Mémoire d'habilitation à diriger des recherches en sciences

Université Paris Diderot

Complétude en Logiques

Olivier Hermant
20 Avril 2017

Rapporteurs

Mme	Delia Kesner	Professeur des universités
M	Dale Miller	Directeur de recherche
Mme	Sara Negri	Professeur

Examinateurs

M	Gilles Dowek	Directeur de recherche
Mme	Catherine Dubois	Professeur des universités
M	Alessio Guglielmi	Professeur
M	Alexander Leitsch	Professeur
M	Pierre Jouvelot	Maître de recherche

Avant-Propos

Ce mémoire d'Habilitation à Diriger des Recherches introduit les articles [1, 18, 19, 22, 24, 37, 50, 51, 65, 80, 81, 82] et quelques résultats non encore publiés [83, 21, 20]. Le travail d'implémentation [44, 42, 43, 27], même s'il est très peu évoqué dans ce manuscrit, y a aussi sa place. Le manuscrit ne discute pas nombre d'autres articles qui ne font pas partie de la thématique abordée dans ce mémoire, même s'ils ont constitué, ou, dans le cas de Dedukti, constituent actuellement, une partie importante de mon travail d'encadrement et de recherche. Enfin, il ne discute pas non plus les articles [75, 76, 78, 79] qui ont fait l'objet de ma thèse de doctorat [77] — même si, dans le cas de [79], le théorème de Skolémisation (en déduction modulo) a été démontré depuis lors alors qu'il figurait en tant que simple axiome dans la thèse.

Afin de comprendre les problématiques qui sont abordées ici, il vaut mieux être familier avec au moins une méthode de démonstration automatique ou de preuve au premier ordre (tableaux, calcul des séquents, déduction naturelle de préférence).

Résumé des travaux de recherche :

Les travaux présentés succinctement dans ce mémoire concernent ce que l'on peut présenter en première lecture comme des démonstrations de la complétude des méthodes de démonstration automatique, en particulier de la méthode des tableaux. La complétude est un résultat permettant d'assurer que, si une assertion est prouvable – ou encore universellement valide–, alors la recherche de preuve exhaustive, associée à la méthode de démonstration automatique, aboutira.

Comme nous le verrons, ces résultats de complétude se traduisent, tout d'abord, en des résultats de complétude *sans coupure* pour le calcul des séquents, puis en des théorèmes d'admissibilité de la coupure de ce même calcul des séquents, ce qui est l'un des résultats centraux de toute logique. Afin de l'obtenir dans un plus grand nombre de cas, nous développerons ensuite des méthodes plus algébriques.

Nous approfondirons notre examen de ces démonstrations d'admissibilité de la coupure en les rapprochant des preuves de normalisation : dans un premier temps en étudiant la manière dont l'admissibilité, lorsqu'elle est démontrée de manière constructive, génère des preuves sans coupure, et donc contient un algorithme d'élimination des coupures. Dans un second temps, nous étudierons les structures sémantiques associées à la normalisation.

Ces travaux ont été menés dans des systèmes logiques, tels que la logique d'ordre supérieure, classique, intuitionniste ou linéaire, en tenant compte de l'intentionnalité. Le domaine d'application le plus important des techniques développées dans ce manuscrit est cependant la Déduction modulo théorie, qui ajoute une relation de réécriture à la logique du premier ordre et aux systèmes de preuve. Cette relation permet de prendre en compte l'aspect calculatoire des preuves, et d'éviter l'introduction d'axiomes. L'étude de la règle de coupure et de son élimination devient uniforme ; de plus, l'on peut, jusqu'à un certain point, s'abstraire des systèmes de réécritures concret.

Table des matières

Chapitre 1

Introduction

La cohérence est une propriété essentielle de tout système formel de raisonnement logique. Sans elle, il devient possible de démontrer tout et son contraire, ce qui limite l'intérêt pratique d'un tel système *incohérent*, en particulier lorsque l'on souhaite en faire usage pour mécaniser des preuves, soit pour les construire (cas des outils de démonstration automatique), soit pour les vérifier (cas des assistants de preuve).

Ce résultat peut être établi par diverses méthodes, notamment en passant par une analyse de la structure des preuves : on peut alors retrouver des appels à d'autres propriétés fondamentales telles que l'élimination des coupures ou la propriété de la sous-formule, qui énonce que, pour une assertion prouvable donnée, on peut toujours trouver une preuve qui ne fait intervenir que des sous-formules de l'assertion initiale.

Une démonstration alternative de la cohérence est de considérer un espace sémantique, essentiellement un ensemble de valeurs de vérité possédant une certaine structure, qui dépend de la logique considérée, dans lequel on interprète la logique. L'existence d'un seul *modèle* suffit à la cohérence, à condition que celui-ci soit non dégénéré, c'est-à-dire qu'il contienne au moins deux valeurs de vérité différentes, que l'on peut appeler "vrai" et "faux". Ce résultat de cohérence sémantique est ensuite transféré au système de raisonnement syntaxique par la contraposée du théorème de correction.

Ceci nous fournit un exemple du rôle central que joue en logique la paire syntaxe–sémantique, comme l'a souligné la thèse d'Alberto Naibo [106]. Cette dualité s'appuie sur deux théorèmes fondamentaux, ceux de correction et de complétude, qui énoncent que tout ce qui est prouvable est *universellement* valide et vice versa.

Ainsi, dans ce manuscrit et plus généralement en logique, nous ferons une distinction claire entre la *validité*, un concept sémantique qui énonce que ce que l'on a *interprété* est équivalent à la valeur de vérité "vrai" dans un modèle (ou dans tous les modèles pour la validité universelle), et la *prou-*

vabilité, un concept syntaxique qui énonce la possibilité de construire une *preuve* selon les *règles d'inférence* d'un système formel de raisonnement (fixé à l'avance), qui sont autant de règles de grammaire édictant la manière de construire les preuves.

Les deux théorèmes de correction et de complétude sont importants pour tout système de raisonnement ainsi que la démonstration automatique, puisque l'on a, à la fois, le besoin essentiel de savoir que l'on ne peut prouver que les formules qui sont valides (théorème de correction) et le besoin secondaire de s'assurer que l'algorithme que l'on implémente est capable, éventuellement en un temps très long, de trouver une preuve de n'importe quelle formule valide (théorème de complétude).

Si la correction est en général relativement simple à démontrer, il en va autrement de la complétude — la première démonstration est due à Gödel [68], en 1929, alors que des systèmes syntaxiques de raisonnement ont été mis en place dès la fin du XIXe siècle par Frege et Hilbert [60, 137].

Une des applications de ces théorèmes est l'admissibilité de la règle de coupure, qui affirme que, pour prouver une formule logique, il n'est pas besoin de passer par des lemmes intermédiaires, que l'on peut voir comme des détours. Cette élimination des coupures est la clef de nombreuses autres propriétés : outre les propriétés de la sous-formule et de la cohérence, on peut aussi rappeler la propriété de la disjonction ou du témoin, en logique intuitionniste.

Pour obtenir ce résultat, une stratégie consiste à combiner la correction avec une version renforcée du théorème de complétude [128, 4, 124, 112, 113] : tout ce qui est universellement vrai est démontrable *sans coupure*. En composant les deux théorèmes de correction et de complétude, on obtient bien que, étant donné une preuve d'une certaine assertion, il existe une preuve de cette même assertion sans coupure. Des variantes plus ou moins constructives de cette méthode existent ; historiquement il s'agissait, étant donné un ensemble de formules vérifiant la propriété de cohérence abstraite, autrement dit un ensemble de formules à la fois

— saturé, par exemple par une méthode de complétion à la Henkin [77] ou par la méthode de démonstration automatique dite des tableaux [88, 12, 123, 125, 110, 59, 99],
— et cohérent syntaxiquement *en interdisant* la règle de coupure,

de construire un modèle non dégénéré. Cette toute dernière contrainte (modèle non dégénéré) apparaît comme un frein à la constructivité des démonstrations et a été au fil du temps assouplie — il est intéressant de remarquer que le modèle est *réellement* non dégénéré, mais qu'il n'est possible de le démontrer qu'a posteriori, une fois l'élimination des coupures obtenue.

D'un point de vue opérationnel, c'est plutôt l'étude minutieuse d'un procédé d'élimination des coupures qui est mis en avant. En théorie des types, on cherche à établir pour la réduction des coupures (qui correspond, dans la vision Curry–De Bruijn–Howard, à l'exécution d'un programme bien typé) des propriétés telles que confluence, réduction du sujet ou terminaison. Cela provient des résultats sur la normalisation des termes de preuves dont on a équipé le système de preuve sous une relation de réduction, éventuellement en suivant une stratégie définie à l'avance (appel par nom, appel par valeur).

On peut alors légitimement se demander quels sont les liens entre ces approches sémantiques et opérationnelles de l'élimination des coupures. Certains travaux [31, 39, 74, 87] s'intéressent à cette question, et nous essaierons à notre tour d'apporter certains éléments de réponse.

Un système de preuve syntaxique pur est, en pratique, insuffisant : la formalisation de raisonnements complexes fait appel à un certain nombre d'hypothèses et d'axiomes, par exemple ceux de l'arithmétique, d'une théorie des ensembles ou, d'un point de vue informatique, d'un modèle mémoire particulier, etc.

Raisonner modulo un ensemble d'axiomes remet en question toutes les propriétés que nous avons évoquées ci-dessus, à commencer par la cohérence : pour donner deux exemples célèbres, la théorie naïve des ensembles est incohérente [117], tandis que la théorie des ensembles de Zermelo-Fraenkel souffre d'absence de normalisation [38].

La Déduction modulo théorie est un cadre logique qui permet une étude uniforme de ces théories axiomatiques, car ces théories deviennent un *paramètre* de la logique. En exprimant une théorie par des règles de réécriture, tout en conservant le même pouvoir expressif, on peut considérer un raisonnement *modulo* une théorie dorénavant implicite, tapie dans la relation d'équivalence par réécriture entre propositions : deux propositions pouvant se réécrire l'une en l'autre seront considérées comme faisant partie de la même classe d'équivalence.

Le système de réécriture est, en Déduction modulo théorie, un paramètre et il est possible de l'étudier sans le spécifier concrètement, au moins jusqu'à un certain point. On gagne en généricité ; de plus, on se retrouve de nouveau dans un cadre sans axiome. En particulier, la notion de coupure axiomatique se confond avec la coupure usuelle.

Bien entendu, puisque l'expressivité de la logique devient la même que celle de la théorie modulo laquelle on raisonne, les propriétés théoriques de la Déduction modulo théorie sont impactées. Certains systèmes de réécritures sont cohérents et ont la propriété d'élimination des coupures et celle de normalisation, comme ceux exprimant l'arithmétique [53] ou la logique d'ordre supérieur [52]. Mais on sait, par exemple, que certains systèmes de réécriture sont incohérents [52], que d'autres n'ont pas la propriété d'élimination des coupures [52] et que d'autres encore possèdent la propriété d'élimination des coupures mais pas celle de normalisation, faible ou forte [77]. Ceci reste vrai même si l'on suppose que le système de réécriture seul, les règles d'inférence de la logique mises à part, possède de "bonnes" propriétés, telles que confluence et terminaison [78, 77]. En réalité, toutes ces propriétés sont indécidables [26]. Nous revenons donc à la nécessité d'étudier de près chaque combinaison système de déduction/système de réécriture.

Malgré cela, la Déduction modulo théorie offre l'avantage de l'abstraction : au lieu d'étudier un système de réécriture concret, nous pouvons tenter de définir des *critères* que ceux-ci devront vérifier. Chaque critère sera alors une condition suffisante pour qu'un système de réécriture qui le satisfait jouisse des propriétés théoriques correspondantes.

C'est en particulier dans ce cadre que se place ce mémoire : l'étude des propriétés fondamentales de la Déduction modulo théorie et l'établissement de critères permettant d'assurer celles-ci. Nous appliquerons aussi nos techniques à des formalismes logiques autres que la Déduction modulo théorie, en particulier à des logiques d'ordre supérieur. L'accent sera particulièrement mis sur la propriété de complétude et les conséquences que l'on peut en tirer si l'on est assez précis dans les définitions et les démonstrations.

Plan

Dans un premier chapitre, nous présenterons les différentes démonstrations de complétude en logique du premier ordre, et nous montrerons comment, à partir de la preuve de Henkin du théorème de complétude, arriver à une méthode de démonstration automatique, appelée la *méthode des tableaux*, que l'on peut démontrer complète en définissant un algorithme systématique de recherche de preuve. Ces démonstrations de complétude dans la méthode des tableaux en logique du premier ordre [125, 110] peuvent être grossièrement assimilées à la construction d'un modèle de Herbrand [59].

Nous étendrons ensuite la méthode des tableaux, ainsi que les définitions, les théorèmes de complétude et les algorithme de recherche de preuve associés, à la Déduction modulo théorie. Nous ferons ce travail en logique classique et en logique intuitionniste. Afin d'être aussi proche que possible de l'implémentation de la méthode des tableaux en Déduction modulo théorie [27], nous introduirons une méthode des tableaux avec variables libres, appelée TaMeD, ainsi que certaines notions, adaptées de celles au premier ordre [123, 125] et où l'usage des règles de réécriture a été optimisé.

L'étude concrète du système de réécriture sera faite le plus tard possible, au moment même de construire le modèle réclamé, in fine, par le théorème de complétude. Cette construction demandera, dans certains cas, la mise en œuvre de techniques spécifiques, tirées de l'ordre supérieur [4, 129, 115, 99]. Nous nous intéresserons enfin à la constructivité de ces démonstrations, en préparation des chapitres suivants.

Dans un second chapitre, nous verrons comment ces preuves de complétude peuvent être transformées en preuves d'élimination des coupures pour le calcul des séquents. Tout d'abord, nous énoncerons et prouverons des théorèmes de correction de la méthode des tableaux par rapport au calcul des séquents *sans coupure*. Dans le cadre de la méthode des tableaux avec variables libres, la mise en œuvre de cette stratégie demandera quelques efforts afin de "désoptimiser" la preuve de tableaux.

Ce second chapitre sera aussi l'occasion de discuter de la manière dont l'élimination des coupures peut être prouvée plus directement, sans le détour par la méthode des tableaux. Cela nous conduira à développer des techniques plus algébriques, s'inspirant des travaux d'Okada [112, 113], qui ont l'avantage d'être directement constructives. Au delà de la Déduction modulo théorie, nous étudierons ces propriétés et la construction de tels modèles pour les logiques d'ordre supérieur, intuitionniste et linéaire. Nous conclurons cette partie par une étude des liens entre logique classique et logique intuitionniste, entre autres dans le cadre des systèmes de réécriture de la Déduction modulo théorie.

Dans un troisième chapitre, plus prospectif, nous nous intéresserons à la normalisation en déduction naturelle et en calcul des séquents et à la manière dont, à la fois,

— les démonstrations de normalisation donnent naissance à des démonstrations d'élimination des coupures et
— les démonstrations d'élimination des coupures, dans le cas où elles sont constructives, fonctionnent comme un algorithme permettant de calculer, à partir d'une preuve quelconque, une preuve sans coupure.

Dans cet objectif, nous construirons des modèles algébriques plus complexes que ceux des chapitres précédents, mais plus simples que les modèles usuels utilisés pour démontrer la normalisation forte des termes de preuve. Tout d'abord, nous nous intéresserons aux modèles pour la normalisation faible, puis nous nous débarrasserons des termes de preuve pour construire des algèbres de *séquents*, en déduction naturelle modulo théorie puis dans le calcul des séquents classique modulo théorie.

Une brève conclusion récapitulera l'essentiel des éléments présentés dans ce document, et proposera certaines persepctives de développement futur.

Chapitre 2

Complétude

Nous commençons ce chapitre par exposer les méthodes classiquement utilisées pour démontrer la complétude du calcul des séquents, notamment par la complétion de théorie. Puis nous raffinons ce procédé et montrons que la complétion de théorie peut se réduire à une recherche exhaustive de preuve dans la méthode des tableaux, dont nous démontrons ensuite la complétude dans le cadre de la Déduction modulo théorie, sous certaines conditions sur le système de réécriture. Un travail similaire est ensuite fait pour le cas de la méthode des tableaux intuitionnistes, et enfin, nous tentons d'améliorer la constructivité de ces preuves.

Les sections 2.1 et 2.2 présentent des définitions et des résultats adaptés de la littérature. Ceux-ci sont de nos jours standards, et ils sont apparus entre le début des années 30 et la fin des années 60 du XXe siècle. Ces deux parties mettent en place le cadre nécessaire au reste du chapitre.

La section 2.3 présente l'extension à la Déduction modulo théorie des notions développées précédemment, notamment la méthode des tableaux modulo théorie en logique classique TaMeD, introduite par Richard Bonichon [17] et retravaillée dans les articles [19] et [20] (non encore publié) Les preuves de complétude de TaMeD de la section 2.4 sont tirées de ces deux mêmes articles. Notons qu'une implémentation de la méthode des tableaux modulo théorie est l'objet des articles [27, 43, 42, 44].

La section 2.5, qui clôt ce chapitre, esquisse l'extension de ces résultats aux tableaux intuitionnistes en Déduction modulo théorie, introduits dans l'article [18].

2.1 Calcul des séquents

Les preuves de complétude pour le calcul des séquents sont relativement élémentaires, lorsqu'on ne s'impose aucune contrainte. Elles fonctionnent sur le principe suivant : la construction d'une algèbre de Boole ou de Heyting particulière, que l'on peut qualifier d'algèbre universelle.

2.1.1 Système d'inférence

La figure 2.1 rappelle le calcul des séquents en logique du premier ordre à une sorte. Nous serons dans ce cadre tout au long de ce chapitre, à l'exception de la section 2.4.3, qui fait appel à plusieurs sortes.

Les énoncés (séquents) ont la forme $\Gamma \vdash \Delta$, ce qui signifie que, sous les hypothèses Γ, au moins une des formules de Δ est prouvable. La figure 2.1 présente le calcul des séquents classique. Mais on peut aussi bien y voir le calcul des séquents intuitionniste, en restreignant le membre droit d'un séquent à au plus une formule en conclusion, c'est-à-dire soit du type $\Gamma \vdash A$, soit du type $\Gamma \vdash$, en éliminant les règles qui n'ont plus de sens, comme contr_R, et en dupliquant la règle $\vee_R$.

Le calcul des séquents classique de la figure 2.1 est une variante du système introduit par Ketonen [11, 89]. Il possède des propriétés particulières, dont nous ferons régulièrement usage.

- Les règles dites structurelles, c'est à dire weak_L,weak_R,contr_L et contr_R sont superflues, à condition de contracter explicitement la formule principale lors de l'application des règles $\forall_L$ et $\exists_R$ (ce que nous n'avons pas imposé dans la figure 2.1). Un séquent est alors démontrable si, et seulement si, il est démontrable sans ces règles.
- La majorité des règles est *inversible*, selon la définition d'inversibilité ci-dessous. Nous ferons une analyse plus détaillée à la section 3.3.1.

Définition 2.1.1 (Règle d'inférence inversible). *Une règle d'inférence est dite inversible si ses prémisses peuvent se déduire de la conclusion.*

Plus précisément, soit $*$ un connecteur ou un quantificateur. La règle $*_L$ (respectivement, $*_R$) est inversible si, et seulement si, pour toute formule A qui a $*$ pour connecteur (ou quantificateur) principal, et tous contextes Γ, Δ, le séquent $\Gamma, A \vdash \Delta$ (respectivement, $\Gamma \vdash A, \Delta$) a une démonstration seulement *s'il a une démonstration qui commence par la règle $*_L$ (respectivement $*_R$).*

Dans le calcul des séquents classique de la figure 2.1, toutes les règles sont inversibles, sauf $\forall_R$ et $\exists_L$ [89] ; il est tout à fait possible de se passer de la règle de coupure pour démontrer cette inversibilité [122] ; il est de plus possible de préserver la hauteur des démonstrations [40, 91], à condition de contraindre la règle axiome à ne s'appliquer qu'à des formules atomiques.

Dans le cas intuitionniste, certaines règles à droite et à gauche perdent leur inversibilité, par exemple $\vee_R$ ou $\Rightarrow_L$ dans le calcul des séquents de la figure 2.1. Cependant, ces mêmes deux règles redeviennent inversibles si l'on considère un calcul des séquents intuitionniste multiconclusion [54, 55, 130, 139]. D'autres règles devenant alors non-inversibles à leur tour, comme la

$$\frac{}{\Gamma, A \vdash A, \Delta}\,\mathrm{ax}$$

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

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

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

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

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

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

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

$$\frac{\Gamma, A, A \vdash \Delta}{\Gamma, A \vdash \Delta}\,\mathrm{contr}_L \qquad \frac{\Gamma \vdash A, A, \Delta}{\Gamma \vdash A, \Delta}\,\mathrm{contr}_R$$

$$\frac{\Gamma \vdash \Delta}{\Gamma, A \vdash \Delta}\,\mathrm{weak}_L \qquad \frac{\Gamma \vdash \Delta}{\Gamma \vdash A, \Delta}\,\mathrm{weak}_R$$

FIGURE 2.1 – Calcul des séquents classique

règle $\Rightarrow_R$, voir la section 3.1.2. De plus, prouver la redondance des règles structurelles demande plus de précision dans le design des règles d'inférence [57, 108].

2.1.2 Structures algébriques

Maintenant que nous savons, grâce au calcul des séquents, caractériser les preuves de manière *syntaxique*, il nous faut définir la notion *sémantique* de vérité. Cela se fait à travers les algèbres de Heyting (en logique intuitionniste) et de Boole (en logique classique).

Définition 2.1.2 (Algèbre de Heyting complète)**.** *Une algèbre de Heyting complète est une structure* $\langle \Omega, \leq, \top, \wedge, \forall, \bot, \vee, \exists, \Rightarrow \rangle$ *telle que la sous-*

structure $\langle \Omega, \leq, \top, \wedge, \forall, \bot, \vee, \exists \rangle$ *est un treillis borné complet et que* $\Rightarrow$ *est un opérateur binaire, le* pseudo-complément relatif. *Par abus de langage, l'algèbre de Heyting elle-même sera aussi appelée* Ω.

La relation binaire $\leq$ *est donc une relation d'ordre, et les opérateurs sémantiques vérifient, pour tous* $a, b, c \in \Omega$ *et* $B \in \wp(\Omega)$ *(les parties de* Ω*),*

— $a \leq \top$;
— $a \wedge b \leq a$ *et* $a \wedge b \leq b$;
— *si* $c \leq a$ *et* $c \leq b$*, alors* $c \leq a \wedge b$;
— $\forall B \leq b_i$*, pour tout* $b_i \in B$;
— *si* $c \leq b_i$ *pour tout* $b_i \in B$*, alors* $c \leq \forall B$;
— $\bot \leq a$;
— $a \leq a \vee b$ *et* $b \leq a \vee b$;
— *si* $a \leq c$ *et* $b \leq c$*, alors* $a \vee b \leq c$;
— $b_i \leq \exists B$*, pour tout* $b_i \in B$;
— *si* $b_i \leq c$ *pour tout* $b_i \in B$*, alors* $\exists B \leq c$;
— $a \leq b \Rightarrow c$ *si, et seulement si,* $a \wedge b \leq c$.

Dans ce treillis, $\top$*,* $\wedge$ *et* $\forall$ *sont respectivement les bornes inférieures nullaires, binaires et arbitraires, tandis que* $\bot$*,* $\vee$ *et* $\exists$ *sont respectivement les bornes supérieures nullaires, binaires et arbitraires.*

Notons que nous utilisons, pour les connecteurs logiques et quantificateurs et pour les opérateurs sémantiques, des symboles identiques.

On peut montrer que le treillis associé à une algèbre de Heyting est semi-distributif, c'est-à-dire que, pour tout $a \in \Omega$ et $B \in \wp(\Omega)$, l'équation suivante est vérifiée :

$$a \wedge \exists B = \exists \{a \wedge b \mid b \in B\}.$$

Les algèbres de Boole sont des algèbres de Heyting possédant une contrainte supplémentaire.

Définition 2.1.3 (Algèbre de Boole complète)**.** *Une algèbre de Boole complète est une algèbre de Heyting complète* Ω*, qui vérifie de plus, pour tous* $a, b \in \Omega$*,*

$$(a \Rightarrow \bot) \vee a = \top.$$

$a \Rightarrow \bot$ *est appelé le* complément *de* a*, noté* $\neg a$.

Grâce à cet opérateur de complément, le treillis sous-jacent à une algèbre de Boole est distributif. Un exemple d'algèbre de Boole (complète) est l'algèbre définie par $\Omega = \{\bot, \top\}$, souvent appelée $\{0, 1\}$. Il faut maintenant définir l'interprétation des formules syntaxiques dans nos structures algébriques.

Définition 2.1.4 (Intepétation dans une structure algébrique)**.** *Soit une structure algébrique de Boole ou de Heyting* Ω *et un ensemble* D *appelé domaine.*

Une interprétation $\hat{}$ *des symboles de fonction et de prédicat dans* Ω *est une fonction telle que, pour tout symbole de fonction* n*-aire* f *et tout symbole de prédicat* n*-aire* P*,*

— $\hat{f}$ *est une fonction de* D^n *dans* D *et*

— $\hat{P}$ *est une fonction de* D^n *dans* Ω*.*

L'interprétation (on dit aussi la structure) $\hat{}$ *détermine, de manière univoque, une fonction d'interprétation* $[\![\,]\!]$ *des termes et formules. Le premier argument de* $[\![\,]\!]$*, noté en indice, est une valuation* φ *qui associe à toute variable* x *un élément de* D*.* $[\![\,]\!]$ *est définie par induction sur son second argument, qui est soit un terme, soit une formule.*

— $[\![x]\!]_\varphi = \varphi(x)$,

— $[\![f(t_1, \cdots, t_n)]\!]_\varphi = \hat{f}([\![t_1]\!]_\varphi, \cdots, [\![t_n]\!]_\varphi)$,

— $[\![P(t_1, \cdots, t_n)]\!]_\varphi = \hat{P}([\![t_1]\!]_\varphi, \cdots, [\![t_n]\!]_\varphi)$,

— $[\![\bot]\!]_\varphi = \bot$,

— $[\![\top]\!]_\varphi = \top$,

— $[\![A \wedge B]\!]_\varphi = [\![A]\!]_\varphi \wedge [\![B]\!]_\varphi$,

— $[\![A \vee B]\!]_\varphi = [\![A]\!]_\varphi \vee [\![B]\!]_\varphi$,

— $[\![A \Rightarrow B]\!]_\varphi = [\![A]\!]_\varphi \Rightarrow [\![B]\!]_\varphi$,

— $[\![\forall x A]\!]_\varphi = \forall\{[\![A]\!]_{\varphi+[d/x]} \mid d \in D\}$,

— $[\![\exists x A]\!]_\varphi = \exists\{[\![A]\!]_{\varphi+[d/x]} \mid d \in D\}$.

La fonction d'interprétation $[\![\,]\!]$*, considérée comme représentante de* Ω, D *et* $\hat{}$*, est aussi appelée un* modèle.

Notons qu'à gauche des égalités de la définition 2.1.4 les symboles $\bot$, $\top$, $\wedge$, $\vee$, $\Rightarrow$, $\forall$ et $\exists$ dénotent les connecteurs et quantificateurs de la syntaxe, alors qu'à droite des égalités, ils dénotent les opérateurs de la structure algébrique Ω.

Par souci de simplicité, nous avons défini une unique fonction d'interprétation hétérogène, au lieu de deux fonctions séparées, un pour les termes et une pour les formules. La définition 2.1.4 s'étend aisément au cas d'une logique à plusieurs sortes, auquel cas il y a besoin d'un domaine D_α pour chaque sorte α.

On obtient ensuite la correction, un résultat standard, qui se démontre par induction sur la preuve du séquent considéré. Nous énonçons un unique théorème pour la logique classique et intuitionniste, entendu que dans ce dernier cas, le séquent a au plus une conclusion.

Théorème 2.1.5 (Correction)**.** *Soit* $A_1, \cdots, A_n \vdash B_1, \cdots, B_m$ *un séquent prouvable en calcul des séquents classique (intuitionniste, respectivement). Alors, pour toute algèbre de Boole (de Heyting, respectivement), pour toute interprétation* $[\![\,]\!]$ *et toute valuation* φ*, on a*

$$[\![A_1]\!]_\varphi \wedge \cdots \wedge [\![A_n]\!]_\varphi \leq [\![B_1]\!]_\varphi \vee \cdots \vee [\![B_m]\!]_\varphi\ .$$

2.1.3 Algèbre de Lindenbaum

La plus simple des constructions algébriques permettant de démontrer la complétude est l'*algèbre de Lindenbaum* [116].

Définition 2.1.6 (Classe d'équivalence)**.** *Soit A une formule close. On définit $[A]$ comme la* classe d'équivalence de A*, c'est-à-dire l'ensemble de toutes les formules équiprouvables à A.*

$$[A] = \{B \mid (A \Rightarrow B) \wedge (B \Rightarrow A) \text{ est prouvable}\}.$$

Comme nous raisonnons en calcul des séquents, la relation de prouvabilité de A sous les hypothèses Γ sera notée $\Gamma \vdash A$. L'utilisation de classes d'équivalences permet de définir l'algèbre suivante.

Définition 2.1.7 (Algèbre de Lindenbaum)**.** *On définit l'algèbre de Lindenbaum comme $\Omega = \{[A] \mid A \text{ formule close}\}$. Ω est ordonnée par la relation $\vdash$ elle-même : $[A] \leq [B]$ si, et seulement si, $A \vdash B$. On munit cette algèbre des opérateurs*

— $\bot = [\bot]$,
— $\top = [\top]$,
— $[A] \wedge [B] = [A \wedge B]$,
— $[A] \vee [B] = [A \vee B]$,
— $[A] \Rightarrow [B] = [A \Rightarrow B]$,
— $\neg[A] = [\neg A]$,
— $\forall\{[A[t/x]] \mid t \text{ terme clos }\} = [\forall x A]$,
— $\exists\{[A[t/x]] \mid t \text{ terme clos }\} = [\exists x A]$.

Cette définition est correcte, car le contenu d'une classe d'équivalence ne dépend pas du représentant choisi. En d'autres termes, pour tout $B \in [A]$, on a $[B] = [A]$. Elle s'applique au cas de la logique classique aussi bien qu'intuitionniste ; il faut bien entendu vérifier que les opérateurs introduits dans la définition sont réellement des opérateurs d'algèbre de Boole (cas classique) ou de Heyting (cas intuitionniste). Notons ici que l'algèbre de Lindenbaum n'est pas complète, ce qui pose potentiellement problème pour l'interprétation des quantificateurs $\forall$ et $\exists$. Cependant, elle est "suffisamment complète", en ce sens qu'elle permet de définir un nombre suffisant de bornes inférieures et supérieures pour que les quantificateurs soient interprétés. On dit *définitionnellement complète* [133]. Enfin, il faut pour terminer vérifier que $\leq$ est une relation d'ordre.

Cela nous permet de définir le modèle suivant :

Définition 2.1.8 (Interprétation dans l'algèbre de Lindenbaum)**.** *Le domaine D est l'ensemble des termes ; l'interprétation des symboles de fonction et de prédicat est définie par*

$$\begin{array}{rcl} \hat{f} : t_1, \cdots, t_n & \longmapsto & f(t_1, \cdots, t_n), \\ \hat{P} : t_1, \cdots, t_n & \longmapsto & [P(t_1, \cdots, t_n)]. \end{array}$$

Pour l'interprétation $[\![\;]\!]$ engendrée par la définition 2.1.8 (voir la définition 2.1.4), on vérifie que, pour tout terme t, toute formule A et toute valuation φ qui lie au moins les variables libres de t et A, on a $[\![t]\!]_\varphi = t\varphi$ et $[\![A]\!]_\varphi = [A\varphi]$. Ces égalités sont bien formées à la condition d'identifier à une substitution la valuation φ, qui associe des éléments de D, c'est à dire des termes clos, à des variables. Cette algèbre permet ensuite de démontrer le théorème de complétude.

Théorème 2.1.9 (Complétude du calcul des séquents)**.** *Soient A et B deux formules. Supposons que, dans toute algèbre, pour toute interprétation et toute valuation φ, $[\![A]\!]_\varphi \leq [\![B]\!]_\varphi$. Alors $A \vdash B$.*

De même, soit Γ un ensemble de formules. Supposons que, dans toute algèbre, pour toute interprétation et toute valuation φ, $\bigwedge [\![\Gamma]\!]_\varphi \leq [\![B]\!]_\varphi$. Alors $\Gamma \vdash B$.

Ici, étant donné $\Gamma = A_1, \cdots, A_n$, nous utilisons la notation $\bigwedge [\![\Gamma]\!]_\varphi$ pour $[\![A_1]\!]_\varphi \wedge \cdots \wedge [\![A_n]\!]_\varphi$. La preuve découle immédiatement du travail qui a été fait précédemment. $[\![A]\!]_\varphi \leq [\![B]\!]_\varphi$ est vrai dans tous les modèles, en particulier le modèle défini ci-dessus. En prenant l'identité pour φ, on obtient $[A] \leq [B]$, soit par définition $A \vdash B$.

Le théorème et sa preuve ne dépendent pas de la nature classique ou intuitionniste du calcul des séquents. Bien entendu, l'algèbre de Lindenbaum est dans le cas de la logique classique, une algèbre de Boole et, dans le cas de la logique intuitionniste une algèbre de Heyting. Dans le cas de la logique classique, il est possible de prouver le même résultat de complétude en considérant un nombre arbitraire de formules Δ à droite.

2.1.4 Complétion d'une théorie

Nous nous plaçons ici dans le cadre de la logique classique uniquement. La sémantique étudiée sera donc celle des algèbres de Boole. Comme nous le verrons à la section 2.5, un tel travail de complétion est aussi possible en logique intuitionniste mais demande des outils différents. Cela rendrait une présentation unifiée, sur le modèle de la section précédente, peu intéressante, voire contre-productive.

Une seconde méthode de démonstration du théorème de complétude, historiquement la première à avoir été définie par Gödel [68] puis modernisée par Henkin [9], consiste à s'intéresser à l'algèbre de Boole non triviale la plus élémentaire $\{\bot, \top\}$ (ou encore $\{0, 1\}$) munie des opérateurs booléens standards. Cette méthode est non constructive et, dans cette section, s'applique à la logique classique.

Cela permet de démontrer le théorème suivant, a priori légèrement plus fort que le théorème 2.1.9, puisque l'hypothèse de validité est plus faible. Notons que, en cohérence avec la signification du séquent $\Gamma \vdash \Delta$ rappelée au tout début de la section 2.1, nous interprétons Γ comme une conjonction et Δ comme une disjonction.

Théorème 2.1.10 (Complétude)**.** *Soient Γ et Δ deux ensembles de formules. Supposons que, pour tout domaine, toute interprétation dans l'algèbre $\{\bot, \top\}$ et toute valuation φ, si $\bigwedge [\![\Gamma]\!]_\varphi = 1$ alors $\bigvee [\![\Delta]\!]_\varphi = 1$.*

Alors $\Gamma \vdash \Delta$ est prouvable en calcul des séquents classique.

En réalité, l'algèbre de Boole $\{\bot, \top\}$ est *initiale*, ce qui signifie que, étant donnée une algèbre de Boole quelconque, on peut, par un quotient approprié et généralement non constructif, toujours se ramener à celle-ci [33], tout en préservant suffisamment de propriétés de l'interprétation $[\![\ ]\!]$ pour pouvoir conclure. Ce théorème est donc strictement équivalent au théorème 2.1.9. L'existence d'une algèbre initiale est spécifique aux algèbres de Boole et ne s'étend ni aux algèbres de Heyting, ni aux structures de Kripke, par exemple.

La méthode est cependant instructive, et promise à un brillant avenir. L'algèbre de Lindenbaum n'est pas une algèbre de Boole à deux éléments ; par exemple, dans l'algèbre de Lindenbaum de la définition 2.1.7, on a $[A \wedge B] < [A] < [A \vee B]$, ce qui implique que nous ne pouvons plus procéder comme à la section 2.1.3. En effet, on n'y prend aucun parti sur la validité d'une formule, alors que l'algèbre $\{\bot, \top\}$ nous impose de décider pour chaque formule A si elle est vraie ou fausse.

La méthode décrite ici fonctionne par contradiction : supposons que le séquent $\Gamma \vdash \Delta$ ne soit pas prouvable, ce qui est noté $\Gamma \nvdash \Delta$. Cet énoncé est équivalent à la non prouvabilité du séquent $\Gamma, \neg\Delta \vdash$ (noté, donc, $\Gamma, \neg\Delta \nvdash$), il y a de nombreuses manières de s'en persuader, dont l'utilisation de la règle de coupure ou d'inversion de Kleene [91] (valable y compris en Déduction modulo théorie [79]) si l'on souhaite s'en passer.

Définition 2.1.11 (Cohérence)**.** *Un ensemble de formules Γ est dit* cohérent *lorsque le séquent $\Gamma \vdash$ n'est pas prouvable.*

La terminologie “cohérent” est justifiée par le fait classique (au sens de logique classique) que si Γ n'est pas cohérent alors, par la règle d'affaiblissment weak$_R$, il est possible de prouver $\Gamma \vdash C$ pour tout C. Comme nous allons le voir, tout ensemble de formules cohérent admet au moins un modèle.

Étant donné un ensemble de formules cohérent $\Gamma, \neg\Delta$, nous cherchons une interprétation dans $\{0, 1\}$ telle que $\bigwedge [\![\Gamma]\!] = \bigwedge [\![\neg\Delta]\!] = 1$. Il existera donc une interprétation dans $\{0, 1\}$ qui ne validera pas l'hypothèse “si $\bigwedge [\![\Gamma]\!]_\varphi = 1$, alors $\bigvee [\![\Delta]\!]_\varphi = 1$”, et le théorème sera prouvé. Le domaine de ce modèle sera, comme de coutume, l'ensemble des termes clos.

Pour définir l'interprétation $[\![\ ]\!]$, nous allons énumérer toutes les formules une à une et, à chaque pas, définir $[\![\ ]\!]$ sur la formule courante.

1. On commence par fixer $\Gamma_0 = \Gamma, \neg\Delta$, et on pose, pour tout $A \in \Gamma_0$, $[\![A]\!] = 1$. Pour l'instant, rien ne nous dit que $[\![\ ]\!]$ est une fonction d'interprétation et, en effet, ce n'en est pas une. En revanche, nous savons que Γ_0 est cohérent.
2. Supposons Γ_i déjà défini et cohérent, tel que, pour tout $A \in \Gamma_i$, $[\![A]\!] = 1$. Considérons la formule A_i. Un des deux ensembles Γ_i, A_i et $\Gamma_i, \neg A_i$ est cohérent, sinon (en utilisant la coupure) Γ_i n'est pas cohérent.

   Dans le premier cas, on pose $\Gamma_{i+1} := \Gamma_i, A_i$ et $[\![A_i]\!] = 1$; dans le second, $\Gamma_{i+1} := \Gamma_i, \neg A_i$ et $[\![A_i]\!] = 0$.
3. Deux cas particuliers. Si Γ_i, A_i est cohérent et $A_i = \exists x B$, on prend une constante fraîche (appelé témoin de Henkin) c et on pose $\Gamma_{i+1} := \Gamma_i, A_i, B[c/x]$ ainsi que $[\![A_i]\!] = 1$ et $[\![B[c/x]]\!] = 1$. De même, si $\Gamma_i, \neg A_i$ est cohérent et $\neg A_i = \neg\forall x B$, on prend une constante fraîche (témoin de Henkin) c et on pose $\Gamma_{i+1} := \Gamma_i, \neg A_i, \neg B[c/x]$ ainsi que $[\![A_i]\!] = 0$ et $[\![B[c/x]]\!] = 0$.

Nous obtenons une chaîne dénombrable d'ensembles finis

$$\Gamma_0 \subseteq \Gamma_1 \subseteq \cdots \subseteq \Gamma_n \subseteq \cdots ,$$

ce qui nous permet de poser $\Gamma_\infty := \bigcup_{i \in \mathbb{N}} \Gamma_i$.

Tous les ensembles Γ_i étant finis, cette construction est correcte à la seule condition d'avoir un ensemble dénombrable de constantes fraîches dans le langage. On définit la fonction $[\![\ ]\!]$ par l'appartenance, ou non, à Γ_∞. $[\![\ ]\!]$ est compatible par construction avec chacun des Γ_i, et on vérifie qu'elle respecte bien les conditions de compatibilité avec les connecteurs et les quantificateurs (définition 2.1.4). De plus, grâce à la compatibilité de $[\![\ ]\!]$ avec les formules de Γ_0, nous obtenons $\bigwedge [\![\Gamma]\!] = 1$ et $\bigvee [\![\Delta]\!] = 0$, soit une interprétation qui ne satisfait pas les hypothèses du théorème 2.1.10. C'est exactement ce que nous cherchions.

Techniquement, Γ_∞ est une théorie *cohérente maximale* ou *complète* : pour toute formule A, soit $A \in \Gamma_\infty$, soit $\neg A \in \Gamma_\infty$, et tous les sous-ensembles finis de Γ_∞ sont cohérents ce qui, par compacité [90, 59, 84], est équivalent à dire que Γ_∞ est cohérent. Nous avons construit un ultrafiltre [133, 33].

2.1.5 Calcul des séquents sans coupure

Les méthodes des sections 2.1.3 et 2.1.4 précédentes ne s'appliquent pas telles quelles au calcul des séquents sans coupure. Nous verrons plus tard comment étendre les idées sous-jacentes aux algèbres de Lindenbaum ; pour le moment, intéressons nous aux méthodes de complétude de la section 2.1.4.

La règle de coupure est nécessaire pour choisir entre Γ_i, A_i et $\Gamma_i, \neg A_i$, ce qui est à son tour nécessaire pour imposer une valeur correcte à $[\![A_i]\!]$. Une telle agressivité peut cependant être évitée, ce qui permet dans le même temps d'éviter le recours à la règle de coupure dans les preuves. On cherche en effet avant tout à assurer les propriétés suivantes pour la fonction $[\![\ ]\!]$:

1. $[\![\ ]\!]$ doit être en accord avec Γ_0 : pour tout $A \in \Gamma_0$, $[\![A]\!] = 1$;
2. $[\![\ ]\!]$ doit être en accord avec la définition d'interprétation pour les connecteurs et les quantificateurs.

En particulier, si les deux propriétés ci-dessus sont respectées, mais que $[\![\ ]\!]$ est une fonction *partielle*, il est possible d'étendre $[\![\ ]\!]$ de manière à ce qu'elle devienne *totale* : la définition 2.1.4 étant une définition inductive, il suffit de donner une interprétation par défaut, quelconque (nous reviendrons sur ce point à la section 2.3.1), à tous les atomes $P(t_1, \cdots, t_n)$ n'appartenant déjà pas au domaine de $[\![\ ]\!]$, c'est-à-dire tels que $[\![P(t_1, \cdots, t_n)]\!]$ n'est pas encore défini. La compatibilité de la fonction $[\![\ ]\!]$ ainsi rendue totale avec Γ_0 est assurée, parce que nous avons étendu la fonction partielle $[\![\ ]\!]$. Il s'agit bien d'une interprétation, parce que $[\![\ ]\!]$ respecte les conditions de connecteurs et de quantificateurs, soit par construction lorsque $[\![A]\!]$ n'était pas fixée au départ, soit par hypothèse, car la fonction partielle initiale les respectait.

C'est la seconde condition ci-dessus qui forme la définition de *valuation partielle*, due à Schütte [123, 125, 124].

Définition 2.1.12 (Valuation partielle)**.** *Une* valuation partielle *est une fonction partielle des formules dans* $\{0, 1\}$ *qui, quand elle est définie, respecte les conditions de la définition 2.1.4.*

Par exemple, $[\![A \wedge B]\!]$ est définie et vaut 1 si, et seulement si, $[\![A]\!]$ et $[\![B]\!]$ sont définies et valent 1, alors que $[\![A \wedge B]\!]$ est définie et vaut 0 si, et seulement si, $[\![A]\!]$ est définie et vaut 0 ou $[\![B]\!]$ est définie et vaut 0. Voir aussi la définition 2.2.1 ci-dessous, très similaire.

Proposition 2.1.13. *Soit $\mathcal{V}$ une valuation partielle. Il existe une fonction d'interprétation $[\![\ ]\!]$ telle que, pour toute formule A telle que $\mathcal{V}(A)$ est défini, $[\![A]\!] = \mathcal{V}(A)$.*

Munis de ces outils, nous devons maintenant définir une valuation partielle telle que $[\![A]\!] = 1$ pour tout $A \in \Gamma_0$. Pour ce faire, il nous faut encore compléter Γ_0, mais on peut le faire de manière plus minimale qu'à la section 2.1.4, en ne s'intéressant qu'aux seules formules telles que Γ_i, A_i est cohérente à l'étape 2, sans se poser la question pour $\neg A_i$. En définissant $\Gamma_\infty := \bigcup_{i \in \mathbb{N}} \Gamma_i$ et en posant $[\![A]\!] = 1$, si $A \in \Gamma_\infty$, et $[\![A]\!] = 0$, si $\neg A \in \Gamma_\infty$, on démontre que $[\![\ ]\!]$ est une valuation partielle, et ce *sans utiliser la règle de coupure*.

