] LÄUCHLI, H. An abstract notion of realizability for which intuitionistic predicate calculus is complete. In ***Intuitionism and Proof Theory*** (1970), Proceedings of the Summer Conference at Buffalo N.Y., North-Holland, pp. 227–234.

[36] MARTIN-LÖF, P. An intuitionistic theory of types: predicative part. In ***Logic colloquium '73*** (1975), H. E. Rose and J. C. Stepherdson, Eds., North-Holland.

[37] MARTIN-LÖF, P. Constructive mathematics and computer programming. In ***Logic, Methodology and Philosophy of Science IV*** (1982), North-Holland, pp. 153–175.

[38] MARTIN-LÖF, P. ***Intuitionistic Type Theory***. Bibliopolis, 1984.

[39] MELLIES, P. A. Typed λ-calculi with explicit substitutions may not terminate. In ***Second International Conference on Typed Lambda Calculi and Applications*** (Edinburgh, United Kingdom, 1995), LNCS, pp. 328–334.

[40] MURTHY, C. R. ***Extracting Constructive Content from Classical proofs***. PhD thesis, Cornell University, Department of Computer Science, 1990.

[41] MURTHY, C. R. Classical proofs as programs: How, when, and why. Tech. Rep. 91-1215, Cornell University, Department of Computer Science, 1991.

[42] MURTHY, C. R. Control operators, hierarchies, and pseudo-classical type systems: A-translation at work, 1992. Accepted to Continuations '92.

[43] NAKANO, H. A constructive formalization of the catch and throw mechanism. In ***Proceedings, Seventh Annual IEEE Symposium on Logic in Computer Science*** (Santa Cruz, California, 1992), pp. 82–89.

[44] NAKANO, H. The non-deterministic catch and throw mechanism and its subject reduction property. In ***Logic, Language and Computation*** (1994), vol. 592 of ***LNCS***, Springer-Verlag, pp. 61–72.

[45] NAKANO, H. ***The Logical Structures of the Catch and Throw Mechanism***. PhD thesis, The University of Tokyo, 1995.

[46] PARIGOT, M. Free deduction: an analysis of computation in classical logic. In ***Proc. Logic Prog. and Autom. Reasoning*** (1991), vol. 592 of ***LNCS***, pp. 361–380.

[47] Parigot, M. $\lambda\mu$-calculus: an algorithmic interpretation of classical natural deduction. In ***Proc. Logic Prog. and Autom. Reasoning*** (1992), vol. 624 of ***LNCS***, pp. 190–201.

[48] Parigot, M. Classical proofs as programs. In ***Computationnal logic and theory*** (1993), vol. 713 of ***LNCS***, Springer-Verlag, pp. 263–276.

[49] Parigot, M. Strong normalisation for second order classical natural deduction. In ***Proceedings of the eighth annual IEEE symposium on logic in computer science*** (1993).

[50] Paulin-Mohring, C. ***Extraction de programmes dans le calcul des conctructions***. PhD thesis, Université Denis Diderot - Paris 7, 1989.

[51] Paulin-Mohring, C. Réalisabilité et extraction de programmes. In ***Logique et informatique : une introduction*** (1989), B. Courcelle, Ed., collection didactique, INRIA, pp. 163–180.

[52] Queinnec, C. ***Les langages Lisp***. InterEditions, Paris, 1994.

[53] Rasiowa, H. ***An algebraic approach to non-classical logics***. North-Holland, 1974.

[54] Rauszer, C. Semi-boolean algebras and their applications to intuitionistic logic with dual operations. In ***Fundamenta Mathematicae*** (1974), vol. 83, pp. 219–249.

[55] Rauszer, C. On the strong semantical completeness of any extension of the intuitionistic predicate calculus. In ***Bulletin de l'Académie Polonaises des Sciences***, vol. 24-2 of ***Série des sciences math., astr. et phys.*** 1976, pp. 81–87.

[56] Rauszer, C. An algebraic and Kripke-style approach to a certain extension of intuitionistic logic. In ***Dissertationes Mathematicae***, vol. 167. Institut Mathématique de l'Académie Polonaise des Sciences, 1980.

[57] Rehof, N. J., and Sørensen, M. H. The λ_Δ-calculus. In ***Theoritical Aspects of Computer Software*** (1994), vol. 542 of ***LNCS***, Springer-Verlag, pp. 516–542.

[58] Reynolds, J. C. The discoveries of continuations. ***Lisp and Symbolic Computation 6*** (1993), 233–248.

[59] Sato, M. Intuitionistic and classical natural deduction systems with the Catch and the Throw rules. In ***Second Workshop on Non-Standard Logic and Logical Aspects of Computer Science*** (Irkutsk, Russia, 1995), NSL'95.

[60] Swaen, M. D. G. ***Weak and strong sum-elimination in intuitionistic type theory***. PhD thesis, The University of Amsterdam, 1989.

[61] Swaen, M. D. G. A characterization of ML in many-sorted arithmetic with conditional application. ***The Journal of Symbolic Logic 57***, 3 (1992), 924–953.

[62] Szabo, M. E. ***Gentzen Collected work***. North-Holland, Amsterdam, 1969.

[63] Szabo, M. E. A categorical characterization of boolean algebras. ***Algebra Universalis 4*** (1974), 192–194.

[64] Troelstra, A. S. ***Metamathematical Investigation of Intuitionistic Arithmetic and Analysis***, vol. 344 of ***Lecture Notes in Mathematics***. Springer-Verlag, Berlin, 1973.

[65] Troelstra, A. S., and van Dalen, D. ***Constructivism in Mathematics***, first ed. North-Holland, Amsterdam, 1988.

[66] van Dalen, D. Intuitionistic logic. In ***Handbook of Philosophical Logic***, vol. 3. D. Reidel Publishing Company, 1986, pp. 225–339.