Lors de la construction, nous ne forçons pas, absolument sciemment, à ce que soit A soit $\neg A$ appartienne à Γ_∞. C'est la clef de l'amélioration de la construction. A posteriori, nous savons cependant que Γ_∞ comporte en réalité soit A, soit $\neg A$, pour toute formule A, de manière similaire au résultat de la section 2.1.4. Mais il est impossible de le démontrer sans se servir de la règle de coupure et il faut obligatoirement autoriser, dans la construction, la possibilité que Γ_i soit cohérente et, en même temps, qu'à la fois Γ_i, A_i et $\Gamma_i, \neg A_i$ soient incohérentes. Ceci ne nous dérange aucunement, au contraire, et nous permet de conclure, comme à la section 2.1.4, à la complétude, mais cette fois-ci du calcul des séquents classique sans coupure.

Théorème 2.1.14 (Complétude forte)**.** *Soient Γ et Δ deux ensembles de formules. Supposons que, pour tout domaine, toute interprétation dans l'algèbre $\{0, 1\}$ et toute valuation φ, si $\bigwedge [\![\Gamma]\!]_\varphi = 1$, alors $\bigvee [\![\Delta]\!]_\varphi = 1$.*

Alors $\Gamma \vdash \Delta$ est prouvable en calcul des séquents classique, sans coupure.

C'est exactement cette approche qui a été développée dans la thèse de Doctorat [77], étendue à la Déduction modulo théorie en logique classique et intuitionniste. Ce sont à leur tour ces dernières méthodes que nous étendons dans le reste de ce chapitre.

2.2 Tableaux classiques

Comme nous allons le voir dans cette section, derrière la construction de modèle par complétion de la section précédente, se cache un algorithme de recherche de preuve utilisant la méthode des tableaux. Cela devient apparent après quelques optimisations sur le processus de complétion sans coupure d'une théorie.

2.2.1 Valuations partielles épurées : semi-valuations

Savoir construire une valuation partielle en accord avec le séquent suffit à prouver la complétude du calcul des séquents, y compris en l'absence de la règle de coupure. En effet, une fois cet objectif atteint, nous sommes en mesure d'étendre la valuation partielle en une valuation totale, c'est-à-dire une interprétation dans le modèle syntaxique ("de termes"), c'est-à-dire le modèle où l'algèbre de Boole est $\{0, 1\}$ et où le domaine est formé des termes clos.

La construction d'une valuation partielle est plus aisée que la construction d'une valuation totale ; c'est la différence entre la méthode de la section 2.1.4 et celle de la section 2.1.5. La conséquence immédiate est de passer de la complétude du calcul des séquents à la complétude du calcul des séquents sans coupure. Il est possible d'affaiblir encore la contrainte et de demander simplement la possibilité de construire une *semi-valuation* ou, de manière équivalente, un ensemble de formules Γ qui vérifie une "abstract consistency property" [59], encore appelée "analytic consistency property" [125].

Définition 2.2.1 (Semi-valuation)**.** *Une* semi-valuation $\mathcal{V}$ *est une fonction partielle de* $\mathcal{F}orm$*, l'ensemble des formules, dans* $\{0, 1\}$ *qui respecte les conditions du tableau ci-dessous.*

Si	*Alors*
$\mathcal{V}(A \wedge B) = 1$	$\mathcal{V}(A) = 1$ *et* $\mathcal{V}(B) = 1$
$\mathcal{V}(A \vee B) = 1$	$\mathcal{V}(A) = 1$ *ou* $\mathcal{V}(B) = 1$
$\mathcal{V}(A \Rightarrow B) = 1$	$\mathcal{V}(A) = 0$ *ou* $\mathcal{V}(B) = 1$
$\mathcal{V}(\neg A) = 1$	$\mathcal{V}(A) = 0$
$\mathcal{V}(\forall x A) = 1$	$\mathcal{V}(A[t/x]) = 1$ *pour tout t*
$\mathcal{V}(\exists x A) = 1$	$\mathcal{V}(A[t/x]) = 1$ *pour au moins un t*
$\mathcal{V}(A \wedge B) = 0$	$\mathcal{V}(A) = 0$ *ou* $\mathcal{V}(B) = 0$
$\mathcal{V}(A \vee B) = 0$	$\mathcal{V}(A) = 0$ *et* $\mathcal{V}(B) = 0$
$\mathcal{V}(A \Rightarrow B) = 0$	$\mathcal{V}(A) = 1$ *et* $\mathcal{V}(B) = 0$
$\mathcal{V}(\neg A) = 0$	$\mathcal{V}(A) = 1$
$\mathcal{V}(\forall x A) = 0$	$\mathcal{V}(A[t/x]) = 0$ *pour au moins un t*
$\mathcal{V}(\exists x A) = 0$	$\mathcal{V}(A[t/x]) = 0$ *pour tout t*

La différence entre valuation partielle et semi-valuation tient à ce que, là où une valuation partielle impose que $[\![A]\!]$ est définie si, et seulement si, un nombre suffisant de ses sous-formules a une interprétation définie et cohérente avec $[\![A]\!]$, une semi-valuation impose seulement que, si $[\![A]\!]$ est définie, alors un nombre suffisant de ses sous-formules a une interprétation définie et cohérente avec $[\![A]\!]$. Par exemple, supposons que $[\![\forall x A]\!] = 1$. Que $[\![\ ]\!]$ soit une valuation partielle ou une semi-valuation, on aura $[\![A[t/x]]\!] = 1$ pour tout terme t. De plus, si $[\![\ ]\!]$ est une valuation partielle, on aura $[\![\forall x(A \wedge A)]\!] = 1$, alors que si $[\![\ ]\!]$ est seulement une semi-valuation, $[\![\forall x(A \wedge A)]\!]$ peut ne pas être définie (si elle l'est, alors elle vaudra forcément 1).

Comme nous l'avons vu, il est possible, dans le modèle booléen ayant pour domaine les termes, de construire à partir d'une valuation partielle une interprétation qui en est l'extension totale. De manière similaire, il est possible d'étendre, par induction, une semi-valuation en valuation partielle.

Proposition 2.2.2. *Soit* $\mathcal{V}$ *une semi-valuation. Il existe une valuation partielle* $\hat{\mathcal{V}}$ *telle que, pour toute formule* A *telle que* $\mathcal{V}(A)$ *est défini,* $\hat{\mathcal{V}}(A) = \mathcal{V}(A)$*.*

L'extension se fait par induction sur la taille des formules pour définir $\hat{\mathcal{V}}(A)$, toujours en s'assurant que $\hat{\mathcal{V}}$ respecte les conditions sur les connecteurs et les quantificateurs pour être une interprétation.

2.2.2 Construction d'une semi-valuation

Le problème initial de la complétude du calcul des séquents, qui était de trouver une interprétation satisfaisant le séquent $\Gamma \vdash \Delta$, s'est successivement réduit à

— compléter l'ensemble des formules $\Gamma, \neg\Delta$ par énumération de toutes les formules, afin de construire un ensemble cohérent maximal qui est une valuation ;
— compléter l'ensemble des formules $\Gamma, \neg\Delta$ par énumération de toutes les formules, afin de construire un ensemble cohérent qui est une valuation partielle ;
— compléter l'ensemble des formules $\Gamma, \neg\Delta$ (par une procédé à définir) afin de construire un ensemble cohérent qui est une semi-valuation.

La question qui subsiste, puisque nous cherchons dorénavant une semi-valuation, est celle de l'énumération des sous-formules de $\Gamma, \neg\Delta$ afin de se conformer à la définition 2.2.1 en fixant la valeur de certaines de ces sous-formules. Il serait possible de construire une énumération comme à la section 2.1.5, mais nous pouvons faire beaucoup mieux.

En effet, l'énumération des sous-formules est *exactement* ce que nous fournit la méthode des tableaux : des règles d'inférence qui, étant donnée une condition sur une formule, par exemple $T\ A \wedge B$ (qui se lit "$A \wedge B$ doit être vrai – T), nous indique les conditions nécessaires sur les sous-formules pour que nous ayons une semi-valuation, dans notre exemple $T\ A$ et $T\ B$ (A doit être vrai et B aussi). F joue le rôle de "faux", de manière similaire. T et F sont appelés *signes* et les formules du tableau sont donc *signées*.

Un tableau avec racine $T\ A_1, \cdots, T\ A_n, F\ B_1, \cdots, F\ B_m$ peut donc se comprendre comme une tentative de construire une semi-valuation dans laquelle $\mathcal{V}(A_i) = 1$ et $\mathcal{V}(B_j) = 0$, c'est à dire en accord avec le séquent $\Gamma, \neg\Delta \nvdash$, supposé cohérent. Le branchement correspond à deux possibilités, en accord avec la définition 2.2.1. Par exemple $T\ A \vee B$ créé une première branche $T\ A$ et une seconde $T\ B$, ce qui correspond à deux semi-valuations différentes, une où l'on aurait $\mathcal{V}(A) = 1$ et $\mathcal{V}(B)$ inconnu, et l'autre où cela serait l'inverse. La définition 2.2.1 est dans les deux cas satisfaite. Au niveau des séquents, cela signifie que, si nous supposons $\Gamma, A \vee B, \neg\Delta \nvdash$, alors au moins l'une des deux branches est cohérente : soit $\Gamma, A \vee B, A, \neg\Delta \nvdash$ soit $\Gamma, A \vee B, B, \neg\Delta \nvdash$. Il faut vérifier cela pour toutes les règles de tableau, rappelés à la figure 2.2, mais cela est une formalité. Notons en particulier la ressemblance entre les règles de tableaux et les règles de séquent, à gauche pour les formules signées par T et à droite pour celles signées par F.

Ainsi, si nous supposons que la racine $T\ \Gamma, F\ \Delta$ est cohérente (c'est à dire $\Gamma, \neg\Delta \nvdash$), au moins une des branches doit rester ouverte et celle-ci génère une semi-valuation à la condition suivante : l'énumération des sous-formules est effectuée correctement. Puisque nous avons l'identité "sous-formule = conclusion de la règle des tableaux", cela se traduit en : pour toute formule signée non atomique $T/F\ A$, la règle de tableau correspondante lui est appliquée.

Tout ce dont nous avons besoin est donc d'une procédure d'application

Règles α

$$\frac{T\ A \wedge B}{T\ A, T\ B}\,T_\wedge \qquad \frac{F\ A \vee B}{F\ A, F\ B}\,F_\vee \qquad \frac{F\ A \Rightarrow B}{T\ A, F\ B}\,F_\Rightarrow$$

$$\frac{T\ \neg A}{F\ A}\,T_\neg \qquad \frac{F\ \neg A}{T\ A}\,F_\neg$$

Règles β

$$\frac{T\ A \vee B}{T\ A \qquad T\ B}\,T_\vee \qquad \frac{F\ A \wedge B}{F\ A \qquad F\ B}\,F_\wedge \qquad \frac{T\ A \Rightarrow B}{F\ A \qquad T\ B}\,T_\Rightarrow$$

Règles δ

$$\frac{T\ \exists x A}{T\ A[c/x]}\,T_\forall \qquad \frac{F\ \forall x A}{F\ A[c/x]}\,F_\forall$$

Règles γ

$$\frac{T\ \forall x A}{T\ \forall x A, T\ A[t/x]}\,T_\forall \qquad \frac{F\ \exists x A}{F\ \exists x A, F\ A[t/x]}\,F_\exists$$

FIGURE 2.2 – Règles des tableaux en logique classique

des règles de tableaux (on dit procédure d'expansion) qui soit *équitable* ; qui assure que toutes les règles applicables seront, in fine, appliquées au moins une fois, à moins qu'une contradiction ne soit trouvée. Tous les algorithmes ne sont pas équitables, considérons l'exemple caricatural suivant : prendre la première formule de la racine, appliquer la règle de tableaux correspondante, et recommencer. Elle construira par exemple le tableau suivant.

$$\cfrac{T\ \neg A \wedge B, T\ A \wedge C}{\cfrac{T\ \neg A, T\ B}{\cfrac{T\ \neg A, T\ B}{\vdots}\,\alpha_\wedge}\,\alpha_\wedge}\,\alpha_\wedge$$

Ce tableau peut évidemment être fermé, mais l'algorithme décrit ici n'y arrivera jamais. Ce même algorithme, appliqué cette fois-ci à un tableau qui a une racine cohérente, comme $T\ A \vee B, F\ A \wedge C$ ne permettra jamais d'atteindre une semi-valuation. Une première tentative de procédure équitable peut être la procédure suivante.

Définition 2.2.3 (Procédure d'expansion de tableaux équitable)**.** *Étant donné un tableau où les formules signées sont marquées "utilisée" ou "non utilisée", prendre la première formule signée non utilisée, en parcourant l'arbre en largeur. Appliquer la règle de tableau correspondante, et étendre simultanément toutes les feuilles ouvertes qui sont situées sous cette formule.*

Le parcours en largeur d'abord n'est pas nécessaire, si nous savons à l'avance quelle est la branche qui est cohérente. Dans ce cas, nous pouvons nous concentrer sur cette branche par un parcours préfixe en profondeur. Mais cette information est en réalité très difficile à obtenir ; la détection de contradiction est justement un des buts de la méthode des tableaux. Pour des raisons pratiques, le parcours en largeur est donc important. Notons aussi que cette procédure d'expansion ne cherche pas à fermer les branches qui comportent une contradiction explicite ; ceci pourrait être fait, notamment pour des raisons d'efficacité, mais n'est pas une nécessité.

La procédure de la définition 2.2.3 est tout ce qui nous manquait pour démontrer le théorème de complétude du calcul des séquents, avec ou sans coupure. Il faut simplement bien faire attention à énumérer toutes les instances des variables lors de l'application des règles $T_\forall$ et $F_\exists$ sur une même formule afin de garantir

Définition 2.2.4 (Branche complète)**.** *Une branche ouverte est dite* complète *si toutes les formules universelles positives $T\ \forall x A$ et toutes les formules existentielles négatives $F\ \exists x B$ sont instanciées par tous les termes clos : pour tout terme clos t, $T\ A[t/x]$ et $F\ B[t/x]$ apparaissent sur cette branche.*

Une branche complète est infinie, sauf dans le cas rare où ni formule universelle positive ni formule existentielle négative n'apparaît sur celle-ci. C'est donc la limite d'une suite de branches finies construites par une procédure complète.

Définition 2.2.5 (Procédure complète)**.** *Une procédure équitable de tableaux est dite* complète *lorsque toute branche ouverte infinie est complète.*

Une manière usuelle d'avoir une procédure complète est de considérer une procédure équitable, par exemple celle de la définition 2.2.3, et de demander à ce qu'elle soit "super-équitable" pour les formules universelles positives et les formules existentielles négatives : les règles $T_\forall$ et $F_\exists$ doivent leur être appliquées un nombre arbitraire de fois, en instanciant à l'itération n par le terme t_n, selon une énumération des termes clos du langage.

Afin de ne pas mettre en danger la contrainte d'équité vis-à-vis des autres formules signées du tableau, une méthode standard consiste à répéter les formules signées en *bas* de tableau à chaque fois que l'on applique la règle et de marquer la formule *courante* comme utilisée. Ainsi, nous sommes certains de revenir traiter la formule signée, mais seulement *après* avoir traité toutes les autres. Ce sont ces règles qui ont été présentées figure 2.2.

Proposition 2.2.6. *Soit une procédure complète et un tableau. Si cette procédure n'arrive jamais à fermer le tableau, alors elle génère une branche complète, et une semi-valuation $\mathcal{V}$ telle que si les formules signées $T\ A$ et $F\ B$ apparaissent sur cette branche, $\mathcal{V}(A) = 1$ et $\mathcal{V}(B) = 0$.*

La preuve de cette proposition est la même que la preuve du théorème 2.2.7 ci-dessous.

Les méthodes développées cette section sont quasiment constructives ; par exemple, l'étape d'énumération des sous-formules est très exactement spécifiée, et la procédure de la définition 2.2.3 ne demande pas à savoir laquelle des branches est cohérente pour pouvoir continuer à s'appliquer : elle s'occupe de toutes les branches en même temps, du moins celles qu'elle n'a pas encore explicitement identifiées comme incohérentes. Il manque encore quelques éléments — notamment la notion de branche ouverte infinie est non constructive — qui seront discutés plus tard dans ce chapitre.

Nous obtenons tout de même le résultat voulu, à savoir la complétude forte du calcul des séquents (théorème 2.1.14), pour le moment par contraposée. Supposons le séquent $\Gamma \vdash \Delta$ non prouvable (sans coupure) ; nous sommes en mesure de construire un tableau ayant pour racine $T\ \Gamma, F\ \Delta$ et qui est cohérent, ce qui par la proposition 2.2.6 génère une semi-valuation, à la suite de l'application de la procédure complète, que l'on étend par les propositions 2.2.2 et 2.1.13 en une fonction d'interprétation $[\![\ ]\!]$. Cette fonction d'interprétation est telle que $[\![\Gamma]\!] = 1$ et $[\![\Delta]\!] = 0$, ce qui contredit les hypothèses du théorème 2.1.14.

2.2.3 Complétude des tableaux classiques

Au lieu de nous intéresser à la complétude du calcul des séquents, nous pouvons nous intéresser directement à la complétude de la méthode des tableaux. Cette étude a été menée à l'origine par Beth [12], Schütte [123], Kanger [88], Hintikka et Smullyan [125] ; elle s'applique aujourd'hui à de nombreuses autres logiques, comme la logique d'ordre supérieur.

Le résultat à démontrer est l'équivalent du théorème 2.1.14, mais, comme plus haut, nous préférerons travailler avec sa contraposée et, puisque avoir une semi-valuation est suffisant pour construire un modèle booléen, nous démontrerons le théorème suivant.

Théorème 2.2.7 (Complétude de la méthode des tableaux)**.** *Si aucune procédure d'expansion de tableaux ne parvient à fermer toutes les branches du tableau dont la racine est $T\ A_1, \cdots, T\ A_n, F\ B_1, \cdots, F\ B_m$, alors il existe une semi-valuation $\mathcal{V}$ telle que $\mathcal{V}(A_i) = 1$ et $\mathcal{V}(B_j) = 0$.*

La preuve fait appel à une procédure de tableaux complète, par exemple la procédure de la définition 2.2.3, modifiée pour instancier les γ-formules par tous les termes successivement, comme mentionné après la définition 2.2.5. Puisque, par hypothèse, aucune procédure d'expansion ne termine, cette procédure complète ne termine pas non plus. En considérant une branche ouverte du tableau limite $\mathcal{T}_\infty$ — l'union de toutes les tableaux finis obtenus lors de l'application de la procédure —, nous obtenons exactement ce qu'il nous faut pour définir une semi-valuation $\mathcal{V}$, en posant $\mathcal{V}(A) = 1$ si $T\ A$ apparaît sur la branche, $\mathcal{V}(A) = 0$ si $F\ A$ apparaît, et en laissant $\mathcal{V}(A)$ indéterminé sinon. La branche étant ouverte, on ne peut définir $\mathcal{V}(A)$ de deux manières contradictoires, et la semi-valuation est en accord avec toutes les formules de cette branche, en particulier la racine $T\ A_1, \cdots, T\ A_n, F\ B_1, \cdots, F\ B_m$. La compatibilité avec la définition 2.2.1 est assurée par les règles de tableaux et la nature complète de la branche.

En pratique, toutes les procédures complètes ne sont pas aussi efficaces les unes que les autres. Notamment, il est plus économique d'introduire des branchements le plus tard possible ; aussi, on préférera appliquer les règles dans l'ordre de la définition ci-dessous.

Définition 2.2.8 (Procédure complète)**.** *Appliquer les règles de tableaux selon l'ordre de priorité suivant :*

1. *clôture ;*
2. *α et δ, avec un parcours en largeur total ;*
3. *β, avec un parcours en largeur total, en revenant éventuellement au point 1 si nécessaire ;*
4. *γ, sur la/les premières k γ-formules non utilisées, rencontrées par un parcours en largeur.*

Cette procédure reste équitable, car l'application d'une règle α, β ou δ, si elle génère l'application potentielle d'autres règles, fait décroître la complexité des sous-formules. Les étapes 2 et 3 terminent donc, au contraire des règles γ, qui recopient la formule non instanciée en bas de tableau. Nous devons donc arrêter au bout d'un certain temps l'application des γ-règles, d'où le paramètre k, qui permet de revenir à l'étape initiale 1.

2.3 Raisonner modulo théorie

Nous cherchons dans cette section à étendre les notions des sections 2.1 et 2.2 à la Déduction modulo théorie.

2.3.1 Séquents et modèles

En Déduction modulo théorie, on souhaite travailler avec des règles de réécriture sur les termes, par exemple

$$0 + x \longrightarrow_{\mathcal{R}} x,$$

et des règles de réécriture sur les formules, par exemple

$$Even(0) \longrightarrow_{\mathcal{R}} \top.$$

Ces règles de réécriture sont prises en compte dans le calcul syntaxique en identifiant les formules et, dans une moindre mesure, les termes, modulo la relation de congruence $\equiv$ engendrée par les règles de réécriture. Concrètement, on peut modifier le calcul des séquents de la figure 2.1 en ajoutant les deux règles de conversion de la figure 2.3. Les versions classiques et intuitionnistes, comme à la section 2.1, diffèrent en ce qu'elles autorisent, respectivement, plusieurs ou au plus une formule à droite des séquents.

$$\frac{\Gamma, A \vdash \Delta}{\Gamma, B \vdash \Delta}\ \mathrm{conv}_L, A \equiv B \qquad \frac{\Gamma \vdash A, \Delta}{\Gamma \vdash B, \Delta}\ \mathrm{conv}_R, A \equiv B$$

FIGURE 2.3 – Règles de conversion du calcul des séquents modulo théorie

Les règles de conversion à gauche et à droite permettent de convertir une formule A en une autre formule B, dès lors qu'elles sont équivalentes modulo la relation de réécriture générée par le système $\mathcal{RE}$ considéré. D'autres présentations du calcul des séquents en Déduction modulo théorie sont possibles. Pour plus de détails sur la réécriture ou le calcul des séquents, voir par exemple [49, 77, 85].

En ce qui concerne la sémantique, les notions d'algèbre de Boole et de Heyting restent rigoureusement identiques à celles de la section 2.1.2. La congruence générée par la relation de réécriture se reflète au niveau de la définition d'interprétation, qui doit refléter l'identité que l'on souhaite imposer aux formules et termes convertibles par réécriture.

Définition 2.3.1 (Interprétation en Déduction modulo théorie)**.** *Soit $\mathcal{RE}$ un système de réécriture. Étant donnée une algèbre, un domaine et une interprétation $\hat{}$, soit $[\![\ ]\!]$ l'interprétation générée. On dit que cette interprétation est un* modèle de $\mathcal{RE}$ *si, et seulement si, pour tous les termes t, u et formules A, B, on a :*

- *si $t \equiv u$, alors $[\![t]\!] = [\![u]\!]$,*
- *si $A \equiv B$, alors $[\![A]\!] = [\![B]\!]$.*

Les interprétations $[\![\ ]\!]$, pour être compatibles avec un système de réécriture, doivent donc respecter certaines contraintes. Ces dernières sont exactement ce dont nous avions besoin pour démontrer les théorèmes de correction de la même manière que le théorème 2.1.5.

Théorème 2.3.2 (Correction)**.** *Soit $\mathcal{RE}$ un système de réécriture. Soit $A_1, \cdots, A_n \vdash B_1, \cdots, B_m$ un séquent prouvable en calcul des séquents classique (intuitionniste, respectivement) modulo $\mathcal{RE}$. Alors, pour toute algèbre de Boole (de Heyting, respectivement), pour toute interprétation $[\![\ ]\!]$* qui est un modèle de $\mathcal{RE}$ *et toute valuation φ, on a*

$$[\![A_1]\!]_\varphi \wedge \cdots \wedge [\![A_n]\!]_\varphi \leq [\![B_1]\!]_\varphi \vee \cdots \vee [\![B_m]\!]_\varphi .$$

En ce qui concerne la complétude, les contraintes imposées par la définition 2.3.1 pèsent plus lourd, non seulement sur la notion de valuation partielle et de semi-valuation, mais aussi sur la manière de définir et d'étendre celles-ci, de semi-valuation à valuation partielle, et de valuation partielle à interprétation.

Reprenons la méthode de complétion "sans coupure" de la section 2.1.5, qui permet d'obtenir une valuation partielle : nous y étendons la valuation partielle en fixant de manière libre $[\![A]\!]$ pour toutes les formules atomiques A telles que la valuation partielle était indéfinie. Ceci est impossible en Déduction modulo théorie, si l'on veut respecter les règles de réécriture, comme le montre l'exemple de la règle de réécriture $A \longrightarrow \neg B$, avec A et B atomiques. Si nous choisissons (librement) $[\![A]\!] = 1$, cela force $[\![B]\!] = 0$; ce dernier choix n'est donc plus libre. Il n'est même plus évident que, en fonction de l'ensemble de règles $\mathcal{RE}$ choisi, il soit possible d'obtenir une interprétation ; par exemple la règle de réécriture $A \longrightarrow \neg A$ introduirait des contraintes insatisfiables.

C'est cette nouvelle contrainte qui fait la difficulté de l'extension des démonstration de complétude en Déduction modulo théorie, et qui sera étudiée par la suite ; nous commencerons par la complétude des tableaux en logique classique modulo théorie, pour laquelle nous nous appuierons sur les résultats de la section 2.2, en ajoutant la difficulté de la Déduction modulo théorie.

En revanche, la complétude des tableaux avec variables libres, celle des tableaux intuitionnistes, du calcul des séquents intuitionnistes, etc., seront étudiées directement dans le cadre plus large de la Déduction modulo théorie, sans passer préalablement par des résultats en logique du premier ordre. L'expérience tirée de l'ajout de la réécriture au cas de la complétude de la méthode des tableaux classique modulo théorie des sections 2.3 et 2.4 sera suffisante pour comprendre les difficultés supplémentaires liées au passage à un système différent, y compris en la présence de la technicité requise par la réécriture.

2.3.2 Algèbre de Lindenbaum

Nous nous plaçons maintenant dans le système de preuve du calcul des séquents modulo théorie. Tous les résultats de la section 2.1.3 s'appliquent directement à la Déduction modulo théorie. Seuls le domaine de base et l'interprétation des termes demandent une légère modification. Étant donné un système de réécriture et un terme t, on pose $[t] := \{u \mid t \equiv u\}$, c'est la *classe d'équivalence de réécriture* de t, à ne pas confondre avec la *classe d'équivalence de prouvabilité* $[A]$ des algèbres de Lindenbaum de la définition 2.1.6.

Définition 2.3.3 (Interprétation dans l'algèbre de Lindenbaum en Déduction modulo théorie)**.** *Soit $\mathcal{RE}$ un système de réécriture. Le domaine D du modèle de Lindenbaum est l'ensemble des classes d'équivalence de termes modulo $\mathcal{RE}$; l'interprétation des symboles de fonction et de prédicat est définie par*

$$\begin{array}{rcl} \hat{f} : t_1, \cdots, t_n & \longmapsto & [f(t_1, \cdots, t_n)], \\ \hat{P} : t_1, \cdots, t_n & \longmapsto & [P(t_1, \cdots, t_n)]. \end{array}$$

Tous les autres définitions et résultats sont identiques, à ceci près que la relation de prouvabilité $\vdash$ est celle du calcul des séquents modulo théorie, et non plus du calcul des séquents. La définition 2.3.3 pourrait aussi conserver le même énoncé que la définition 2.1.8, au prix d'une modification de la notion de modèle en Déduction modulo théorie (définition 2.3.1), qui, pour le moment, impose $[\![t]\!] = [\![u]\!]$ dès lors que $t \equiv u$, c'est à dire que t et u sont liés par la relation de réécriture. Cette contrainte n'est strictement nécessaire que pour les formules, et nous pourrions être plus permissifs au niveau de l'interprétation des termes. On obtient donc la complétude du calcul des séquents (intuitionniste ou classique) modulo théorie.

Théorème 2.3.4 (Complétude du calcul des séquents modulo théorie)**.** *Soit $\mathcal{RE}$ un système de réécriture. Soient Γ un ensemble de formules et B une formule. Supposons que, dans toute algèbre, pour toute interprétation qui est un modèle de $\mathcal{RE}$ et toute valuation φ, $\bigwedge [\![\Gamma]\!]_\varphi \leq [\![B]\!]_\varphi$. Alors $\Gamma \vdash B$ est prouvable en calcul des séquents.*

2.3.3 Semi-valuations, valuations partielles

La notion de semi-valuation doit prendre en compte la réécriture. La définition la plus stricte serait la suivante.

Définition 2.3.5 (Semi-valuation et valuation partielle en Déduction modulo théorie)**.** *Soit $\mathcal{RE}$ un système de réécriture. Une semi-valuation $\mathcal{V}$ (respectivement une valuation partielle $[\![\ ]\!]$) est dite* compatible *avec $\mathcal{RE}$ si, pour tout A tel que $\mathcal{V}(A)$ (respectivement $[\![A]\!]$) est défini, alors pour tout B tel que $A \equiv B$, $\mathcal{V}(B) = \mathcal{V}(A)$ (respectivement, $[\![B]\!] = [\![A]\!]$).*

Par la suite, nous utiliserons la terminologie semi-valuation pour indiquer une semi-valuation compatible avec $\mathcal{RE}$, s'il n'y a pas besoin de mettre l'accent sur le système de réécriture.

Cette définition est très contraignante pour les procédures d'énumération, ainsi que pour les méthodes de tableaux. Pour chaque formule A apparaissant sur le tableau, il nous faudrait énumérer toutes les formules B convertibles à A, ce qui peut consommer encore plus de ressources que les règles γ.

D'autre part, nous ne sommes plus certains de savoir étendre une semi-valuation en une interprétation, comme le montre l'exemple de la règle $A \longrightarrow B$ de la section 2.3.1. Considérons le tableau dont la racine est $F\ B$: nous essayons de démontrer B. Aucune règle de tableau ne s'applique ; la branche est ouverte et complète, nous avons donc une semi-valuation ; elle est telle que $\mathcal{V}(B) = 0$. De plus, $\mathcal{V}$ est bien compatible avec la règle de réécriture, puisque $\mathcal{V}(A)$ n'est pas défini. Cependant, on ne peut étendre $\mathcal{V}$ en une fonction d'interprétation qui soit compatible avec la règle de réécriture. En effet, on peut montrer qu'une telle interprétation doit forcément satisfaire $[\![B]\!] = 1$ et $[\![A]\!] = 0$ [52, 77]. Le problème soulevé ici est que la méthode des tableaux n'admet pas la coupure, qui serait indispensable pour pouvoir fermer ce tableau.

Nous avons donc besoin de conditions spécifiques sur le système de réécriture $\mathcal{RE}$ qui permettent d'étendre les semi-valuations en interprétation dans un modèle de $\mathcal{RE}$. Ces conditions seront discutées plus loin dans ce chapitre ; pour le moment nous nous intéressons à l'affaiblissement des conditions de la définition 2.3.5, c'est-à-dire de demander à ce que $\mathcal{V}(B)$ ne soit défini et égal à $\mathcal{V}(A)$ que pour un nombre restreint de formules $B \equiv A$. Nous allons pour cela supposer que le système de réécriture $\mathcal{RE}$ *termine* et est *confluent*.

Définition 2.3.6. *Soit $\mathcal{RE}$ un système de réécriture. Il est dit*

- terminant, *lorsqu'il n'existe pas de chaîne de réécriture $t_0 \longrightarrow t_1 \longrightarrow \cdots \longrightarrow$ infinie ;*
- confluent, *lorsque, pour tous les termes $t_0 \equiv t_1$, il existe un terme u tel que $t_0 \longrightarrow^* u$ et $t_1 \longrightarrow^* u$;*
- convergent, *lorsqu'il est confluent et terminant.*

Une des propriétés des systèmes convergents est que pour tout terme t et pour toute formule A, il existe un unique terme u et une unique formule B tels que

- $t \longrightarrow^* u$ et $A \longrightarrow^* B$;
- il n'existe aucun terme u' et aucune formule C tels que $u \longrightarrow u'$ et $B \longrightarrow C$.

Ce terme et cette formule s'appellent la *forme normale* de t et de A, notées $\downarrow t$ et $\downarrow A$, respectivement. Leur existence est garantie par la terminaison et l'unicité est garantie par la confluence. Pour être précis, on a unicité de la classe d'équivalence modulo $\mathcal{E}$ des formes normales, dans le cas où $\mathcal{E}$ contient des axiomes équationnels. Si l'unification modulo $\mathcal{E}$ est décidable, alors la conversion modulo $\mathcal{RE}$ dans un système convergent est décidable [49].

En présence d'un système de réécriture convergent, une stratégie possible est de conserver toutes les formules en *forme normale* et de raisonner seulement sur celles-ci. Cette stratégie conduit à une procédure d'expansion complète raisonnablement efficace. Attention cependant à ne pas simplement vouloir mettre dans un premier temps les formules en forme normale pour ne plus les réécrire ensuite (prénormalisation) ; cette stratégie naïve échoue dans le cas général, car les règles $T_\forall$ et $F_\exists$ peuvent introduire des formules qui ne sont plus en forme normale, par exemple lorsque l'on instancie $T\ \forall x P(x)$ par 0 et que l'on a la règle de réécriture $P(0) \longrightarrow \bot$. Il faut donc veiller à renormaliser les formules introduites par les γ-règles ; il n'y a pas besoin de le faire pour les autres types de règles.

Nous pouvons nous servir de cette discussion pour raffiner la notion de semi-valuation.

Définition 2.3.7 (Semi-valuation normale)**.** *Soit $\mathcal{RE}$ un système de réécriture convergent. Une semi-valuation $\mathcal{V}$ est dite* normale *si :*
- *pour toute formule A telle que $\mathcal{V}(A)$ est définie, on a $\mathcal{V}(\downarrow A) = \mathcal{V}(A)$;*
- *pour toutes formules A et B unifiable par les équation de $\mathcal{E}$ (noté $A =_\mathcal{E} B$), si $\mathcal{V}(A)$ et $\mathcal{V}(B)$ sont définies, alors $\mathcal{V}(A) = \mathcal{V}(B)$.*

Une branche complète est *normale* si, pour toute formule signée, elle contient aussi la forme normale de cette formule signée. Une telle branche permet de définir une semi-valuation normale.

Cette dernière notion étant plus faible que celle de semi-valuation de la définition 2.3.5, il nous faut une étape supplémentaire : savoir étendre une semi-valuation normale en une semi-valuation compatible avec $\mathcal{RE}$. Un résultat critique qui permet de le faire est la propriété de non-confusion.

Lemme 2.3.8 (Non-confusion des systèmes confluents)**.** *Soit $\mathcal{RE}$ un système de réécriture confluent, et A et B deux formules non atomiques telles que $A \equiv B$. Alors le connecteur principal de A et de B est identique (non-confusion). De plus, les sous-formules sont deux à deux convertibles modulo $\mathcal{RE}$.*

Cette propriété est utilisée pour démontrer que la complétion par réécriture de la semi-valuation normale n'introduit pas de contradiction avec les autres conditions de la définition 2.2.1. Supposons que $\mathcal{V}(A)$ respecte les conditions sur les connecteurs et quantificateurs des semi-valuations sur A, et soit une formule $B \equiv A$. Poser $\mathcal{V}(B) = \mathcal{V}(A)$ continue de respecter les mêmes conditions sur B.

Au lieu de nous intéresser aux semi-valuations normales, nous nous intéresserons de préférence à la notion de semi-valuation suivante :

Définition 2.3.9 (Semi-valuation atomique)**.** *Soit $\mathcal{RE}$ un système de réécriture confluent. Une semi-valuation $\mathcal{V}$ est dite* atomique *si :*

— *pour toute formule atomique A telle que $\mathcal{V}(A)$ est définie, pour toute règle de réécriture $l \longrightarrow r$ applicable à A, il existe au moins une formule B telle que $A \overset{l \longrightarrow r}{\longrightarrow} B$ et $\mathcal{V}(B)$ est définie et égale à $\mathcal{V}(A)$;*

— *pour toutes formules atomiques A et B telles que $\mathcal{V}(A)$ et $\mathcal{V}(B)$ sont définies et $A =_{\mathcal{E}} B$, alors on doit avoir $\mathcal{V}(B) = \mathcal{V}(A)$.*

Nous avons encore affaibli la définition 2.3.7, puisque nous ne nous intéressons plus qu'aux formules atomiques. D'un autre côté, nous appliquons toutes les règles de réécriture possibles, et seulement pas à pas, ce qui est moins optimal que de demander une mise en forme normale.

En l'absence de terminaison, réécrire pas à pas avec chacune des règles applicables est une nécessité. Considérons un système comportant les règles de réécriture $A \longrightarrow B$ et $A \longrightarrow C$, ainsi que d'autres règles assurant la confluence, il existera donc une formule D telle que $B \longrightarrow^* D$ et $C \longrightarrow^* D$. Le tableau $T\ B, F\ C$ doit pouvoir être fermé et, comme nous réécrivons de manière directe, il n'est pas possible de trouver une contradiction qui impliquerait A. Tout ce que nous pouvons faire est réécrire B et C en avant jusqu'à obtenir D, mais nous ignorons quelle suite de règles appliquer ; nous devons donc tout tenter.

L'avantage de la définition 2.3.9 est qu'elle s'applique aussi au cas des systèmes de réécriture non terminant, pour autant qu'ils soient confluents, ce qui est parfois le cas, par exemple les règles de réécriture permettant de définir l'arithmétique [53].

Dans le cas d'un système convergent, il est sans doute possible de définir la notion encore plus faible de semi-valuation atomique normale, en prenant la conjonction des définitions 2.3.7 et 2.3.9. Ou encore il devrait être possible d'introduire une notion de semi-valuation qui ne réécrit les formules atomiques que jusqu'à *une* formule non atomique, non nécessairement normale.

Nous conjecturons qu'il est possible d'étendre ces deux dernières notions de semi-valuation très optimisée en des semi-valuation compatibles avec $\mathcal{RE}$. Dans la section suivante nous ne discutons que de la manière d'étendre une semi-valuation atomique (définition 2.3.9) en semi-valuation compatible avec $\mathcal{RE}$ (définition 2.3.5).

2.3.4 Extension des semi-valuations atomiques

Les résultats de cette section sont valides pour tout système de réécriture $\mathcal{RE}$ confluent. Nous nous plaçons donc dans ce cadre et supposons de plus avoir une semi-valuation atomique $\mathcal{V}$.

L'extension de $\mathcal{V}$ se fait par étapes. Nous commençons par l'étendre par

$\mathcal{E}$-équivalence : on pose $\mathcal{V}_{\mathcal{E}}(B) := \mathcal{V}(A)$ dès lors que $\mathcal{V}(A)$ est définie et $B =_{\mathcal{E}} A$. $\mathcal{V}_{\mathcal{E}}$ possède de nombreuses propriétés, parmi lesquelles :

— être telle que, si $A =_{\mathcal{E}} B$ et $\mathcal{V}_{\mathcal{E}}(A)$ est définie, alors $\mathcal{V}_{\mathcal{E}}(B) = \mathcal{V}_{\mathcal{E}}(A)$, ce qui découle soit des propriétés de la définition 2.3.9 si $\mathcal{V}(B)$ est définie, soit de la définition de $\mathcal{V}_{\mathcal{E}}$;

— être une semi-valuation, y compris pour les formules B telles que $\mathcal{V}(B)$ est non définie ;

— être compatible avec la réécriture directe : si $A \longrightarrow B$ et $\mathcal{V}_{\mathcal{E}}(A)$ est définie, alors $\mathcal{V}_{\mathcal{E}}(B) = \mathcal{V}_{\mathcal{E}}(A)$;

— assurer que, si $A \equiv B$, alors $\mathcal{V}_{\mathcal{E}}(A) = \mathcal{V}_{\mathcal{E}}(B)$ dès que les deux valeurs sont définies.

$\mathcal{V}_{\mathcal{E}}$ ne possède pas toutes les propriétés voulues ; en particulier nous n'avons pour l'instant que la compatibilité avec la réécriture dans le sens gauche–droite ; c'est pourquoi nous devons continuer à étendre $\mathcal{V}_{\mathcal{E}}$ en une semi-valuation $\mathcal{V}_\omega$ compatible avec $\mathcal{RE}$, par construction successive de $\mathcal{V}_0 = \mathcal{V}_{\mathcal{E}}, \mathcal{V}_1, \cdots, \mathcal{V}_n$. L'extension de $\mathcal{V}_\omega$ à une valuation partielle n'est pas aussi simple qu'à la section 2.2.1, car les règles de réécriture peuvent interférer avec l'induction sur les formules ; par exemple si $C \longrightarrow \forall x P(x) \wedge Q(x)$ et que $\mathcal{V}_{\mathcal{E}}(\forall x P(x)) = \mathcal{V}_{\mathcal{E}}(\forall x Q(x)) = 1$. La complétion inductive de $\mathcal{V}_\omega$ donnerait simplement $\hat{\mathcal{V}}_\omega(\forall x P(x) \wedge Q(x))$, sans possibilité de définir $\hat{\mathcal{V}}_\omega(C)$.

Aussi, l'extension de $\mathcal{V}_{\mathcal{E}}$ en $\mathcal{V}_\omega$ comporte-t-elle deux étapes entrelacées, et s'appuie sur une énumération des formules avec répétition, similaire aux énumérations de la section 2.1.4 mais où chaque formule est maintenant visitée un nombre arbitraire de fois. Pour chacune de ces formules, à tour de rôle, nous

— complétons par réécriture et

— complétons par les lois des valuations partielles.

Nous devons évidemment assurer la compatibilité de ces deux étapes vis-à-vis de la réécriture et de la définition 2.2.1 de semi-valuation. Cette étape d'extension de $\mathcal{V}_{\mathcal{E}}$ en $\mathcal{V}_\omega$ doit elle-même être répétée afin d'atteindre une valuation partielle, et ce car l'extension n'est plus définie par induction (impossible à cause de la réécriture) mais par énumération, comme indiqué plus haut. Ce processus nous permet d'atteindre, au final, une valuation partielle à partir d'une semi-valuation atomique, que nous obtiendrons dans les sections 2.3.5 et 2.3.6.

Proposition 2.3.10 (De semi-valuation atomique à valuation partielle)**.** *Soit $\mathcal{V}$ une semi-valuation atomique, au sens de la définition 2.3.9, alors il existe une valuation partielle $\hat{\mathcal{V}}$ telle que, pour toute formule A telle que $\mathcal{V}(A)$ est défini, $\hat{\mathcal{V}}(A) = \mathcal{V}(A)$.*

Il nous faut ensuite passer à des conditions plus précises sur le système de réécriture $\mathcal{RE}$ pour pouvoir étendre une valuation partielle telle que celle obtenue ici en une interprétation dans un modèle de $\mathcal{RE}$. Ceci sera l'objet des sections suivantes.

2.3.5 Tableaux sans variables libres

Une méthode des tableaux en Déduction modulo théorie peut être tirée des règles de tableaux en logique classique de la figure 2.2 en ajoutant les deux règles de conversion de la figure 2.4. On peut aussi vouloir étendre la règle de clôture des branches, de manière à pouvoir l'appliquer dès que l'on rencontre $T\ A$ et $F\ B$ avec $A =_{\mathcal{E}} B$.

$$\frac{T\ A}{T\ B}\ T\text{conv}, A \equiv B \qquad \frac{F\ A}{F\ B}\ F\text{conv}, A \equiv B$$

FIGURE 2.4 – Règles de conversion de la méthode des tableaux modulo théorie classique

Comme vu dans le cadre de la logique du premier ordre, l'idée des preuves de complétude est que chaque branche ouverte d'un tableau représente un modèle potentiel de l'ensemble de formules initial. Ce modèle prend la forme d'une semi-valuation, ou plus exactement d'une ébauche de semi-valuation, qui devient une semi-valuation à part entière si la branche est complète.

Notre objectif, étant donné un tableau qui ne peut être fermé par aucune stratégie, est de construire une branche complète, de laquelle nous pouvons tirer une semi-valuation atomique, que les résultats de la section 2.3.3 précédente permettent d'étendre en valuation partielle. Pour construire cette branche, nous avons besoin d'une procédure d'expansion de tableau complète. Cependant les notions de branche et de procédure complète sont en Déduction modulo théorie plus fortes qu'à la section 2.2, puisqu'il faut prendre en compte la réécriture de manière à pouvoir ensuite se conformer à la définition 2.3.9.

Définition 2.3.11 (Branche et procédure complètes en Déduction modulo théorie)**.** *Soit $\mathcal{RE}$ un système de réécriture. Une branche ouverte est dite* complète *si toutes les formules universelles signées positivement $T\ \forall x A$ et toutes les formules existentielles signées négativement $F\ \exists x B$ sont instanciées par tous les termes clos, et si toutes les règles de réécritures de $\mathcal{RE}$ applicables aux atomes signés sont appliquées au moins une fois (avec la règle de conversion).*

Une procédure de tableaux est dite complète *lorsque pour tout tableau construit par celle-ci, toute branche ouverte infinie est complète.*

Cela requiert aussi l'élargissement de la notion de branche ouverte.

Définition 2.3.12 (Branche ouverte)**.** *Une branche d'un tableau est dite* ouverte *si elle ne contient pas deux formules signées $T\ A$ et $F\ B$ telles que $A =_{\mathcal{E}} B$.*

Comme on le voit, ces deux définitions correspondent exactement à la définition 2.3.9 de semi-valuation atomique ; avoir une branche complète permet donc aisément de définir une telle semi-valuation. On peut, par exemple, étendre la procédure de la section 2.2, en choisissant l'ordre d'application suivant :

1. règle de clôture étendue (si $T\ A$ et $F\ B$ apparaissent sur la branche avec $A =_{\mathcal{E}} B$, on ferme la branche) ;
2. règles α et δ, avec un parcours en largeur total, en revenant éventuellement au point 1 si nécessaire ;
3. règles β, avec un parcours en largeur total, en revenant éventuellement aux points 1 et 2 si nécessaire ;
4. règles γ, sur la/les premières k γ-formules non utilisées, rencontrées par un parcours en largeur ;
5. règles de conversion, en alternance avec les règles γ, soit toutes les règles de réécriture applicables aux k premières formules atomiques non utilisées rencontrées par un parcours en largeur.

Si les règles de réécriture sont intégrées aux règles de tableaux au lieu de former une catégorie à part, alors la dernière étape est supprimée, mais toutes les autres sont modifiées, y compris l'étape de clôture 1.

La procédure d'expansion définie ici ne dépend pas de la confluence de $\mathcal{RE}$, ni de sa terminaison ; cependant elle suppose de manière implicite la décidabilité de $\mathcal{E}$. Il s'agit d'une procédure de semi-décision : si le tableau peut être fermé, et que l'on a confluence, alors elle y arrive en un temps fini ; s'il ne l'est pas, alors elle peut prendre un temps infini, y compris en l'absence de règle de réécriture, puisque la logique du premier ordre est indécidable [30, 134, 90, 58]. C'est entre autres ce qui arrive dès lors que nous sommes en présence d'une γ-formule.

Proposition 2.3.13. *Soit $\mathcal{RE}$ un système de réécriture. Soit une procédure complète et un tableau en Déduction modulo théorie. Si cette procédure n'arrive jamais à fermer le tableau, alors elle génère une branche complète et une semi-valuation atomique $\mathcal{V}$ telle que si les formules signées $T\ A$ et $F\ B$ apparaissent sur cette branche, $\mathcal{V}(A) = 1$ et $\mathcal{V}(B) = 0$.*

2.3.6 Tableaux avec variables libres

La méthode des tableaux en Déduction modulo théorie a été introduite par Bonichon [17]. Elle est rappelée plus loin, figure 2.5, car elle appelle certaines remarques préliminaires. Une implémentation de la méthode des tableaux en Déduction modulo théorie, qui s'inspire de la méthode des tableaux de la figure 2.5, ainsi que ses résultats lorsqu'elle est appliquée à des problèmes concrets, sont discutés dans les articles [27, 43, 42, 44].

La méthode considérée ici, appelée TaMeD, diffère des présentations usuelles des tableaux avec variables libres.

— Tout d'abord les tableaux sont non-destructifs : toute feuille du tableau contient toutes les formules apparaissant sur la branche, ce qui permet une implémentation plate et non sous forme d'arbre.

— Deuxièmement, les tableaux ne sont pas signés ; au lieu d'avoir une règle $T_\wedge$ et une règle $F_\wedge$, on aura donc une règle $\wedge$ qui décompose $A \wedge B$ en A, B sur la même branche (règle α) et une règle $\neg\wedge$ qui décompose $\neg(A \wedge B)$ en $\neg A$ et $\neg B$ en créant un branchement (règle β), et ainsi de suite pour tous les connecteurs et quantificateurs, à l'exception de $T_\neg$, qui disparaît, et de $F_\neg$, qui revient à une règle α d'élimination de la double négation.

— Troisièmement, les règles sont factorisées selon le groupe auquel elles appartiennent. En effet, les règles α sont toutes similaires entre elles, de même que les règles β, δ et γ. La catégorisation est rappelée dans le tableau ci-dessous.

α	$A \wedge B, \neg(A \vee B), \neg(A \Rightarrow B), \neg\neg A$
β	$A \vee B, A \Rightarrow B, \neg(A \wedge B)$
δ	$\exists x\ A, \neg(\forall x\ A)$
γ	$\forall x\ A, \neg(\exists x\ A)$

$\alpha(A, B)$ dénote ainsi indifféremment $A \wedge B$, $\neg(A \vee B)$, $\neg(A \Rightarrow B)$ ou $\neg\neg A$. Dans les règles, A dénotera alors, selon le cas, soit A, soit $\neg A$; en l'occurrence A, $\neg A$, A et A, respectivement.

— La skolémisation se fait de manière interne, par rapport aux variables libres présentes dans la formule skolémisée, ce qui est dénoté par le terme de Skolem `sko(args)`. Dans certaines versions de TaMeD, notamment [19], la skolémisation se fait de manière externe, par rapport à l'ensemble plus grand de toutes les variables universellement quantifiées de la formule initiale, que l'on doit enregistrer dans un label associé à chaque formule. C'est aussi ce choix de skolémisation externe qui est fait dans la résolution modulo théorie [49].

Mais le plus grand changement, par rapport à la présentation usuelle des tableaux avec variables libres , consiste en l'introduction de *contraintes d'$\mathcal{E}$-unification* (unification modulo les équations de $\mathcal{E}$) [49]. Ce changement de présentation est dû à l'interaction entre les variables libres et les règles de réécriture. Une contrainte d'$\mathcal{E}$-unification entre deux termes t et u sera notée $t \approx_\mathcal{E} u$, nous avons aussi besoin de contraintes d'unification entre formules, sur le même modèle.

— Le premier niveau de contraintes, local, porte sur les formules, qui sont maintenant de la forme A_c, où c est un ensemble de contraintes. La réécriture d'une formule en une autre formule à l'aide de la règle $l \longrightarrow r$ est possible à condition que l'on puisse trouver une occurrence ω dans A, A_ω, qui s'unifie à l. Cette condition d'unification, qui vient s'ajouter aux contraintes précédentes c, remplace et généralise l'application de la substitution de réécriture. En effet, la contrainte d'unification porte *aussi bien* sur les variables de l que sur les variables libres de $A_{|\omega}$. Techniquement, il s'agit de narrowing [111, 49]. Cette généralisation est nécessaire si nous voulons une méthode de preuve complète. En effet, nous ne connaissons pas la "vraie" valeur des variables libres X ; il faut donc se réserver la possibilité de pouvoir appliquer toutes les règles *potentiellement* applicables, où "potentiellement" doit être compris comme "contraintes d'unification satisfiables". Par exemple, avec les règles

$$\begin{array}{rcl} Null(0) & \longrightarrow & \top \text{ et} \\ Null(s(X)) & \longrightarrow & \bot, \end{array}$$

deux règles de réécriture sont potentiellement applicables à la formule $Null(Y)$. L'une d'entre elle est utile, si nous cherchons à prouver $\exists y Null(y)$, et l'autre, si nous cherchons à prouver $\neg\forall y Null(y)$.

Nous avons donc, pour chaque formule, un ensemble de contraintes de réécriture associées. Notons que l'ensemble des contraintes sur toutes les formules n'est pas forcément satisfiable dans sa globalité : en reprenant l'exemple précédent, si nous cherchons à prouver $\exists y Null(y)$, nous pouvons générer les deux formules contraintes suivantes :

$$\top_{\{Y \approx_{\mathcal{E}} 0\}} \text{ et } \bot_{\{Y \approx_{\mathcal{E}} s(X)\}}.$$

L'ensemble des deux contraintes est incohérent, ce qui veut dire que l'application *effective* (et non plus potentielle) d'une des deux règles à $Null(Y)$ rend l'*autre* règle inapplicable, en imposant une structure particulière à Y. Cela ne signifie pas que le tableau lui-même est incohérent, mais simplement que certaines formules hypothétiques peuvent se concrétiser, et d'autres non. Nous sommes avec la règle conv dans une étape prospective, où nous essayons tout ce qui est possible. Les conditions de cette potentialité sont mémorisées dans les contraintes *locales* aux formules, qui deviennent une réalité (appliquées au tableau entier) lors de l'application de la règle de clôture, ce qui forme notre second point.

— Le second niveau de contraintes porte sur les tableaux eux-mêmes, qui auront donc la forme de n branches accompagnées d'un ensemble de contraintes globales, comme ceci :

$$(\Gamma_1 \mid \cdots \mid \Gamma_n) \cdot \mathcal{C}.$$

Le moment où les contraintes locales deviennent des contraintes globales est lors de la fermeture d'une branche, car c'est le seul moment où l'on se sert explicitement des formules en jeu : les contraintes d'unification portées par les formules doivent être satisfaites *globalement*, ainsi que les contraintes d'unification entre les deux formules.

Utiliser la règle de clôture est donc un engagement fort, qui a un impact global, contrairement aux autres règles d'expansion de tableau, y compris la réécriture. Lors de la recherche d'une preuve — nous le verrons dans la procédure d'expansion complète —, il faut faire très attention à où et comment l'appliquer. Il est en effet très facile de générer des contraintes d'unification globales non satisfiables, ce qui nous mène à une impasse. Ceci est apparent dans la notion de tableau fermé ci-dessous.

$$\frac{\Gamma_1, \alpha(A,B)_c \mid \ldots \mid \Gamma_n \cdot \mathcal{C}}{\Gamma_1, \alpha(A,B)_c, A_c, B_c \mid \ldots \mid \Gamma_n \cdot \mathcal{C}}\ \alpha \qquad \frac{\Gamma_1, \beta(A,B)_c \mid \ldots \mid \Gamma_n \cdot \mathcal{C}}{\Gamma_1, \beta(A,B)_c, A_c \mid \Gamma_1, \beta(A,B)_c, B_c \mid \ldots \mid \Gamma_n \cdot \mathcal{C}}\ \beta$$

$$\frac{\Gamma_1, \gamma(x,A)_c \mid \ldots \mid \Gamma_n \cdot \mathcal{C}}{\Gamma_1, A[X/x]_c, \gamma(x,A)_c \mid \ldots \mid \Gamma_n \cdot \mathcal{C}}\ \gamma \qquad \frac{\Gamma_1, \delta(x,A)_c \mid \ldots \mid \Gamma_n \cdot \mathcal{C}}{\Gamma_1, A_c[\texttt{sko(args)}/x], \delta(x,A)_c \mid \ldots \mid \Gamma_n \cdot \mathcal{C}}\ \delta$$

$$\frac{\Gamma_1, A_c \mid \ldots \mid \Gamma_n \cdot \mathcal{C}}{\Gamma_1, A_c, A_{\mathcal{K}}[r]_\omega \mid \ldots \mid \Gamma_n \cdot \mathcal{C}}\ \text{conv si } l \longrightarrow_{\mathcal{R}} r, \text{ et } \mathcal{K} = c \cup \{A_{|\omega} \approx_{\mathcal{E}} l\}$$

$$\frac{\Gamma_1, A_{c_1}, \neg A'_{c_2} \mid \ldots \mid \Gamma_n \cdot \mathcal{C}}{(\Gamma_2 \mid \ldots \mid \Gamma_n) \cdot \mathcal{C} \cup c_1 \cup c_2 \cup \{A \approx_{\mathcal{E}} A'\}}\ \text{clôture } (\odot)$$

FIGURE 2.5 – Règles d'expansion et de clôture des tableaux modulo théorie

Définition 2.3.14. *Dans* TaMeD*, un tableau est dit* fermé *si*
— *toutes les branches sont fermées et*
— *l'ensemble global de contraintes est satisfiable.*

Très souvent, en fermant un tableau, certaines contraintes *locales* deviennent incompatibles avec l'ensemble de contraintes globales. Pour s'en convaincre, reprenons l'exemple précédent de la preuve de $\exists y Null(y)$. Si nous tentons les deux règles de réécriture, puis nous fermons la branche avec la règle de clôture, qui s'applique aussi dans le cas où l'on rencontre $\neg\top$ ou $\bot$, l'ensemble global de contraintes $\{Y \approx_{\mathcal{E}} 0\}$ est incompatible avec la formule $\top_{Y \approx_{\mathcal{E}} s(X)}$:

$$\cfrac{\cfrac{\cfrac{\cfrac{\neg\exists y Null(y) \cdot \{\}}{\neg\exists y Null(y), \neg Null(Y) \cdot \{\}}\ \gamma}{\neg\exists y Null(y), \neg Null(Y), \neg\top_{\{Y \approx_{\mathcal{E}} 0\}} \cdot \{\}}\ \text{conv}}{\neg\exists y Null(y), \neg Null(Y), \neg\top_{\{Y \approx_{\mathcal{E}} 0\}}, \neg\bot_{\{Y \approx_{\mathcal{E}} s(X)\}} \cdot \{\}}\ \text{conv}}{\emptyset \cdot \{Y \approx_{\mathcal{E}} 0\}}\ \text{clôture}$$

Ceci n'est pas un problème, au contraire : cela signifie simplement que la formule $\neg\bot_{Y\approx_{\mathcal{E}} s(X)}$ que nous avons obtenue n'a aucune utilité dans la preuve finale. À un moment donné, nous avons tenté d'appliquer une règle de réécriture (en l'occurrence $Null(s(X)) \longrightarrow \bot$) qui n'aura servi à rien. Mais il ne nous a pas été possible *a priori* de renoncer à appliquer cette règle. Les seules formules "réelles" sont celles pour lesquelles les contraintes sont concrètement satisfaites, et ce sont celles-ci qui nous serviront dans la preuve du théorème de complétude. Pour le moment, nous pouvons vérifier qu'il est possible de construire une preuve de tableau qui n'appelle pas les règles de réécriture incompatibles avec la contrainte globale finale.

2.3.7 Branches ouvertes, complètes et semi-valuations

Il nous faut définir la notion de branche ouverte, mais cette notion n'est pas absolue, comme le montre l'exemple suivant, où nous adoptons la convention qui ne répète pas les sous-formules (destructive) pour des questions de place :

$$\dfrac{\dfrac{\forall x(\neg P(a)\wedge P(x))\vee(\neg P(b)\wedge P(x))}{\neg P(a)\wedge P(X)\vee(\neg P(b)\wedge P(X))}\,\gamma}{\alpha\,\dfrac{\neg P(b)\wedge P(X)}{\neg P(b), P(X)} \qquad \alpha\,\dfrac{\neg P(a)\wedge P(X)}{\neg P(a), P(X)}}\,\beta$$

Ce tableau n'est pas complet, c'est à dire qu'il possède au moins une branche ouverte ou non complète ; on pourrait appliquer la règle γ indéfiniment, mais ce qui est intéressant ici est que la branche gauche peut être fermée, en générant la contrainte globale $\{X \approx_{\mathcal{E}} b\}$, ainsi que la branche droite, en générant la contrainte globale $\{X \approx_{\mathcal{E}} a\}$. Ces deux contraintes sont incompatibles entre elles ; les deux branches ne peuvent pas être fermées simultanément, même si elles le peuvent individuellement.

Il existe des modèles de la formule $\forall x(\neg P(a)\wedge P(x))\vee(\neg P(b)\wedge P(x))$, par exemple celui où $[\![P(a)]\!] = 1$ et $[\![P(b)]\!] = 0$, ce tableau ne pourra donc pas être fermé, en faisant l'hypothèse, que nous démontrerons plus tard, de la correction de la méthode des tableaux modulo avec variables libres. On peut, comme dans le cas des tableaux sans variables libres, voir le tableau précédent comme une tentative de décrire tous les modèles, chaque branche correspondant à un modèle potentiel.

— Choisir de fermer la branche de gauche "force" la substitution $[b/X]$, la branche de droite restant ouverte, cela induit le modèle où $[\![P(a)]\!] = 0$ et $[\![P(b)]\!] = 1$.
— Choisir au contraire de fermer la branche de droite induit le modèle où $[\![P(a)]\!] = 1$ et $[\![P(b)]\!] = 0$.

Nous avons donc deux choix, la question est laquelle des deux branches est vraiment ouverte ? En réalité, ni l'une ni l'autre n'est ouverte ou fermée dans l'absolu. La réflexion sous-jacente qui permet de répondre à la question précédente est : nous savons par la définition 2.3.14 ce qu'est un tableau fermé, mais qu'est-ce qu'un tableau ouvert ? Au lieu de définir ouvert comme "non fermé", nous optons pour une définition directe, plus adaptée à nos besoins.

Définition 2.3.15 (Tableau ouvert, substitution totale)**.** *Soit $\mathcal{T} \cdot \mathcal{C}$ un tableau. Il est dit* ouvert *si, pour toute substitution* totale *σ qui satisfait $\mathcal{C}$, au moins une branche du tableau instancié $\mathcal{T}\sigma$ n'est pas fermée.*

Une substitution σ est totale *pour un tableau $\mathcal{T} \cdot \mathcal{C}$ si, et seulement si, elle substitue des termes clos à toutes les variables libres. Elle* satisfait *une contrainte d'unification $t \approx_{\mathcal{E}} u$ si, et seulement si, $t\sigma =_{\mathcal{E}} u\sigma$, et un ensemble de contraintes si, et seulement si, elle satisfait toutes les contraintes.*

Une branche du tableau instancié est fermée *si, et seulement si, il existe sur cette branches deux formules $A\sigma_c$ et $\neg A'\sigma_{c'}$ telles que $A\sigma =_{\mathcal{E}} A'\sigma$ et que σ satisfait c et c'. Une branche du tableau $\mathcal{T} \cdot \mathcal{C}$ instancié par σ qui n'est pas fermée est dite* ouverte*.*

La notion de branche ouverte dans un tableau avec variables libres est donc maintenant une notion *relative* à une substitution. Un tableau est lui-même ouvert si, pour toute substitution compatible, au moins une de ses branches reste ouverte. Cette définition résout nos interrogations.

En particulier, si l'ensemble de contraintes $\mathcal{C}$ est insatisfiable, alors le tableau est déclaré ouvert, mais ceci est surtout un signe que la règle de clôture a été appliquée de manière incorrecte. Les tableaux ayant un ensemble de contraintes $\mathcal{C}$ satisfiable, par exemple l'ensemble de contraintes vide, seront beaucoup plus intéressants. Notamment, si $\mathcal{C}$ est vide, nous avons toute latitude pour définir la substitution σ, afin de définir, à partir d'une branche instanciée (par σ) et ouverte, une semi-valuation atomique.

Enfin, nous voyons que les formules A_c telles que σ ne satisfait pas c ne comptent pas, puisqu'elles ne peuvent servir à fermer les branches du tableau instancié. Elles sont ignorées et ce, toujours de manière relative à σ.

Voyons maintenant quelles sont les conditions qui nous permettent de définir une semi-valuation atomique, et leur impact sur une procédure d'expansion complète en TaMeD.

— Une des conséquences importantes de la définition 2.3.15 est que le choix de σ est totalement libre, à condition qu'il respecte les contraintes globales. Afin de ne se priver d'aucun degré de liberté, nous avons intérêt à ne générer *aucune* contrainte globale, sauf si celles-ci sont triviales ou bien si elles permettent de fermer *toutes* les branches simultanément, auquel cas nous avons affaire à un tableau fermé.

— Les semi-valuations atomiques demandent à instancier les γ-formules par tous les termes possibles. Le seul endroit où il est possible d'énumérer les termes est la substitution σ, qui, rappelons-le, est un paramètre et que l'on pourrait choisir afin d'énumérer ces termes. Mais pour cela, il nous faut, pour chaque γ-formule, un nombre dénombrable de γ-variables : nous n'échapperons pas, dans la procédure complète, à une répétition des γ-formules et des instances, même par des variables libres. Les définitions de σ et de la procédure complète sont donc interdépendantes.

— Une fois σ fixé, nous pouvons ignorer les formules A_c, lorsque σ ne satisfait pas c, et oublier c, lorsque σ les satisfait, puisque $c\sigma$ devient un ensemble de contraintes trivial.

— En l'ignorance de σ, nous ne savons pas quelles contraintes seront satisfaites ou non ; nous devons donc tout essayer. Chacune des règles de réécriture applicable sur les littéraux (formules atomiques ou leur négation) doit être appliquée. Cela ne concerne que les littéraux dont l'ensemble de contraintes est satisfiable (voir le point précédent) et que les règles de réécriture qui ne rendent pas directement incohérentes les contraintes locales.

Étant données toutes ces remarques, nous aboutissons à la définition d'une procédure complète en TaMeD.

Définition 2.3.16 (Procédure complète en TaMeD)**.** *Soit $\mathcal{RE}$ un système de réécriture. Une procédure d'expansion est dite* complète *lorsque :*

1. *aucune contrainte globale n'est générée par les règles de clôture, sauf si l'ensemble de contraintes généré est satisfiable et permet de fermer le tableau entier ;*
2. *pour chaque formule composée qui n'est pas une γ-formule, la règle correspondante lui est appliquée pour étendre toutes les branches ouvertes ;*
3. *pour chaque γ-formule, pour tout entier m, il existe une étape telle que la γ-règle aura été applique m fois avec m variables libres différentes pour étendre toutes les sous-branches ouvertes ;*
4. *pour chaque littéral de la branche, P_c ou $\neg P_c$, et chaque règle de réécriture $l \longrightarrow r$ telle que*
   — *soit $P \approx_{\mathcal{E}} l \cup c$ est satisfiable, si $l \longrightarrow r$ est une règle de réécriture propositionnelle,*
   — *soit pour une certaine position ω, $P_{|\omega} \approx_{\mathcal{E}} l \cup c$ est satisfiable, si $l \longrightarrow r$ est une règle de réécriture de terme,*

   alors il existe une étape pour laquelle la règle de réécriture est appliquée au littéral pour étendre toutes les sous-branches ouvertes.

La définition 2.3.16 est valide même si la branche que l'on étend est fermée. Il n'est pas possible de donner la définition d'une branche complète, puisqu'il s'agit d'une notion relative, qui dépend fortement de la substitution que l'on applique au tableau.

Un exemple concret d'une procédure complète est la procédure d'expansion systématique que nous appellerons STEP (pour "Systematic Tableaux Expansion Procedure"). Elle s'appuie sur l'ordre suivant entre les différentes branches : $\mathcal{B}_1 \prec \mathcal{B}_2$ si, et seulement si, le nombre de formules de $\mathcal{B}_1$ est inférieur au nombre de formules de $\mathcal{B}_2$. STEP s'appuie aussi sur un ordre local à chaque branche, entre les types de formules ; cet ordre est *variable*, mais n'est pas forcément hérité par la/les branches filles :

— soit l'ordre est $\alpha \prec \delta \prec \beta \prec \gamma \prec \text{conv}$;

— soit c'est $\alpha \prec \delta \prec \beta \prec \text{conv} \prec \gamma$.

Rappelons que les règles de réécriture ne sont autorisées, dans TaMeD, que sur les littéraux. Comme dans les procédures d'expansion de tableaux au premier ordre et sans variables libres de la section 2.2.2, toutes les formules sont marquées soit utilisées, soit non utilisées ; à l'étape initiale, toutes les formules sont non utilisées. Ensuite, nous appliquons l'algorithme suivant.

1. Pour chaque branche $\mathcal{B}$ et chaque couple de littéraux positif/négatif unifiable sur $\mathcal{B}$, générer les contraintes d'unification permettant d'appliquer la règle de fermeture. Fermer d'office les branches telles qu'il existe un couple générant un ensemble de contraintes trivial. S'il est possible de trouver, pour chaque branche restante $\mathcal{B}_i$, un ensemble de contraintes de fermeture c_i tel que $\bigcup c_i$ est satisfiable, fermer le tableau entièrement.

2. Considérer, selon l'ordre sur les branches, la plus petite branche $\mathcal{B}$ qui est extensible, c'est à dire non fermée et avec au moins une formule non utilisée. Considérer, selon l'ordre local à $\mathcal{B}$, le premier *ensemble* de formules non utilisées qui est non vide. Appliquer toutes les règles correspondantes.

   (a) Dans le cas des règles α, δ et β, marquer les formules décomposées comme étant "utilisées", les produits de la décomposition comme étant "non utilisés", et munir la/les branches filles du même ordre local que la branche mère.

   (b) Dans le cas des règles γ, marquer les formules décomposées ainsi que leurs produits comme "non utilisés", et munir la branche fille de l'ordre $\text{conv} \prec \gamma$.

   (c) Dans le cas des règles conv, réécrire les littéraux par la première règle de réécriture applicable et non déjà utilisée ; marquer le littéral comme "utilisé avec r", où r est le nom de la règle de réécriture, et, le cas échéant, la position du littéral où cette règle a été utilisée. Marquer la formule réécrite "non utilisée", et munir la branche fille de l'ordre $\gamma \prec \text{conv}$.

Cette procédure est assez lourde et peu souple, mais elle nous permet de progresser vers la complétude. Cela appelle quelques commentaires.

- L'étape 1 termine, les combinaisons étant en nombre fini. Si elle réussit, elle trouve (par unification) une substitution qui permet de fermer le tableau.
- L'étape 2 garantit que toute branche sera visitée, à condition que celle-ci soit vivante, c'est à dire non fermée. En effet, TaMeD étant une méthode non destructive, STEP fait grossir la taille des branches visitées.
- L'étape 2b laisse les γ-formules “non utilisées”; chacune de ces formules sera donc décomposée un nombre arbitraire de fois.
- La notion de “règle de réécriture applicable” de l'étape 2c prend en compte les contraintes locales : si celles-ci sont incompatibles avec l'application de la réécriture, alors la règle n'est pas applicable. Le choix de la règle est libre, du moment qu'il est équitable dans le cas d'un nombre infini de règles.
- Dans le cas d'un nombre fini de règles, on peut générer toutes les formules réécrites en une seule passe.
- Dans le cas où le système $\mathcal{RE}$ est confluent et termine, on pourrait vouloir normaliser les littéraux, ou au moins les réécrire jusqu'à obtenir une formule non atomique, ce qui est, par exemple, la stratégie implémentée par Zenon Modulo [44, 71].
  Cette stratégie est difficilement applicable à TaMeD, à cause des variables libres, qui autorisent *plusieurs* formes normales *potentielles*. Par exemple, $Null(X)$ a deux formes normales potentielles, $\bot_{X \approx_{\mathcal{E}} s(Y)}$ et $\top_{X \approx_{\mathcal{E}} 0}$, et il faut, en toute généralité, les considérer toutes les deux. Zenon a une stratégie qui mélange les variables libres et les termes clos; il peut même être considéré comme sans variables libres, mais la stratégie d'instanciation des γ-formules doit tout de même être modifiée pour tenir compte de la réécriture possible (narrowing), si l'on veut rester complet.
- L'alternance des étape 2b et 2c peut être contrôlée, par exemple en donnant une plus grande fréquence aux étapes de réécriture.
- Le changement (potentiel) de l'ordre local lors des étapes 2b et 2c offre une alternance entre les γ-règles et les règles de réécriture, ce qui permet de rester équitable.

STEP est effectivement complète au sens de la définition 2.3.16, y compris si l'on ignore l'étape 1, qui cherche à construire une preuve lorsque cela est possible. Cette étape est facultative si l'on a un tableau ouvert au sens de la définition 2.3.15. Comme il n'existe pas, par hypothèse, de substitution qui permette de fermer toutes les branches, l'étape 1 échoue de toute façons, soit parce que toute combinaison de contraintes d'unifications est insatisfiable, soit parce que certaines branches ne sont même pas en mesure de générer de contrainte de fermeture locale. Il n'est pas gênant, même si inefficace, de garder ouvertes les branches qui pourraient être fermées d'office
.

Voyons maintenant comment, étant donnée une procédure complète, et un tableau auquel on l'applique et que l'on suppose *ouvert indéfiniment*, on peut construire une semi-valuation atomique. Nous supposerons avoir un tableau ouvert infini, que l'on peut voir comme la limite de la suite finie de tableaux construits par STEP ; il est tel que toutes les règles applicables ont été appliquées, en particulier toutes les formules y sont marquées “utilisées”. La définition 2.3.15 nous laisse toute latitude pour choisir la substitution qui nous convient le mieux. Nous allons donc construire une substitution *énumérante*, afin d'instancier chaque γ-formule par tous les termes possibles. Cette construction s'appuie sur la manière dont STEP construit le tableau.

1. Considérons une énumération de tous les termes clos du langage, y compris ceux comportant des symboles de Skolem, notée $(t_j)_{j\in\mathbb{N}}$.
2. Nous commençons par la substitution vide $\sigma_0 = \emptyset$, que nous associons au tableau initial $\mathcal{T}_0$.
3. À chaque étape de STEP où nous appliquons une ou des γ-règles, nous augmentons σ_n de la manière suivante : soit γ_i la formule décomposée, X_i la variable (fraîche) introduite pour γ_i et k_i le nombre de fois que γ_i a déjà été décomposée. Nous posons
$$\sigma_{n+1} = \sigma_n \cup [t_{k_i}/X_i].$$

Lorsque nous appliquons $\sigma = \bigcup_{n\in\mathbb{N}} \sigma_n$ à $\mathcal{T}$, nous obtenons une branche instanciée ouverte $\mathcal{B}$ qui nous permet de définir une semi-valuation atomique $\mathcal{V}$ en posant $\mathcal{V}(A) = 1$, si A_c apparaît sur la branche ouverte instanciée avec c satisfaite par σ (on dira que A_c est satisfaite), et $\mathcal{V}(A) = 0$, si $\neg A_c$ apparaît et est satisfaite par σ. Vérifions maintenant les deux points cruciaux de la définition 2.3.9, qui portent sur la réécriture et les quantificateurs.

— Supposons $\mathcal{V}(\forall x A) = 1$. C'est donc que $(\forall x A_0)_c$, avec $A = A_0\sigma$, apparaît sur $\mathcal{B}$ et que c est satisfaite. Puisque $\mathcal{B}$ est obtenue par limite de l'application d'une procédure complète, il existe un ensemble dénombrable de variables fraîches et spécifiques à $(\forall x A_0)_c$, nommons-les $(X_k)_{k\in\mathbb{N}}$, telles que tous les $A_0[X_k/x]_c$ apparaissent sur cette branche. c étant satisfaite par σ, ces formules sont elles aussi satisfaites et, par la définition de σ, $\mathcal{V}(A_0[X_k/x]\sigma) = \mathcal{V}(A[t_k/x]) = 1$. $(t_k)_{k\in\mathbb{N}}$ étant une énumération des termes, on obtient $\mathcal{V}(A[t/x]) = 1$ pour tout terme clos t.

— supposons $\mathcal{V}(A) = 1$, avec A formule atomique, et $A \xrightarrow{l \longrightarrow r} B$, avec $l \longrightarrow r \in \mathcal{R}$. Soit A_{0_c} la formule de la branche telle que $A_{0_c}\sigma = A$. La règle $l \longrightarrow r$ est applicable à A_{0_c} : les contraintes d'unification avec l sont compatibles avec c et satisfaites par σ, la réécriture $A = A_{0_c}\sigma \longrightarrow B$ en étant justement le témoin. Nous avons donc $\mathcal{V}(B) = 1$.

Nous avons donc réussi à construire une semi-valuation atomique dans le cas de TaMeD, une méthode de tableaux classiques en Déduction modulo théorie avec variables libres.

Proposition 2.3.17. *Soit $\mathcal{RE}$ un système de réécriture. Soit une procédure complète (par exemple STEP) et un tableau en* TaMeD*. Si cette procédure n'arrive jamais à fermer le tableau, alors il existe une substitution énumérante σ, une branche complète et une semi-valuation atomique $\mathcal{V}$ telle que si les formules signées $T\ A_c$ et $F\ B_{c'}$ apparaissent sur cette branche avec c et c' satisfaites par σ, $\mathcal{V}(A\sigma) = 1$ et $\mathcal{V}(B\sigma) = 0$.*

2.4 De valuation partielle à modèle d'une théorie

Nous avons pris soin, dans la partie précédente, de construire une semi-valuation sans aborder la construction de modèle. C'est l'objet de cette partie.

2.4.1 Prise en compte de la réécriture

Nous nous sommes abstraits de la méthode de tableaux sous-jacente ainsi que de la réécriture : qu'il s'agisse de tableaux avec variables libres (section 2.3.7) ou de tableaux sans variables libres (section 2.3.5), nous sommes arrivés à construire une semi-valuation atomique. Par la section 2.3.4, nous savons aussi comment étendre cette semi-valuation atomique en valuation partielle au sens de la définition 2.3.5. Cette dernière étape suppose simplement le système de réécriture confluent, nous supposerons donc implicitement confluents tous les systèmes de réécriture discutés dans cette section 2.4.

Il nous faut maintenant étendre ces valuations partielles en fonction d'interprétation. La méthode de la section 2.1.5 n'est pas assez précise, car l'interprétation doit respecter la réécriture. Nous ne pouvons pas fixer librement l'interprétation des formules atomiques.

En conséquence, l'extension d'une valuation partielle en une fonction d'interprétation qui respecte le système de réécriture $\mathcal{RE}$ dépend fortement de $\mathcal{RE}$ lui-même. Par exemple, nous pouvons reprendre l'exemple de la section 2.3.3, et considérer la règle de réécriture $A \longrightarrow A \Rightarrow B$. Le tableau $\neg B$ est ouvert et complet ; en posant $\mathcal{V}(B) = 0$, on obtient bien une valuation partielle en accord avec la branche ouverte, mais il est cependant impossible de construire un modèle dans lequel on aurait $[\![B]\!] = 0$. Nous pouvons en conclure que, dans le cas de la règle de réécriture $A \longrightarrow A \Rightarrow B$, la méthode des tableaux n'est pas complète, tandis que le calcul des séquents, *avec coupure*, l'est, par exemple par l'algèbre de Lindenbaum de la section 2.1.3. Le lecteur intéressé peut vérifier que le séquent $\vdash B$ est prouvable, mais nécessite la règle de coupure [52]. Comme annoncé à la section 2.3.3, la méthode des tableaux n'admet pas la coupure dans ce cas précis.

Le problème de savoir si, sous un système de réécriture $\mathcal{RE}$ donné, la méthode des tableaux est complète, a même été démontré indécidable [26], y compris dans le cas où $\mathcal{RE}$ est supposé confluent et terminant. Notons que par "méthode des tableaux" nous sous-entendons une méthode de tableaux sans règle de coupure.

Il est donc temps, pour toutes ces raisons, de spécialiser notre étude à des classes de systèmes de réécriture particuliers.

2.4.2 Systèmes positifs, systèmes ordonnés

Construire un modèle des règles de réécriture est possible dans le cas des systèmes de réécriture positifs.

Définition 2.4.1 (Positivité)**.** *Soit pol* : $\mathcal{A} \to \{-,+\}$ *une fonction qui classe les prédicats en deux catégories, positifs et négatifs. Une formule B est* positive *si tous ses atomes sont soit positifs avec une occurrence positive, soit négatifs avec une occurrence négative. B est* négative *si tous ses atomes sont, soit négatifs avec une occurrence positive, soit positifs avec une occurrence négative.*

La notion d'occurrence positive et négative est la notion standard, l'inversion de polarité ayant lieu à gauche de l'implication et sous la négation. Nous aurions aussi pu choisir une définition inductive plus classique, calquée sur la notion d'occurrence positive ou négative. La positivité et négativité des symboles de prédicat n'a pas de lien direct avec la réécriture polarisée [46].

Définition 2.4.2 (Système de réécriture positif)**.** *Soit $\mathcal{RE}$ un système de réécriture. Il est dit* positif *s'il existe une fonction pol* : $\mathcal{A} \to \{-,+\}$ *qui classe les prédicats en deux catégories, positifs et négatifs, telle que, pour toute règle de réécriture sur les formules $A \longrightarrow B \in \mathcal{R}$:*

- *si A est une instance d'un prédicat positif, alors B est positive ;*
- *si A est une instance d'un prédicat négatif, alors B est négative.*

Sous ces hypothèses, on peut construire un modèle de $\mathcal{RE}$ qui est en accord avec la valuation partielle $\mathcal{V}$ en posant :

- $[\![A]\!] = \mathcal{V}(A)$ si $\mathcal{V}(A)$ est défini ;
- $[\![A]\!] = 0$ si A est atomique, négative et $\mathcal{V}(A)$ n'est pas défini ;
- $[\![A]\!] = 1$ si A est atomique, positive et $\mathcal{V}(A)$ n'est pas défini.

Cette interprétation valide les règles de réécriture, car elle est uniforme. Considérons une (application d'une) règle de réécriture $A \longrightarrow B$, telle que A est positive et $\mathcal{V}(A)$ est non défini. $\mathcal{V}(B)$ est non défini par définition de valuation partielle (définition 2.3.5). Il y a donc suffisamment d'espace libre dans B, c'est à dire que suffisamment de sous-formules atomiques de B sont non interprétées. Par construction de l'interprétation, et par hypothèse de positivité sur B, ces atomes seront interprétés de telle sorte que $[\![B]\!] = 1$.

Nous conjecturons qu'il est possible de spécialiser la fonction de polarisation pour qu'elle catégorise non pas les prédicats, mais leurs *instances*, plus exactement les classes d'équivalence modulo les règles de termes de $\mathcal{R}$ et $\mathcal{E}$. Par exemple, $Null(0)$ pourrait être positive et toutes les formules du type $Null(s(t))$ négative. Ainsi, le système suivant respecterait la condition de positivité étendue.

$$\begin{array}{rcl} Null(0) & \longrightarrow & \top, \\ Null(s(y)) & \longrightarrow & \bot. \end{array}$$

Cela imposerait entre autres un travail plus précis sur le traitement des quantificateurs, $\forall x A$ étant positive, si toutes les instances de A par des termes clos sont positives et négative, si au moins une de ses instances l'est. De plus, il devrait être interdit de réécrire $s(t)$ sur 0.

Une seconde condition pour laquelle il est possible de construire un modèle est de disposer d'un ordre bien fondé à la fois compatible avec la réécriture et les sous-formules [126].

Définition 2.4.3 (Système ordonné)**.** *Un système de réécriture $\mathcal{RE}$ est* ordonné *s'il existe un ordre bien fondé $\prec$ tel que :*

— *si $A \longrightarrow B$, alors $B \prec A$;*

— *et, si B est une sous-formule stricte de A, alors $B \prec A$.*

Cet ordre bien fondé est exactement ce qu'il nous faut pour procéder comme à la section 2.1.5.

— Si $\mathcal{V}(A)$ est défini, on pose $[\![A]\!] = \mathcal{V}(A)$.

— Si $\mathcal{V}(A)$ n'est pas défini mais que A n'est pas minimal pour $\prec$ alors, si A est une formule composée, on considère l'interprétation de ses sous-formules, sinon on la réécrit en B et on pose $[\![A]\!] = [\![B]\!]$.

— Enfin, si $\mathcal{V}(A)$ n'est pas défini et que A est minimale pour $\prec$, donc forcément atomique, on fixe $[\![A]\!]$ comme on le souhaite.

Cette toute dernière étape de choix indifférent nous indique que nous avons encore un point d'amélioration potentiel et, en effet, nous pouvons combiner les deux conditions de positivité et d'ordre. Supposons que $\mathcal{RE}$ termine (et est confluent) et qu'il peut être décomposé en $\mathcal{R}_{\succ} \cup \mathcal{R}_{+}\mathcal{E}_{+}$ tel que :

— $\mathcal{R}_{\succ}$ peut être ordonné au sens de la définition 2.4.3 ;

— $\mathcal{R}_{+}\mathcal{E}_{+}$ est un système positif au sens de la définition 2.4.2 ;

— $\mathcal{R}_{+}\mathcal{E}_{+}$ est normal à droite par rapport à $\mathcal{R}_{\succ}$. Pour toute règle de réécriture $l \longrightarrow r \in \mathcal{R}_{+}\mathcal{E}_{+}$, pour toute substitution σ telle que, pour tout x, $\sigma(x)$ est en forme normale pour $\mathcal{R}_{\succ}$, $r\sigma$ est en forme normale pour $\mathcal{R}_{\succ}$.

Dans ce cas, il est possible de combiner la définition de $[\![\ ]\!]$ par induction sur $\prec$ et par positivité.

Pour tous ces systèmes de réécriture, nous obtenons donc la complétude, en combinant les résultats d'existence des valuation partielles des propositions 2.3.13 et 2.3.17 et de la proposition 2.3.10 avec les extensions ce celles-ci en fonction d'interprétation décrites ci-dessus.

Théorème 2.4.4 (Complétude de la méthode des tableaux modulo théorie)**.** *Soit $\mathcal{RE}$ un système de réécriture qui respecte une des conditions mentionnée dans cette section. Si aucune procédure d'expansion de tableaux ne parvient à fermer toutes les branches du tableau (avec ou sans variable libre) dont la racine est $T\ A_1, \cdots, T\ A_n, F\ B_1, \cdots, F\ B_m$, alors dans l'algèbre de Boole $\{0,1\}$, il existe un modèle de $\mathcal{RE}$ tel que $[\![A_i]\!] = 1$ et $[\![B_j]\!] = 0$.*

2.4.3 Logique d'ordre supérieur

La logique d'ordre supérieur [129, 115, 4] peut s'exprimer en Déduction modulo théorie dans le cadre d'une logique multi-sortée [47]. Les combinateurs permettent d'éviter d'avoir une copie du λ-calcul au niveau des termes [49].

Les types des termes sont ι (termes de base), o (termes propositionnels, appelés à être promus au niveau des formules) et les types fonctionnels composés $\alpha_1 \to \alpha_2$.

$$\alpha, \beta, \gamma \quad := \quad \iota \mid o \mid \alpha \to \beta.$$

D'autre part, la logique d'ordre supérieur possède un certain nombre de symboles spécifiques, qui permettent notamment de refléter la logique au niveau des termes :

— les combinateurs $K_{\alpha,\beta} : \alpha \to \beta \to \alpha$ et $S_{\alpha,\beta,\gamma} : (\alpha \to \beta \to \gamma) \to (\alpha \to \beta) \to \alpha \to \gamma$, pour chaque type α, β, γ ;

— les constantes logiques $\dot{\vee}, \dot{\wedge}, \dot{\Rightarrow}$ de type $o \to o \to o$, $\dot{\neg}$ de type $o \to o$, $\dot{\bot}, \dot{\top}$ de type o et les symboles de quantificateurs, pour chaque type α, $\dot{\forall}_\alpha, \dot{\exists}_\alpha$, de type $(\alpha \to o) \to o$;

— un symbole d'application $\mathsf{app}_{\alpha,\beta} : (\alpha \to \beta) \to \alpha \to \beta$ pour chaque type α, β. $\mathsf{app}_{\alpha,\beta}(f, a)$ sera noté de la manière non typée, infixe et associative à gauche $f \cdot a$, ou plus simplement encore fa.

Un seul symbole de prédicat unaire fait partie de la représentation de la logique d'ordre supérieur en Déduction modulo théorie, le prédicat $\varepsilon(\,)$, qui attend un argument de type o. On dira que $\varepsilon()$ est de rang $\langle o \rangle$. $\varepsilon()$ permet de promouvoir les termes de type o en des formules. Il interagit bien entendu avec les termes représentant des formules non atomiques, entre autres, ce qui est reflété dans le système de réécriture $\mathcal{R}$ de la figure 2.6.

Le système de réécriture de la figure 2.6 termine et est confluent. Le calcul des combinateurs portant sur les termes ne comporte que deux règles de réduction relatives à S et K, équivalent à la β-réduction dans le calcul

$$\begin{array}{rcl}
S \cdot x \cdot y \cdot z & \longrightarrow & (x \cdot z) \cdot (y \cdot z) \\
K \cdot x \cdot y & \longrightarrow & x \\
\varepsilon(\dot{\Rightarrow} \cdot x \cdot y) & \longrightarrow & \varepsilon(x) \Rightarrow \varepsilon(y) \\
\varepsilon(\dot{\vee} \cdot x \cdot y) & \longrightarrow & \varepsilon(x) \vee \varepsilon(y) \\
\varepsilon(\dot{\wedge} \cdot x \cdot y) & \longrightarrow & \varepsilon(x) \wedge \varepsilon(y) \\
\varepsilon(\dot{\neg} \cdot x) & \longrightarrow & \neg\, \varepsilon(x) \\
\varepsilon(\dot{\bot}) & \longrightarrow & \bot \\
\varepsilon(\dot{\top}) & \longrightarrow & \top \\
\varepsilon(\dot{\forall}_\alpha \cdot x) & \longrightarrow & \forall_\alpha\, y\, \varepsilon(x \cdot y) \\
\varepsilon(\dot{\exists}_\alpha \cdot x) & \longrightarrow & \exists_\alpha\, y\, \varepsilon(x \cdot y)
\end{array}$$

FIGURE 2.6 – Règles de réécriture permettant l'expression de la logique d'ordre supérieur

des combinateurs.

Étant donné une valuation partielle, la construction d'un modèle se fait par le biais des V-complexes introduits par Takahashi [129] et Prawitz [115], qui répond aux deux préoccupations principales de la logique d'ordre supérieur.

— La logique d'ordre supérieur, y compris la version en Déduction modulo théorie présentée ici, est imprédicative. La formule $\forall_o\, x\ (\varepsilon(x) \Rightarrow \varepsilon(x))$ peut être instanciée par un terme t tel que $\varepsilon(t) \Rightarrow \varepsilon(t)$ est une formule plus complexe que la formule initiale. t peut, entre autres, être le terme $\dot{\forall}_o\ (\overline{\lambda}x.(\dot{\Rightarrow} \cdot x \cdot x))$, où $\overline{\lambda}$ est une fonction d'abstraction par rapport à la variable x, qui retourne un terme composé des combinateurs S, K et de $\dot{\Rightarrow}$ qui, appliqué à x, se réduit sur $\dot{\Rightarrow} \cdot x \cdot x$. On obtient, pour ce terme précis, $\varepsilon(t) \Rightarrow \varepsilon(t) \equiv (\forall_o\, x\ (\varepsilon(x) \Rightarrow \varepsilon(x)) \Rightarrow (\forall_o\, x\ (\varepsilon(x) \Rightarrow \varepsilon(x)))$, une formule bien plus complexe que $\forall_o x(\varepsilon(x) \Rightarrow \varepsilon(x))$. Définir $[\![\,]\!]$ ne peut plus se faire par induction naïve sur les formules et les termes.
— La logique d'ordre supérieur est intentionnelle. Si l'on peut démontrer $(\varepsilon(t) \Rightarrow \varepsilon(t')) \wedge (\varepsilon(t') \Rightarrow \varepsilon(t))$, alors t et t', tous deux de type o, sont dits logiquement équivalents, mais ils ne sont en aucun cas strictement égaux. Par exemple $\dot{\top}$ et $\dot{\wedge} \cdot \dot{\top} \cdot \dot{\top}$ sont logiquement équivalents ; cependant ils sont différents, y compris pour la logique, car on peut définir des formules p tel que $\varepsilon(p \cdot \dot{\top})$ est démontrable, alors que $\varepsilon(p \cdot (\dot{\wedge} \cdot \dot{\top} \cdot \dot{\top}))$ ne l'est pas. Un exemple de telle formule serait $\overline{\lambda}x.(x =_o \dot{\top})$, où $=_o$ est l'égalité de Leibniz de type o [92]

$$\overline{\lambda}x, y.(\dot{\forall}_{o \to o}(\overline{\lambda}P.\ \dot{\Rightarrow} \cdot (P \cdot x) \cdot (P \cdot y))).$$

Cette définition implique la caractérisation plus usuelle $\varepsilon(x =_o y) \equiv$

$\forall_{o\to o} P\,(\varepsilon(P\cdot x)\Rightarrow\varepsilon(P\cdot y))$.
Plus simplement, pour se persuader de la nécessité de faire la distinction entre $\dot{\top}$ et $\dot{\wedge}\ \cdot\dot{\top}\ \cdot\ \dot{\top})$, il suffit de s'apercevoir que ni le séquent $\varepsilon(p\cdot\dot{\top})\vdash\varepsilon(p\cdot(\dot{\wedge}\ \cdot\dot{\top}\ \cdot\ \dot{\top}))$, ni son jumeau $\varepsilon(p\cdot(\dot{\wedge}\ \cdot\dot{\top}\ \cdot\dot{\top}))\vdash\varepsilon(p\cdot\dot{\top})$, n'a de preuve, quel que soit le symbole de prédicat p de type $o\to o$.

Les solutions à ces problèmes ont été introduites successivement.
- Le problème de l'imprédicativité apparaît dès le second ordre [128, 114]. On doit pouvoir quantifier sur toutes les valeurs *possibles*, c'est-à-dire autorisées par la valuation partielle.
- le problème de l'intentionnalité indique qu'on ne doit pas interpréter les termes de type o comme des valeurs de vérité, auquel cas la distinction entre t et $\dot{\wedge}\ \cdot t\cdot t$ disparaîtrait.

Ces deux remarques nous amènent à introduire une définition d'un *domaine pour les termes* en premier lieu, avant même d'y interpréter quoi que ce soit. Ce domaine doit contenir à la fois une information qui permette de distinguer deux termes syntaxiquement différents (intentionnalité) et une information qui permette de donner une valeur sémantique aux termes. De plus, un lien doit exister entre l'information syntaxique et l'information sémantique : les éléments de type o doivent recevoir un traitement compatible avec la valuation partielle. Cette information doit se propager aux termes d'ordre supérieur, et notamment ceux qui produisent au final un terme de type o. Cela doit en particulier, mais pas seulement, être le cas des éléments du domaine dénotant les connecteurs $\dot{\wedge},\dot{\vee}$ ou à ceux dénotant des quantificateurs $\dot{\forall},\dot{\exists}$.

Nous aboutissons à une définition des domaines qui doit s'appuyer sur une induction *sur les types*, bien fondée (contrairement à une induction sur les termes), et qui comporte deux parties : une partie syntaxique (intentionalité) et une partie sémantique (imprédicativité, adéquation à la valuation partielle). La partie sémantique des termes de type ι est irrelevante.

Définition 2.4.5 (V-complexes)**.** *Soit α un type simple. Nous définissons le domaine des V-complexes de type α, $\mathcal{D}_\alpha$, comme suit.*
- *$\mathcal{D}_\iota$ est l'ensemble des couples $\langle t,\iota\rangle$, tels que t est un terme de type ι en forme normale et ι est une constante qui n'a aucune signification particulière.*
- *$\mathcal{D}_o$ est l'ensemble des couples $\langle t,v\rangle$, tels que t est un terme de type o en forme normale et $v=\mathcal{V}(\varepsilon(t))$, si cette dernière valeur est définie ; si ce n'est pas le cas, v peut valoir soit 0, soit 1.*
- *$D_{\alpha\to\beta}$ est l'ensemble des couples $\langle t,f\rangle$, tels que t est un terme de type $\alpha\to\beta$ en forme normale et f est une fonction de $\mathcal{D}_\beta^{\mathcal{D}_\alpha}$ telle que pour tout $\langle u,k\rangle\in\mathcal{D}_\alpha$, $f(\langle u,k\rangle)\in\mathcal{D}_\beta$ et il existe g tel que $f(\langle u,k\rangle)=$*

$\langle {\downarrow}(t \cdot u), g\rangle$.

Munis de cette définition, nous pouvons procéder à la construction d'une interprétation dans le modèle booléen où toute constante c de type α est interprétée par un représentant canonique $\langle c, k\rangle \in D_\alpha$ (remarquons que $c = {\downarrow}c$), dont il faut encore prouver qu'il existe.

Les symboles "distingués" qui ont été définis au début de la section 2.4.3 sont interprétés de manière à correspondre à leur sémantique attendue ; par exemple, $[\![\dot{\wedge}]\!]$ doit avoir une valeur telle que, lorsqu'il est appliqué à $\langle a, v_a\rangle$, il doit produire un V-complexe qui, lorsqu'il est appliqué à $\langle b, v_b\rangle$, doit produire un V-complexe dont la seconde composante (la valeur de vérité) doit être égale à la conjonction booléenne $v_a \wedge v_b$.

Vérifier que tout est bien défini, notamment l'interprétation des symboles distingués, et que l'on obtient bien un modèle de $\mathcal{RE}$ est ensuite une affaire de technique. Notons la similarité des arguments avec les arguments initiaux, tels qu'exposés dans [4]. La présence de la réécriture n'induit pas une modification substantielle des démonstrations ; cependant celles-ci sont réorganisées de manière plus compréhensibles. En particulier, dans les articles originaux [129, 115, 4], on doit *modifier* la notion de modèle, car il y avait confusion entre *terme* de type o et formule : le modèle n'est donc plus un modèle à valeurs dans les algèbres de Boole, mais dans les V-complexes de type o. Voir la section 3.2.4 pour les détails. La Déduction modulo théorie, en insistant sur la séparation entre termes et formules, nous permet de rester dans le cadre de départ, en posant

$$[\![\varepsilon(t)]\!] = \pi_1([\![t]\!]),$$

avec π_1 la projection sur la seconde composante, c'est-à-dire l'extraction de la valeur de vérité de l'interprétation du terme t.

Théorème 2.4.6 (Complétude de la méthode des tableaux modulo la logique d'ordre supérieur)**.** *Soit $\mathcal{RE}$ le système de réécriture de la figure 2.6. Si aucune procédure d'expansion de tableaux ne parvient à fermer toutes les branches du tableau (avec ou sans variable libre) dont la racine est $T\ A_1, \cdots, T\ A_n, F\ B_1, \cdots, F\ B_m$, alors dans l'algèbre de Boole $\{0, 1\}$, il existe un modèle de $\mathcal{RE}$ tel que $[\![A_i]\!] = 1$ et $[\![B_j]\!] = 0$.*

2.5 Tableaux intuitionnistes

Nous souhaitons maintenant démontrer la complétude des tableaux intuitionnistes modulo théorie. Ceux-ci peuvent être définis à partir des tableaux intuitionnistes [59] en rajoutant les deux règles de conversion de la figure 2.7, de manière totalement similaire à l'extension de la section 2.3.5 des tableaux classiques à la Déduction modulo théorie.

$$\frac{T\ p\ A}{T\ p\ B}\ T_{\text{conv}}, A \equiv B \qquad \frac{F\ p\ A}{F\ p\ B}\ F_{\text{conv}}, A \equiv B$$

FIGURE 2.7 – Règles de conversion de la méthode des tableaux modulo théorie intuitionniste

La sémantique qui est intimement liée aux tableaux intuitionniste n'est plus une structure algébrique comme à la section 2.1.2. Il s'agit des structures de Kripke [133, 59, 99], non rappelées ici, que l'on adapte à la Déduction modulo théorie d'une manière qui ne devrait plus surprendre maintenant.

Définition 2.5.1 (Structure de Kripke en Déduction modulo théorie)**.** *Soit $\mathcal{RE}$ un système de réécriture. Une structure de Kripke $\langle W, \leq, D, \hat{}, \Vdash\rangle$ est un* modèle de $\mathcal{RE}$ *si, et seulement si, pour toutes les formules A, B, telles que $A conv B$, pour tout monde $w \in W$, pour toute valuation φ, on a*

$$w \Vdash_\varphi A \textit{ si, et seulement si, } w \Vdash_\varphi B$$

Munis de cette définition, il est alors aisé de démontrer le théorème de correction suivant, par rapport au calcul des séquents intuitionniste modulo théorie, dont nous nous servirons plus tard.

Théorème 2.5.2 (Correction)**.** *Soit $\mathcal{RE}$ un système de réécriture. Soit $A_1, \cdots, A_n \vdash B$ un séquent prouvable en calcul des séquents intuitionniste modulo théorie. Soit $\langle W, \leq, D, \hat{}, \Vdash\rangle$ une structure de Kripke qui est un modèle de $\mathcal{RE}$. Pour tout monde $w \in W$, pour toute valuation φ, si $w \Vdash_\varphi A_i$ pour tout i, alors $w \Vdash_\varphi B$.*

Pour aller plus loin vers la complétude de la méthode des tableaux par rapports aux structures de Kripke, nous avons besoin d'étendre le travail de la section 2.3.3 au cas de la Déduction modulo théorie intuitionniste, à commencer par la définition d'une semi-valuation. L'extension la plus immédiate, non sans lien avec la méthode des tableaux intuitionniste, est la définition de structure de Kripke partielle.

Définition 2.5.3 (Semi-valuation intuitionniste)**.** *Soient W un ensemble de mondes partiellement ordonné par $\leq$ et un ensemble de domaines D indexés par W, monotones par rapport à $\leq$, et dont les éléments sont des termes du langage.*

Une semi-valuation intuitionniste*, ou* semi-valuation de Kripke *dans W est une relation partielle ternaire entre l'ensemble de signes $\{T, F\}$, l'ensemble des mondes $p \in W$ et l'ensemble des formules closes de D_p, notée $\Vdash$, qui vérifie :*

— *si $T\ p \Vdash A \wedge B$, alors $T\ p \Vdash A$ et $T\ p \Vdash B$;*

— *si* $F\ p \Vdash A \wedge B$*, alors soit* $F\ p \Vdash A$*, soit* $F\ p \Vdash B$ *;*
— *si* $T\ p \Vdash A \vee B$*, alors soit* $T\ p \Vdash A$*, soit* $T\ p \Vdash B$ *;*
— *si* $F\ p \Vdash A \vee B$*, alors* $F\ p \Vdash A$ *et* $F\ p \Vdash B$ *;*
— *si* $T\ p \Vdash A \Rightarrow B$*, alors pour tout* $p' \geq p$*, soit* $F\ p' \Vdash A$*, soit* $T\ p' \Vdash B$ *;*
— *si* $F\ p \Vdash A \Rightarrow B$*, alors il existe au moins un* $p' \geq p$ *tel que* $T\ p' \Vdash A$ *et* $F\ p' \Vdash B$ *;*
— *si* $T\ p \Vdash \neg A$*, alors pour tout* $p' \geq p$*,* $F\ p' \Vdash A$ *;*
— *si* $F\ p \Vdash \neg A$*, alors il existe au moins un* $p' \geq p$ *tel que* $T\ p' \Vdash A$ *;*
— *si* $T\ p \Vdash \exists x A$*, alors il existe au moins un terme* $t \in D_p$ *tel que* $T\ p \Vdash A[t/x]$ *;*
— *si* $F\ p \Vdash \exists x A$*, alors, pour tous les termes* $t \in D_p$*,* $F\ p \Vdash A[t/x]$ *;*
— *si* $T\ p \Vdash \forall x A$*, alors, pour tous les mondes* $p' \geq p$ *et tous les termes* $t \in D_{p'}$*,* $T\ p' \Vdash A[t/x]$ *;*
— *si* $F\ p \Vdash \forall x A$*, alors il existe au moins un monde* $p' \geq p$ *et au moins un terme* $t \in D_{p'}$ *tels que* $F\ p' \Vdash A[t/x]$*.*

La notation utilisée, $\Vdash$, est la même que pour la relation de forcing des structures de Kripke, à l'exception notable que, dans les semi-valuations, la relation est *signée* par T ou F. Ainsi $T\ p \Vdash A$ doit se lire comme “le monde p doit forcer A” et $F\ p \Vdash A$, “le monde p doit ne pas forcer A”, ce qui est plus fort que “le monde p ne doit pas forcer A” et plus faible que “le monde p doit forcer $\neg A$” (ce qui serait un retour à un monde classique).

On peut adapter les notions de valuation partielle, de semi-valuation compatible avec un système de réécriture $\mathcal{RE}$ (définition 2.3.5) et de semi-valuation atomique (définition 2.3.9) de la section 2.3.3 au cas intuitionniste. Nous ne définissons ici que la notion de semi-valuation atomique intuitionniste ; les autres extensions suivent le même modèle.

Définition 2.5.4 (Semi-valuation intuitionniste atomique)**.** *Soit* $\mathcal{RE}$ *un système de réécriture confluent. Une semi-valuation intuitionniste* $\Vdash$ *est dite* atomique *si :*

— *pour toute formule atomique* A*, tout* $b \in \{T, F\}$ *et tout monde tel que* $b\ p \Vdash A$ *est défini, pour toute règle de réécriture* $l \longrightarrow r$ *applicable à* A*, il existe au moins une formule* B *telle que* $A \overset{l \longrightarrow r}{\longrightarrow} B$ *et* $b\ p \Vdash B$ *;*
— *pour toutes formules atomiques* A *et* B *telles que* $A =_{\mathcal{E}} B$*, pour tous* $b, b' \in \{T, F\}$ *et tout monde tel que* $b\ p \Vdash A$ *et* $b'\ p \Vdash B$ *sont définis, alors* $b = b'$*.*

Du fait de ces nouvelles définitions, et de la nature différente des tableaux intuitionnistes, il faut aussi modifier la procédure d'expansion de la définition 2.2.8.

Définition 2.5.5 (Procédure complète pour les tableaux intuitionnistes modulo théorie)**.** *Appliquer en priorité les règles de clôture,* $T_\wedge$*,* $F_\vee$ *et* $T_\exists$ *puis*

$F_\neg$, $F_\Rightarrow$ et $F_\forall$, puis $F_\wedge$ et $T_\vee$ et, enfin, $T_\neg$, $T_\Rightarrow$, $T_\forall$, $F_\exists$ et la réécriture (en alternance). Étendre à chaque fois toutes les sous-branches encore ouvertes, marquer la formule signée comme utilisée et, dans le cas du groupe $F_\neg$, $F_\Rightarrow$ et $F_\forall$:

— *créer un nouveau monde, situé au dessus du monde courant (celui de la formule signée) ;*
— *pour la règle $F_\forall$, choisir une constante fraîche.*

Dans le cas du dernier groupe :

— *répéter la formule signée (cela fait déjà partie des règles de tableaux associées) ;*
— *dans le cas des règles $T_\neg$ et $T_\Rightarrow$, énumérer les mondes apparaissant sur la sous-branche étendue, et choisir parmi ceux-ci le premier monde pour lequel la règle n'a pas encore été appliquée ;*
— *dans le cas de la règle $F_\exists$, énumérer les termes composés des symboles apparaissant sur la branche étendue au monde p en question, et choisir le premier terme pour lequel la règle n'a pas encore été appliquée ;*
— *dans le cas de la règle $T_\forall$, énumérer les couples mondes/termes pour les mondes apparaissant sur la sous-branche étendue et les termes clos composés de symboles apparaissant à ce monde sur la branche étendue ; choisir parmi ces couples le premier pour lequel la règle n'a pas encore été appliquée ;*
— *dans le cas de la règle de réécriture, énumérer les règles de réécriture s'appliquant à la formule, et choisir la première règle qui n'a pas encore été appliquée.*

Notons en particulier l'étape de recopie de la formule signée lors de l'application des règles $T_\neg$, $T_\Rightarrow$, $T_\forall$ et $F_\exists$, qui correspond *exactement* à la contraction explicite faite dans les règles $\neg_L$, $\Rightarrow_L$, $\forall_L$ et $\exists_R$ des calculs de séquents intutionnistes multiconclusion [54, 56, 108]. Notons aussi la création d'un nouveau monde dans les règles $F_\neg$, $F_\Rightarrow$ et $F_\forall$, qui correspond, là aussi, dans le calcul des séquents multiconclusion, à l'abandon des formules située à droite du séquent, c'est à dire signées négativement dans le monde d'origine.

La procédure de la définition 2.5.5 est équitable et complète, y compris dans le cas de la Déduction modulo théorie, et on peut compléter une semi-valuation de Kripke (atomique, en Déduction modulo théorie) en une valuation partielle de Kripke (compatible avec le système de réécriture $\mathcal{RE}$, respectivement) de manière similaire à la section 2.3.4. L'énumération des couples mondes/termes peut être faite de façon générique, sans inspection des symboles apparaissant sur la branche, au détriment de l'efficacité. Pour cela, il suffit d'énumérer les couples séquences finies d'entiers / termes de ce monde, avec un ensemble dénombrable de constantes fraîches propres à chaque monde.

Il est ensuite possible, aux mêmes conditions d'ordre et/ou de positivité sur le système de réécriture $\mathcal{RE}$ qu'à la section 2.4.2, et par un raisonnement similaire, de démontrer le théorème de complétude pour la méthode des tableaux intuitionniste.

Théorème 2.5.6 (Complétude de la méthode des tableaux modulo théorie)**.** *Soit $\mathcal{RE}$ un système de réécriture qui respecte une des conditions de la section 2.4.2. Soit un tableau dont la racine est $T\ \emptyset\ A_1, \cdots, T\ \emptyset\ A_n$, $F\ \emptyset\ B_1, \cdots, F\ \emptyset\ B_m$. Si aucune procédure (en particulier, procédure complète) ne termine, alors il existe une structure de Kripke qui est un modèle de $\mathcal{RE}$ et un monde w tels que $w \Vdash A_1, \cdots, w \Vdash A_n$ et $w \nVdash B_1, \cdots, w \nVdash B_m$.*

Le cas de la logique d'ordre supérieur ne peut plus être directement traité par ces méthodes, et sera vu dans le chapitre suivant, uniquement pour le calcul des séquents. Si nous souhaitions obtenir la complétude de la méthode des tableaux intuitionniste modulo la théorie définissant la logique d'ordre supérieur, alors nous aurions besoin d'un travail similaire à [99, 82], ce qui reste à faire.

2.6 Constructivisation des démonstrations

Les différentes démonstrations de complétude que nous avons esquissées sont non constructives, que cela soit par la méthode de Henkin pour le calcul des séquents (section 2.1.4), la même méthode adaptée au calcul des séquents sans coupure (section 2.1.5) ou celles, plus optimisées, des différentes méthodes de tableaux (sections 2.2, 2.3 et 2.4, ainsi que 2.5). Les étapes en cause sont, en particulier :

- la construction d'une semi-valuation (ou pire, d'une valuation partielle), qui demande soit une énumération de toutes les formules, soit "d'attendre" qu'une procédure complète nous fournisse une branche complète et ouverte ;
- dans le cas de la Déduction modulo théorie, la complétion d'une semi-valuation (atomique) en une valuation partielle (validant $\mathcal{RE}$), qui fait appel à une énumération de toutes les formules (section 2.1.5), elle-même répétée (section 2.3.4) — l'extension n'est pas définissable de manière inductive, dans le cas général —.

Le contenu calculatoire des preuves à la Henkin a pu cependant être récemment analysé [73], à l'aide d'opérateurs de contrôle.

En revanche, la preuve de complétude à l'aide des algèbres de Lindenbaum de la section 2.1.3 est, elle, constructive. Elle peut, par exemple, être formalisée dans un assistant de preuve tel qu'Agda [131] ou Coq [103] et nous exploiterons cette possibilité plus tard. Cependant, la complétude dans ce cas est pour l'instant celle du calcul des séquents avec coupure, un résultat plus faible que la complétude du calcul des séquents sans coupure.

Ce qui est en cause dans ce manque de constructivité, c'est la volonté de construire explicitement un modèle. Nous nous engageons donc à définir une interprétation pour tous les termes et formules du langage, et donc à une certaine forme d'énumération, même si nous avons affaibli cette contrainte le plus possible avec les semi-valuations.

C'est donc l'idée même de devoir construire un modèle qu'il faut critiquer. Cette idée tire son origine dans l'énoncé du théorème lui-même.

Théorème (Complétude)**.** *Soit un tableau dont la racine est $T\ \emptyset\ A_1, \cdots, T\ \emptyset\ A_n, F\ \emptyset\ B_1, \cdots, F\ \emptyset\ B_m$. Si aucune procédure ne termine, alors il existe une structure de Kripke et un monde w tels que $w \Vdash A_1, \cdots, w \Vdash A_n$ et $w \nVdash B_1, \cdots, w \nVdash B_m$.*

Deux idées, utilisées conjointement, permettent de contourner ce problème. La première est intrinsèque à la notion de modèle : en imposant la contrainte, au moins implicitement dans leur définition, qu'un modèle booléen ou une structure de Kripke doit être cohérent (c'est-à-dire, $\top \neq \bot$), cela nous impose de décider agressivement, pour toute formule, si elle est vraie ou fausse, ce qui est notoirement indécidable.

Cette contrainte de cohérence des modèles peut être relâchée, comme cela a été le cas tout d'abord dans les travaux de Veldman [138] sur les structures de Kripke faillibles, où certains mondes peuvent forcer $\bot$. Dans ce cas, si $w \Vdash \bot$, on a $w \Vdash A$ pour tout A ; on parle alors de monde qui explose. Cela permet, en particulier, une définition positive, plus constructive, de la négation, sur le modèle de celle de l'implication : $w \Vdash \neg A$ si, et seulement si, $w \Vdash A$ implique $w \Vdash \bot$. En ne cherchant surtout pas à savoir, lors de la construction, que sont ces mondes, nous gagnons un degré de liberté. Un travail similaire a été fait dans le cadre des algèbres de Boole [94], où sont autorisés les modèles booléens qui explosent, c'est-à-dire, outre les modèles usuels, le modèle réduit à un point, où $\bot = \top$.

L'idée que nous avons retenue dans le cas intuitionniste est de renforcer la condition d'explosion : si $w \Vdash \bot$, alors, pour tout monde w' (y compris incomparable avec w), pour toute formule A, $w' \Vdash A$. La structure de Kripke elle-même explose, et plus seulement les mondes. Dans les faits, cela revient donc à autoriser, à la manière de Krivine [94], une seule structure de Kripke supplémentaire, appelée structure dégénérée, ou impropre.

La seconde idée vient de l'affaiblissement de l'énoncé même du théorème de complétude. Au lieu de démontrer que, si une formule est cohérente (on ne peut en dériver de contradiction), alors il existe un modèle de celle-ci, nous cherchons à démontrer la contraposée du théorème de complétude.

Théorème 2.6.1 (Complétude faible)**.** *Supposons que toutes les structures de Kripke telles que* $w \Vdash A_1, \cdots, w \Vdash A_n$ *et* $w \nVdash B_1, \cdots, w \nVdash B_m$ *explosent. Soit un tableau dont la racine est* $T\ \emptyset\ A_1, \cdots, T\ \emptyset\ A_n, F\ \emptyset\ B_1, \cdots, F\ \emptyset\ B_m$ *et auquel on applique une procédure d'expansion complète. Alors toute branche de ce tableau sera fermée par cette procédure.*

Cet énoncé est plus faible que l'énoncé fort. S'il est classiquement équivalent aux théorèmes 2.1.14 (pour le calcul des séquents) et 2.2.7, il est, en logique constructive, strictement plus faible. Cependant, une procédure d'expansion complète se satisfait de n'importe lequel des deux énoncés : les arguments utilisés n'ont absolument aucun impact sur l'algorithme complet de recherche de preuve lui-même. D'autre part, les preuves constructives sont plus directement informatives, ce qui est un avantage lorsque l'on veut les analyser, comme cela sera notre cas.

Une esquisse de démonstration du théorème 2.6.1 est la suivante. Plaçons nous dans le cas sans règle de réécriture pour plus de simplicité, et considérons une branche à laquelle on a ainsi appliqué la procédure d'expansion jusqu'à obtenir une branche complète. Cela nous permet de définir une Structure de Kripke qui est en accord avec les formules signées rencontrées sur la branche. D'après l'hypothèse, cette structure est la structure de Kripke impropre, c'est donc qu'on a rencontré une contradiction sur la branche.

La notion de branche complète doit être ici comprise d'une manière légèrement modifiée : une branche où Íon rencontre une contradiction est définie comme complète, et par convention toutes les formules signées apparaissent sur cette branche. En effet, une telle branche contradictoire ne nous permet de définir que la structure de Kripke impropre. La notion de semi-valuation intuitionniste doit aussi être adaptée pour prendre en compte ce cas de figure.

Notons aussi que nous avons légèrement modifié la notion de preuve elle-même : il ne s'agit plus d'un tableau fini dont toutes les branches sont fermées, mais d'un tableau dont chaque branche est fermée. Enfin, tous ces arguments s'appliquent aussi au cas des tableaux classiques.

Chapitre 3

Séquents

Nous discutons plusieurs manières de démontrer l'admissibilité de la règle de coupure en calcul des séquents. La première passe par la correction syntaxique de la méthode des tableaux, qui fait l'objet d'une première partie, la seconde est plus algébrique et permet le passage à l'ordre supérieur dans les logiques intuitionnistes et linéaires, ce qui fait l'objet d'une seconde partie. Dans une dernière partie, nous introduisons des traduction par double négation, qui permettent de porter les résultats d'élimination des coupures de la logique intuitionniste vers la logique classique.

La section 3.1 introduit la première partie et les développements qui suivent, elle est tirée du folklore et de la littérature. La section 3.1.2 est tirée des articles [18, 82] alors que la section 3.1.3 suivante fait l'objet de l'article [21] (non encore publié).

De même, la section 3.2.1 introduit la deuxième partie en mettant bout à bout tous les résultats obtenus jusqu'à présent afin d'obtenir l'élimination des coupures, selon un schéma standard. Les sections 3.2.2 et 3.2.3 sont tirées de [20] et d'une simplification de [81, 80] au cas du calcul des séquents au premier ordre. Ces deux derniers papiers donnent aussi la matière discutée à la section 3.2.4 suivante. L'extension de cette section à la logique linéaire, traitée à la section 3.2.5, est l'objet de l'article [83] (non encore publié).

Dans dernière partie de ce chapitre, la section 3.3.1 s'appuie sur [22] pour et la section 3.3.2 s'appuie sur [1].

3.1 Tableaux et calcul des séquents

Nous commençons ce chapitre par une étude des relations syntaxiques entre le calcul des séquents sans coupure et la méthode des tableaux classique, la méthode des tableaux intuitionniste, et la méthode des tableaux classiques avec variables libres.

3.1.1 Logique classique

Un tableau en logique classique correspond exactement à une preuve en calcul des séquents sans coupure ; les règles T_* correspondent aux règles gauche sur le connecteur $*$ et les règles F_*, aux règles droites. Prenons l'exemple de la règle $T_\vee$. Elle partage une grande similarité avec la règle $\vee_L$ du calcul des séquents. L'analogie est forte, si l'on compare

$$T_\vee \frac{T\ A \vee B}{T\ B \qquad T\ A} \quad \text{et} \quad \vee_L \frac{\Gamma, B \vdash \Delta \qquad \Gamma, A \vdash \Delta}{\Gamma, A \vee B \vdash \Delta} .$$

La seule différence réside dans la présentation des règles, du haut vers le bas pour les tableaux et du bas vers le haut pour les séquents. Quelques points de détails supplémentaires sont à régler.

— La règle $T_\vee$ prend une formule située à un certain nœud de l'arbre, et étend une ou plusieurs feuilles ouvertes situées en dessous de ce nœud. La règle $\vee_L$ du calcul des séquents est au contraire purement locale, et étend une feuille à la fois. La méthode adaptée pour ceci est la méthode des tableaux non destructives, où l'on collecte toutes les formules d'un chemin au niveau de chaque nœud, y compris bien sûr les feuilles. Transformer un tableau destructif en tableau non destructif est immédiat.

— En collectant toutes les formules à chaque nœud, on obtient une règle de séquent non destructive. Par exemple, la règle $T_\vee$ des tableaux sera, en calcul des séquents,
$$\frac{\Gamma, A \vee B, B \vdash \Delta \qquad \Gamma, A \vee B, A \vdash \Delta}{\Gamma, A \vee B \vdash \Delta} \vee'_L .$$
Pour dériver cette règle $\vee'_L$, il suffit de systématiser l'usage de la règle de contraction dans le calcul des séquents, comme montré dans la preuve ci-dessous.
$$\dfrac{\dfrac{\Gamma, A \vee B, B \vdash \Delta \qquad \Gamma, A \vee B, A \vdash \Delta}{\Gamma, A \vee B, A \vee B \vdash \Delta} \vee_L}{\Gamma, A \vee B \vdash \Delta} \mathrm{contr}_L$$

Après ces quelques ajustements, nous obtenons directement une démonstration du résultat suivant.

Théorème 3.1.1 (Correction syntaxique de la méthode des tableaux)**.** *Si le tableau ayant pour racine $T\ \Gamma$, $F\ \Delta$ peut être fermé, alors il existe une preuve du séquent $\Gamma \vdash \Delta$ sans coupure.*

3.1.2 Logique intuitionniste

Le théorème 3.1.1 précédent ne se généralise pas aussi aisément à la méthode des tableaux intuitionniste. Intéressons nous au cas de la règle $F_\vee$ intuitionniste, qui concentre la majorité des difficultés,

$$\frac{F\ p\ A \vee B}{F\ p\ A, F\ p\ B}\ F_{\vee}.$$

Si l'on tente une traduction directe en termes de séquents, nous devrions pouvoir permettre la dérivation

$$\frac{\Gamma \vdash A, B}{\Gamma \vdash A \vee B}.$$

Or, le calcul des séquents intuitionniste tel que nous l'avons défini à la section 2.1.1 ne permet pas d'avoir plusieurs conclusions en même temps. La réponse à ce problème vient du calcul des séquents intuitionniste *multi-conclusion* [55, 139], qui permet justement ce genre de règle.

Ce calcul des séquents intuitionniste multiconclusion a été introduit dès les années 1950 [54, 90, 101, 130] et se rapproche beaucoup des tableaux intuitionnistes, de la même manière que le calcul des séquents classique se rapproche des tableaux classiques. De manière équivalente, nous pouvons démontrer, dans le calcul des séquents intuitionniste de la section 2.1.1, les résultats suivants [82]. Ce sont aussi ces résultats qui permettent de passer syntaxiquement d'un calcul des séquents intuitionniste multiconclusion à un calcul des séquents intuitionniste monoconclusion.

Proposition 3.1.2. *Les règles suivantes sont admissibles, en calcul des séquents intuitionniste, avec ou sans la règle de coupure.*

$$\wedge_R\text{-}ext\ \frac{\Gamma \vdash A \vee C \qquad \Gamma \vdash B \vee C}{\Gamma \vdash (A \wedge B) \vee C} \qquad\qquad \exists_R\text{-}ext\ \frac{\Gamma \vdash A[t/x] \vee C}{\Gamma \vdash (\exists x A) \vee C}$$

$$\Rightarrow_R\text{-}ext\ \frac{\Gamma \vdash A \vee C \qquad \Gamma, B \vdash C}{\Gamma, A \Rightarrow B \vdash C} \qquad\qquad contr_R\text{-}ext\ \frac{\Gamma \vdash A \vee B}{\Gamma \vdash (A \vee A) \vee B}$$

$$comm_R\ \frac{\Gamma \vdash A \vee B}{\Gamma \vdash B \vee A} \qquad\qquad assoc_R\ \frac{\Gamma \vdash A \vee (B \vee C)}{\Gamma \vdash (A \vee B) \vee C}$$

Les deux dernières règles admissibles montrent que nous pouvons utiliser la disjonction (au moins à droite) comme un symbole associatif et commutatif, et justifie la notation n-aire de ce symbole. Cela nous permet de traduire $T\ p\ A_1, \cdots, T\ p\ A_n\ F\ p\ B_1, \cdots, F\ p\ B_m$ par le séquent $A_1, \cdots, A_n \vdash B_1 \vee \cdots \vee B_m$. La règle $F_{\vee}$ se traduit alors par rien, ou éventuellement une contraction à droite contr$_R$-ext, admissible comme on l'a vu.

Le second problème, que nous avons soigneusement évité d'aborder jusqu'à maintenant, est celui des mondes. Dans le paragraphe précédent, nous avons considéré des formules signées $T\,p\,A_1, \cdots, T\,p\,A_n, F\,p\,B_1, \cdots, F\,p\,B_m$ avec un monde p fixe, mais en réalité sur une branche du tableau apparaissent les formules $T\ p_1\ A_1, \cdots, T\ p_n\ A_n, F\ q_1\ B_1, \cdots, F\ q_m\ B_m$. Aussi, il faut se fixer un monde q, et on traduira alors la séquence ci-dessus en le séquent

$\{A_i \mid p_i \leq q\} \vdash \bigvee\{B_j \mid q_j = q\}$. En effet, d'après l'interprétation dans les structure de Kripke, si une formule est forcée à un monde, elle le sera pour tout monde futur ; ainsi on choisit toutes les formules forcées à des mondes inférieurs à q. On pourrait éventuellement choisir tous les mondes *après* q dans le membre droit du séquent.

La question qui se pose ensuite est la suivante : sur quel monde appliquer la transformation en séquent ci-dessus ? Le choix est facile à faire dans le cas de la fermeture d'une branche ; dans le cas général, le seul impératif est de choisir un monde qui apparaît sur le chemin de la racine au nœud π que l'on étend. En posant les notations

$\mathcal{T}_p(\pi){:=}\{A \mid T\ q\ A$ apparaît sur le chemin menant à π, avec $q \leq p\}$ et
$\mathcal{F}_p(\pi){:=}\{B \mid F\ p\ B$ apparaît sur le chemin menant à $\pi\},\}$

nous démontrons le théorème suivant.

Théorème 3.1.3 (Correction syntaxique locale de la méthode des tableaux intuitionniste)**.** *Considérons un tableau intuitionniste fermé, et un nœud interne π. Alors il existe un monde p apparaissant sur le chemin menant à π tel que $\mathcal{T}_p(\pi) \vdash \bigvee \mathcal{F}_p(\pi)$ est prouvable.*

La démonstration s'appuie sur une induction, le cas de base étant celui d'un nœud qui est une feuille close de l'arbre, ce qui signifie que l'on a rencontré $T\ p\ A$ et $F\ q\ A$ pour deux mondes $p \leq q$; on choisit alors le monde q, et nous pouvons appliquer la règle axiome (étendue) au séquent $\mathcal{T}_q(\pi) \vdash \bigvee \mathcal{F}_q(\pi)$.

Dans le cas général, le choix du monde dépend fortement de ce que l'on obtient par l'hypothèse d'induction. Notamment, l'hypothèse d'un monde nouveau et incomparable aux précédents dans le cas des règles $F_\Rightarrow$ et $F_\forall$ est primordiale. Considérons le cas de la règle

$$\frac{T\ p\ A \Rightarrow B}{T\ p.q\ B \qquad F\ p.q\ A}\, T_\Rightarrow.$$

L'hypothèse d'induction sur la branche de droite π_r nous donne, pour un certain monde q_r, une preuve de $\mathcal{T}_{q_r}(\pi_r) \vdash \mathcal{F}_{q_r}(\pi_r)$. Nous pouvons supposer $q_r = p.q$; dans le cas contraire, $A \notin \mathcal{F}_{q_r}(\pi_r)$ et la preuve de séquent s'applique aussi au nœud père π avec le monde q_r, car on a alors $\mathcal{T}_{q_r}(\pi) = \mathcal{T}_{q_r}(\pi_r)$ et $\mathcal{F}_{q_r}(\pi) = \mathcal{F}_{q_r}(\pi_r)$.

De même, l'hypothèse d'induction sur la branche de gauche nous donne $\mathcal{T}_{q_l}(\pi_l) \vdash \mathcal{F}_{q_l}(\pi_l)$, et nous pouvons supposer $q_l \geq p.q$; sinon B n'apparaît pas dans $\mathcal{T}_{q_l}(\pi_l)$. Nous avons malheureusement besoin de l'hypothèse plus forte $q_l = p.q$ pour pouvoir conclure. Supposons pour l'instant que cela soit le cas. On peut alors utiliser la règle $\Rightarrow_R$-ext de la proposition 3.1.2 pour obtenir

$$\frac{\mathcal{T}_{p.q}(\pi) \vdash A, \mathcal{F}_{p.q}(\pi) \qquad \mathcal{T}_{p.q}(\pi), B \vdash \mathcal{F}_{p.q}(\pi)}{\mathcal{T}_{p.q}(\pi), A \Rightarrow B \vdash \mathcal{F}_{p.q}(\pi)}\, \Rightarrow_R \text{-ext},$$

en se servant des faits $A \Rightarrow B \in \mathcal{T}_{p.q}(\pi)$, $\mathcal{F}_{p.q}(\pi_r) = A, \mathcal{F}_{p.q}(\pi)$ et $\mathcal{T}_{p.q}(\pi_l) = \mathcal{T}_{p.q}(\pi), B$. Nous pouvons donc conclure, car nous savons que $p.q$ est un monde qui apparaît sur le chemin π, sinon la règle $T_{\Rightarrow}$ ne se serait pas appliquée avec le monde $p.q$.

Notre problème final est donc de s'assurer que l'on peut choisir $q_l = p.q$ lors de l'application de l'hypothèse d'induction à la sous-preuve gauche. Pour ceci, nous devons faire en sorte qu'il n'existe aucun monde q' sur π tel que $q' > p.q$, c'est à dire $q' = p.q.r$. Ces mondes sont introduits par les règles $F_\forall$ et $F_\Rightarrow$, et il est possible de faire permuter vers le bas toutes les formules (et les règles) mettant en jeu ces mondes. Ceci est possible, car la règle $T_\Rightarrow$ au monde p est complètement indépendante de ces mondes ; en particulier les constantes fraîches introduites par les règles $F_\forall$ n'ont d'existence que dans les mondes supérieurs à $p.q.r$, et ne peuvent apparaître dans les mondes inférieurs. Si on “fait descendre” toutes ces règles, en conservant leur ordre relatif, aucune contrainte de fraîcheur ne sera invalidée.

Ainsi à partir d'un tableau intuitionniste clos, nous pouvons effectivement récupérer une preuve sans coupure en calcul des séquents par la démarche présentée dans cette section. Ce que nous avons présenté de manière informelle ici demande encore à être décrit plus précisément et reste donc pour le moment au statut de conjecture. Le théorème 3.1.3 a ensuite le corollaire suivant.

Corollaire 3.1.4 (Correction des tableaux intuitionnistes)**.** *Si le tableau intuitionniste dont la racine comporte les formules signées $T\ \emptyset\ \Gamma$, $F\ \emptyset\ B$ peut être fermé, alors le séquent $\Gamma \vdash B$ est prouvable sans coupure en calcul des séquents intuitionniste.*

3.1.3 Tableaux classiques avec variables libres

Revenons maintenant aux tableaux classiques, mais cette fois-ci avec variables libres. Comme nous le savons, celles-ci, grâce à la skolémisation, permettent de trouver des preuves plus compactes [7, 70, 28] qu'en calcul des séquents, par exemple la preuve de l'énoncé du buveur ci-dessous.

$$\cfrac{\cfrac{\cfrac{F\ \exists y(P(y) \Rightarrow \forall x P(x))}{F\ \exists y(P(y) \Rightarrow \forall x P(x)), F\ P(Y) \Rightarrow \forall x P(x)}\,F_\exists}{T\ P(Y), F\ \forall x P(x)}\,F_\Rightarrow}{\cfrac{F\ P(a)}{\odot\ [a/Y]}}\,T_\forall \ .$$

Cette preuve est strictement plus courte que la plus courte preuve existante en calcul des séquents ou par la méthode des tableaux sans variables libres. Et en effet, il n'est pas possible de traduire ce tableau de manière inductive, car on se heurte à un problème de fraîcheur. Prenons le tableau ci-dessus, et appliquons lui la substitution fermante $[a/Y]$. Nous remplaçons toutes les occurrences de Y par a, puis nous traduisons directement chacune des étapes en une preuve du calcul des séquents, en commençant par les feuilles ou la racine. On obtient alors la pseudo-preuve, où les règles de contraction sont implicites,

$$\dfrac{\dfrac{\dfrac{\dfrac{}{P(a) \vdash P(a), \forall y P(y), P(a) \Rightarrow \forall y P(y), \exists x(P(x) \Rightarrow \forall y P(y))}\text{ax}}{P(a) \vdash \forall y P(y), P(a) \Rightarrow \forall y P(y), \exists x(P(x) \Rightarrow \forall y P(y))}\forall_R}{\vdash P(a) \Rightarrow \forall y P(y), \exists x(P(x) \Rightarrow \forall y P(y))}\Rightarrow_R}{\vdash \exists x(P(x) \Rightarrow \forall y P(y))}\exists_R \; .$$

Le problème est que la règle $\forall_R$ ne vérifie pas la condition de fraîcheur requise : la constante a introduite est censée être fraîche, mais elle ne l'est pas. Dit autrement, les termes de Skolem ne sont pas nécessairement frais dans le tableau après instanciation de celui-ci, ce qui invalide la structure de la preuve en tant que preuve du calcul des séquents. Cependant, la méthode des tableaux avec variables libres est correcte, mais la cohérence de cette méthode est *globale*, lorsque l'on donne une valeur aux métavariables, alors que la contrainte de fraîcheur est purement *locale*.

C'est cette tension local/global qu'il faut résoudre, en remettant les règles d'inférence dans le bon ordre. Comme la taille des preuves peut décroître avec l'introduction de termes de Skolem et de variables libres, retourner à un monde avec constantes fraîches et termes clos aura forcément comme effet de bord de faire grossir les preuves. Une simple permutation des règles, à la Kleene, ne nous suffira donc pas.

Nous sommes dans le cadre du calcul des séquents sans coupure, dans le cas contraire, nous pourrions tenter d'utiliser la stratégie suivante : avant chaque traduction d'une règle $T_\forall$ ou $F_\exists$ qui introduit un certain terme t, faire une coupure sur toutes les formules quantifiées sur lesquelles, dans le tableau instancié, on introduit un terme de Skolem qui est un sous-terme de t. Puis, on applique la règle $\forall_L$ ou $\exists_R$ correspondante. Dans notre exemple cela signifie une coupure sur la formule $\forall x P(x)$ et l'application de la règle $\forall_R$ pour introduire a, juste avant la règle $\exists_R$.

Ensuite, il faudrait recopier, dans toutes les nouvelles branches ainsi introduites, le tableau entier et gérer les clôtures différenciées selon que l'on est dans la banche de gauche ou de droite. Dans le cas général, cela est plus difficile : un des formules sur lesquelles on coupe pourrait contenir un terme de Skolem b qui n'est pas un sous-terme de t et provoquer un nouveau conflit de fraîcheur. Pour que cela fonctionne bien, il faudrait au final appliquer des techniques similaires à celles que nous développons dans le cadre sans coupure.

Avant de discuter le cas général, voyons comment cette traduction est possible sur l'exemple précédent. La preuve de l'énoncé du buveur en calcul des séquents demande d'appliquer plusieurs fois certaines règles. Comme nous cherchons à traduire un arbre de preuve, et non pas à rechercher une preuve à partir de rien, nous devons nous appuyer sur l'arbre de dérivation du tableau lui-même, ce qui se traduit en une recopie à certains moments de tout ou partie du tableau global, ou du moins de son squelette, c'est-à-dire ses règles (les formules pouvant être différentes).

Plus précisément, nous ne procéderons plus par une induction classique sur la preuve dont nous disposons par la méthode des tableaux, qui commencerait par les feuilles, mais par une induction qui commence par la racine, et qui traduit les règles une à une. Nous donnerons le nom technique de greffe à la recopie d'une structure de preuve au niveau d'un nœud interne. Cette preuve sera correcte, au sens que toutes ses règles seront appliquées correctement, mais en général partielle, c'est-à-dire que ses feuilles seront ouvertes.

La figure 3.1 présente les trois premières étapes de l'application de cette procédure donne sur l'exemple du théorème du buveur, en notation abrégée. Chaque étape correspond à la traduction d'une règle de l'arbre de preuve des tableaux.

Étape 1	$\vdash \exists x(P(x) \Rightarrow \forall y P(y))$
Étape 2	$\dfrac{\vdash P(a) \Rightarrow \forall y P(y), \exists x(P(x) \Rightarrow \forall y P(y))}{\vdash \exists x(P(x) \Rightarrow \forall y P(y))}\ \exists_R$
Étape 3	$\dfrac{\dfrac{P(a) \vdash \forall y P(y), P(a) \Rightarrow \forall y P(y), \exists x(P(x) \Rightarrow \forall y P(y))}{\vdash P(a) \Rightarrow \forall y P(y), \exists x(P(x) \Rightarrow \forall y P(y))}\ \Rightarrow_R}{\vdash \exists x(P(x) \Rightarrow \forall y P(y))}\ \exists_R$

FIGURE 3.1 – Les trois premières étapes de la traduction de la preuve de l'énoncé du buveur

À ce point de la traduction, nous ne pouvons pas appliquer la règle $\forall_R$, pour les problèmes de fraîcheur mentionnés plus haut. Appelons π la preuve partielle de séquents à l'étape 3 de la figure 3.1. La stratégie consiste à :

— *nettoyer* le séquent avec la règle weak, de manière à éliminer les formules comportant le terme de Skolem (a dans notre exemple),

figure 3.2a ;
- appliquer la règle $\forall_R$, ce qui est maintenant possible, car a est frais, qui introduit la “formule de Skolem” $P(a)$, figure 3.2b ;
- *greffer* la structure de π intégralement au niveau de ce nœud, figure 3.2c, ce qui est possible puisque la conclusion de π ne comporte aucun symbole de Skolem : les formules nécessaires à la greffe sont donc toutes disponibles, et nous conservons *en plus* la formule de Skolem $P(a)$ (figure 3.2b), ce qui ne pose pas de problème de fraîcheur. C’est cette formule préservée, que nous distinguons en gras, qui nous permettra de continuer à appliquer les règles ensuite.

$$\dfrac{\dfrac{\dfrac{\vdash \Delta, \forall y P(y)}{P(a) \vdash \Delta, P(a) \Rightarrow \forall y P(y), \forall y P(y)}\,\text{weak}}{\vdash \Delta, (P(a) \Rightarrow \forall y P(y))}\,\Rightarrow_R}{\vdash \Delta}\,\exists_R$$

(a) Nettoyage

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\vdash \Delta, \mathbf{P(a)}}{\vdash \Delta, \forall y P(y), P(a)}\,\text{weak}}{\vdash \Delta, \forall y P(y)}\,\forall_R}{P(a) \vdash \Delta, P(a) \Rightarrow \forall y P(y), \forall y P(y)}\,\text{weak}}{\vdash \Delta, (P(a) \Rightarrow \forall y P(y))}\,\Rightarrow_R}{\vdash \Delta}\,\exists_R$$

(b) Application de la δ-règle $\forall_R$ et second nettoyage

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{P(a) \vdash \Delta, \mathbf{P(a)}, P(a) \Rightarrow \forall y P(y), \forall y P(y)}{\vdash \Delta, \mathbf{P(a)}, P(a) \Rightarrow \forall y P(y)}\,\Rightarrow_R}{\vdash \Delta, \mathbf{P(a)}}\,\exists_R}{\vdash \Delta, \forall y P(y), P(a)}\,\text{weak}}{\vdash \Delta, \forall y P(y)}\,\forall_R}{P(a) \vdash \Delta, P(a) \Rightarrow \forall y P(y), \forall y P(y)}\,\text{weak}}{\vdash \Delta, P(a) \Rightarrow \forall y P(y)}\,\Rightarrow_R}{\vdash \Delta}\,\exists_R$$

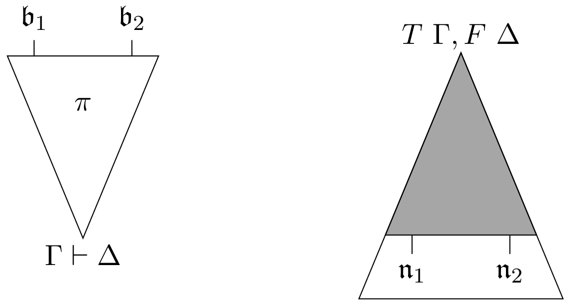
(c) Greffe

FIGURE 3.2 – Traduction de la dernière règle du théorème du buveur

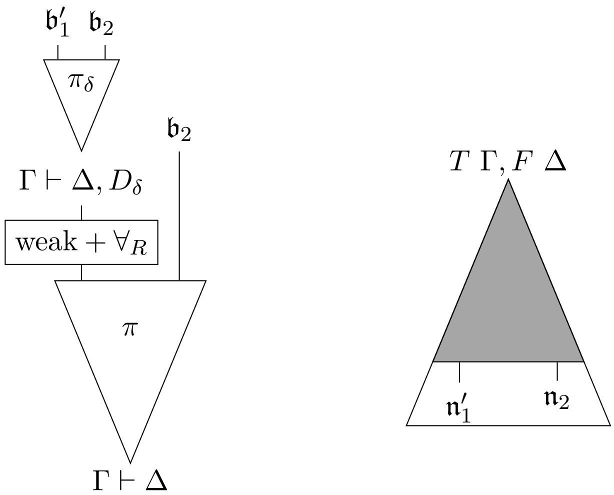
Nous avons donc progressé d’une étape dans la traduction de la preuve de tableaux, et pouvons procéder à la dernière étape, à savoir la recopie de la règle de clôture du tableau, qui devient un axiome. Nous obtenons au final la preuve suivante qui, si elle n’est pas la preuve optimale en calcul des séquents, a été construite de manière mécanique et fournit un point de départ à la solution générale.

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{P(a) \vdash \Delta, \mathbf{P}(\mathbf{a}), P(a) \Rightarrow \forall y P(y), \forall y P(y)}\,\text{ax}}{\vdash \Delta, \mathbf{P}(\mathbf{a}), P(a) \Rightarrow \forall y P(y)}\,\Rightarrow_R}{\vdash \Delta, \mathbf{P}(\mathbf{a})}\,\exists_R}{\vdash \Delta, \forall y P(y), P(a)}\,\text{weak}}{\vdash \Delta, \forall y P(y)}\,\forall_R}{P(a) \vdash \Delta, P(a) \Rightarrow \forall y P(y), \forall y P(y)}\,\text{weak}}{\vdash \Delta, P(a) \Rightarrow \forall y P(y)}\,\Rightarrow_R}{\vdash \Delta}\,\exists_R
$$

Lorsque nous généralisons cette approche, nous nous heurtons au premier problème suivant. Supposons que nous soyons arrivés à une ébauche de preuve, où nous avons distingué deux feuilles ouvertes, $\mathfrak{b}_1$ et $\mathfrak{b}_2$ de la preuve des séquents, liées à deux nœuds internes $\mathfrak{n}_1$ et $\mathfrak{n}_2$ de la preuve tableau. Nous appellerons aussi $\mathfrak{n}_1$ et $\mathfrak{n}_2$ directement des branches.



Supposons qu'une δ-règle s'applique dans le tableau au nœud $\mathfrak{n}_1$ (auquel $\mathfrak{b}_1$ fait référence). Appelons D_δ la δ-formule et $\mathfrak{n}'_1$ le nœud père. Nous voulons traduire cette règle. Par la procédure ci-dessus, nous passerons à la preuve suivante, où $\mathfrak{b}'_1$ est maintenant liée à $\mathfrak{n}'_1$, la prémisse de la δ-règle.



Nous pouvons remarquer plusieurs phénomènes.

1. Nous avons bien progressé, puisque nous avons consommé une règle du tableau d'origine, et nous avons maintenant un tableau dont les branches ouvertes sont liées soit à $\mathfrak{n}_1'$, soit à $\mathfrak{n}_2$.
2. La branche $\mathfrak{b}_2$ est dupliquée. Dans le cas général, l'ébauche de preuve en calcul des séquents peut donc avoir *plusieurs* feuilles liées à la même branche de tableau. Dans notre exemple nous devrons traduire la règle de tableaux qui s'applique à la branche $\mathfrak{n}_2$ deux fois, et généralement un nombre arbitraire de fois.
3. La partie de la preuve notée π_δ a la même structure que π, mais comporte la δ-formule D_δ en plus, $P(a)$ dans notre exemple. Celle-ci est inutile pour la branche $\mathfrak{b}_2$ de π_δ et peut en être éliminée.
4. Dans le cas le plus général, π peut contenir des termes de Skolem contenus dans le δ-terme D_δ que nous désirons préserver dans π_δ. Ajouter cette δ-formule aux séquents de π_δ, même si on se restreint aux séquents descendants de $\mathfrak{b}_1'$, peut donc invalider certaines conditions de fraîcheur dans π_δ.

Notre approche doit donc être encore généralisée si nous voulons qu'elle aboutisse à une preuve valide. Par exemple, nous sommes amenés à développer la notion de partie initiale d'un tableau (puisque l'induction se fait à partir de la racine) et le point 2 demande une fonction d'association, non injective (mais surjective), entre les branches ouvertes de la preuve de séquent et les branches ouvertes de la partie initiale du tableau. Cela permet de suivre avec précision quelles sont les branches de la preuve en calcul des séquents qui doivent être étendues lorsque l'on traite une règle de la preuve de tableaux.

Le point 4 est de loin le plus ardu ; sa résolution a pour corollaire la résolution de tous les autres. Pour parvenir à nos fins, nous devons étendre la notion de lien entre preuve partielle de séquent et partie initiale d'un tableau (supposé instancié) à la notion d'une liaison double : une preuve partielle de séquent (la preuve courante, π_0) est doublement liée à un couple "preuve partielle de séquent π / partie initiale de cette même preuve partielle π" si

— certaines branches, "gelées", sont liées aux branches ouvertes de la preuve π, π elle-même étant liée à la partie initiale du tableau ;

— les autres branches sont liées à une partie initiale de π, qui sert à construire la preuve courante. Lorsque nous arrivons à la fin de la construction, c'est-à-dire quand la partie initiale de π est π elle-même, nous aurons un lien double avec π/π et les branches pourront être collectées et rattachées à la partie initiale du tableau.

Ceci est nécessaire pour savoir ajouter par induction double, primaire sur la taille du δ-terme et secondaire sur la taille de la preuve partielle (partie initiale de π), une δ-formule donnée dans les branches de π “qui nous intéressent”, c'est-à-dire liées à la branche de tableau à laquelle s'applique la δ-règle. Nous commençons avec $\pi_0 := \pi$, puis par lier les branches “qui nous intéressent” à la partie initiale vide de π, en rajoutant la δ-formule D_δ par nettoyage et application (figures 3.2a et 3.2b), tandis que nous gelons et rattachons toutes les autres à π. Puis nous faisons croître π par ces premières branches, en faisant grossir la partie initiale règle par règle. Si nous rencontrons une δ-règle introduisant un δ-terme qui apparaît dans D_δ, ce δ-terme est structurellement plus petit et nous pouvons donc appliquer l'hypothèse d'induction qui, si nous faisons attention, nous fournira un nouveau tableau qui aura un lien double de la bonne nature, nous permettant ainsi de continuer notre tâche. La correction de ce processus est assurée par le tableau lui-même et l'existence de la substitution fermante : la contrainte *globale* d'existence nous assure que l'ordre sur les termes de Skolem est bien fondé, et la manière dont les termes sont définis assure que les règles de quantificateurs seront appliquées dans le bon ordre. Il est important de remarquer que tous les termes et formules que nous manipulons proviennent de ce tableau.

Il suffit ensuite d'applique ce résultat à π, comme preuve courante, et π, comme preuve partielle, pour conclure, dans le cas général, que nous savons progresser y compris dans le cas d'une δ-règle dans le tableau. Il s'agit au final de notions, d'énoncés et de démonstrations relativement techniques, dont nous espérons avoir pu retracer les grandes lignes. Une alternative serait de s'intéresser aux tableaux connectés comme objet intermédiaire, et à une double preuve syntaxique de correction [13]. On obtient dans les deux cas le théorème de correction syntaxique voulu.

Théorème 3.1.5 (Correction des tableaux avec variables libres)**.** *Soit un tableau ayant pour racine l'ensemble de formules signées $T\ \Gamma$, $F\ \Delta$. S'il peut être fermé, alors le séquent $\Gamma \vdash \Delta$ est prouvable sans coupure en calcul des séquents classique.*

3.2 Élimination des coupures

Nous avons terminé de développer tous les outils nécessaires qui nous permettront de démontrer l'admissibilité des coupures en calcul des séquents. Ce point central du manuscrit fait l'objet de cette partie. Nous commencerons par nous servir de la méthode des tableaux, puis nous passerons aux preuves algébriques, que nous appliquerons au cas de l'ordre supérieur, avec une incursion en logique linéaire.

3.2.1 Correction + complétude + correction = élimination

Nous venons de dépenser une certaine énergie à démontrer la correction *syntaxique* de la méthode des tableaux, alors qu'une preuve de correction sémantique aurait été plus immédiate. La raison tient au théorème suivant

Théorème 3.2.1 (Élimination des coupures en calcul des séquents)**.** *Si le séquent $\Gamma \vdash \Delta$ a une preuve en calcul des séquents (classique, intuitionniste), alors ce séquent a une preuve sans coupure.*

Ce théorème est aussi démontrable dans le cas de la Déduction modulo théorie, si l'on considère un système de réécriture $\mathcal{RE}$ qui vérifie l'une des conditions de la section 2.4. Le seul système de réécriture dont les résultats sont spécifiques à la logique classique (pour l'instant) est celui permettant d'exprimer la logique d'ordre supérieur.

La preuve tient à la mise bout à bout de trois résultats : la correction du calcul des séquents vis-à-vis de la sémantique (algèbres de Boole, structures de Kripke), la complétude de la méthode des tableaux, et la correction *syntaxique* de la méthode des tableaux, qui génère des preuves sans coupure.

- Si $\Gamma \vdash \Delta$ a une preuve, nous obtenons par les théorèmes de correction les faits suivants.
  - Dans le cas classique, voir le théorème 2.1.5 et en Déduction modulo théorie le théorème 2.3.2, $\bigwedge [\![\Gamma]\!]_\varphi \leq \bigvee [\![\Delta]\!]_\varphi$ pour toute algèbre de Boole, toute valuation et toute interprétation (qui est un modèle de $\mathcal{RE}$ dans le cas de la Déduction modulo théorie).
  - Dans le cas intuitionniste, nous obtenons par le théorème 2.5.2 que, pour toute structure de Kripke, toute valuation indexée φ et tout monde w, si, pour tout $A \in \Gamma, w \Vdash_\varphi A$, alors $w \Vdash_\varphi \Delta$. Rappelons que Δ ne comporte alors qu'au plus une formule. En Déduction modulo théorie, nous obtenons ce résultat pour les structures qui sont des modèles de $\mathcal{RE}$ uniquement (voir la définition 2.3.1 et la définition 2.5.1).
- D'après les théorèmes de complétude du chapitre 2, dans le cas classique, le tableau ayant pour racine $T\ \Gamma, F\ \Delta$ a une preuve et, dans le cas intuitionniste, le tableau ayant pour racine $T\ \emptyset\ \Gamma, F\ \emptyset\ \Delta$ a une preuve. Dans le cas classique, on peut choisir les tableaux avec ou sans variables libres.
- D'après les résultats de la section 3.1 précédente, il est possible de transformer cette preuve de tableaux en preuve du calcul des séquents, et la règle de coupure n'est pas utilisée lors de cette traduction.
  - Dans le cas classique, on obtient directement une preuve de $\Gamma \vdash \Delta$ par le théorème 3.1.1 dans le cas des tableaux sans variables libres ou de $\Gamma\sigma \vdash \Delta\sigma$ dans le cas de tableaux avec variables libres. Comme σ est une substitution fermante et que Γ et Δ

ne contiennent aucune variable libre, on obtient bien une preuve du séquent $\Gamma \vdash \Delta$. Dans les deux cas, aucune coupure n'est introduite.

— Dans le cas intuitionniste, le théorème 3.1.3 nous garantit que, pour les formules à la racine du tableau, il existe un monde p tel que $\mathcal{T}_p(\varepsilon) \vdash \bigvee \mathcal{F}_p(\varepsilon)$ est prouvable. Or, le seul monde apparaissant à la racine est la séquence vide $\emptyset$, et d'autre part il n'y a au plus qu'une seule formule signée négativement à la racine. On obtient donc bien une preuve de $\Gamma \vdash \Delta$ et, encore une fois, aucune coupure n'a été introduite dans la traduction.

Dans le chapitre 2, nous avons fait des efforts pour aller vers des preuves de complétude constructives. On remarque qu'on peut tout à fait utiliser les théorèmes de complétude faible de la section 2.6 dans les arguments ci-dessus. Comme les preuves de correction sémantique du calcul des séquents et de correction syntaxique de la méthode des tableaux sont constructives, nous obtenons donc un théorème d'élimination des coupures presque entièrement constructif.

Dans le cas le plus simple, celui des tableaux en logique classique sans variable libre, la traduction “tableaux vers séquent” est à peu de choses près l'identité ; il ne nous reste plus qu'à examiner la correction du calcul des séquents et la complétude de la méthode des tableaux. L'algorithme derrière ce dernier théorème est accessible : il s'agit de la procédure complète d'expansion de tableaux (par exemple, la procédure STEP introduite section 2.3.7). L'algorithme d'élimination des coupures reconstruit ainsi, à partir de rien, une preuve sans coupure du séquent. Il ne s'appuie en particulier pas sur la preuve d'origine (avec coupures).

Cependant, la preuve d'origine sert à démontrer le théorème de correction, et la validité universelle du séquent $\Gamma \vdash \Delta$ sert ensuite à justifier le fait que toute branche du tableau est fermée (explose) et témoigne de la terminaison de la procédure complète. Nous savons “sémantiquement” que la branche sur laquelle nous travaillons est contradictoire, et que nous finirons donc par aboutir.

Cette justification — on peut aussi parler de preuve au niveau du discours — est le reflet de la preuve d'origine ; elle comporte en particulier ses coupures, traduites par la transitivité de la relation $\leq$. Elle ne modifie en rien le comportement de la procédure d'expansion de tableaux, mais nous fournit un moyen d'observer celle-ci. Afin d'étudier plus précisément son interaction avec la procédure, nous aurions besoin de formaliser la preuve de correction, et même de réifier la justification, afin de pouvoir la manipuler explicitement.

3.2.2 Méthodes algébriques d'élimination des coupures

Les théorèmes d'élimination des coupures précédents font référence soit à l'algèbre de Boole $\{\bot, \top\}$, éventuellement à l'algèbre triviale $\{\bot\}$, soit à leur pendant intuitionniste que sont les structures de Kripke, éventuellement impropres.

Nous avons cependant à notre disposition une sémantique bien plus riche que la simple algèbre de Boole $\{0, 1\}$ dans le cas classique, et une sémantique alternative, les algèbres de Heyting, dans le cas intuitionniste. Nous allons obtenir dans cette partie de notre travail l'élimination des coupures à partir de la même idée correction/complétude, mais cette fois ci sans passer par un calcul intermédiaire (la méthode des tableaux), ce qui nous épargne une étape.

Nous pouvons d'ores et déjà appliquer cette méthode, grâce aux algèbres de Lindenbaum :

— en logique classique, en combinant les théorèmes 2.1.5 et 2.1.9 ;

— en logique intuitionniste, en combinant les même théorèmes appliqués au calcul des séquents intuitionniste et aux algèbres de Heyting.

Cependant, l'algèbre de Lindenbaum est définie à partir d'un calcul des séquents *avec* coupures ; il n'est donc pas possible de dériver le théorème d'élimination des coupures et on se retrouve, peu ou prou, avec la même preuve qu'au départ. L'algèbre construite avec les classes d'équivalences $[A]$ n'est pas suffisante.

Mais nous pouvons généraliser l'algèbre de Lindenbaum en faisant appel aux *contextes* plutôt qu'aux formules équiprouvables. L'idée est la suivante : dans une algèbre de Lindenbaum, nous avons $B \in [A]$, si, et seulement si, $B \vdash A$ et $A \vdash B$ sont prouvables ; au lieu de cela, on cherche à imposer la seule condition $B \vdash A$ prouvable et, même mieux, prouvable sans coupure. Ce qui définit A ne sont plus les "pseudo-A" mais les "prouveurs de A", qui sont l'ensemble des formules plus fortes que A logiquement. Cela fonctionne à deux conditions.

— Premièrement, généralisons aux contextes. Comme B est à gauche du séquent seulement, il n'y a plus de raison de demander à ce que B soit une *formule*. Par exemple, si $B \in [A]$, c'est-à-dire $B \vdash A$ est prouvable (sans coupure), on souhaite aussi que $C, C \Rightarrow B \in [A]$, etc.

— Deuxièmement, nous n'arriverons à obtenir $[\![A]\!] = [A]$ que lorsque A est atomique, mais cela n'est pas un problème. En analysant les preuves de complétude par les algèbres de Lindenbaum, on s'aperçoit que nous n'utilisons que deux propriétés sur $[\![A]\!]$, qui sont : $A \in [\![A]\!]$ et, si $\Gamma \in [\![A]\!]$, alors $\Gamma \vdash A$ est prouvable (sans coupure). La condition, plus faible que $[\![A]\!] = [A]$, que $[\![A]\!]$ doit respecter est donc $A \in [\![A]\!] \subseteq [A]$. Nous l'exprimerons d'une manière équivalente dans le lemme 3.2.8 d'adéquation ci-dessous.

Cette idée a plusieurs origines, à commencer par la logique linéaire et la sémantique des phases pour des preuves avec coupures [66] et la logique d'ordre supérieur extensionnelle [102] pour des preuves sans coupure et un théorème d'élimination des coupures, ce qui a donné lieu à des preuves d'élimination des coupures pour différentes logiques linéaires [112, 113], y compris d'ordre supérieur (toujours limitées à l'extensionalité). Ce sont ces définitions que nous allons étendre. L'idée est donc de partir de la définition suivante, où $\vdash^*$ dénote l'existence d'une preuve sans coupure ou, mieux, la relation de dérivabilité du calcul des séquents auquel on a *retranché* la règle de coupure.

Définition 3.2.2 (Extraction sans coupure)**.** *Soit A une formule close. Nous appelons* extraction *de A, et nous notons $\lceil A \rceil$, l'ensemble des contextes*

$$\lceil A \rceil := \{\Gamma \mid \Gamma \vdash^* A\}.$$

Il est parfois confortable de généraliser l'extraction aux séquents, en définissant

$$\lceil \Gamma \vdash \Delta \rceil = \{\Theta \mid \Theta, \Gamma \vdash \Delta\},$$

ou, encore plus généralement,

$$\lceil \Gamma \vdash \Delta \rceil = \{\Theta \vdash \Xi \mid \Theta, \Gamma \vdash \Delta, \Xi\},$$

et en se limitant, comme d'habitude, à au plus une formule à droite du séquent dans le cas de la logique intuitionniste. On pose ensuite la définition suivante.

Définition 3.2.3 (Algèbre universelle)**.** *Ω est la clôture de l'ensemble des extractions sans coupure par intersections (ensemblistes) arbitraires. Autrement dit, Ω est le plus petit ensemble tel que :*
- *$\lceil A \rceil \in \Omega$ pour toute formule A ;*
- *si $\omega_i \in \Omega$ pour tout $i \in I$, alors $\bigcap_{i \in I} \omega_i \in \Omega$.*

Les éléments de Ω peuvent toujours s'écrire sous la forme $\bigcap_{i \in I} \lceil A_i \rceil$ pour un certain ensemble de formules $\{A_i\}_{i \in I}$. Nous pouvons munir Ω de sa structure de treillis naturelle.

Lemme 3.2.4 (Treillis universel)**.** *$\langle \Omega, \leq, \cap, \cup \rangle$ est un treillis, avec les opérations suivantes :*
- *$\leq$ est l'inclusion ensembliste ;*
- *$a \cap b$ est l'intersection ensembliste de $a \in \Omega$ et $b \in \Omega$;*
- *$a \cup b$ est la borne inférieure des majorants : $\bigcap\{d \mid d \geq a \text{ et } d \geq b\}$.*

Ω est aussi un treillis complet*, car les bornes inférieures et supérieures existent pour tout ensemble arbitraire d'éléments de Ω. Enfin, le treillis Ω est borné.*

La démonstration de ce lemme est aisée. Il faut toutefois vérifier que, pour tout ensemble d'éléments de Ω, il existe au moins un majorant. Il s'agit de $\lceil\top\rceil$, qui contient tous les contextes et est donc l'élément maximum. De même, $\lceil\bot\rceil$ est l'élément minimum. Notons aussi que nous pouvons simplifier la définition de $a \cup b$ en $\bigcap\{\lceil D\rceil \mid \lceil D\rceil \geq a \text{ et } \lceil D\rceil \geq b\}$.

Nous pouvons transformer ce treillis complet en une algèbre de Boole complète, dans le cas de la logique classique, en définissant le complément booléen.

Définition 3.2.5 (Complément dans l'algèbre universelle (cas classique))**.** *Soit $\langle\Omega, \leq, \bot, \top, \wedge, \vee, \forall, \exists\rangle$ le treillis universel défini ci-dessus. On pose, pour tout $a \in \Omega$,*

$$\neg a := \exists\{b \mid a \wedge b \leq \bot\}$$

De même, nous obtenons une algèbre de Heyting dans le cas intuitionniste en définissant une opération de pseudo-complément.

Définition 3.2.6 (Pseudo-complément dans l'algèbre universelle (cas intuitionniste))**.** *Soit $\langle\Omega, \leq, \bot, \top, \wedge, \vee, \forall, \exists\rangle$ le treillis universel défini ci-dessus. On pose, pour tout $a, b \in \Omega$,*

$$a \Rightarrow b := \exists\{c \mid a \wedge c \leq b\}$$

La démonstration que les deux opérateurs définis ci-dessus sont bien les opérateurs d'algèbre attendus est assez délicate, car le calcul des séquents sous-jacent est sans coupure. Elle repose sur la validité de lemmes d'inversion à la Kleene [91]. Ceci tient bien entendu au fait que a et b peuvent être exprimés comme intersection d'extraction de formules. Des définitions plus simples peuvent ensuite être données, mais, dans ce cas, il faut encore démontrer que l'on a bien affaire à un élément de Ω :

$$a \Rightarrow b = \{\Gamma \mid \text{pour tout } \Delta \in a, (\Gamma, \Delta) \in b\}.$$

Pour toute formule A, nous définissons aussi le plus petit élément de Ω le contenant. Attention : ici A doit être entendu en tant que contexte.

Définition 3.2.7. *On pose $\lfloor A\rfloor := \bigcap\{a \in \Omega \mid A \in a\}$.*

Il est possible de montrer

$$\lfloor A\rfloor = \{\Gamma \mid \text{pour tout } B, \text{ si } A \vdash^* B, \text{ alors } \Gamma \vdash^* B\}.$$

Autrement dit, $\lfloor A\rfloor$ représente l'ensemble des *contextes coupables* avec A dans un calcul qui est, rappelons-le, sans coupure.

Munis de ces définitions, nous pouvons définir une interprétation, comme d'habitude par induction sur la taille des formules. Considérons pour le moment les seuls termes clos, ce qui est suffisant au premier ordre, car on peut toujours se réduire à celui-ci par l'identité $[\![A]\!]_\varphi = [\![A\varphi]\!]$, qui joue sur l'ambiguïté de φ, qui est à la fois une valuation dans le domaine (les termes clos) et une substitution. Le traitement des formules atomiques doit simplement respecter $\lfloor P(t_1, \cdots, t_n) \rfloor \leq [\![P(t_1, \cdots, t_n)]\!] \leq \lceil P(t_1, \cdots, t_n) \rceil$. Nous pouvons ensuite démontrer que cet invariant est respecté pour toutes les formules.

Lemme 3.2.8 (Adéquation)**.** *Pour toute formule (close) A,*

$$\lfloor A \rfloor \leq [\![A]\!] \leq \lceil A \rceil.$$

La preuve de ce lemme, par induction sur la formule A, repose exactement sur les règles du calcul des séquents ; la règle droite du connecteur/quantificateur $*$ est utilisée pour démontrer $[\![A * B]\!] \leq \lceil A * B \rceil$ et la règle gauche est utilisée pour démontrer $\lfloor A * B \rfloor \leq [\![A * B]\!]$. Ce résultat permet ensuite de démontrer de manière immédiate le théorème de complétude forte.

Théorème 3.2.9 (Complétude forte)**.** *Soient Γ, Δ deux contextes. Si, dans toutes les algèbres (de Boole, de Heyting), $\bigwedge [\![\Gamma]\!] \leq \bigvee [\![\Delta]\!]$, alors le séquent $\Gamma \vdash \Delta$ a une preuve sans coupure en calcul des séquents (classique, intuitionniste).*

Considérons un contexte Δ à un élément. Pour démontrer le théorème, on se sert de la chaîne $\Gamma \in \bigcap \lfloor \Gamma \rfloor \leq \bigwedge [\![\Gamma]\!] \leq [\![\Delta]\!] \leq \lceil \Delta \rceil$. Par définition de $\lceil \Delta \rceil$, vu que la relation d'ordre $\leq$ est l'inclusion, on obtient la preuve recherchée. La généralisation, en logique classique, au cas où Δ a plus d'une formule est immédiate, car $\bigvee \lceil \Delta \rceil \leq \lceil \bigvee \Delta \rceil$. Cela nous permet de conclure.

Corollaire 3.2.10 (Élimination des coupures)**.** *Soit $\Gamma \vdash \Delta$ un séquent prouvable en calcul des séquents (avec coupures). Alors $\Gamma \vdash \Delta$ est prouvable sans coupure.*

Ce corollaire se démontre par une simple combinaison du théorème 2.1.5 de correction par rapport à la sémantique algébrique (Boole, Heyting) et du théorème 3.2.9.

Nous pouvons généraliser les notions $\lfloor\ \rfloor$ et $\lceil\ \rceil$ et définir le pendant des semi-valuation classiques de la définition 2.2.1 du chapitre 2 dans les algèbres de Heyting.

Définition 3.2.11 (Semi-valuation de Heyting)**.** *Une* semi-valuation *dans une algèbre de Heyting Ω est une paire de fonctions $\lfloor\ \rfloor / \lceil\ \rceil$ définie sur les formules, à valeurs dans Ω et qui vérifie, pour toutes les formules A et B, la contrainte de séparation $\lfloor A \rfloor \leq \lceil A \rceil$ ainsi que toutes les contraintes suivantes* *sur les connecteurs et les quantificateurs.*

$$
\begin{array}{rcl@{\qquad}rcl}
\lfloor \bot \rfloor & = & \bot & \lceil \bot \rceil & = & \bot \\
\lfloor \top \rfloor & = & \top & \lceil \top \rceil & = & \top \\
\lfloor A \wedge B \rfloor & \leq & \lfloor A \rfloor \wedge \lfloor B \rfloor & \lceil A \rceil \wedge \lceil B \rceil & \leq & \lceil A \wedge B \rceil \\
\lfloor A \vee B \rfloor & \leq & \lfloor A \rfloor \vee \lfloor B \rfloor & \lceil A \rceil \vee \lceil B \rceil & \leq & \lceil A \vee B \rceil \\
\lfloor \forall x A \rfloor & \leq & \forall \{ \lfloor A[t/x] \rfloor \mid t \in \mathcal{T} \} & \forall \{ \lceil A[t/x] \rceil \mid t \in \mathcal{T} \} & \leq & \lceil \forall x A \rceil \\
\lfloor \exists x A \rfloor & \leq & \exists \{ \lfloor A[t/x] \rfloor \mid t \in \mathcal{T} \} & \exists \{ \lceil A[t/x] \rceil \mid t \in \mathcal{T} \} & \leq & \lceil \exists x A \rceil \\
\lfloor A \Rightarrow B \rfloor \wedge \lceil A \rceil & \leq & \lfloor B \rfloor & \lceil A \rceil \Rightarrow \lfloor B \rfloor & \leq & \lceil A \Rightarrow B \rceil
\end{array}
$$

Remarquons les deux dernières conditions portant sur l'implication. Les conditions de la définition 3.2.11 sont exactement celles qui permettent de démontrer chaque cas du lemme 3.2.8 ; toute la difficulté est donc de trouver une telle paire : nous en avons une avec les définitions précédentes (définitions 3.2.7 et 3.2.2), mais ce n'est pas le seul choix possible [99, 82]. Enfin, notons aussi que certaines des conditions de la définition 3.2.11 sont redondantes.

3.2.3 Méthodes algébriques en Déduction modulo théorie

Nous pouvons étendre les résultats d'élimination des coupures de la section 3.2.2 précédente à la Déduction modulo théorie pour un certain nombre de conditions sur les règles de réécriture. Malgré la modification de la relation de prouvabilité $\vdash$, qui prend maintenant en compte la congruence modulo $\mathcal{RE}$, l'algèbre universelle de contextes de la définition 3.2.3 est toujours une algèbre de Heyting. Le travail supplémentaire qui doit être fourni se faisant lors de la définition de l'interprétation $[\![A]\!]$, afin de s'assurer que nous obtenons bien une interprétation qui valide les règles de réécriture $\mathcal{RE}$.

Par exemple, dans le cas d'un système de réécriture respectant la condition d'ordre de la définition 2.4.3, nous ne définirons $[\![P(t_1, \cdots, t_n)]\!]$ que pour les formules minimales pour l'ordre $\prec$, l'extension aux autre formules se fait ensuite par induction sur $\prec$, comme à la section 2.4.2.

De manière similaire, en présence d'un système de réécriture $\mathcal{RE}$ qui respecte la condition de positivité de la définition 2.4.2, nous poserons :

$$
\begin{array}{ll}
[\![P(t_1, \cdots, t_n)]\!] = \lceil P(t_1, \cdots, t_n) \rceil & \text{pour } P(t_1, \cdots, t_n) \text{ positif ;} \\
[\![P(t_1, \cdots, t_n)]\!] = \lfloor P(t_1, \cdots, t_n) \rfloor & \text{pour } P(t_1, \cdots, t_n) \text{ négatif.}
\end{array}
$$

On démontre par induction les mêmes résultats qu'à la section 2.4.2, énonçant que toute formule positive A (respectivement, négative) sont telles que $[\![A]\!] = \lceil A \rceil$ (respectivement $[\![A]\!] = \lfloor A \rfloor$). En conséquence, les règles de réécriture sont valides.

De plus, un nombre plus important de systèmes respecte cette condition de positivité, par exemple dans le système

$$
\begin{array}{rcl}
Null(0) & \longrightarrow & \top \\
Null(s(y)) & \longrightarrow & \bot
\end{array}
$$

de la section 2.4.2, le prédicat $Null$ peut être défini indifféremment comme positif ou négatif, car pour le cas particulier de $\perp$ et $\top$ il est possible de démontrer $\lfloor \perp \rfloor = \lceil \perp \rceil = \perp$ et $\lfloor \top \rfloor = \lceil \top \rceil = \top$.

Théorème 3.2.12 (Élimination des coupures en calcul des séquents intuitionniste modulo théorie)**.** *Soit $\mathcal{RE}$ un système de réécriture qui respecte une des conditions de la section 2.4.2. Si le séquent $\Gamma \vdash B$ est prouvable en calcul des séquents intuitionniste modulo $\mathcal{RE}$ (avec coupures), alors il est prouvable sans coupure.*

De manière plus intéressante, à l'aide de ces algèbres, il devient possible de passer à l'ordre supérieur dans le cas intuitionniste (nous savions déjà comment faire dans le cas classique, voir la section 2.4.3). Ces résultats seront présentés dans les sections suivante, et dans le cas de la logique d'ordre supérieur exprimée sans règle de réécriture, donc, sans l'aide de la Déduction modulo théorie. Cela nous impose en particulier de présenter pour cette logique une nouvelle syntaxe pour les termes, les formules, qui seront des termes presque comme les autres, et les preuves.

La démonstration de l'élimination des coupures de l'expression de cette logique avec les règles de réécriture de la section 2.4.3, mais en calcul des séquents intuitionniste modulo, se déduit du travail que nous allons faire en combinant les résultats de cette même section 2.4.3 et de la section 3.2.4 ci-dessous.

3.2.4 Logique intuitionniste d'ordre supérieur

Une expression possible des règles d'inférence de la logique intuitionniste d'ordre supérieur est celle de la figure 3.3 ; puisque nous souhaitons travailler dans un cadre sans coupure, la règle de coupure n'y est pas présentée. Sans entrer dans les détails, au lieu d'utiliser les combinateurs, nous utilisons maintenant le λ-calcul [8] pour représenter les termes. Cela change peu par rapport à ce qui a été décrit à la section 2.4.3 ; le système de types simples est le même, mais il y a une absence dont nous reparlerons : le symbole de prédicat $\varepsilon(\;)$. En effet, dans l'expression "directe" de la logique d'ordre supérieur, nous n'avons pas de règle de réécriture et un séquent $A_1, \cdots A_n \vdash B$ manipule directement des formules $A_1 : o, \cdots, A_n : o, B : o$. Ainsi, il n'y a plus de distinction entre les connecteurs logiques et quantificateurs $(\wedge, \vee, \cdots)$ et leur dénotation dans le monde des termes $(\dot\wedge, \dot\vee, \cdots)$.

Plusieurs éléments sont à noter dans ces règles. Tout d'abord la variable x doit bien sûr être fraîche dans les deux règles marquées d'une astérisque, $\forall_R$ et $\exists_L$; de plus nous n'avons pas rappelé les règles structurelles d'affaiblissement et de contraction. La règle de contraction à gauche est essentielle, notamment pour les formules quantifiées universellement. De manière plus intéressante, les règles d'inférence de la logique d'ordre supérieur rappellent

$$\frac{}{\Gamma \vdash \top}\ \top_L \qquad \frac{}{\Gamma, A \vdash A}\ \mathrm{ax}$$

$$\frac{\Gamma, B, C \vdash A}{\Gamma, B \wedge C \vdash A}\ \wedge_L \qquad \frac{\Gamma \vdash B \qquad \Gamma \vdash C}{\Gamma \vdash B \wedge C}\ \wedge_R$$

$$\frac{\Gamma, B \vdash A \qquad \Gamma, C \vdash A}{\Gamma, B \vee C \vdash A}\ \vee_L \qquad \frac{\Gamma \vdash B_i}{\Gamma \vdash B_1 \vee B_2}\ \vee_R$$

$$\frac{\Gamma \vdash B \qquad \Gamma, C \vdash A}{\Gamma, B \Rightarrow C \vdash A}\ \Rightarrow_L \qquad \frac{\Gamma, B \vdash C}{\Gamma \vdash B \Rightarrow C}\ \Rightarrow_R$$

$$\frac{\Gamma, A[t/x] \vdash B}{\Gamma, \forall x.A \vdash B}\ \forall_L \qquad \frac{\Gamma \vdash A}{\Gamma \vdash \forall x.A}\ \forall_R^*$$

$$\frac{\Gamma, A \vdash B}{\Gamma, \exists x.A \vdash B}\ \exists_L^* \qquad \frac{\Gamma \vdash A[t/x]}{\Gamma \vdash \exists x.A}\ \exists_R$$

$$\frac{\Gamma' \vdash A'}{\Gamma \vdash A}\ \lambda, \quad \Gamma' \equiv_\lambda \Gamma, A' \equiv_\lambda A \qquad \frac{\Gamma \vdash \bot}{\Gamma \vdash B}\ \bot_R$$

FIGURE 3.3 – Règle d'inférence du calcul des séquents en logique d'ordre supérieur intuitionniste

furieusement celles du calcul des séquents intuitionniste au premier ordre de la figure 2.1.

— La différence majeure réside surtout dans la syntaxe des termes et le fait que nous autorisons la quantification sur les termes de tous les types, y compris des prédicats. Cette différence n'est pas apparente dans les règles de déduction elles-mêmes ; en revanche, elle est bien présente lorsque l'on définit les formules.

— Nous avons une règle supplémentaire de conversion,

$$\frac{\Gamma' \vdash A'}{\Gamma \vdash A}\ \lambda, \quad \Gamma' \equiv_\lambda \Gamma, A' \equiv_\lambda A.$$

Cette règle peut être déclenchée pour tous Γ, Γ' et A, A' convertibles deux à deux par $\alpha\beta\eta$-équivalence, dénoté par la relation $\equiv_\lambda$. Elle permet, entre autres, d'obtenir une forme normale lors de l'instanciation d'une formule. En effet, en réalité, ce que l'on écrit avec $\forall x.A$ est la formule $\forall_T(\lambda x.A)$ et l'instance de cette formule avec t peut être définie soit comme $A[t/x]$, ce qui est fait ici, soit comme $(\lambda x.A)\ t$. Dans les deux cas, il est nécessaire de pouvoir calculer, car A peut très bien être de la forme $x\ u$, et t, un prédicat complexe.

Pour nous préparer à la construction du modèle pour la complétude forte, commençons par établir un cahier des charges. Comme dans la section 2.4.3 nous voulons interpréter les termes dans le domaine des V-complexes, à la différence, mineure, que nous pouvons directement utiliser l'abstraction λ plutôt que de définir une fastidieuse opération sur les termes $\overline{\lambda}$.

Ce passage par les V-complexes est requis par la double contrainte de l'imprédicativité, qui demande à autoriser toutes les valeurs possible *a priori* (seconde composante des V-complexes), et de l'intentionnalité qui demande à ce que la dénotation de deux termes syntaxiquement différents soit deux valeurs sémantiques différentes (première composante des V-complexes).

En respectant ces deux dernières contraintes, et en visant le domaine des V-complexes pour les termes, nous perdons cependant quelque chose : les formules logiques (de type o) seront interprétées dans le domaine D_o, qui est une paire et n'est pas une algèbre de Heyting. En effet, toute interprétation dans une algèbre de Heyting doit vérifier l'identité $[\![\top]\!] = \top = \top \wedge \top = [\![\top \wedge \top]\!]$; plus généralement, d'après le théorème de correction, dès que A et B sont équiprouvables ($\vdash A \Rightarrow B$ et $\vdash B \Rightarrow A$ dérivables), on doit avoir $[\![A]\!] = [\![B]\!]$. Ceci n'est plus le cas dans D_o où, si nous procédons par analogie avec la section 2.4.3, $[\![\top]\!]$ devra être un V-complexe de la forme $\langle \top, v_1 \rangle$ et $[\![\top \wedge \top]\!]$ un V-complexe de la forme $\langle \top \wedge \top, v_2 \rangle$, qui seront donc différents selon leur première composante.

Cependant, les formules logiques doivent aussi avoir une *valeur de vérité* — dans notre exemple ci-dessus, il s'agit de v_1 et v_2, que l'on peut imposer égales. Il y a donc besoin de faire la distinction entre la *dénotation* ou interprétation de base, dans D_o, plus riche, et l'*interprétation* (sémantique) d'une formule logique dans une algèbre de Heyting.

Ceci nous amène à modifier la notion même de modèle en logique d'ordre supérieur, en introduisant un étage supplémentaire d'effacement qui, étant donnée une dénotation des termes de type o (dans les V-complexes ou dans toute autre structure), en extrait la valeur sémantique (l'interprétation propositionnelle).

Définition 3.2.13 (Structure applicative de Heyting)**.** *Une structure applicative de Heyting $\langle \mathcal{D}, \cdot, \hat{}, \omega, \Omega \rangle$ est composée de :*

- *un ensemble de domaines $\mathcal{D}_\alpha$, pour chaque type simple α ;*
- *une fonction $\cdot$, dite d'application et notée de manière infixe, de type $\mathcal{D}_{\alpha \to \beta} \times \mathcal{D}_\alpha \to \mathcal{D}_\beta$, pour chaque paire de types simples α, β ;*
- *une fonction d'interprétation des constantes $\hat{}$, de type $\alpha \to D_\alpha$;*
- *une algèbre de Heyting Ω ;*
- *une fonction d'effacement ω, de type $\mathcal{D}_o \to \Omega$.*

Cette structure doit vérifier les équations

$$\begin{array}{rcl}
\omega(\hat{\top}_o) & = & \top_\Omega, \\
\omega(\hat{\bot}_o) & = & \bot_\Omega, \\
\omega(\hat{\wedge} \cdot d_1 \cdot d_2) & = & \omega(d_1) \wedge \omega(d_2), \\
\omega(\hat{\vee} \cdot d_1 \cdot d_2) & = & \omega(d_1) \vee \omega(d_2), \\
\omega(\hat{\Rightarrow} \cdot d_1 \cdot d_2) & = & \omega(d_1) \Rightarrow \omega(d_2), \\
\omega(\hat{\exists}_T \cdot f) & = & \exists\{\omega(f \cdot d) : d \in \mathcal{D}_T\}, \\
\omega(\hat{\forall}_T \cdot f) & = & \forall\{\omega(f \cdot d) : d \in \mathcal{D}_T\}.
\end{array}$$

Dans toute structure applicative [104, 105] on peut définir une interprétation des termes du λ-calcul.

Le théorème de complétude du calcul des séquents sans coupure aura ainsi la forme suivante.

Théorème 3.2.14 (Complétude forte)**.** *Si pour toute structure applicative de Heyting et toute valuation φ, nous avons $\omega(\llbracket \Gamma \rrbracket_\varphi) \leq \omega(\llbracket A \rrbracket_\varphi)$, alors le séquent $\Gamma \vdash A$ est prouvable sans coupure.*

La façon de démontrer ce théorème est très largement identique à ce qui a été fait section 2.4.3 ; elle fait, en effet, appel à un domaine défini en étendant la définition 2.4.5 au cas des semi-valuation intuitionnistes.

Définition 3.2.15 (V-complexes intuitionnistes)**.** *Soient une algèbre de Heyting Ω et une semi-valuation de Heyting $\lfloor\ \rfloor / \lceil\ \rceil$. Nous définissons le domaine des V-complexes de type α, $\mathcal{D}_\alpha$, comme suit.*

- *$\mathcal{D}_\iota$ est l'ensemble des couples $\langle t, \iota \rangle$ tels que t est un terme de type ι en forme normale et ι est une constante qui n'a aucune signification particulière.*
- *$\mathcal{D}_o$ est l'ensemble des couples $\langle t, v \rangle$ tels que t est un terme de type o en forme normale, $v \in \Omega$ et $\lfloor t \rfloor \leq v \leq \lceil t \rceil$.*
- *$D_{\alpha \to \beta}$ est l'ensemble des couples $\langle t, f \rangle$ tels que t est un terme de type $\alpha \to \beta$ en forme normale et f est une fonction de $\mathcal{D}_\beta^{\mathcal{D}_\alpha}$ telle que, pour tout $\langle u, k \rangle \in \mathcal{D}_\alpha$, $f(\langle u, k \rangle) \in \mathcal{D}_\beta$ et $f(\langle u, k \rangle) = \langle \downarrow(t \cdot u), g \rangle$.*

Comme à la section 2.4.3, il est très important de définir ces domaines d'interprétation

- par une approche a priori qui considère toutes les valeurs de vérité potentielles d'un terme t_o, la marge de manœuvre étant l'espace entre $\lfloor t_o \rfloor$ et $\lceil t_o \rceil$,
- par induction sur les types, pour éviter la circularité d'une induction sur les termes — cette induction propage la marge de manœuvre aux types complexes —

— et par une approche semi-syntaxique qui intègre une partie syntaxique (la première composante) dans les éléments du domaine, ceci afin de traiter correctement la question de l'intentionnalité.

Cette définition n'est pas qu'un simple copier/coller de la définition 2.4.5, car la condition portant sur le domaine D_o est généralisée fortement. Notamment, la condition sur D_o revient à forcer l'*adéquation* du lemme 3.2.8 pour les V-complexes et ce, sans induction sur la formule. Pour $\langle t, v\rangle \in D_o$, v est appelé *candidat* pour t par [113].

Si l'on remplace "algèbre de Heyting" par "algèbre de Boole" dans la définition 3.2.11, nous obtenons une généralisation de la définition 2.2.1 où l'algèbre de Boole bien particulière $\{\bot, \top\}$ est remplacée par n'importe quelle algèbre. C'est cela qui nous permettra d'avoir des preuves encore plus constructives que celles que nous avions à la section 2.6.

Afin de conclure à l'élimination des coupures, nous devons cependant nous assurer que le théorème de correction s'étend bien à la classe de modèles décrite à la définition 3.2.13.

Théorème 3.2.16 (Correction)**.** *Si le séquent $\Gamma \vdash A$ est prouvable en logique intuitionniste d'ordre supérieur (avec coupures), alors, pour toute structure applicative de Heyting et toute valuation φ, nous avons $\omega([\![\Gamma]\!]_\varphi) \leq \omega([\![A]\!]_\varphi)$.*

Enfin, nous devons, préalablement à la définition 3.2.15, généraliser la notion de semi-valuation de Heyting de la définition 3.2.11 à la logique d'ordre supérieur, en définissant les semi-structures applicatives de Heyting

$$\langle \mathcal{D}, \cdot, \hat{}, \lfloor\ \rfloor, \lceil\ \rceil, \Omega\rangle,$$

où, cette fois-ci, les équations qui doivent être vérifiées correspondent à la transposition sémantique, dans l'esprit de la définition 3.2.13, des conditions de la définition 3.2.11. Nous laissons le soin au lecteur de faire ce travail instructif.

Corollaire 3.2.17 (Élimination des coupures)**.** *Si le séquent $\Gamma \vdash A$ est prouvable en logique intuitionniste d'ordre supérieur (avec coupures), alors il est prouvable sans coupure.*

Il est intéressant de remarquer que, dans le cas de l'expression de la logique d'ordre supérieur en Déduction modulo théorie et de la preuve de complétude de la section 2.4.3, nous n'avons pas eu besoin de définir une nouvelle classe de modèle à la façon de la définition 3.2.13, où les formules seraient interprétées dans un domaine, puis l'interprétation effacée dans une algèbre de Boole/Heyting. Cet avantage de la Déduction modulo théorie est dû à la barrière entre "terme de type o" et "formule" créée par le symbole de prédicat unaire $\varepsilon(\ )$. L'interprétation de $\varepsilon(\ )$ joue exactement le même rôle que la fonction d'effacement ω de la définition 3.2.13. En rendant la liaison terme–formule explicite, nous sommes autorisés à interpréter un terme de type o et la formule qu'il dénote de deux manières différentes, ce que nous sommes obligés de forcer dans le cas de cette section, car cette différence est maintenant *implicite*.

Cela ne signifie pas que les preuves soient plus simples (ou plus compliquées) dans le cas de la Déduction modulo théorie. Les difficultés sont déplacées, en particulier dans la nécessité de construire un modèle qui valide des règles de réécriture. Les énoncés intermédiaires, ainsi que les arguments dans leurs preuves, sont d'ailleurs très largement similaire, voire isomorphes. Nous pouvons, par exemple, remarquer la ressemblance entre les conditions (sémantiques) de la définition 3.2.13 et les règles de réécriture (syntaxiques) de la figure 2.6.

3.2.5 Logique linéaire d'ordre supérieur

Tous les arguments développés dans la section 3.2.4 s'appliquent aussi au cas de la logique d'ordre supérieur classique, que celle-ci soit exprimée de la manière directe, comme à la section 3.2.4 précédente, ou bien en Déduction modulo théorie [48, 52]. La structure de la démonstration, y compris la notion de V-complexe, est rigoureusement identique, à l'unique différence que l'algèbre de Heyting est remplacée par une algèbre de Boole dans la définition 3.2.13.

Nous pouvons alors nous demander s'il est possible de systématiser cette approche. Comme nous venons de le mentionner, cela est possible dans le cas de la logique classique ; historiquement cela a même été la première application [129, 115, 4]. Une telle généralisation peut aussi être faite dans le cadre de la logique linéaire [66, 112, 113], ce qui est l'objet de cette partie. La sémantique algébrique usuelle en logique linéaire est fondée sur les *espaces de phases*.

Définition 3.2.18 (Espace de phases (intuitionniste))**.** *Un* espace de phases intuitionniste $\langle M, \cdot, 1, cl(), J\rangle$ *est composé d'un monoïde commutatif* $\langle M, \cdot, 1\rangle$ *et d'un opérateur* $cl()$ *de* clôture*, opérant sur l'ensemble* $\wp(M)$ *de parties de* M*, qui vérifie, pour tous* $\alpha, \beta \in \wp(M)$ *:*

- $\alpha \subseteq cl(\alpha)$ *;*
- *si* $\alpha \subseteq \beta$*, alors* $cl(\alpha) \subseteq cl(\beta)$ *(monotonicité) ;*
- $cl(cl(\alpha)) \subseteq cl(\alpha)$*.*

De plus, $cl()$ *doit être tel que* $cl(\alpha) \cdot \beta \subseteq cl(\alpha \cdot \beta)$ *(stabilité). Enfin,* J *est un sous-monoïde de* M *qui vérifie,*

$$\textit{pour tout } a \in J, cl(\{a\}) \subseteq cl(\{a.a\}) \textit{(idempotence faible).}$$

Cette définition correspond aux règles d'inférence de la logique linéaire intuitionniste présentées à la figure 3.4 où, comme de coutume, la variable

x est supposée fraîche dans les règles $\forall_R$ et $\exists_L$. La logique linéaire intuitionniste ne considère pas tous les connecteurs et modalités de la logique linéaire classique ; $⅋$ et ? sont absents, car ils n'ont pas de sens dans un calcul des séquents monoconclusion. Quant à $\perp$, on peut décider de le supprimer de l'ensemble des formules, auquel cas la négation $A^\perp$ disparaît aussi, et tous les séquents linéaires intuitionnistes ont exactement une conclusion [132]. Dans notre cas, les séquents à membre droit vide $\Gamma \vdash$ doivent être compris, notamment en vue de leur interprétation dans les espaces de phases, comme des séquents $\Gamma \vdash \perp$. Enfin, contrairement à la logique linéaire classique, la négation $(\ )^\perp$ n'est *pas* un opérateur, mais un connecteur.

Les espaces de phases ne sont pas une généralisation directe des algèbres de Heyting ; cependant, de la même manière que la logique linéaire décompose les connecteurs usuels de la logique en opérateurs plus élémentaires (et mieux contrôlés), dans tout espace de phases, il est possible de trouver des sous-espaces qui sont des algèbres de Heyting.

Comme dans le cas des algèbres de Heyting et de Boole, pour les logiques intuitionniste et classique respectivement, on obtient la notion d'espace de phases classique, en restreignant l'opérateur de clôture à un usage particulier.

Définition 3.2.19 (Espace de phases (classique))**.** *Un* espace de phases classique $\langle M, \cdot, 1, \perp, J\rangle$ *est composé d'un monoïde commutatif* $\langle M, \cdot, 1\rangle$*, d'un ensemble* $\perp \subseteq M$*, nommé le* pôle*, et d'un sous-monoïde* J *de* M *tels que la structure* $\langle M, \cdot, 1, (\ )^{\perp\perp}, J\rangle$ *est un une espace de phases intuitionniste, où :*

- *l'opérateur* $(\ )^\perp$ *est défini comme* $\alpha^\perp := \{d \mid d.a \in \perp \text{ pour tout } a \in \alpha\}$ *;*
- *l'opérateur de clôture* $(\ )^{\perp\perp}$ *est* $((\ )^\perp)^\perp$*.*

Il est aisé de montrer que $()^{\perp\perp}$ est un opérateur de clôture. Les espaces de phases classiques sont donc un cas particulier des espaces de phases intuitionnistes ; pour cette raison, nous discuterons d'élimination des coupures uniquement dans le cas de la logique linéaire intuitionniste. Les résultats obtenus s'étendent ensuite directement à la logique linéaire classique. Définissons tout d'abord les opérateurs sémantiques.

Définition 3.2.20 (Opérateurs dans les espaces de phases)**.** *Soit un espace de phases* $\langle M, \cdot, 1, cl(), J\rangle$*. On définit les opérateurs suivants sur* $\Omega_M \subseteq \wp(M)$*, l'ensemble des parties closes de* M*, comme*

- $\top_M = M$,
- $\mathbf{0}_M = cl(\{\emptyset\})$,
- $\mathbf{1}_M = cl(\{1\})$,
- $\alpha \,\&\, \beta = \alpha \cap \beta$,
- $\alpha \oplus \beta = cl(\alpha \cup \beta)$,
- $\alpha \otimes \beta = cl(\alpha \cdot \beta)$,

identités

$$\frac{}{A \vdash A}\ \mathrm{ax} \qquad \frac{\Gamma \vdash A \qquad \Delta, A \vdash C}{\Gamma, \Delta \vdash C}\ \mathrm{cut}$$

connecteurs multiplicatifs

$$\frac{\Gamma \vdash A}{\Gamma, \mathbf{1} \vdash A}\ \mathbf{1}_L \qquad \frac{}{\vdash \mathbf{1}}\ \mathbf{1}_R$$

$$\frac{}{\bot \vdash}\ \bot_L \qquad \frac{\Gamma \vdash}{\Gamma \vdash \bot}\ \bot_R$$

$$\frac{\Gamma, A, B \vdash C}{\Gamma, A \otimes B \vdash C}\ \otimes_L \qquad \frac{\Gamma \vdash A \qquad \Delta \vdash B}{\Gamma, \Delta \vdash A \otimes B}\ \otimes_R$$

$$\frac{\Gamma \vdash A \qquad \Delta, B \vdash C}{\Gamma, \Delta, A \multimap B \vdash C}\ \multimap_L \qquad \frac{\Gamma, A \vdash B}{\Gamma \vdash A \multimap B}\ \multimap_R$$

connecteurs additifs

$$\frac{}{\Gamma \vdash \top}\ \top_R \qquad \frac{}{\mathbf{0}, \Gamma \vdash C}\ \mathbf{0}_L$$

$$\frac{\Gamma, A \vdash C}{\Gamma, A \,\&\, B \vdash C}\ \&_L \qquad \frac{\Gamma, B \vdash C}{\Gamma, A \,\&\, B \vdash C}\ \&_L \qquad \frac{\Gamma \vdash A \qquad \Gamma \vdash B}{\Gamma \vdash A \,\&\, B}\ \&_R$$

$$\frac{\Gamma, A \vdash C \qquad \Gamma, B \vdash C}{\Gamma, A \oplus B \vdash C}\ \oplus_L \qquad \frac{\Gamma \vdash A}{\Gamma \vdash A \oplus B}\ \oplus_R \qquad \frac{\Gamma \vdash B}{\Gamma \vdash A \oplus B}\ \oplus_R$$

négation

$$\frac{\Gamma \vdash A}{\Gamma, A^{\perp} \vdash}\ \mathrm{negation}_L \qquad \frac{\Gamma, A \vdash}{\Gamma \vdash A^{\perp}}\ \mathrm{negation}_R$$

modalités

$$\frac{\Gamma, A \vdash C}{\Gamma, !A \vdash C}\ !_L \qquad \frac{!\Gamma \vdash C}{!\Gamma \vdash !C}\ !_R$$

$$\frac{\Gamma \vdash C}{\Gamma, !A \vdash C}\ \mathrm{weak}_L \qquad \frac{\Gamma, !A, !A \vdash C}{\Gamma, !A \vdash C}\ \mathrm{contr}_L$$

quantificateurs

$$\frac{\Gamma, A[t/x] \vdash C}{\Gamma, \forall x A \vdash C}\ \forall_L \qquad \frac{\Gamma \vdash A}{\Gamma \vdash \forall x A}\ \forall_L^*$$

$$\frac{\Gamma, A \vdash C}{\Gamma, \exists x A \vdash C}\ \exists_L^* \qquad \frac{\Gamma \vdash A[t/x]}{\Gamma \vdash \exists x A}\ \exists_R$$

FIGURE 3.4 – Logique linéaire intuitionniste d'ordre supérieur

— $\alpha \multimap \beta = \{d \in M \mid \textit{pour tout } a \in \alpha, d \cdot a \in \beta\}$,
— $!\,\alpha = cl(\alpha \cap \mathbf{1}_M \cap J)$.

La définition 3.2.20 engendre de manière naturelle, et inductive, la définition d'interprétation. Notons que $\perp$ peut être choisi comme n'importe quel sous-ensemble de M, sans autre contrainte. Lorsque l'on passe au premier ordre (et à l'ordre supérieur) et que l'on veut interpréter les quantificateurs, on le fera additivement ; c'est à dire, étant donné un domaine D_T, et une fonction d'interprétation que l'on définit inductivement, on posera

$$\begin{array}{rcl} [\![\forall_T x A]\!]_\varphi & := & \bigcap_{d \in D_T} [\![A]\!]_{\varphi+(d/x)} \text{ et} \\ [\![\exists_T x A]\!]_\varphi & := & cl(\bigcup_{d \in D_T} [\![A]\!]_{\varphi+(d/x)}). \end{array}$$

Dans le but de démontrer le théorème d'élimination des coupures à l'ordre supérieur, nous avons, encore une fois, besoin d'un domaine d'interprétation pour les termes qui supporte l'intentionnalité, ce qui n'est pas le cas des espaces de phases tels qu'ils sont présentés dans la définition 3.2.18 : dans tout espace de phases, il est aisé de vérifier que $\alpha \otimes \mathbf{1} = \alpha$. Comme à la section 3.2.4, nous avons donc besoin d'étendre la notion de modèle pour avoir la possibilité de plonger le domaine de support des formules, D_o, dans les espaces de phases usuels sans que ces deux notions soient confondues. C'est principalement cette notion qu'il manque à [112, 113] pour pouvoir démontrer l'admissibilité des coupures en logique d'ordre supérieur *intentionnelle*. Ainsi, nous devons adapter la définition 3.2.13 aux espaces de phases.

Définition 3.2.21 (Structures applicatives de phases)**.** *Une* structure applicative de phases $\langle \mathcal{D}, \cdot, \hat{}, \omega, \rangle M, \cdot, 1, cl(), J\rangle$ *est composée de :*

— *un ensemble de domaines $\mathcal{D}_\alpha$, pour chaque type simple α ;*
— *une fonction $\cdot$, dite d'application et notée de manière infixe, de type $\mathcal{D}_{\alpha\to\beta} \times \mathcal{D}_\alpha \to \mathcal{D}_\beta$, pour chaque paire de types α, β ;*
— *une fonction d'interprétation des constantes $\hat{}$, de type $\alpha \to D_\alpha$;*
— *un espace de phases $\langle M, \cdot, 1, cl(), J\rangle$;*
— *une fonction d'effacement ω, de type $\mathcal{D}_o \to \Omega_M$, où Ω_M est l'ensemble des parties closes de M.*

Cette structure doit vérifier les équations

$$
\begin{array}{rcl}
\omega(\hat{\top}_o) & = & \top_M, \\
\omega(\hat{\mathbf{0}}_o) & = & \mathbf{0}_M, \\
\omega(\hat{\mathbf{1}}_o) & = & \mathbf{1}_M, \\
\omega(\hat{\&} \cdot d_1 \cdot d_2) & = & \omega(d_1) \,\&\, \omega(d_2), \\
\omega(\hat{\oplus} \cdot d_1 \cdot d_2) & = & \omega(d_1) \oplus \omega(d_2), \\
\omega(\hat{\otimes} \cdot d_1 \cdot d_2) & = & \omega(d_1) \otimes \omega(d_2), \\
\omega(\hat{\multimap} \cdot d_1 \cdot d_2) & = & \omega(d_1) \multimap \omega(d_2), \\
\omega(\hat{\perp}_{o\to o} \cdot d) & = & \omega(d) \multimap \omega(\hat{\perp}_o) \\
\omega(\hat{!} \cdot d_1) & = & !\,\omega(d_1), \\
\omega(\hat{\exists}_T \cdot f) & = & cl(\bigcup\{\omega(f \cdot d) : d \in \mathcal{D}_T\}), \\
\omega(\hat{\forall}_T \cdot f) & = & \bigcap\{\omega(f \cdot d) : d \in \mathcal{D}_T\}.
\end{array}
$$

Cette définition est une adaptation directe de la définition 3.2.13. Le seul point qui mérite un commentaire est la condition portant sur le *connecteur unaire* $\perp_{o\to o}$. Ce symbole (qui a pour type simple $o \to o$) est le symbole de la *négation* linéaire ; ainsi, $\perp_{o\to o}A$ a été jusqu'à présent noté $A^\perp$. $\perp_{o\to o}$ est donc différent de la constante $\perp_o$, de type o. Bien entendu, leurs interprétations sont liées (voir l'équation correspondante). Nous pouvons remarquer, par la même occasion, que $\omega(\hat{\perp}_o)$ ne doit obéir à aucune contrainte particulière. On vérifie ensuite que le théorème de correction est bien valide.

Théorème 3.2.22 (Correction)**.** *Si le séquent $\Gamma \vdash A$ est prouvable en logique linéaire intuitionniste d'ordre supérieur (avec coupures), alors, pour toute structure applicative de phases et pour toute valuation φ, nous avons $\omega([\![\Gamma]\!]_\varphi) \subseteq \omega([\![A]\!]_\varphi)$.*

Ce théorème est une généralisation du théorème de correction usuelle, dans lequel ω est l'identité et D_o, Ω_M. Nous pouvons ensuite relancer ensuite la machinerie des V-complexes.

Définition 3.2.23 (V-complexes dans les espaces de phases intuitionnistes)**.** *Soient $\langle M, \cdot, 1, cl(), J\rangle$ un espace de phases et une semi-valuation de phases $\lfloor\,\rfloor/\lceil\,\rceil$. Nous définissons les V-complexes de type α, $\mathcal{D}_\alpha$, comme suit.*

- *$\mathcal{D}_\iota$ est l'ensemble des couples $\langle t, \iota\rangle$ tels que t est un terme de type ι en forme normale et ι une constante qui n'a aucune signification particulière.*
- *$\mathcal{D}_o$ est l'ensemble des couples $\langle t, v\rangle$ tels que t est un terme de type o en forme normale, v est une partie close de M et $\lfloor t\rfloor \leq v \subseteq \lceil t\rceil$.*
- *$D_{\alpha\to\beta}$ est l'ensemble des couples $\langle t, f\rangle$ tels que t est un terme de type $\alpha \to \beta$ en forme normale et f est une fonction de $\mathcal{D}_\beta^{\mathcal{D}_\alpha}$ telle que, pour tout $\langle u, k\rangle \in \mathcal{D}_\alpha$, $f(\langle u, k\rangle) \in \mathcal{D}_\beta$ et $f(\langle u, k\rangle) = \langle \downarrow(t \cdot u), g\rangle$.*

Comme à la section 3.2.2, nous :

— définissions les extractions *sans coupure* de contextes à la manière e la définition 3.2.2 ;
— formons l'ensemble M des contextes (linéaires), muni de l'opération de concaténation (linéaire) de contextes, dont l'élément neutre 1 est le contexte vide ;
— et définissons les ensembles clos comme étant les intersections arbitraires d'extractions, à la manière de la définition 3.2.3.

Cela induit l'opérateur de clôture, sur les ensembles de contextes,

$$cl(\omega) := \bigcap\{\lceil A \rceil \mid \omega \subseteq A\},$$

et permet ensuite de définir un espace de phases universel de contextes $\langle M, \cdot, 1, cl(), J\rangle$, en posant $J := !M$. J est l'ensemble des contextes dont tous les éléments sont de la forme $!A$.

Comme nous sommes à l'ordre supérieur, nous devons aussi définir la notion de semi-valuation de phases et procéder à toutes les vérifications d'usage de la définition 2.4.3 sur les V-complexes, afin de démontrer le théorème de complétude. Ce travail est une adaptation simple de celui de la section 3.2.4.

Théorème 3.2.24 (Complétude forte)**.** *Si, pour toute structure applicative de phases et toute valuation φ, nous avons $\omega(\llbracket\Gamma\rrbracket_\varphi) \subseteq \omega(\llbracket A\rrbracket_\varphi)$, alors le séquent $\Gamma \vdash A$ est prouvable sans coupure en logique linéaire intuitionniste d'ordre supérieur.*

Cela nous permet de conclure directement, comme à la section 3.2.2, à l'élimination des coupures en logique linéaire intuitionniste d'ordre supérieur.

Corollaire 3.2.25 (Élimination des coupures)**.** *Si le séquent $\Gamma \vdash A$ est prouvable en logique linéaire intuitionniste d'ordre supérieur (avec coupures), alors il est prouvable sans coupure.*

3.3 Logique classique en logique intuitionniste

Nous explorons, dans cette partie, les liens entre la logique classique et la logique intuitionniste, en particulier dans le cadre de la Déduction modulo théorie et toujours en travaillant sans coupure. Plus précisément, nous définissons des traductions de la logique intuitionniste vers la logique classique par *double négation*, une technique introduite par les travaux de Kolmogorov [93], Gentzen [63] et Gödel [69]. C'est un outil qu'il est intéressant d'avoir à disposition lorsque l'on cherche à éliminer les coupures, car il permet d'obtenir cela en logique classique par le biais de l'élimination des coupures en logique intuitionniste.

3.3.1 Traduction par double négation polarisée

Commençons par considérer la traduction par double négation de la figure 3.5, qui est celle de Kolmogorov [93] allégée [52] où, au lieu d'introduire une double négation en tête, on introduit celle-ci en profondeur, sous les connecteurs.

$$
\begin{array}{rcl@{\qquad}rcl}
A^{Ko} & := & A, \textit{ si } A \textit{ atomique} & (\neg A)^{Ko} & := & \neg(\neg\neg A^{Ko}) \\
(A \wedge B)^{Ko} & := & (\neg\neg A^{Ko} \wedge \neg\neg B^{Ko}) & (\forall x A)^{Ko} & := & \forall x \neg\neg A^{Ko} \\
(A \vee B)^{Ko} & := & (\neg\neg A^{Ko} \vee \neg\neg B^{Ko}) & (\exists x A)^{Ko} & := & \exists x \neg\neg A^{Ko} \\
(A \Rightarrow B)^{Ko} & := & \neg\neg A^{Ko} \Rightarrow \neg\neg B^{Ko} & & &
\end{array}
$$

FIGURE 3.5 – Traduction de Kolmogorov légère par double négation

Le théorème que l'on cherche ensuite à démontrer est une variation sur le thème ci-dessous.

Théorème 3.3.1. *Soit $(\,)^{\neg}$ une traduction par double négation. Le séquent $\Gamma \vdash \Delta$ a une preuve en calcul des séquents classique si, et seulement si, le séquent $\Gamma^{\neg}, \neg\Delta^{\neg} \vdash \bot$ a une preuve en calcul des séquents intuitionniste.*

La méthode utilisée s'appuie sur le schéma suivant. Soit r une règle du calcul des séquents classique ayant pour formule principale A ; nous voulons la traduire en un ensemble équivalent de règles du calcul des séquents intuitionniste, agissant sur la formule $A^{\neg}$ (ou $\neg A^{\neg}$). De plus, nous désirons faire en sorte, dans la/les prémisses, de conserver la forme générale $\Gamma'^{\neg}, \neg\Delta'^{\neg} \vdash \bot$, qui est l'invariant global de la traduction et permet de procéder à la traduction par induction.

Pour simuler la règle r, il nous faut extraire le connecteur principal de $A^{\neg}$ (ou de $\neg A^{\neg}$), à l'aide des règles $\neg_L/\neg_R$ du calcul des séquents si besoin, appliquer la règle de connecteur r, puis reconstruire les traduction des sous-formules produites (c'est-à-dire $A_i^{\neg}$ ou $\neg A_i^{\neg}$), à l'aide des règles $\neg_R/\neg_L$ si besoin.

La traduction inverse est plus simple, et demande une légère généralisation, car l'induction peut faire apparaître des séquents de la forme $\Gamma^{\neg}, \neg\Delta^{\neg} \vdash D^{\neg}$.

La plupart des traductions par double négation suivent ce modèle, mais portent une plus grande attention aux connecteurs et aux quantificateurs, en tentant notamment de n'introduire des négations que là où elles sont nécessaires (le connecteur $\vee$ et le quantificateur existentiel $\exists$ pour la traduction de Gödel–Gentzen ; juste après le quantificateur universel $\forall$ pour la traduction de Kuroda). Un des objectifs poursuivi est de pouvoir traduire des classes de formules par elles-mêmes afin que, si une telle formule est prouvable, ou prouvée, à l'aide de la logique classique, alors elle le sera aussi en logique intuitionniste. Cependant, peu de traductions s'intéressent à une information sur les connecteurs que l'on peut remarquer en comparant le calcul des séquents intuitionniste et classique : les règles gauches sont absolument identiques dans les deux logiques (exceptées les règles $\Rightarrow_L$ et $\neg_L$), ce qui nous a par exemple permis de donner une présentation unifiée dans la figure 2.1.

Cette remarque suggère une traduction différenciée des formules à gauche du séquent classique $\Gamma \vdash \Delta$ de celles à droite, où on ne rajoute pas de négation pour traduire les connecteurs qui ont vocation à apparaître à gauche du séquent. Nous voici de retour dans la *polarité* gauche/droite des formules, que nous appelons positive/négative dans [22]. Une terminologie plus adaptée pourrait être *latéralité* gauche/droite. Rappelons qu'un changement de polarité/latéralité intervient lorsque l'on passe sous une négation ou à gauche d'une implication, que ces traductions sont adaptées au cas du calcul des séquents sans coupure et qu'il est aussi préférable de limiter la règle axiome aux formules atomiques.

Si l'on applique cette idée à la traduction de Kolmogorov de la figure 3.5, on obtient la traduction de la figure 3.6, où K^- dénote la traduction *gauche* (ou négative) des formules, et K^+ dénote la traduction *droite* (ou positive).Quant à la traduction de Gödel–Gentzen, il est aussi possible de la polariser de la manière de la figure 3.7.

$$
\begin{array}{rclrcl}
A^{K^-} & := & A,\ \textit{si}\ A\ \textit{atomique} & A^{K^+} & := & A,\ \textit{si}\ A\ \textit{atomique} \\
(A \wedge B)^{K^-} & := & A^{K^-} \wedge B^{K^-} & (A \wedge B)^{K^+} & := & \neg\neg A^{K^+} \wedge \neg\neg B^{K^+} \\
(A \vee B)^{K^-} & := & A^{K^-} \vee B^{K^-} & (A \vee B)^{K^+} & := & \neg\neg A^{K^+} \vee \neg\neg B^{K^+} \\
(A \Rightarrow B)^{K^-} & := & \neg\neg A^{K^+} \Rightarrow B^{K^-} & (A \Rightarrow B)^{K^+} & := & A^{K^-} \Rightarrow \neg\neg B^{K^+} \\
(\neg A)^{K^-} & := & \neg A^{K^+} & (\neg A)^{K^+} & := & \neg A^{K^-} \\
(\forall x A)^{K^-} & := & \forall x A^{K^-} & (\forall x A)^{K^+} & := & \forall x \neg\neg A^{K^+} \\
(\exists x A)^{K^-} & := & \exists x A^{K^-} & (\exists x A)^{K^+} & := & \exists x \neg\neg A^{K^+}
\end{array}
$$

FIGURE 3.6 – Traduction de Kolmogorov polarisée

Ces nouvelles traductions introduisent moins de négations ; il n'est donc plus possible d'être aussi agressif que dans le schéma de démonstration qui suit le théorème 3.3.1 et qui applique brutalement "$\neg_L$-r-$\neg_R$" pour traduire une règle r. La solution à cette difficulté vient du contrôle de la preuve (en logique classique) du séquent $\Gamma \vdash \Delta$ par le *focusing* [3, 97] : lorsque nous travaillons sur une formule (nous lui appliquons une règle r), nous souhai-

$$
\begin{array}{rcl@{\qquad}rcl}
A^n & := & A,\ \textit{si } A \textit{ atomique} & A^p & := & \neg\neg A,\ \textit{si } A \textit{ atomique} \\
(A \wedge B)^n & := & A^n \wedge B^n & (A \wedge B)^p & := & A^p \wedge B^p \\
(A \vee B)^n & := & A^n \vee B^n & (A \vee B)^p & := & \neg(\neg A^p \wedge \neg B^p) \\
(A \Rightarrow B)^n & := & A^p \Rightarrow B^n & (A \Rightarrow B)^p & := & A^n \Rightarrow B^p \\
(\neg A)^n & := & \neg A^p & (\neg A)^p & := & \neg A^n \\
(\forall x A)^n & := & \forall x A^n & (\forall x A)^p & := & \forall x A^p \\
(\exists x A)^n & := & \exists x A^n & (\exists x A)^p & := & \neg \forall x \neg A^p
\end{array}
$$

FIGURE 3.7 – Traduction de Gödel–Gentzen polarisée

tons continuer de travailler sur celle-ci, plus exactement ses sous-formules, le plus longtemps possible.

Ainsi, au lieu de traduire une règle r sur une formule, nous traduisons la *chaîne de règles* $r_1, \cdots, r_n$, qui satisfait la relation suivante : la formule principale de r_{i+1} est la formule produite par r_i, et la règle qui suit r_n n'a pas cette propriété. La notion de chaîne est une approximation, puisque les preuves ont une structure arborescente, et ce qui intéresse la discipline du focusing est de fixer des règles de dérivation qui permettent de faire en sorte que cette chaîne soit maximale et qu'elle commence et s'arrête à des endroits précis, déterminés à l'avance, d'une formule.

La restriction syntaxique associée au focusing est la distinction d'une position particulière dans le séquent classique, dite *bénitier* [72], celle de la formule B dans le séquent $\Gamma \vdash B; \Delta$. Nous n'autorisons les règles de dérivation à décomposer que la seule formule du bénitier ; aucune autre règle ne peut s'appliquer tant que celui-ci n'est pas vide.

Si nous avons un focusing suffisamment précis, nous pouvons de nouveau appliquer le schéma de preuve "$\neg_L$-$r_1, \cdots, r_n$-$\neg_R$", la négation initiale faisant office de "placement dans le bénitier" (étape appelée *focus* ou *decide* [97]), et la négation finale de retrait de cet emplacement (étape appelée *release*). Ne pas appliquer de négation à l'intérieur de la chaîne $r_1, \cdots, r_n$ revient à considérer des preuves qui s'interdisent de "changer le focus", ce qui est une contrainte que doivent maintenant vérifier les preuves du calcul des séquents classique. Notons aussi, dans cette approche, l'analogie entre le bénitier du focusing en calcul des séquents classique et la position à droite du séquent en calcul des séquents intuitionniste.

Le focusing, en tant que regroupement de règles portant sur une formule et ses descendants, peut être obtenu en adoptant l'approche d'inversibilité des règles de preuve en calcul des séquents classiques [91]. Cela permet justement de regrouper, soit vers le haut (pour les règles synchrones/non inversibles), soit vers le bas (pour les règles asynchrones/inversibles), toutes les règles portant sur une formule donnée et ses sous-formules. Pour la polarisation de la traduction de Gödel–Gentzen, un focusing très lâche est requis et le calcul des séquents focusé correspondant est présenté à la figure 3.8. On a bien sûr équivalence entre prouvabilité dans le calcul des séquents usuel et dans le calcul des séquents focusé. D'autres traductions, encore plus optimisées, demandent plus de travail [64, 29].

Un dernier outil, qui permet de simplifier les démonstrations et d'exprimer de manière concise les théorèmes, est l'opérateur d'*antinégation*.

Définition 3.3.2 (Antinégation)**.** *Soit A une formule, ; $\lrcorner A$ est la formule*
- *B, si $A = \neg B$;*
- *$\neg A$ sinon.*

Par exemple, l'énoncé à démontrer dans le cas de la polarisation de la traduction de Gödel–Gentzen est, avec l'usage de l'antinégation et du focusing, celui du théorème suivant, qui prend bien en compte toutes les configurations possibles.

Théorème 3.3.3. *Si le séquent $\Gamma \vdash B; \Delta$ a une preuve en calcul des séquents classique, alors le séquent $\Gamma^n, \lrcorner\Delta^p \vdash B^p$ a une preuve en calcul des séquents intuitionniste.*

Ces traductions par double négation polarisée ne peuvent pas s'appliquer pas à la Déduction modulo théorie, car, lors de la traduction, il faut traduire toutes les formules, y compris celles des règles de réécriture. Or, une règle de réécriture peut s'appliquer à la fois à gauche et à droite d'un séquent ; il faudrait donc traduire les règles à la fois par la traduction négative/gauche et par la traduction positive/droite. En revanche, la polarisation des traductions est parfaitement adaptée à la Déduction modulo théorie polarisée [45], où l'idée est de traduire toute règle positive $A \longrightarrow^+ B$ par la règle positive $A \longrightarrow^+ B^p$ (si l'on parle de traduction de Gödel–Gentzen). Une telle extension est un exercice facile.

3.3.2 Traduction par double négation sémantique

La supercohérence [45] est une propriété sémantique des systèmes de réécriture extrêmement forte qui permet de garantir la propriété d'élimination des coupures et de normalisation forte des systèmes de preuve en Déduction modulo théorie. Son énoncé est le suivant.

Définition 3.3.4 (Supercohérence)**.** *Un système de réécriture est dit* supercohérent *lorsque, dans toute préalgèbre de Heyting (ordonnée et complète), il est possible de construire un modèle de ce système de réécriture.*

$$\frac{}{\Gamma, A \vdash .; A, \Delta}\text{ax}$$

$$\frac{\Gamma, A, B \vdash .; \Delta}{\Gamma, A \wedge B \vdash .; \Delta}\wedge_L \qquad \frac{\Gamma \vdash A; \Delta \qquad \Gamma \vdash B; \Delta}{\Gamma \vdash A \wedge B; \Delta}\wedge_R$$

$$\frac{\Gamma, A \vdash .; \Delta \qquad \Gamma, B \vdash .; \Delta}{\Gamma, A \vee B \vdash .; \Delta}\vee_L \qquad \frac{\Gamma \vdash .; A, B, \Delta}{\Gamma \vdash .; A \vee B, \Delta}\vee_R$$

$$\frac{\Gamma \vdash A; \Delta \qquad \Gamma, B \vdash .; \Delta}{\Gamma, A \Rightarrow B \vdash .; \Delta}\Rightarrow_L \qquad \frac{\Gamma, A \vdash B; \Delta}{\Gamma \vdash A \Rightarrow B; \Delta}\Rightarrow_R$$

$$\frac{\Gamma \vdash A; \Delta}{\Gamma, \neg A \vdash .; \Delta}\neg_L \qquad \frac{\Gamma, A \vdash .; \Delta}{\Gamma \vdash .; \neg A, \Delta}\neg_R$$

$$\frac{\Gamma, A[c/x] \vdash .; \Delta}{\Gamma, \exists x A \vdash .; \Delta}\exists_L \qquad \frac{\Gamma \vdash .; A[t/x], \Delta}{\Gamma \vdash .; \exists x A, \Delta}\exists_R$$

$$\frac{\Gamma, A[t/x] \vdash .; \Delta}{\Gamma, \forall x A \vdash .; \Delta}\forall_L \qquad \frac{\Gamma \vdash A[c/x]; \Delta}{\Gamma \vdash \forall x A; \Delta}\forall_R$$

$$\frac{\Gamma, A, A \vdash .; \Delta}{\Gamma, A \vdash .; \Delta}\text{contr}_L \qquad \frac{\Gamma \vdash .; A, \Delta}{\Gamma \vdash .; A, A, \Delta}\text{contr}_R$$

$$\frac{\Gamma \vdash .; \Delta}{\Gamma, A \vdash .; \Delta}\text{weak}_L \qquad \frac{\Gamma \vdash .; \Delta}{\Gamma \vdash A; \Delta}\text{weak}_R$$

$$\frac{\Gamma \vdash A; \Delta}{\Gamma \vdash .; A, \Delta}\text{focus} \qquad \frac{\Gamma \vdash .; A, \Delta}{\Gamma \vdash A; \Delta}\text{release}$$

avec :
- A atomique dans la règle axiome ;
- A atomique, ou de la forme $\exists x B$, $B \vee C$, $\neg B$ dans la règle release ;
- A pas de la forme précédente dans la règle focus ;
- les restrictions de fraîcheur pour les règles $\forall_L$ et $\exists_R$.

FIGURE 3.8 – Un calcul des séquent classique sans coupure avec focus

Par rapport à la définition 2.1.11 du chapitre précédent, qui impliquait l'existence d'*un* modèle, nous imposons ici l'existence d'un modèle pour *toute* structure.

En des termes plus explicites, une préalgèbre de Heyting a exactement la même définition qu'une algèbre de Heyting (définition 2.1.2), à l'unique différence que l'ordre $\leq$ est affaibli en un *préordre*, ce qui autorise par exemple l'existence des deux éléments $\top \wedge \top$ et $\top$, sans qu'ils soient nécessairement confondus, tous deux étant par ailleurs supérieur à $\top$. Une préalgèbre (ou une algèbre) est complète, si les bornes supérieures et inférieures arbitraires existent, tandis que la condition d'ordre stipule l'existence d'un ordre $\sqsubseteq$, parallèle au préordre $\leq$, qui doit vérifier certaines conditions de compatibilité avec *leq* [45].

La notion de modèle du système de réécriture impose, quant à elle, que, pour toutes formules $A \equiv B$ et toute valuation φ, $[\![A]\!]_\varphi = [\![B]\!]_\varphi$. Notons que, par le théorème de correction (qui se démontre pour les préalgèbres de Heyting aussi), si l'on sait prouver $\vdash A \Leftrightarrow B$, alors on a seulement $[\![A]\!]_\varphi \leq [\![B]\!]_\varphi$ et $[\![B]\!]_\varphi \leq [\![A]\!]_\varphi$, ce qui n'implique *plus* $[\![A]\!]_\varphi = [\![B]\!]_\varphi$. La condition d'être un modèle du système de réécriture est donc strictement plus forte que la condition d'être une sémantique correcte. Nous pourrions réunifier les deux notions en réduisant les préalgèbres à des algèbres de Heyting, avec un quotient par $\cong$, la relation d'équivalence associée au préordre $\leq$, mais cela nous priverait de la richesse de certaines préalgèbres, telle que celle des candidats de réductibilité, dont nous ferons usage au chapitre suivant.

Un système de réécriture $\mathcal{RE}$ supercohérent possède donc la propriété de normalisation forte pour la déduction naturelle modulo théorie [45]. Nous dérivons ensuite le résultat d'élimination des coupures en calcul des séquents intuitionniste modulo théorie par traduction vers et depuis la déduction naturelle modulo théorie [52]. Nous retrouvons nos traductions par double négation (non polarisées, comme il a été expliqué) lors du passage à l'élimination des coupures en calcul des séquents classique. Soient $\mathcal{RE}$ un système de réécriture et une preuve de $\Gamma \vdash \Delta$ en calcul des séquents classique modulo théorie.

- On traduit cette preuve en une preuve de $\Gamma^\neg, \neg\Delta^\neg \vdash$ en calcul des séquents intuitionniste modulo $\mathcal{R}^\neg\mathcal{E}$; on a donc traduit le système de réécriture.
- Par supercohérence de $\mathcal{R}^\neg\mathcal{E}$ et les résultats d'élimination des coupures afférents, on obtient une preuve sans coupure, que l'on retraduit en preuve, sans coupure, de $\Gamma \vdash \Delta$ en calcul des séquents classique modulo $\mathcal{RE}$.

Il est embêtant de devoir s'intéresser à la supercohérence de $\mathcal{R}^\neg\mathcal{E}$ plutôt qu'à celle de $\mathcal{RE}$ directement ; c'est une hypothèse supplémentaire dont nous aimerions nous débarrasser. C'est ce vide que nous allons combler, en démontrant que, si $\mathcal{RE}$ est supercohérent, alors $\mathcal{R}^{\neg}\mathcal{E}$ l'est aussi. La méthode adoptée pour ce faire, puisque la supercohérence est une propriété sémantique, est d'introduire une traduction par *double négation sémantique* des préalgèbres de Heyting.

Théorème 3.3.5. *Si le système de réécriture $\mathcal{RE}$ est supercohérent, alors le système de réécriture $\mathcal{R}^{\neg}\mathcal{E}$ est supercohérent.*

Le choix de la traduction de $\mathcal{RE}$ par double négation (non polarisée) est libre ; nous prendrons ici l'exemple de la traduction de Kolmogorov de la figure 3.5 ; nous aurons donc $A^{\neg} = A^{Ko}$. La démonstration de ce théorème fait appel aux étapes suivantes, étant donné un système de réécriture $\mathcal{RE}$ supercohérent.

Soit $\mathcal{H}$ n'importe quelle préalgèbre de Heyting complète et ordonnée. Nous définissons la préalgèbre $\mathcal{H}^{\neg}$, traduite de $\mathcal{H}$ par double négation sémantique, dans laquelle tous les opérateurs sont définis sur le modèle de la figure 3.5. Nous utilisons ensuite l'hypothèse de supercohérence de $\mathcal{RE}$ pour trouver une interprétation dans $\mathcal{H}^{\neg}$, appelée $[\![\ ]\!]^{\neg}$, qui est donc un modèle de la réécriture de $\mathcal{RE}$. On a donc, pour toutes formules $A \equiv_{\mathcal{RE}} B$ et toute valuation φ,

$$[\![A]\!]^{\neg}_{\varphi} = [\![B]\!]^{\neg}_{\varphi}.$$

D'autre part, par construction de $\mathcal{H}^{\neg}$, nous pouvons démontrer, pour toute formule A et tout terme t, que

$$[\![t]\!]^{\neg}_{\varphi} = [\![t]\!]_{\varphi} \quad \text{et} \quad [\![A]\!]^{\neg}_{\varphi} = [\![A^{\neg}]\!]_{\varphi},$$

où $[\![\ ]\!]$ est l'interprétation dans $\mathcal{H}$ qui a été générée par $[\![\ ]\!]^{\neg}$. Une telle définition est possible, car, comme nous le verrons dans quelques lignes, l'ensemble qui sert de support à $\mathcal{H}^{\neg}$, ainsi que le domaine d'interprétation des termes, sont rigoureusement les mêmes que ceux de $\mathcal{H}$. On dérive de l'équation ci-dessus la validité de toute règle de réécriture de $\mathcal{R}^{\neg}\mathcal{E}$. Le cas des règles de réécriture sur les termes et des axiomes équationnels étant immédiat, considérons $P \longrightarrow A \in \mathcal{R}$; nous aurons $P \longrightarrow A^{\neg} \in \mathcal{R}^{\neg}$ et nous cherchons à démontrer

$$[\![P]\!]_{\varphi} = [\![A^{\neg}]\!]_{\varphi},$$

ce qui est une conséquence directe des égalités $P^{\neg} = P$ et $[\![P^{\neg}]\!]_{\varphi} = [\![P]\!]^{\neg}_{\varphi}$, de l'hypothèse $[\![P]\!]^{\neg}_{\varphi} = [\![A]\!]^{\neg}_{\varphi}$ (supercohérence de $\mathcal{RE}$ dans $\mathcal{H}^{\neg}$) et de l'égalité $[\![A]\!]^{\neg}_{\varphi} = [\![A^{\neg}]\!]_{\varphi}$.

Ainsi, nous avons trouvé une interprétation dans $\mathcal{H}$ qui est un modèle de $\mathcal{RE}$ et ce, pour toute préalgèbre $\mathcal{H}$. Le système de réécriture $\mathcal{RE}$ est donc supercohérent et le théorème 3.3.5 est démontré, à condition de savoir construire une algèbre $\mathcal{H}^{\neg}$ qui possède les propriétés requises. Cette construction est esquissée dans le paragraphe suivant.

La traduction par double négation sémantique considère une préalgèbre de Heyting $\langle \Omega, \leq, \top, \bot, \wedge, \vee, \Rightarrow, \forall, \exists \rangle$ complète et ordonnée par $\sqsubseteq$, et la transforme en la structure $\langle \Omega, \overset{\neg}{\leq}, \overset{\neg}{\top}, \overset{\neg}{\bot}, \overset{\neg}{\wedge}, \overset{\neg}{\vee}, \overset{\neg}{\Rightarrow}, \overset{\neg}{\forall}, \overset{\neg}{\exists} \rangle$ telle que

— $b \overset{\neg}{\leq} c$ si, et seulement si, $\neg\neg b \leq \neg\neg c$,
— $\overset{\neg}{\top} = \top$,
— $\overset{\neg}{\bot} = \bot$,
— $b \overset{\neg}{\Rightarrow} c = (\neg\neg b) \Rightarrow (\neg\neg c)$,
— $b \overset{\neg}{\wedge} c = (\neg\neg b) \wedge (\neg\neg c)$,
— $b \overset{\neg}{\vee} c = (\neg\neg b) \vee (\neg\neg c)$,
— $\overset{\neg}{\forall} A = \forall\, (\neg\neg A)$,
— $\overset{\neg}{\exists} A = \exists\, (\neg\neg A)$,

où $\neg a$ est défini comme $a \Rightarrow \bot$ et $\neg A := \{\neg a \mid a \in A\}$. Nous remaquons bien entendu la similarité annoncée avec la traduction syntaxique de Kolmogorov de la figure 3.5, ce qui permet ensuite de tisser les liens nécessaires, notamment la correspondance $[\![A]\!]^{\neg} = [\![A^{\neg}]\!]$. $\mathcal{H}^{\neg}$ est bien une préalgèbre de Heyting complète et l'ordre $\sqsubseteq$ de $\mathcal{H}$ peut être directement hérité par $\mathcal{H}^{\neg}$.

Proposition 3.3.6. *Si $\mathcal{H}$ est une préalgèbre de Heyting complète et ordonnée par $\sqsubseteq$, alors $\mathcal{H}^{\neg}$ l'est aussi.*

Il est possible de généraliser cette approche de traduction par double négation en une approche par A-traduction. Cette idée, due à Friedman [61], peut être expliquée de la manière suivante : $\neg\neg B$ est, en logique intuitionniste, rigoureusement équivalent à $(B \Rightarrow \bot) \Rightarrow \bot$. Au lieu d'utiliser $\bot$ dans la traduction, il est tout à fait possible de le remplacer par n'importe quelle formule close A pour obtenir $(B \Rightarrow A) \Rightarrow A$. En systématisant cette approche, nous obtenons la A-traduction de la figure 3.9.

$$
\begin{array}{rcl}
B^A & := & B \quad \textit{si } A \textit{ atomique} \\
(\neg B)^A & := & \neg((B^A \Rightarrow A) \Rightarrow A) \\
(B \wedge C)^A & := & ((B^A \Rightarrow A) \Rightarrow A) \wedge ((C^A \Rightarrow A) \Rightarrow A) \\
(B \vee C)^A & := & ((B^A \Rightarrow A) \Rightarrow A) \vee ((C^A \Rightarrow A) \Rightarrow A) \\
(B \Rightarrow C)^A & := & ((B^A \Rightarrow A) \Rightarrow A) \Rightarrow ((C^A \Rightarrow A) \Rightarrow A) \\
(\forall x B)^A & := & \forall x((B^A \Rightarrow A) \Rightarrow A) \\
(\exists x B)^A & := & \exists x((B^A \Rightarrow A) \Rightarrow A)
\end{array}
$$

FIGURE 3.9 – A-traduction de Friedman légère

L'hypothèse que A est une formule close permet de passer sans problème la barrière des quantificateurs. Il est, par ailleurs, possible de faire le même travail avec les autres traductions par double négation, dont celles de la section 3.3.1, polarisées ou non, auquel cas nous obtiendrons des A-traductions plus légères, qui satisferont toujours une version étendue de théorème 3.3.1.

La A-traduction peut se refléter dans une a-traduction sémantique. De même que la A-traduction syntaxique généralise la traduction par double négation syntaxique, la a-traduction sémantique généralise la traduction sémantique par double négation vue plus haut.

Les démonstrations de conservativité de l'interprétation, de type $[\![B^A]\!] = [\![B]\!]^a$ où $[\![\ ]\!]^a$ est l'interprétation dans la structure $\mathcal{H}^a$, l'algèbre a-traduite de $\mathcal{H}$, demandent de la précision supplémentaire : il faut pouvoir poser $a = [\![A]\!]$, mais, pour cela, il faut que $[\![A]\!]$ soit déjà définie, et donc déjà disposer d'une interprétation $[\![\ ]\!]_0$, qui n'a pas loisir de dépendre de la supercohérence et qui doit donc être fournie a priori. Ce problème n'existait pas dans le cas de la traduction par double négation, puisque nous savons que toute interprétation, quelle qu'elle soit, doit vérifier $[\![\bot]\!] = \bot$.

Il faut ensuite supposer que A, ses sous-termes et sous-formules n'apparaissent pas dans le système de réécriture $\mathcal{RE}$ pour pouvoir revenir dans l'algèbre $\mathcal{H}$ tout en conservant l'interprétation fournie par la supercohérence, la seule (contrairement à $[\![\ ]\!]_0$) à valider les règles de réécriture.

Chapitre 4

Normalisation

Après avoir rappelé le principe des démonstrations de normalisation des termes de preuve en déduction naturelle modulo théorie, nous en dérivons des preuves simplifiées d'élimination des coupures, puis nous étudions le fonctionnement des algorithmes sous-jacents à toutes les preuves d'admissibilité des coupures qui ont été développées dans ce manuscrit.

La section 4.1 contient une présentation rapide des résultats qui nous seront utiles dans les deux parties suivantes.

La section 4.2 discute de nombreuses contributions. La section 4.2.1, qui discute de normalisation faible en introduisant notamment les candidats de réductibilité faible, s'appuie sur [37]. La section 4.2.2, qui simplifie la section précédente au cas des algèbres de séquents en Déduction naturelle, est tirée de [50, 51], ainsi que la section 4.2.3 suivante. L'adaptation de ces derniers résultats au calcul des séquents de la section 4.2.4 a été l'objet de l'article [24].

Toute la section 4.2 s'applique à la Déduction modulo théorie, mais l'impact des règles de réécriture y est minime grâce à l'hypothèse de supercohérence, qui offre une coucher d'abstraction du traitement de la réécriture et de la construction effective de modèle, que nous ne verrons explicitement qu'à la section 4.2.3. La section 4.2 peut donc aussi être lue comme s'appliquant directement à la logique du premier ordre.

La formalisation et l'extraction d'algorithme présentées à la section 4.3.1 a été effectuée dans l'article [65] et la discussion de la section 4.3.2 est tirée de [37].

4.1 Introduction : réduction et termes de preuve

Dans les deux chapitres précédents, nous avons démontré l'admissibilité de la règle de coupure, en passant par la correction et la complétude forte. Mais il existe une autre manière d'éliminer les coupures, plus procédurale :

expliquer comment *réduire* n'importe quelle coupure, pas à pas. Par exemple, l'arbre de preuve

$$\dfrac{\wedge_R \dfrac{\dfrac{\pi_A}{\Gamma \vdash A} \quad \dfrac{\pi_B}{\Gamma \vdash B}}{\Gamma \vdash A \wedge B} \qquad \dfrac{\dfrac{\pi}{\Gamma, A, B \vdash C}}{\Gamma, A \wedge B \vdash C} \wedge_L}{\Gamma \vdash C}\,\text{cut}$$

comporte une coupure sur la formule $A \wedge B$, et il est possible de la réduire en l'arbre de preuve

$$\dfrac{\dfrac{\pi_A}{\Gamma \vdash A} \qquad \dfrac{\dfrac{\pi_B}{\Gamma, A \vdash B} \quad \dfrac{\pi}{\Gamma, A, B \vdash C}}{\Gamma, A \vdash C}\,\text{cut}}{\Gamma \vdash C}\,\text{cut}$$

qui comporte deux coupures, structurellement plus petites, une sur A et une sur B, qu'il faut à leur tour éliminer. Les questions sont alors les suivantes.

1. Lorsque la procédure termine, toutes les coupures sont-elles éliminées ?
2. La procédure termine-t-elle dans tous les cas ?

C'est, par exemple, la manière dont Gentzen [62, 127] a démontré l'élimination des coupures en calcul des séquents, un résultat plus fort que la simple admissibilité. La question de la terminaison est particulièrement délicate : si on ne prend pas garde, la procédure peut tourner en rond [67].

Lorsqu'elle s'intéresse aux procédures d'élimination des coupures, la logique moderne s'appuie très largement sur la correspondance de Curry–De Bruijn–Howard, notamment dans le cadre de la déduction naturelle et de la théorie des types. Dans cette approche, on démontre que tous les programmes bien typés que l'on peut écrire dans le langage des *termes de preuve* terminent ; on dit aussi que ces derniers se réduisent en une forme normale (en un nombre fini d'étapes). La preuve finale, correspondant au terme en forme normale, est alors une preuve sans coupure. Il faut choisir alors avec le plus grand soin le langage des termes et la *relation de réduction*, qui conditionne à la fois la terminaison de la procédure d'élimination, mais aussi (point 1 ci-dessus) la quantité de coupures qui sont effectivement éliminées. Citons, à ce titre, la notion de *coupure commutative* en déduction naturelle, que nous retrouverons plus tard. Un langage possible pour les termes de preuve en déduction naturelle modulo théorie est tiré de [52] et présenté figure 4.1.

Ce langage, tiré de [52], est relativement standard en ce qui concerne les connecteurs, mais fait certains choix au niveau des quantificateurs, notamment celui de faire intervenir explicitement le terme (de la logique) t dans le terme de preuve. Il est possible de faire le choix inverse [35]. Pour chacun des connecteurs et quantificateurs, nous présentons le ou les termes correspondant à la ou les règles d'introduction, puis celui ou ceux correspondant

π	$:=$	α	axiome
		$\mid \lambda\alpha\ \pi \mid (\pi\ \pi')$	implication
		$\mid \langle \pi, \pi' \rangle \mid \mathit{fst}(\pi) \mid \mathit{snd}(\pi)$	conjonction
		$\mid i(\pi) \mid j(\pi) \mid (\delta_\vee\ \pi_1\ \alpha_2\pi_2\ \alpha_3\pi_3)$	disjonction
		$\mid (\delta_\perp\ \pi)$	absurde
		$\mid \lambda x\ \pi \mid (\pi\ t)$	quantificateur universel
		$\mid \langle t, \pi \rangle \mid (\delta_\exists\ \pi\ x\alpha\pi')$	quantificateur existentiel

FIGURE 4.1 – Termes de preuve non typés

à la ou les règles d'élimination, comme il apparaît dans les règles de typage de la figure 4.2, la dynamique de réduction (le calcul des programmes) est défini plus loin, à la figure 4.3.

Les termes correspondant à l'élimination de la disjonction, de l'absurde et du quantificateur existentiel ont une notation voisine. Ils commencent tous par le symbole d'élimination δ — rappelons que $\perp, \vee$ et $\exists$ peuvent être respectivement compris comme les bornes supérieures nullaire, binaire et infinitaire —, les trois connecteurs/quantificateurs fonctionnent de manière similaire. Notons aussi, dans ces termes de preuve, le fait que x et α sont liés dans π', dans le cas du quantificateur existentiel, et similairement pour la disjonction.

Une sous-catégorie de termes sont les termes de preuve neutres, qui correspondent soit à un axiome, soit à une règle d'élimination. Lorsqu'un terme neutre n est substitué à une variable α dans un terme de preuve π, les seules redex de $\pi[n/\alpha]$ sont les redex de n et les redex $r[n/\alpha]$ où r est *déjà* une redex de π. Ainsi, substituer un terme neutre dans π ne créé aucune nouvelle redex.

Définition 4.1.1 (Neutralité)**.** *Un terme de preuve π est dit* neutre *s'il est de la forme α, $\pi_1\ \pi_2$, $\mathit{fst}(\pi_1)$, $\mathit{snd}(\pi_2)$, $\delta_\vee\ \pi_1\ \alpha_2\pi_2\ \alpha_3\pi_3$, $\delta_\perp\pi$, $\pi\ t$ ou $\delta_\exists\ \pi\ x\alpha\pi'$.*

Les règles de réduction définissent le calcul à effectuer lorsque l'on rencontre un terme de preuve qui est une élimination et dont le sous-terme *principal* est une introduction. Notons que, pour les termes de preuve bien typés, l'introduction et l'élimination ne peuvent que concerner le même connecteur/quantificateur ; on ne définit donc que les règles correspondantes ; ce sont celles de la figure 4.3.

Il est plus ardu de définir des termes de preuve et des règles de réduction pour le calcul des séquents [72, 135, 120, 5] ; c'est pourquoi une des méthodes usuelles d'élimination réductive des coupures passe par une traduction du

$$\frac{}{\Gamma \vdash \alpha : B} \text{ axiome, avec } \alpha : A \in \Gamma \text{ et } A \equiv B$$

$$\frac{\Gamma, \alpha : A \vdash \pi : B}{\Gamma \vdash \lambda\alpha.\pi : C} \Rightarrow_I, \text{ si } C \equiv A \Rightarrow B$$

$$\frac{\Gamma \vdash \pi : C \qquad \Gamma \vdash \pi' : A}{\Gamma \vdash (\pi\ \pi') : B} \Rightarrow_E, \text{ si } C \equiv A \Rightarrow B$$

$$\frac{\Gamma \vdash \pi : A \qquad \Gamma \vdash \pi' : B}{\Gamma \vdash \langle \pi, \pi' \rangle : C} \wedge_I, \text{ si } C \equiv A \wedge B$$

$$\wedge_{e1}, \text{ si } C \equiv A \wedge B\ \frac{\Gamma \vdash \pi : C}{\Gamma \vdash \mathit{fst}(\pi) : A} \qquad \frac{\Gamma \vdash \pi : C}{\Gamma \vdash \mathit{snd}(\pi) : B} \wedge_{e2}, \text{ si } C \equiv A \wedge B$$

$$\vee_{i1}, \text{ si } C \equiv A \vee B\ \frac{\Gamma \vdash \pi : A}{\Gamma \vdash i(\pi) : C} \qquad \frac{\Gamma \vdash \pi : B}{\Gamma \vdash j(\pi) : C} \vee_{i2}, \text{ si } C \equiv A \vee B$$

$$\frac{\Gamma \vdash \pi_1 : D \qquad \Gamma, \alpha_2 : A \vdash \pi_2 : C \qquad \Gamma, \alpha_3 : B \vdash \pi_3 : C}{\Gamma \vdash (\delta_\vee\ \pi_1\ \alpha_2\pi_2\ \alpha_3\pi_3) : C} \vee_E, \text{ si } D \equiv A \vee B$$

$$\frac{\Gamma \vdash \pi : B}{\Gamma \vdash (\delta_\perp\ \pi) : A} \perp_E, \text{ si } B \equiv \perp$$

$$\frac{\Gamma \vdash \pi : A}{\Gamma \vdash \lambda x.\pi : B} \forall_I, \text{ si } B \equiv \forall x A \text{ et } x \notin FV(\Gamma)$$

$$\frac{\Gamma \vdash \pi : B}{\Gamma \vdash (\pi\ t) : C} \forall_E, \text{ si } B \equiv \forall x A \text{ et } C \equiv (t/x)A$$

$$\frac{\Gamma \vdash \pi : C}{\Gamma \vdash \langle t, \pi \rangle : B} \exists_I, \text{ si } B \equiv \exists x A \text{ et } C \equiv (t/x)A$$

$$\frac{\Gamma \vdash \pi : C \qquad \Gamma, \alpha : A \vdash \pi' : B}{\Gamma \vdash (\delta_\exists\ \pi\ x\alpha\pi') : B} \exists_E, \text{ si } C \equiv \exists x A \text{ et } x \notin FV(\Gamma, B)$$

FIGURE 4.2 – Règles de typage de la déduction naturelle modulo théorie

$$
\begin{array}{rcl}
(\lambda\alpha.\pi_1\ \pi_2) & \rhd & \pi_1[\pi_2/\alpha] \\
fst(\langle \pi_1, \pi_2 \rangle) & \rhd & \pi_1 \\
snd(\langle \pi_1, \pi_2 \rangle) & \rhd & \pi_2 \\
(\delta_\vee\ i(\pi_1)\ \alpha_2\pi_2\ \alpha_3\pi_3) & \rhd & \pi_2[\pi_1/\alpha_2] \\
(\delta_\vee\ j(\pi_1)\ \alpha_2\pi_2\ \alpha_3\pi_3) & \rhd & \pi_3[\pi_1/\alpha_3] \\
\lambda x.\pi\ t & \rhd & \pi[t/x] \\
(\delta_\exists\ \langle t, \pi_1 \rangle\ \alpha x \pi_2) & \rhd & \pi_2[t/x, \pi_1/\alpha]
\end{array}
$$

FIGURE 4.3 – Règle de réduction des termes de preuve

calcul des séquents vers la déduction naturelle, une élimination des coupures dans ce dernier formalisme, puis une traduction inverse des preuves sans coupure de la déduction naturelle en calcul des séquents. Si l'on se limite à la notion de coupure usuelle de la déduction naturelle, on obtient cependant une preuve en calcul des séquents qui comporte encore des coupures. En effet, en déduction naturelle, la preuve suivante ne comporte aucune coupure au sens usuel "élimination suivie d'introduction".

$$
\dfrac{\vee_E\ \dfrac{\dfrac{\begin{matrix}\vdots\\ \Gamma, C, A \vdash B\end{matrix}}{\Gamma, C \vdash A \Rightarrow B}\Rightarrow_I \quad \dfrac{\begin{matrix}\vdots\\ \Gamma, D, A \vdash B\end{matrix}}{\Gamma, D \vdash A \Rightarrow B}\Rightarrow_I \quad \begin{matrix}\vdots\\ \Gamma \vdash C \vee D\end{matrix}}{\Gamma \vdash A \Rightarrow B} \qquad \Gamma \vdash A}{\Gamma \vdash B}\Rightarrow_E
$$

La traduction de cet ensemble de règles en calcul des séquents introduit des coupures, et ce même si seule l'une des deux prémisses mineures de la règle $\vee_E$ commence par la règle $\Rightarrow_I$. Un exemple complet est la preuve suivante, qui souffre du même problème, et où nous supposons que Γ est un contexte contenant $B \vee B$.

$$
\vee_E\ \dfrac{\wedge_I\ \dfrac{\text{ax}\ \dfrac{}{\Gamma, B \vdash B} \quad \text{ax}\ \dfrac{}{\Gamma, B \vdash B}}{\Gamma, B \vdash B \wedge B} \quad \wedge_I\ \dfrac{\dfrac{}{\Gamma, B \vdash B}\ \text{ax} \quad \dfrac{}{\Gamma, B \vdash B}\ \text{ax}}{\Gamma, B \vdash B \wedge B} \quad \dfrac{}{\Gamma \vdash B \vee B}\ \text{ax}}{\dfrac{\Gamma \vdash B \wedge B}{\Gamma \vdash B}\wedge_{E1}}
$$

La traduction standard, par induction, de cette preuve en calcul des séquents donne la preuve ci-dessous.

$$
\dfrac{\text{ax}\ \dfrac{}{\Gamma \vdash B \vee B} \qquad \dfrac{\vee_L\ \dfrac{\wedge_R\ \dfrac{\text{ax}\ \dfrac{}{\Gamma, B \vdash B} \quad \dfrac{}{\Gamma, B \vdash B}\ \text{ax}}{\Gamma, B \vdash B \wedge B} \quad \wedge_R\ \dfrac{\text{ax}\ \dfrac{}{\Gamma, B \vdash B} \quad \dfrac{}{\Gamma, B \vdash B}\ \text{ax}}{\Gamma, B \vdash B \wedge B}}{\Gamma, B \vee B \vdash B \wedge B} \qquad \dfrac{\dfrac{}{\Gamma, B \vee B, B, B \vdash B}\ \text{ax}}{\Gamma, B \vee B, B \wedge B \vdash B}\wedge_L}{\Gamma, B \vee B \vdash B}\ \text{cut}}{\Gamma \vdash B}\ \text{cut}
$$

Afin de pallier ce problème, il faut étendre la notion de coupure et de réduction des termes de preuve aux *coupures commutatives*, dont nous avons deux exemples ci-dessus : une règle d'élimination, séparée de la règle d'introduction correspondante par un nombre arbitraire de règles $\vee_E$ et $\exists_E$. Cela revient aussi, dans une preuve en déduction naturelle, à pousser "vers le bas" les règles $\vee_E$ et $\exists_E$.

Notons par ailleurs que c'est justement dans le but inverse (éliminer les coupures en déduction naturelle) que Gentzen a introduit le calcul des séquents, démontré le théorème d'élimination des coupures, puis traduit les preuves sans coupure obtenues en preuves sans coupure de la déduction naturelle (Hauptsatz [62]). Mais, il n'avait pas à l'époque à disposition les termes de preuve.

Enfin, l'élimination des coupures par réduction des termes de preuve n'est pas la seule approche possible. Mentionnons en particulier l'approche plus géométrique des réseaux de preuve [66].

4.2 Normalisation légère

Dans cette partie du chapitre, nous étudions des versions affaiblies des preuves de normalisation forte des termes de preuve qui permettent d'obtenir des résultats d'élimination des coupures en un sens plus fort que ceux d'admissibilité sémantique des coupures. Pour cela, nous développons des (pré)algèbres qui sont construites à partir de termes de preuve faiblement normalisants, ou directement de séquents. On peut comprendre ces dernières algèbres comme un effacement des termes de preuve dans la première ; elles permettent d'autre part d'obtenir des résultats intéressants à l'ordre supérieur.

Comme il a été rappelé en début de chapitre, une grande partie de cette section peut se lire sans tenir compte de la réécriture (à l'exception de la section 4.2.3), même si, dans le calcul syntaxique (déduction naturelle ou calcul des séquents), la réécriture est présente.

Les difficultés sont majoritairement celles de la logique du premier ordre, déduction naturelle ou en calcul des séquents, alors que les difficultés liées à la présence de la réécriture de la Déduction modulo théorie sont pour la plupart reportées sur la supercohérence (classique ou intuitionniste), qui assure la construction d'un modèle de la réécriture que nous pouvons utiliser librement ailleurs. Ceci est particulièrement vrai dans les sections 4.2.1 et 4.2.4 et nous y indiquerons explicitement, les endroits où la prise en compte de la réécriture est importante.

4.2.1 Normalisation faible des termes de preuve

Puisque nous nous intéressons à la production de preuves sans coupure, il est suffisant de démontrer que tous les termes de preuves bien typés sont *faiblement* normalisants. Les arguments de normalisation faible présentés ici sont des extensions de [34], notamment la définition 4.2.2.

Définition 4.2.1 (Normalisation faible)**.** *Un terme de preuve π est dit* faiblement normalisant *si, et seulement si, il existe une chaîne finie de réductions selon les règles de la figure 4.3,*

$$\pi \rhd \pi_1 \rhd \cdots \rhd \pi_n \rhd \pi',$$

telle que π' est irréductible.

Pour démontrer la normalisation faible, nous adoptons une approche classique qui consiste à interpréter les formules par certains *ensembles de termes*. Les caractéristiques de ces ensembles, appelés candidats de réductibilité, sont définies a priori, sans induction sur les formules, de même que les V-complexes doivent être définis a priori aux sections 2.4.3 et 3.2.4. Notre but étant plus modeste que la normalisation forte, nous pouvons adapter la notion de candidat de réductibilité [67] et l'affaiblir.

Définition 4.2.2 (Candidat de réductibilité faible)**.** *Soit R un ensemble de termes de preuve. R est un* candidat de réductibilité pour la normalisation faible *si, et seulement si :*

- *($\boldsymbol{P}_1$) si $\pi \in R$, alors π est faiblement normalisant ;*
- *($\boldsymbol{P}_{3a}$) si π est neutre et s'il existe $\pi' \in R$ tel que $\pi \rhd \pi'$, alors $\pi \in R$;*
- *($\boldsymbol{P}_{3b}$) R contient tous les termes de preuve isolés et faiblement normalisants.*

Comparons rapidement la définition 4.2.2 avec la notion usuelle de candidat de réductibilité pour la normalisation forte [67]. $\mathbf{CR}_1$ a été remplacé de manière naturelle par $\mathbf{P}_1$. $\mathbf{CR}_2$, qui impose que tous les réduits de $\pi \in R$ sont aussi dans R, n'a plus lieu d'être. $\mathbf{CR}_3$ a été partiellement remplacée par $\mathbf{P}_{3a}$, en remplaçant la quantification universelle en quantification existentielle. Enfin, $\mathbf{P}_{3b}$ complète la condition $\mathbf{P}_{3a}$, qui, seule, n'implique plus qu'un candidat de réductibilité est non vide, ce que $\mathbf{CR}_3$ force (voir $\mathbf{CR}_4$ dans [67]).

Pour que la définition 4.2.2 soit complète, nous devons expliciter ce que nous entendons par terme de preuve isolé. Pour cela, reprenons la définition 4.1.1 de neutralité : substituer un terme neutre n dans π ne créé aucune nouvelle redex. Mais ceci ne dit rien sur les redex “futures” de $\pi[n/\alpha]$; par exemple $n = (\lambda x.\lambda y.y)\ z$ est neutre, mais son réduit $\lambda y.y$ ne l'est plus. En conséquence, si nous réduisons $\pi[n/\alpha]$ en $\pi[\lambda y.y/\alpha]$, de nouvelles redex peuvent apparaître. Ce défaut d'hérédité est corrigé par la notion de terme de preuve isolé.

Définition 4.2.3 (Neutralité héréditaire)**.** *Un terme de preuve π est dit* isolé *ou* neutre héréditairement *s'il est neutre et si tous ses réduits sont neutres.*

Une autre manière de comprendre cette définition est que, si π est isolé et π' est un terme de preuve, alors tout réduit $\pi'[\pi/\alpha] \rhd^* \theta$ peut s'écrire sous la forme $\theta = \pi'_1[\pi_1/\alpha]$, avec $\pi' \rhd^* \pi'_1$ et $\pi \rhd^* \pi_1$.

Nous définissons les opérations suivantes sur les ensembles de termes de preuve.

Définition 4.2.4 (Opérateurs sur les ensembles de termes de preuve)**.** *Nous définissions certains ensembles de termes de preuve, étant donnés des ensembles de termes de preuve a et b et un ensemble d'ensembles de termes de preuve S.*

- *$\top$ et $\bot$ sont tous deux l'ensemble de tous les termes de preuve faiblement normalisants, appelé WN.*
- *$a \Rightarrow b$ est l'ensemble des termes de preuve π qui sont, soit isolés et faiblement normalisants, soit tels qu'il existe π_1 tel que $\pi \rhd^* \lambda\alpha.\pi_1$ et pour tout $\pi' \in a$, $\pi_1[\pi'/\alpha] \in b$.*
- *$a \wedge b$ est l'ensemble des termes de preuve π qui sont, soit isolés et faiblement normalisants, soit tels qu'il existe $\pi_1 \in a$ et $\pi_2 \in b$ tels que $\pi \rhd^* \langle \pi_1, \pi_2 \rangle$.*
- *$a \vee b$ est l'ensemble des termes de preuve π qui sont, soit isolés et faiblement normalisants, soit tels qu'il existe $\pi_1 \in a$ tel que $\pi \rhd^* i(\pi_1)$ ou $\pi_2 \in b$ tel que $\pi \rhd^* j(\pi_2)$.*
- *$\forall S$ est l'ensemble des termes de preuve π qui sont, soit isolés et faiblement normalisants, soit tels qu'il existe π_1 tel que $\pi \rhd^* \lambda x.\pi_1$ et, pour tout terme t et tout $a \in S$, $\pi_1[t/x] \in a$.*
- *$\exists S$ est l'ensemble des termes de preuve π qui sont, soit isolés et faiblement normalisants, soit tels qu'il existe $a \in S$, $\pi_1 \in a$ et un terme t tels que $\pi \rhd^* \langle t, \pi_1 \rangle$.*

Cette définition est bien entendu à mettre en parallèle avec les opérations correspondantes dans le cadre de la normalisation forte [52] (voir aussi la définition 4.2.9 ci-dessous pour une version encore plus faible). Notons que, pour $\bot$, nous aurions pu choisir le plus petit ensemble vérifiant la définition 4.2.2, c'est-à-dire l'intersection de tous les candidats de réductibilité. Il n'y a pas de raison objective de faire ce choix, puisque nous allons, dans la structure algébrique que nous sommes sur le point d'introduire, ignorer la relation d'ordre.

Nous pouvons considérer l'ensemble $\mathcal{WN}$ de tous les candidats de réductibilité ; il est clos par les opérations de la définition 4.2.4. Nous obtenons une structure $\langle \mathcal{WN}, \top, \bot, \Rightarrow, \wedge, \vee, \forall, \exists\rangle$ et, en y ajoutant la relation (de préordre) triviale $\leq$, nous obtenons une préalgèbre de Heyting, ce qui nous permet d'utiliser la supercohérence [45] pour obtenir une interprétation dans cette algèbre. Nous ne sommes pas en présence d'une algèbre de Heyting, car il est impossible de munir cette structure d'une relation d'ordre, $\leq$ étant seulement une relation de préordre. Par exemple, si nous avions une relation d'ordre, nous devrions avoir $\top \wedge \top = \top$; or nous avons $\lambda\alpha.\alpha \in \top$, mais $\lambda\alpha.\alpha \notin \top \wedge \top$.

Définition 4.2.5 (Supercohérence)**.** *Un système de réécriture $\mathcal{RE}$ est* supercohérent *si, et seulement si, pour toute préalgèbre de Heyting Ω, il existe un modèle à valeurs dans Ω (cf. définition 2.3.1) tel que, pour toutes formules $A \equiv B$ et pour toute valuation φ, $[\![A]\!]_\varphi = [\![B]\!]_\varphi$.*

En particulier, il existe un tel modèle dans la préalgèbre de Heyting $\mathcal{WN}$; soit $[\![\ ]\!]$ la fonction d'interprétation associée.

La définition précédente est une simplification de la définition réelle [45], qui se restreint aux préalgèbres complètes et ordonnées (voir la définition 3.3.4), ce qui est bien le cas de $\mathcal{WN}$, qu'il faut alors munir d'un ordre $\sqsubseteq$ qui raffine le préordre $\leq$, voir la discussion de la section 3.3.2. Munis d'un tel modèle, nous pouvons ensuite démontrer le théorème d'adéquation.

Lemme 4.2.6 (Adéquation)**.** *Soit $\Gamma \vdash \pi : A$ un séquent prouvable ; soit σ une substitution et φ une valuation.*

Soit θ une substitution de termes de preuve telle que, pour tout $\alpha_i : A_i \in \Gamma$, nous avons $\theta(\alpha_i) \in [\![A_i]\!]_\varphi$. Alors $\pi\sigma\theta \in [\![A]\!]_\varphi$.

Pour démontrer lemme d'adéquation, il est important d'avoir $[\![A]\!]_\varphi = [\![B]\!]_\varphi$ quand $A\mathrm{conv}B$. Notons qu'aucune condition de synchronisation n'est nécessaire entre la substitution σ et la valuation φ. La conséquence du lemme d'adéquation est immédiate : en choisissant σ et θ comme étant l'identité, on obtient $\pi \in [\![A]\!]_\varphi$. Ce choix est possible car, pour tout α_i, $\theta(\alpha_i) = \alpha_i$ est un terme isolé faiblement normalisant ; il fait donc partie de tous les candidats de réductibilité, et en particulier de $[\![A_i]\!]_\varphi$.

En conclusion, $\pi \in [\![A]\!]_\varphi$; c'est un terme de preuve faiblement normalisant. Étant donné une preuve d'un séquent $\Gamma \vdash A$ (sans termes de preuve), il est aisé de construire le terme de preuve π correspondant, qui est faiblement normalisant et possède donc une forme normale. Ceci génère une preuve de $\Gamma \vdash A$ sans coupure.

Théorème 4.2.7 (Normalisation faible)**.** *Soit $\mathcal{RE}$ un système de réécriture supercohérent. Soit $\Gamma \vdash A$ un séquent prouvable en déduction naturelle modulo $\mathcal{RE}$. Alors il existe une preuve sans coupure de $\Gamma \vdash A$.*

De plus, si π est un terme de preuve tel que $\Gamma \vdash \pi : A$ est dérivable selon les règles de la figure 4.2, alors π est faiblement normalisant.

4.2.2 Algèbres de séquents en déduction naturelle

Démontrer la terminaison de la réduction des termes de preuves demande de donner une structure bien précise aux preuves, indépendante, dans une certaine mesure, des termes de preuve eux-mêmes. Nous pouvons donc spécialiser les notions de la section précédente aux preuves elles-mêmes.

Concrètement, dans le cadre de la déduction naturelle, la notion de terme de preuve en *forme normale* et celle de terme de preuve en *neutre* ne dépendent pas directement de la notion de terme de preuve, et nous pouvons introduire deux types de séquents en déduction naturelle, $\vdash^*$ et $\vdash_{ne}$, dont toutes les preuves seront sans coupure et neutres respectivement. La manière de dériver de tels séquents est présenté dans le calcul de la figure 4.4, adapter ce système d'inférence à la Déduction modulo théorie se fait sur le modèle de la figure 4.2.

$$\frac{\Gamma \vdash_{ne} A}{\Gamma \vdash^* A}\text{coerce} \qquad \frac{A \in \Gamma}{\Gamma \vdash_{ne} A}\text{ax}$$

$$\frac{\Gamma \vdash^* A \qquad \Gamma \vdash^* B}{\Gamma \vdash^* A \wedge B}\wedge_I \qquad \frac{\Gamma \vdash_{ne} A \wedge B}{\Gamma \vdash_{ne} A}\wedge_{E1} \qquad \frac{\Gamma \vdash_{ne} A \wedge B}{\Gamma \vdash_{ne} B}\wedge_{E2}$$

$$\frac{\Gamma \vdash^* A}{\Gamma \vdash^* A \vee B}\vee_{I1} \qquad \frac{\Gamma \vdash^* B}{\Gamma \vdash^* A \vee B}\vee_{I2} \qquad \frac{\Gamma \vdash_{ne} A \vee B \qquad A, \Gamma \vdash^* C \qquad B, \Gamma \vdash^* C}{\Gamma \vdash_{ne} C}\vee_E$$

$$\frac{\Gamma, A \vdash^* B}{\Gamma \vdash^* A \Rightarrow B}\Rightarrow_I \qquad \frac{\Gamma \vdash_{ne} A \Rightarrow B \qquad \Gamma \vdash^* A}{\Gamma \vdash_{ne} B}\Rightarrow_E$$

$$\frac{}{\Gamma \vdash^* \top}\top_I \qquad \frac{\Gamma \vdash_{ne} \bot}{\Gamma \vdash_{ne} A}\bot_E$$

$$\frac{\Gamma \vdash^* A \qquad x \notin FV(\Gamma)}{\Gamma \vdash^* \forall x.A}\forall_I \qquad \frac{\Gamma \vdash_{ne} \forall x.A}{\Gamma \vdash_{ne} A[t/x]}\forall_E$$

$$\frac{\Gamma \vdash^* A[t/x]}{\Gamma \vdash^* \exists x.A}\exists_I \qquad \frac{\Gamma \vdash_{ne} \exists x.A \qquad A, \Gamma \vdash^* C \qquad x \notin FV(C, \Gamma)}{\Gamma \vdash_{ne} C}\exists_E$$

FIGURE 4.4 – Déduction naturelle sans coupure

Ici, neutre est considéré dans son acceptation "règle d'élimination *et* sans coupure". On utilise parfois "neutre" pour les preuves qui se terminent par une simple règle d'élimination, ce qui correspond directement à la définition 4.1.1 pour les termes de preuve. L'intégration de la notion "sans coupure" se traduit par une définition par induction mutuelle, qui est le reflet de la figure 4.4.

Définition 4.2.8 (Neutre et sans coupure)**.** *Une preuve en déduction naturelle est dite* sans coupure *lorsque :*

- *c'est une preuve neutre ;*
- *c'est une application d'une règle d'introduction, et la (les) prémisse(s) est (sont) sans coupure.*

Une preuve est dite neutre *lorsque :*

- *c'est une application de la règle axiome ;*
- *c'est une application d'une règle d'élimination, la preuve de la prémisse principale est neutre et la (les) prémisses secondaires sont des preuves sans coupure.*

Munis de ces définitions, nous pouvons simplifier la notion de candidats de réductibilité de les termes de preuves au cas des séquents. La définition qui suit fait explicitement intervenir la réécriture et la déduction naturelle modulo théorie, mais l'on peut évidemment se restreindre au cas de la déduction naturelle au premier ordre en considérant le système de réécriture vide.

Nous commençons par la définition d'opérateurs sur les ensembles de séquents (au lieu des ensembles de termes de la définition 4.2.4). Ici, les séquents sont dénotés $\Gamma \vdash A$, car l'information de dérivabilité sans coupure ou neutre n'est pas connue et, par dessus tout, nous ne parlons pas ici de preuve concrète, mais de dérivabilité.

Définition 4.2.9 (Opérateurs sur les ensembles de séquents)**.** *Nous définissons certains ensembles de séquents, étant donnés des ensembles de séquents a et b et un ensemble d'ensembles de séquents S.*

- $\top$ *est l'ensemble des séquents $\Gamma \vdash C$ qui ont une preuve neutre ou tels que $C \equiv \top$.*
- $\bot$ *est l'ensemble des séquents $\Gamma \vdash C$ qui ont une preuve neutre.*
- $a \wedge b$ *est l'ensemble des séquents $\Gamma \vdash C$ qui ont une preuve neutre ou tels que $C \equiv (A \wedge B)$, avec $(\Gamma \vdash A) \in a$ et $(\Gamma \vdash B) \in b$.*
- $a \vee b$ *est l'ensemble des séquents $\Gamma \vdash C$ qui ont une preuve neutre ou tels que $C \equiv (A \vee B)$, avec $(\Gamma \vdash A) \in a$ ou $(\Gamma \vdash B) \in b$.*
- $a \Rightarrow b$ *est l'ensemble des séquents $\Gamma \vdash C$ qui ont une preuve neutre ou tels que $C \equiv (A \Rightarrow B)$ et, pour tous les contextes Σ tels que $(\Gamma, \Sigma \vdash A) \in a$, $(\Gamma, \Sigma \vdash B) \in b$.*
- $\forall\ S$ *(où S est un ensemble d'ensembles de séquents) est l'ensemble des séquents $\Gamma \vdash C$ qui ont une preuve neutre ou tels que $C \equiv (\forall x\ A)$ et, pour tout terme t et $a \in S$, $(\Gamma \vdash A[t/x]) \in a$.*
- $\exists\ S$ *(où S est un ensemble d'ensembles de séquents) est l'ensemble des séquents $\Gamma \vdash C$ qui ont une preuve neutre ou tels que $C \equiv (\exists x\ A)$ et, pour au moins un terme t et un élément $a \in S$, $(\Gamma \vdash A[t/x]) \in a$.*

Ceci nous amène à considérer le plus petit ensemble $\mathcal{S}$ clos par toutes ces opérations, ainsi que par intersections arbitraires.

Nous obtenons une structure algébrique $\langle \mathcal{S}, \top, \bot, \Rightarrow, \wedge, \vee, \forall, \exists \rangle$, mais elle ne forme pas une algèbre de Heyting. $\mathcal{S}$ est une préalgèbre de Heyting [45], où la relation d'ordre ne respecte pas forcément l'antisymétrie. Par exemple, $\top \wedge \top$ contient le séquent $\vdash \top \wedge \top$, alors que $\top$ lui-même ne le contient pas. $\wedge$ n'est alors plus l'opérateur de borne inférieur usuel, puisqu'on aura $\top \wedge \top \neq \top$. Plus exactement, l'ordre $\leq$ n'est plus qu'un préordre (que nous définissons d'ailleurs comme le préordre trivial — ou, de manière équivalente, tous les éléments de S sont définis comme étant positifs [45]) pour lequel $\wedge$ reste bien un opérateur de borne inférieur.

Cependant, la structure $\mathcal{S}$ se rapproche très fortement d'une algèbre de Heyting. L'idée directe de quotienter cette préalgèbre par la relation d'équivalence générée par le préordre donne, dans notre cas, une algèbre, certes de Heyting, mais triviale car réduite à un point. Nous allons effectuer une analyse plus fine et en extraire une algèbre de Heyting plus intéressante. Pour l'instant, avoir une préalgèbre de Heyting nous suffit pour obtenir par supercohérence (définition 4.2.5) une interprétation dans $\mathcal{S}$ qui satisfait le système de réécriture $\mathcal{RE}$.

Cela est, entre autres, le cas lorsque $\mathcal{RE}$ est vide ; les résultats s'appliqueront alors à la déduction naturelle en logique du premier ordre. Munis d'une telle interprétation, nous pourrions démontrer le résultat suivant, selon une structure tout à fait similaire au théorème 3.1 de [52] ou au lemme 4.2.6 précédent, avec lesquels on remarquera la similarité de la formulation.

Conjecture 4.2.10. *Soit $A_1, \cdots, A_n \vdash A$ un séquent prouvable en déduction naturelle. Soient σ une substitution et φ une valuation dans le modèle donné par la définition 4.2.5. Soit $\Delta_1, \cdots, \Delta_n$ une séquence de contextes tels que pour tout i, $\Delta_i \vdash \sigma A_i \in [\![A_i]\!]_\varphi$.*

Alors $\Delta_1, \cdots, \Delta_n \vdash \sigma A \in [\![A]\!]_\varphi$.

La démonstration d'un tel résultat demanderait l'appel à des résultats intermédiaires, comme l'admissibilité de la contraction et de l'affaiblissement à gauche dans les éléments de $\mathcal{S}$ [51]. Le théorème d'élimination des coupures suivrait ensuite par le fait que $A_i \vdash A_i \in [\![A_i]\!]$.

Cette conjecture, reflet des théorèmes de normalisation qui s'appuient sur les candidat de réductibilité, a une forme bien particulière. On observe en particulier que, dans les conditions $\Delta_i \vdash A_i \in [\![A_i]\!]_\varphi$, les conclusions, A_i, sont les mêmes que celles dont on prend l'interprétation, $[\![A_i]\!]_\varphi$. Il s'agit d'une spécialisation du lemme 4.2.6, qui prend sa source dans l'application que nous avons l'intention de faire du lemme 4.2.6, à savoir prendre l'identité pour θ, ce qui correspond à considérer $\Delta_i = A_i$. Une telle simplification est trop poussée, mais nous pouvons tout de même nous restreindre aux séquents dont la conclusion est A_i.

Cette simplification suggère que l'on peut *extraire* les contextes Δ_i et travailler avec ceux-ci directement. Pour cela, nous supposons avoir le modèle et l'interprétation $[\![\ ]\!]$ définis par la définition 4.2.5.

Définition 4.2.11 (Extraction de contextes)**.** *Soit A une formule. L'extraction $[A]$ est définie comme l'ensemble des contextes $A_1, \cdots, A_n$ tels que, pour toute valuation φ et substitution σ, si, pour tout i, $\Delta \vdash \sigma A_i \in [\![A_i]\!]_\varphi$, alors $\Delta \vdash \sigma A \in [\![A]\!]_\varphi$.*

Cette définition rappelle la définition 3.2.2 de la section 3.2.2, qui définit l'extraction $\lceil A \rceil$ comme tous les contextes Γ tels que $\Gamma \vdash A$ est démontrable sans coupure.

L'adaptation que nous avons faite est de prendre en compte, non pas tous les contextes, mais seulement ceux qui sont dans $[\![A]\!]$, avec une indirection supplémentaire, qui revient à dire que nous restreignons l'ensemble des contextes $A_1, \cdots, A_n$ acceptables à ceux qui "peuvent être branchés avec tout contexte compatible". La conjecture 4.2.10 pourrait d'ailleurs être formulée de la même manière avec un contexte Δ partagé par tous les A_i – ou la définition 4.2.11 ci-dessus énoncée avec un ensemble de contextes Δ_i.

L'algèbre de Heyting est construite à partir de l'ensemble sous-jacent Ω, qui est le plus petit ensemble clos par intersections arbitraires et contenant les extractions de contexte. En munissant cet ensemble de la relation d'ordre induite par l'inclusion, nous obtenons un treillis complet, auquel il faut ensuite ajouter une opération d'implication, qui doit donc respecter les conditions de la définition 2.1.2. Ce dernier point est assez technique à vérifier, mais une fois que cette algèbre a été correctement définie, nous obtenons le résultat suivant.

Lemme 4.2.12. *Pour toutes formules A et B, nous avons, avec $\mathcal{T}$ l'ensemble des termes du langage,*

- $[\top] = \top$,
- $[\bot] = \bot$,
- $[A \wedge B] = [A] \wedge [B]$,
- $[A \vee B] = [A] \vee [B]$,
- $[A \Rightarrow B] = [A] \Rightarrow [B]$,
- $[\forall x A] = \forall \{A[t/x] \mid t \in \mathcal{T}\}$,
- $[\exists x A] = \exists \{A[t/x] \mid t \in \mathcal{T}\}$.

Ce résultat est intéressant de plusieurs points de vue. Tout d'abord, sa démonstration fait appel aux mêmes principes que la preuve de la conjecture 4.2.10 et, plus généralement, que les preuves correspondantes de normalisation par les techniques de candidats de réductibilité. Mais surtout, cela nous permet de construire un modèle $\mathcal{D}$ à base de termes et de définir une interprétation $[\![\ ]\!]^{\mathcal{D}}$ dans ce modèle.

Définition 4.2.13. *Soient t un terme et A une formule. Pour toute substitution φ, nous posons*

- $[\![t]\!]^{\mathcal{D}}_{\varphi} := [t\varphi]$, *la classe d'équivalence modulo $\equiv$ du terme $t\varphi$ et*
- $[\![A]\!]^{\mathcal{D}}_{\varphi} := [A\varphi]$, *selon la définition 4.2.11.*

Il faut ensuite démontrer que cette définition correspond réellement à ce que l'on attend d'elle pour pouvoir parler de modèle.

- Il est clair que nous pouvons définir le domaine de ce modèle comme les classes d'équivalence de termes (potentiellement ouverts). De ce point de vue, nous avons affaire à un modèle syntaxique.
- Le lemme 4.2.12 nous assure que $[\![\,]\!]$ respecte les propriétés des interprétations de la définition 2.1.4. Notons que nous avons procédé à rebours en ne définissant pas $[\![\,]\!]$ par induction. L'approche inductive est possible, mais demande alors de démontrer $[\![A]\!]^{\mathcal{D}}_{\varphi} = [A]^{\mathcal{D}}_{\varphi}$, et fait appel au lemme 4.2.12.
- Considérer les classes d'équivalence de termes nous assure que nous avons un modèle de $\mathcal{RE}$ pour les termes (définition 2.3.1). Pour les formules, la vérification, immédiate, provient de l'invariance de la prouvabilité par conversion modulo $\equiv$ et du fait que nous avons supposé que l'interprétation $[\![\,]\!]$ dans la structure $\mathcal{S}$ (obtenue par supercohérence) était un modèle de $\mathcal{RE}$.

Ici, nous avons l'égalité $[\![A]\!]_{\varphi} = [A\varphi]$, ce qui représente la différence majeure entre la définition 4.2.13 ci-dessus et le lemme 3.2.8, auquel elle fait écho, qui démontre au contraire $\lfloor A \rfloor \leq [\![A]\!] \leq \lceil A \rceil$. De même, nous pouvons faire la comparaison avec la condition de la définition 3.2.15 sur les formules (de type o) qui demande qu'un V-complexe $\langle t, v\rangle$ doit respecter $\lfloor t \rfloor \leq v \leq \lceil t \rceil$. Enfin, nous pouvons conclure.

Théorème 4.2.14 (Complétude)**.** *Soit $\mathcal{RE}$ un système de réécriture tel que $[\![\,]\!]^{\mathcal{D}}$ en soit un modèle. Soient Γ un contexte et A une formule. Si, pour toute valuation φ, $\bigwedge [\![\Gamma]\!]^{\mathcal{D}}_{\varphi} \leq [\![A]\!]^{\mathcal{D}}_{\varphi}$, alors le séquent $\Gamma \vdash A$ a une preuve dans le système de la figure 4.4.*

La démonstration est similaire à la démonstration du théorème 3.2.9, cependant elle utilise l'égalité $[A] = [\![A]\!]^{\mathcal{D}}$ au lieu de l'inclusion, comme mentionné ci-dessus. En choisissant l'identité pour φ, on obtient $\Gamma \in \bigcap[\Gamma] = \bigwedge [\![\Gamma]\!]^{\mathcal{D}} \leq [\![A]\!]^{\mathcal{D}} = [A]$, et donc $\Gamma \in [A]$, ce qui nous permet de conclure.

Corollaire 4.2.15 (Élimination des coupures)**.** *Si le séquent $\Gamma \vdash A$ est prouvable en déduction naturelle modulo un système de réécriture supercohérent $\mathcal{RE}$, alors il est prouvable sans coupure.*

La supercohérence est un moyen de construire une interprétation $[\![\,]\!]$, dans la préalgèbre de Heyting des séquents de la définition 4.2.9 qui est un modèle de $\mathcal{RE}$, et qui engendre à son tour une interprétation $[\![\,]\!]^{\mathcal{D}}$ qui est un modèle de $\mathcal{RE}$, ce qui nous permet de faire appel au théorème 4.2.14.

4.2.3 Application au cas de l'ordre supérieur

Nous nous intéressons maintenant à la manière dont les arguments de la section 4.2.2 s'appliquent au cas de la logique d'ordre supérieur en Déduction modulo théorie. Pour ceci, nous considérons le système de réécriture $\mathcal{RE}$ de la section 2.4.3, en gardant en tête que nous ne sommes plus en calcul des séquents, mais en déduction naturelle modulo $\mathcal{RE}$. Nous sommes donc dans le cas intuitionniste, et la comparaison à faire sera par rapport aux résultats de la section 3.2.4, qui ne sont exprimés ni en Déduction modulo théorie ni pour un calcul de combinateurs, et aux définitions de la section 2.4.3. Malgré ces différences, nous resterons en mesure de faire une comparaison sur les points critiques.

La première hypothèse de la section 4.2.2 est que nous devons être en présence d'un système de réécriture $\mathcal{RE}$ supercohérent. C'est bien le cas du système de réécriture $\mathcal{R}$ ($\mathcal{E}$ est vide) de la figure 2.6 [45]. Le modèle $\mathcal{M}$, construit à valeurs dans $\mathcal{S}$, donne

— $M_\iota := \{0\}$;
— $M_o := \mathcal{S}$, l'ensemble de base de la préalgèbre de Heyting de la section 4.2.2 précédente ;
— $M_{T\to U} := M_U^{M_T}$, l'ensemble des fonctions de M_T dans M_U.

Nous pouvons déjà remarquer que nous interprétons les termes de type o dans le domaine de la préalgèbre $\mathcal{S}$ des ensembles de séquents elle-même, à ne pas confondre avec le combinateur S. Les termes de type fonctionnel sont, eux, interprétés dans le domaine des fonctions : pas de V-complexe ici et, notamment, pas de première composante syntaxique et seconde composante sémantique. L'interprétation des symboles du langage est

— $\hat{S} := a \mapsto (b \mapsto (c \mapsto a(c)(b(c))\,))$, la fonction qui applique ses arguments exactement de la manière dont est définie la règle de réécriture sur S ;
— $\hat{K} := a \mapsto (b \mapsto a(b)\,)$;
— $\hat{\varepsilon} := a \mapsto a$;
— $\hat{\dot{\top}} := \top$, $\hat{\dot{\bot}} := \bot$, $\hat{\dot{\Rightarrow}} := \Rightarrow$, $\hat{\dot{\vee}} := \vee$, $\hat{\dot{\wedge}} := \wedge$, $\hat{\dot{\neg}} := \neg$;
— $\hat{\dot{\forall}}_T := a \mapsto \forall(Im(a))$, $\hat{\dot{\exists}}_T := a \mapsto \exists(Im(a))$, avec $Im(a)$ l'ensemble des valeurs prises par la fonction a (l'image de a).

Il est intéressant de noter que la constante ε est interprétée comme l'identité, en cohérence avec l'étape précédente : les termes de type o sont interprétés dans $M_o = \mathcal{S}$ et, par définition des modèles, ε doit être une fonction de type $\mathcal{S}^{M_o}$, car $\mathcal{S}$ est la préalgèbre dans laquelle nous devons interpréter les symboles de prédicat.

Le modèle que nous venons de construire n'est pas à valeurs dans une algèbre de Heyting, mais dans la préalgèbre $\mathcal{S}$ des ensemble de séquents ;

la comparaison est malaisée pour le moment. Mais la section 4.2.2 nous fournit aussi un modèle à valeur dans l'algèbre de Heyting des ensembles de contextes Ω construite à partir de l'extraction de contextes de la définition 4.2.11. Le modèle construit (définition 4.2.13) est un modèle syntaxique ; on obtient donc en particulier les domaines

- $D_\iota = \mathcal{T}_\iota$, les (classes d'équivalence des) termes de type ι, et
- $D_o = \mathcal{T}_o$, les (classes d'équivalence des) termes de type o.

Ici, nous avons bien une algèbre de Heyting, et toujours pas de V-complexe. La situation est même dramatiquement simple : un modèle syntaxique.

Ce modèle syntaxique répond tout à fait aux attentes de l'intentionnalité de la logique d'ordre supérieur, puisque le terme $\dot{\top} \mathbin{\dot{\wedge}} \dot{\top}$ est interprété de manière différente du terme $\dot{\top}$: $[\dot{\top} \mathbin{\dot{\wedge}} \dot{\top}] \neq [\dot{\top}]$ car $\dot{\top} \mathbin{\dot{\wedge}} \dot{\top}$ et $\dot{\top}$ ne sont pas convertibles modulo $\mathcal{RE}$. De même, pour P un prédicat de type $o \to o$, on aura $P(\dot{\top} \mathbin{\dot{\wedge}} \dot{\top}) \in [\![P(\dot{\top} \mathbin{\dot{\wedge}} \dot{\top})]\!]^{\mathcal{D}}$ et $P(\dot{\top}) \in [\![P(\dot{\top})]\!]$, sans avoir les appartenances croisées dans le cas général. Ainsi $[P(\dot{\top} \mathbin{\dot{\wedge}} \dot{\top})]$ et $[P(\dot{\top})]$ seront différents. La question de la différence entre la dénotation de $\dot{\top}$ dans ce modèle et son interprétation en tant que constante logique, qui apparaît à la section 3.2.4, n'a plus lieu d'être, car $\dot{\top}$ n'est plus une formule ; ce terme n'a donc pas besoin de recevoir une interprétation dans l'algèbre de Heyting $\mathcal{D}$. C'est $\varepsilon(\dot{\top})$ qui doit être interprétée par $\top$, comme à la section 2.4.3.

Mais la seconde raison d'être des V-complexes est de résoudre le problème de la définition imprédicative de la logique, et de l'impossibilité de définir une interprétation par une induction pure. Dans notre cas, nous nous passons de définition inductive dans le modèle $\mathcal{D}$, car la définition 4.2.11 définit $[A]$ pour toutes les formules A, sans induction. Le lemme 4.2.12 démontre ensuite que $[\,]$ possède toutes les propriétés voulues pour être une interprétation, mais, encore une fois, ne se prouve pas par induction sur la taille des formules. Nous pouvons donc ignorer ce problème d'imprédicativité *dans* $\mathcal{D}$.

Pour résumer, la construction du modèle $\mathcal{D}$ ne demande aucune technique spécifique à la logique d'ordre supérieur. Tout le travail a été fait lors de la construction du *premier* modèle sous-jacent $\mathcal{M}$, duquel nous avons *extrait* $\mathcal{D}$. Le modèle $\mathcal{M}$ possède aussi quelques particularités, par rapport aux modèles par V-complexes standards.

- Tout d'abord, l'ensemble de base est une *préalgèbre* de Heyting, et non pas une algèbre. Ce point a déjà été souligné plusieurs fois.
- Les préalgèbres de Heyting ont une structure plus riche, ce qui permet de donner deux interprétation différentes à $\dot{\top} \mathbin{\dot{\wedge}} \dot{\top}$ et à $\dot{\top}$ et, *en même temps*, d'interpréter ε comme l'identité, au contraire de ce qu'il se passe à la section 2.4.3. Par exemple, $[\![\dot{\top} \mathbin{\dot{\wedge}} \dot{\top}]\!]^{\mathcal{M}}$ contiendra le séquent $\vdash \dot{\top} \mathbin{\dot{\wedge}} \dot{\top}$, mais ne contiendra pas le séquent $\vdash \dot{\top}$, qui n'a pas de preuve

neutre, et inversement pour $[\![\dot\top]\!]^{\mathcal{M}}$. Avoir $[\![\varepsilon(\dot\top \mathbin{\dot\wedge} \dot\top)]\!]^{\mathcal{M}} = [\![\dot\top \mathbin{\dot\wedge} \dot\top]\!]^{\mathcal{M}}$ ne pose aucun problème.

— Nous définissons $M_{T\to U}$ comme étant tout simplement $M_U^{M_T}$. Pour comparaison, les conditions des V-complexes $\langle t, f\rangle$ de type $T \to U$, à la définition 2.4.5 aussi bien qu'à la définition 3.2.15, demandent que la fonction f ait le type $M_U^{(T_T\times M_T)}$ et que, de plus, le résultat de l'application de f à un argument $\langle t', f'\rangle$ ait un résultat qui dépend de t, membre gauche auquel f a été associé dans le V-complexe d'origine. La définition que nous avons maintenant est manifestement bien plus simple, et ceci est possible grâce au choix, non pas d'une algèbre de Heyting ou de Boole pour M_o, mais d'une préalgèbre de Heyting.

— Le problème de l'imprédicativité ne se pose pas en tant que tel lors de la construction de $\mathcal{M}$. Il est traité par la supercohérence elle-même, qui définit les domaines M_T, par induction sur le type T. La supercohérence ne cherche pas à définir directement $[\![\ ]\!]^{\mathcal{M}}$, qui est générée de manière inductive et générique à partir des domaines M_T, de l'interprétation des constantes et de la (pré)algèbre spécifique à laquelle on spécialise la construction issue de la supercohérence.

Une question intéressante serait de savoir si la définition qui permet de démontrer la supercohérence de la logique d'ordre supérieur, tirée de [45] et qui a été reprise au début de cette section 4.2.3, est la *seule* définition possible. Serait-il envisageable de construire les domaines M_T d'une manière similaire aux sections 2.4.3 ou 3.2.4, de définir $\hat\varepsilon$ comme la seconde projection, et ainsi de suite, et d'obtenir la supercohérence tout de même ?

Il n'est que peu plausible qu'une telle construction, si elle est possible, appliquée cette fois-ci à l'algèbre de Heyting des contextes de la définition 3.2.3, nous permette de démontrer directement l'admissibilité des coupures, telle qu'elle est faite à la section 2.4.3, par exemple. Il serait en effet très difficile, voire impossible, de prouver l'adéquation $\lfloor A\rfloor \leq [\![A]\!] \leq \lceil A\rceil$, correspondant au lemme 3.2.8. En effet, la supercohérence ne permettrait que d'obtenir $\lfloor B\rfloor \leq [\![A]\!] \leq \lceil B\rceil$, pour un B déconnecté de A. Cette déconnexion n'est pas un problème dans le cadre de la préalgèbre de Heyting de la définition 4.2.9, car elle permet une indirection supplémentaire, qui est ensuite mise à profit par la définition 4.2.11 pour retrouver A en conclusion du séquent. Des efforts supplémentaires sont donc à envisager ; mentionnons, par exemple, que la supercohérence ne se prononce que sur les symboles de prédicat, qui sont sans structure interne, pour le moins tant que $\mathcal{RE}$ est vide.

La construction discutée dans ces trois paragraphes s'appliquerait aussi à la préalgèbre des séquents, et à d'autres telles que celles des candidats de réductibilité forts ou faibles. Observer les structures ainsi générées pourrait être riche d'enseignements.

$$\frac{}{\vdash A, A^\perp}\,\text{ax} \qquad \frac{\vdash A, \Delta_1 \qquad \vdash A^\perp, \Delta_2}{\vdash \Delta_1, \Delta_2}\,\text{cut} \qquad \frac{\vdash A, \Delta \qquad A \equiv B}{\vdash B, \Delta}\,\text{conv}$$

$$\frac{\vdash A, A, \Delta}{\vdash A, \Delta}\,\text{contr} \qquad \frac{\vdash \Delta}{\vdash A, \Delta}\,\text{weak}$$

$$\frac{}{\vdash \top}\,\top \qquad \text{(pas de règle } \bot\text{)}$$

$$\frac{\vdash A, \Delta_1 \qquad \vdash B, \Delta_2}{\vdash A \wedge B, \Delta_1, \Delta_2}\,\wedge \qquad \frac{\vdash A, B, \Delta}{\vdash A \vee B, \Delta}\,\vee$$

$$\frac{\vdash A[t/x], \Delta}{\vdash \exists x.A, \Delta}\,\exists \qquad \frac{\vdash A, \Delta}{\vdash \forall x.A, \Delta}\,\forall^*$$

Figure 4.5 – Calcul des séquents classique modulo théorie monolatéral

4.2.4 Algèbres de séquents en calcul des séquents

Maintenant que nous savons construire des algèbres de séquent en déduction naturelle (modulo théorie), nous pouvons nous demander s'il n'est pas possible de faire un travail similaire en calcul des séquents (modulo théorie). Ceci demande un travail supplémentaire conséquent, car, comme vu à la section 4.1, la notion de coupure en déduction naturelle est très différente de celle de coupure en calcul des séquents.

De plus, nous allons effectuer ce travail dans le cadre du calcul des séquents classique, qui offre plus de symétrie, et permet notamment de travailler avec un calcul des séquents monolatéral et une négation qui est un opérateur involutif. Ce calcul des séquents est présenté à la figure 4.5, où, comme d'habitude, x est supposé frais dans la règle $\forall$. Obtenir de tels résultats dans le cadre du calcul des séquents intuitionniste est un travail qui reste à faire.

Le calcul des séquents de la figure 4.5 est présenté sous sa forme multiplicative, ce qui rend plus aisées les définitions ci-dessous. La logique linéaire et les réseaux de preuve ont en effet influencé nombre des définitions ci-dessous, par exemple les notions de pôle et celle de “mise en relation” entre deux séquents pointés. Un calcul des séquents multiplicatif est bien plus facile à manipuler dans ce cadre. Le travail décrit dans cette section peut aussi être rapproché de celui d'Urban [135, 136] ; toutefois, ce dernier introduit des termes de preuves, ce qui complique grandement son approche.

Pour définir correctement le calcul des séquents de la figure 4.5, nous avons besoin de formules en forme normale négative [110, 59], ce qui est possible avec la définition suivante de la négation comme *opérateur* plutôt que comme connecteur à part entière. Notons que nous avons maintenant des formules atomiques positives, a, et négatives, $\bar{a}$, exactement comme les littéraux positifs et négatifs de la résolution.

Définition 4.2.16 (Négation involutive)**.** *La négation est une fonction* $(.)^\perp$ *sur les formules, définie de la manière suivante :*

$$\begin{array}{llll} a^\perp := \bar{a}, & \perp^\perp := \top, & (A \wedge B)^\perp := A^\perp \vee B^\perp, & (\forall x.A)^\perp := \exists x.A^\perp, \\ \bar{a}^\perp := a, & \top^\perp := \perp, & (A \vee B)^\perp := A^\perp \wedge B^\perp, & (\exists x.A)^\perp := \forall x.A^\perp. \end{array}$$

L'implication est aussi un connecteur absent, comme il est d'usage dans les calculs des séquents monolatéraux avec négation involutive. $A \Rightarrow B$ est alors remplacé par $A^\perp \vee B$.

La sémantique que nous considérons est une restriction des préalgèbres de Heyting, puisque nous sommes en logique classique. Il s'agira des préalgèbres de Boole, dont on peut trouver un exemple d'utilisation dans les travaux de Krivine [95].

Définition 4.2.17 (Préalgèbre de Boole)**.** *Une préalgèbre de Boole est une structure* $\langle \mathcal{B}, \leq, \top, \perp, \wedge, \vee, (.)^\perp, \forall, \exists \rangle$ *qui respecte tous les axiomes des algèbres de Boole (définition 2.1.3), mais où l'ordre* $\leq$ *est remplacé par un* préordre*.*

- $\top, \wedge$ *et* $\forall$ *forment les bornes inférieures nullaire, binaire et arbitraire (quand elle existe), respectivement ;*
- *tandis que* $\perp, \wedge$ *et* $\exists$ *forment les bornes supérieures nullaire, binaire et arbitraire (quand elle existe), respectivement ;*
- $\wedge$ *et* $\vee$ *sont distributifs l'un par rapport à l'autre ;*
- $(.)^\perp$ *est l'opération de complément ; elle vérifie* $\top \leq a \vee a^\perp$ *et* $a \wedge a^\perp \leq \perp$*.*

De plus, nous imposons $a^{\perp\perp} = a$*.*

La condition $a^{\perp\perp} = a$ est une condition plus restrictive que le minimum requis, ce qui rend nos préalgèbres de Boole strictement différentes des préalgèbres de Heyting.

Nous désirons, comme à la section 4.2.1 séparer au maximum le traitement de la réécriture du traitement des règles d'inférence et pour cela, nous introduisons une notion de supercohérence adaptée à logique classique. La notion de supercohérence [45] se spécialise naturellement aux préalgèbres de Boole ; la définition 4.2.5 peut être reprise au mot près, en remplaçant “préalgèbre de Heyting” par “préalgèbre de Boole” ; on obtient aussi que si un système de réécriture est supercohérent, alors il est classiquement supercohérent. En particulier, il peut exister des systèmes de réécriture classiquement supercohérents mais pas supercohérents.

Définition 4.2.18 (Supercohérence classique)**.** *Un système de réécriture* $\mathcal{RE}$ *est* classiquement supercohérent *si, et seulement si, pour toute préalgèbre* *de Boole Ω, il existe un modèle à valeurs dans Ω tel que, pour toutes formules $A \equiv B$ et pour toute valuation φ, $[\![A]\!]_\varphi = [\![B]\!]_\varphi$.*

Afin de définir une notion similaire à l'extraction de la définition 4.2.11 et au "branchement de contexte" de la conjecture 4.2.10 et du lemme 4.2.6 d'adéquation (où le branchement est représenté par θ), nous devons définir précisément la notion d'interaction (le "branchement") entre séquents. En effet, il faut savoir distinguer un endroit précis, qui sera le lieu de l'interaction. Ce lieu et cette interaction, assez naturels en déduction naturelle (la conclusion d'un séquent se branche sur l'hypothèse correspondante d'un autre), sont l'objet des définitions suivantes.

Définition 4.2.19 (Séquents pointés)**.** *Un* séquent pointé *est de la forme $\vdash A^\circ, \Delta$. A° est appelée la formule* distinguée *; sa position dans Δ est libre, et Δ ne contient aucune autre formule distinguée. Ax° est l'ensemble des séquents pointés* axiomatiques*, de la forme $\vdash A^\circ, A^\perp$.*

Si $\vdash A^\circ, \Delta_A$ et $\vdash B^\circ, \Delta_B$ sont deux séquents pointés tels que $A \equiv B^\perp$, alors l'opération $\star$, la coupure*, entre ces deux séquents est*

$$(\vdash A^\circ, \Delta_A) \ \star \ (\vdash B^\circ, \Delta_B) \quad := \ \vdash \Delta_A, \Delta_B.$$

Comme nous allons construire une algèbre composée d'ensembles de séquents, il nous faut définir en tout premier lieu l'opération $(.)^\perp$ sur ceux-ci. Cette opérateur d'orthogonalité est au cœur de la construction.

Définition 4.2.20 (Orthogonal)**.** *Soient $\perp\!\!\!\perp$ (le* pôle*), l'ensemble des séquents qui ont une preuve sans coupure, et $\perp\!\!\!\perp^\circ$ l'ensemble des séquents pointés qui ont une preuve sans coupure. Soit X, un ensemble de séquents pointés ; nous définissons $X^\perp$, l'*orthogonal *de X, comme*

$$\begin{aligned} X^\perp \quad := \quad \{ \vdash A^\circ, \Delta_A \mid & \textit{ pour tout } (\vdash B^\circ, \Delta_B) \in X \textit{ tel que } A \equiv B^\perp, \\ & (\vdash A^\circ, \Delta_A) \ \star \ (\vdash B^\circ, \Delta_B) \in \perp\!\!\!\perp \}. \end{aligned}$$

L'opérateur $()^\perp$ est antimonotone et génère un opérateur de clôture $()^{\perp\perp}$. $X^\perp$ est en particulier un ensemble toujours clos. On appellera *comportement* un tel ensemble, qui contient donc uniquement des séquents pointés. Cet opérateur nous permet de définir les autres opérateurs ; nous présentons ici le cas de $\wedge$. Pour deux comportements X et Y, nous posons $X \wedge Y = ((X.Y) \cup Ax^\circ)^{\perp\perp}$, où $X.Y$ est

$$\{\vdash (A \wedge B)^\circ, \Delta_A, \Delta_B \mid (\vdash A^\circ, \Delta_A) \in X \text{ et } (\vdash B^\circ, \Delta_B) \in Y\}.$$

Il est impossible de définir $X \wedge Y$ comme étant directement $X.Y$, car d'autres séquents doivent être ajoutés en adjoignant les axiomes, puis en prenant la clôture de l'ensemble ainsi obtenu. Nous pouvons faire le parallèle avec la définition de $\wedge$ de la définition 4.2.9 précédente : les séquents ayant une preuve “neutre” (adjonction des axiomes, puis clôture) doivent aussi faire partie de $a \wedge b$.

Tout ceci nous permet de définir une préalgèbre de Boole $\mathcal{D}$, dont l'ensemble de base est

$$\{X \mid X \text{ est un comportement, et } Ax^\circ \subseteq X \subseteq \perp\!\!\!\perp^\circ\}.$$

Une fois que nous avons vérifié la stabilité de $\mathcal{D}$ par tous les opérateurs sémantiques ($\mathcal{D}$ est donc bien une préalgèbre de Boole), nous pouvons démontrer le lemme d'adéquation, à condition de supposer avoir une interprétation dans $\mathcal{D}$, qui, lorsque l'on se place en Déduction modulo théorie, doit respecter le système de réécriture et peut être fournie par la supercohérence.

Lemme 4.2.21 (Adéquation)**.** *Soit $\mathcal{RE}$ un système de réécriture. Soit un séquent $\vdash A_1, \cdots, A_n$ prouvable en calcul des séquents classique modulo théorie. Soient σ, une substitution qui n'entraîne pas de capture de variable par $A_1, \cdots, A_n$, φ, une valuation et $[\![\ ]\!]$, une interprétation dans un modèle de $\mathcal{RE}$ à valeurs dans $\mathcal{D}$.*

Si, pour tout i, $\vdash ((A_i\sigma)^\perp)^\circ, \Delta_i \in [\![A_i]\!]_\varphi^\perp$, alors $\vdash \Delta_1, \cdots, \Delta_n \in \perp\!\!\!\perp$.

On remarquera la similitude avec la conjecture 4.2.10 de la section 4.2.2 et avec le lemme 4.2.6, à la différence que, dans le cas présent, l'interaction se produit à droite du séquent. Par conséquent, il faut considérer les orthogonaux. La complétude et l'élimination des coupures suivent.

Corollaire 4.2.22 (Élimination des coupures)**.** *Si le séquent $\vdash A_1, \cdots, A_n$ est prouvable dans le calcul des séquents de la figure 4.5 modulo un système de réécriture classiquement supercohérent $\mathcal{RE}$, alors il est prouvable sans coupure.*

Par supercohérence (classique), nous avons une interprétation $[\![\ ]\!]$ dans la préalgèbre $\mathcal{D}$ qui est un modèle de $\mathcal{RE}$. Puisque, par hypothèse, Ax° fait partie de tous les comportements, on a $\vdash (A_i^\perp)^\circ, A_i \in [\![A_i]\!]_\varphi^\perp$. Cela nous permet d'appliquer le lemme 4.2.21, en choisissant l'identité pour φ et σ, pour conclure $\vdash A_1, \cdots, A_n \in \perp\!\!\!\perp$.

Enfin, au lieu de démontrer le lemme d'adéquation, nous pouvons aussi procéder exactement comme à la section 4.2.2 et extraire de cette préalgèbre de Boole une algèbre de Boole.

Définition 4.2.23 (Extraction de contextes)**.** *Soit A une formule. $[A]$ est défini comme l'ensemble des contextes $A_1, \cdots, A_n$ qui vérifient la condition suivante.*

Pour toute valuation φ, toute substitution σ, tout contexte Δ tel que $\vdash \Delta, ((A\sigma)^\perp)^\circ \in [\![A]\!]_\varphi^\perp$ et tous contextes Δ_i tels que $\vdash \Delta_i, ((A_i\sigma)^\perp)^\circ \in [\![A_i]\!]_\varphi^\perp$ pour tout i, alors on a

$$\vdash \Delta_1, \cdots, \Delta_n, \Delta \in \perp\!\!\!\perp.$$

Cette définition rappelle directement le lemme 4.2.21, à la différence que l'on a choisi de mettre en avant une formule, A, afin d'en extraire les contextes associés. Si l'on souhaite une définition encore moins symétrique, on peut demander que la condition à respecter soit $\vdash \Delta_1, \cdots, \Delta_n, (A\sigma)^\circ \in [\![A]\!]_\varphi^\perp$. La démonstration de l'équivalence entre ces deux conditions est laissée au lecteur.

Il est ensuite possible de démontrer l'admissibilité de la règle de coupure sans passer le lemme 4.2.21, comme à la section 4.2.2. Si l'on souhaite passer par le lemme 4.2.21, alors il suffit de considérer la substitution σ vide et de remarquer que tous les éléments de la préalgèbre $\mathcal{D}$, donc en particulier $[\![A_i]\!]_\varphi^\perp$, contiennent, par définition, tous les axiomes distingués, donc, en particulier, $\vdash (A_i^\perp)^\circ, A_i$. Nous pouvons donc choisir $\Delta_i = A_i$, ce qui permet de conclure $\vdash A_1, \cdots, A_n \in \perp\!\!\!\perp$; ce séquent a donc une preuve sans coupure.

4.3 Algorithmes d'élimination des coupures

Dans cette dernière partie, nous examinons les algorithmes qui sont à l'oeuvre derrière les théorèmes d'admissibilité des coupures, que ceux-ci soient prouvés de manière sémantique par la méthode des tableaux, à travers des structures algébriques ou par les méthodes de la section précédente.

4.3.1 Contenu calculatoire des preuves d'admissibilité

Les preuves d'admissibilité des coupures produisent, c'est un pléonasme, des preuves sans coupure à partir de la preuve (avec ou sans coupures) donnée en entrée.

À la section 2.6, nous avons vu comment les preuves de complétude peuvent être rendues constructives, et nous obtenons alors un théorème de complétude qui reconstruit une preuve à partir de zéro. Cette reconstruction ne s'appuie en particulier pas sur la preuve fournie en entrée : la procédure de recherche systématique de preuve, qui est le contenu calculatoire du théorème 2.6.1, s'applique quoi qu'il arrive dans l'ordre qui a été déterminé à l'avance.

Si nous voulons analyser le contenu calculatoire des démonstration sémantiques d'élimination des coupures plus finement, alors nous avons besoin de plus d'information dans l'hypothèse du théorème 2.6.1, “toutes les structures de Kripke telles que $w \Vdash A_1, \cdots, w \Vdash A_n$ et $w \nVdash B_1, \cdots, w \nVdash B_m$ explosent”. C'est cette hypothèse qui nous assure l'existence d'une contradiction sur la branche, et donc la terminaison de l'algorithme. Cette hypothèse est donnée par le théorème 2.5.2 de correction qui, lui, s'appuie sur la preuve d'origine. Le théorème de correction étant une induction simple, la preuve d'origine est reflétée dans la justification sémantique, qui, elle-même, sert ensuite pour garantir la terminaison de la recherche de preuve. Cependant, la structure de la justification sémantique elle-même n'est pas clairement isomorphe à la preuve d'origine, car celle-ci fait appel, à chaque étape et de manière implicite, aux propriétés de la structure de Kripke.

Pour progresser, une piste serait de formaliser les théorèmes de correction et de complétude, un travail qui a en partie été fait [74], toutefois dans le cadre de la déduction naturelle, ce qui éloigne de la méthode des tableaux et du calcul des séquents qui est le nôtre aux chapitres 2 et 3. Les fragments logiques considérés sont des sous-ensembles des logiques considérées dans ce manuscrit, notamment le fragment implication/quantification universelle, qui se marient bien avec les structures de Kripke, ce qui fait écho aux travaux de *normalisation par évaluation* [32, 31, 2, 10, 41]. La disjonction, qui correspond aux types somme, est difficile à traiter dans les structures de Kripke : si on a $w \Vdash A \vee B$, alors on doit forcément avoir $w \Vdash A$ ou $w \Vdash B$, ce qui impose, d'un point de vue logique, de compléter la théorie (ce qui est très peu constructif) ou de généraliser les structures de Kripke [86, 87].

Comme nous l'avons vu, il existe un autre type de sémantique qui permet de démontrer l'admissibilité des coupures : les algèbres de Heyting. De plus, c'est la seule sémantique qui, en l'état actuel de nos connaissances, permet de passer à la logique d'ordre supérieur. Un autre avantage est que les preuves de complétude forte sont déjà constructives ; il n'y a aucun besoin de travail de constructivisation supplémentaire. Enfin, parmi les bonnes propriétés des algèbres de Heyting, le traitement de la disjonction, définie par l'opération de borne supérieure, est tout ce qu'il y a de plus facile. À ce dernier sujet, remarquons tout de même que la construction d'algèbre de Heyting universelle, par exemple à la définition 3.2.3, ne définit pas la borne supérieure comme une opération primitive, mais comme la borne inférieure de tous les majorants. L'interprétation du cas de la disjonction sera donc légèrement obscurcie.

C'est ce travail que nous nous proposons de décrire maintenant : étudier le contenu calculatoire des preuves d'admissibilité des coupures qui font appel aux algèbres de Heyting. La première étape est de transcrire les preuves du chapitre 3 au cadre de la déduction naturelle, pour laquelle nous disposons de termes de preuve et d'une procédure d'élimination des coupures standards. Les preuves en forme normale sont définies à la figure 4.4 précédente.

La définition de l'algèbre de Heyting se fait, sans surprise, en partant d'une extraction de contextes tout à fait identique à celle de la définition 3.2.2 du chapitre 3.

Définition 4.3.1 (Extraction)**.** *Soit A une formule close. Nous appelons* extraction de A, *et nous notons* $\lceil A \rceil$, *l'ensemble des contextes*

$$\lceil A \rceil := \{\Gamma \mid \Gamma \vdash^* A\},$$

où la relation de prouvabilité $\vdash^*$ *est celle de la figure 4.4.*

Passer à la prouvabilité en déduction naturelle nous donne toujours une algèbre de Heyting, qui est le plus petit ensemble Ω contenant les extractions et clos par intersections arbitraires. Les opérateurs sémantiques sont définis exactement comme à la section 3.2.2, l'ordre est l'inclusion ensembliste. La démonstration que nous sommes bien en présence d'une algèbre de Heyting conserve la structure globale de la démonstration du lemme 3.2.8 ; c'est-à-dire que, pour chacune des propriétés à respecter, l'une des règles de la figure 4.4 est utilisée, mais cette démonstration demande tout de même un certain travail technique dans les détails. Ceci montre un léger défaut d'adaptation des algèbres de Heyting à la déduction naturelle.

Comme nous ne sommes pas en Déduction modulo théorie, nous avons la possibilité de définir manuellement la fonction d'interprétation, toujours comme à la section 3.2.2, à la différence que, pour plus de maniabilité, nous autorisons ici les termes avec variables libres.

— Le domaine est syntaxique, composé de tous les termes. Les valuations seront donc en même temps des substitutions et, pour tout t, nous aurons $[\![t]\!]_\varphi = t\varphi$.

— Les formules atomiques A sont interprétées par n'importe quel élément de Ω tel que $\lfloor A\varphi \rfloor \subseteq [\![A]\!]_\varphi \subseteq \lceil A\varphi \rceil$. Les deux choix canoniques sont $\lfloor A\varphi \rfloor$ et $\lceil A\varphi \rceil$.

Rappelons que $\lfloor A \rfloor$ est défini comme étant le plus petit élément de l'algèbre Ω contenant (le contexte) A, et est également caractérisé plus simplement par

$$\lfloor A \rfloor = \{\Gamma \mid \text{si } A \vdash^* B, \text{ alors } \Gamma \vdash^* B\}$$

En particulier, si $\Gamma \vdash_{ne} A$ est dérivable, $\Gamma \in \lfloor A \rfloor$. On passe ensuite au résultat principal.

Lemme 4.3.2. *Pour toute formule A et toute substitution/valuation φ,*

$$\lfloor A\varphi \rfloor \leq [\![A]\!]_\varphi \leq \lceil A\varphi \rceil.$$

Nous dérivons ensuite le théorème de complétude forte, puis l'admissibilité des coupures de la manière habituelle.

Théorème 4.3.3 (Complétude forte)**.** *Soit $\Gamma \vdash A$ un séquent. Si, pour toute algèbre de Heyting, toute interprétation et toute valuation φ, $\bigwedge [\![\Gamma]\!]_\varphi \leq [\![A]\!]_\varphi$, alors on peut dériver*

$$\Gamma \vdash^* A.$$

Ce théorème se démontre en choisissant l'algèbre de Heyting et le modèle que nous venons juste de construire, et φ, la fonction identité. Nous avons alors

$$\Gamma \in \bigwedge \lfloor \Gamma\varphi \rfloor \subseteq \bigwedge [\![\Gamma]\!]_\varphi \subseteq [\![A]\!]_\varphi \subseteq \lceil A\varphi \rceil$$

par, successivement, définition et des propriétés basiques, le lemme 4.3.2, hypothèse et, une seconde fois, le lemme 4.3.2. Ici, nous utilisons les notations $\bigwedge \lfloor \Gamma \rfloor := \bigwedge_{C\in\Gamma} \lfloor C \rfloor$ et $\bigwedge [\![\Gamma]\!] := \bigwedge_{C\in\Gamma} [\![C]\!]$.

Théorème 4.3.4 (Correction)**.** *Soit $\Gamma \vdash A$ un séquent. S'il est prouvable en déduction naturelle (avec coupures), alors, pour toute algèbre de Heyting, toute interprétation et toute valuation φ,*

$$\bigwedge [\![\Gamma]\!]_\varphi \leq [\![A]\!]_\varphi.$$

Corollaire 4.3.5 (Élimination des coupures)**.** *Soit $\Gamma \vdash A$ un séquent prouvable en déduction naturelle (avec coupures). Alors, $\Gamma \vdash^* A$ est prouvable dans le système de la figure 4.4 ; autrement dit, il existe une preuve de $\Gamma \vdash A$ sans coupure.*

Ce plan, une fois discuté et démontré sur papier, a fait l'objet d'une formalisation en Coq pour le fragment propositionnel. Ce fragment permet d'éviter les problèmes, habituels, de gestion de variables (capture, domaine des termes ouverts), qui rendraient l'analyse des résultats encore plus difficile d'accès. Le but de cette formalisation est de regarder l'algorithme à l'œuvre.

On peut *extraire un programme*. Cette étape est rendue complexe, voire impossible, sauf à volontairement rendre Coq incohérent, du fait de l'imprédicativité nécessaire à la définition de l'algèbre Ω. Le nœud du problème est la clôture par intersections arbitraires.

On peut aussi travailler par *observation interne* à Coq. Il est possible de demander à Coq de faire le travail lui-même, et même de l'afficher, même s'il reste impossible d'en extraire la composante qui nous intéresse.

Dans les deux cas, nous obtenons un algorithme qui normalise *effectivement* les preuves de la manière attendue. Cependant, l'intrication des différents niveaux est telle qu'il est, pour l'instant, impossible de comprendre ce qu'il se passe dans les détails.

La normalisation des termes de preuves se fait par réflexion dans la sémantique (correction), puis réification à partir de celle-ci (complétude). On obtient donc un algorithme de *normalisation par évaluation* dont la spécificité est qu'il fait appel aux algèbres de Heyting. Celui-ci effectue une réduction des coupures (β-réduction) combinée à une expansion des axiomes (η-expansion), qui provient du fait que le théorème d'admissibilité des coupures fonctionne par induction sur les types/la taille de la formule. Ainsi, la preuve (en une ligne) de $A \Rightarrow B \vdash A \Rightarrow B$ est transformée en la preuve ci-dessous.

$$\dfrac{\dfrac{\dfrac{}{A \Rightarrow B, A \vdash A \Rightarrow B}\,\text{ax} \qquad \dfrac{}{A \Rightarrow B, A \vdash A}\,\text{ax}}{A \Rightarrow B, A \vdash B}\,{\Rightarrow_E}}{A \Rightarrow B \vdash A \Rightarrow B}\,{\Rightarrow_I}$$

Bien entendu, si on applique une nouvelle fois l'algorithme de normalisation sur la preuve précédente, nous obtenons la même preuve. Le cas de la disjonction est cependant différent et mérite que l'on s'y attarde. Pour la preuve élémentaire de $A \vee B \vdash A \vee B$, nous obtenons encore une expansion de l'axiome.

$$\dfrac{\dfrac{}{A \vee B \vdash A \vee B}\,\text{ax} \qquad \dfrac{\dfrac{}{A \vee B, A \vdash A}\,\text{ax}}{A \vee B, A \vdash A \vee B}\,{\vee_{I1}} \qquad \dfrac{\dfrac{}{A \vee B, B \vdash B}\,\text{ax}}{A \vee B, B \vdash A \vee B}\,{\vee_{I2}}}{A \vee B \vdash A \vee B}\,{\vee_E}$$

Mais une seconde application de l'algorithme à cette preuve, que nous qualifierons d'intermédiaire ci-dessous, donne une preuve plus complexe.

$$\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{}{A \vee B, A, A \vee B \vdash A}\,\text{ax}}{A \vee B, A, A \vee B \vdash A \vee B}\,{\vee_{I1}}}{A \vee B, A \vdash (A \vee B) \Rightarrow (A \vee B)}\,{\Rightarrow_I} \qquad \dfrac{\dfrac{\dfrac{}{A \vee B, B, A \vee B \vdash B}\,\text{ax}}{A \vee B, B, A \vee B \vdash A \vee B}\,{\vee_{I2}}}{A \vee B, B \vdash (A \vee B) \Rightarrow (A \vee B)}\,{\Rightarrow_I} \qquad \dfrac{}{A \vee B \vdash A \vee B}\,\text{ax}}{A \vee B \vdash (A \vee B) \Rightarrow (A \vee B)}\,{\vee_E} \qquad \dfrac{}{A \vee B \vdash A \vee B}\,\text{ax}}{A \vee B \vdash A \vee B}\,{\Rightarrow_E}$$

Cette dernière preuve appelle plusieurs commentaires. Tout d'abord, si l'axiome $A \vee B \vdash A \vee B$ de la prémisse principale de la règle $\vee_E$ dans la preuve intermédiaire a été η-expansé, nous n'en retrouvons aucune trace dans la preuve finale, ce qui est le comportement attendu, car cela constituerait une coupure et elle a donc été réduite. Deuxièmement, nous observons une abstraction par rapport au contexte $A \vee B$, à la racine de la preuve ($\Rightarrow_E$). Ce contexte est transporté au plus près possible des feuilles ($\Rightarrow_I$), une fois que la règle d'élimination du symbole $\vee$ a été appliquée.

Ce dernier motif est une instance d'une *coupure commutative* : tout ce qui sépare le couple $\Rightarrow_E$ / $\Rightarrow_I$ d'être une coupure est la règle $\vee_E$, et il est possible de *faire commuter* ces règles pour obtenir une coupure, que l'on réduit ensuite. La preuve finale que nous avons obtenue est bien sans coupure, mais il faudrait faire plus pour obtenir de nouveau la preuve intermédiaire.

Nous pouvons définir les preuves sans coupure commutative comme des preuves sans coupure où, de plus, les règles $\vee_E/\exists_E$ sont placées le plus bas possible, autrement dit, suivies (pour l'ordre de sous-arbre) par une règle d'introduction ou une autre règle $\vee_E/\exists_E$. Les relations $\vdash^*$ et $\vdash_{ne}$ de la figure 4.4 doivent ainsi être modifiées de la manière de la figure 4.6.

Enfin, il s'avère que les preuves sans coupure en calcul des séquents correspondent aux preuves en déduction naturelle sans coupure commutative. Ainsi, il est possible de se débarrasser des coupures, y compris commutatives, en passant par le théorème d'élimination des coupures du calcul des séquents et un va-et-vient avec la déduction naturelle. C'est la stratégie qui

$$\frac{\Gamma, A \vdash^* C \qquad \Gamma, B \vdash^* C \qquad \Gamma \vdash_{ne} A \vee B}{\Gamma \vdash^* C} \vee_E \qquad \frac{\Gamma, A \vdash^* C \qquad \Gamma \vdash_{ne} \exists x A}{\Gamma \vdash^* C} \exists_E$$

FIGURE 4.6 – Règles de dérivation empêchant les coupures commutatives

a été employée par Gentzen [62], mais cela reste peu satisfaisant lorsque l'on veut analyser le comportement d'un algorithme de normalisation des preuves en déduction naturelle. Une formalisation directe, dans le système de dérivation de la figure 4.6, reste encore à faire.

4.3.2 Preuves de normalisation faible

Les preuves de normalisation faible ne contiennent pas directement un algorithme permettant de trouver la preuve en forme normale ; elles se bornent à affirmer que le terme de preuve correspondant à la preuve de $\Gamma \vdash A$ donnée en entrée est faiblement normalisant. Nous obtenons seulement l'existence d'une suite de réductions qui mène à une preuve sans coupure. Cependant, nous savons que la forme normale (preuve sans coupure) résultante sera un réduit du terme de preuve (de la preuve initiale), ce qui est une information supplémentaire par rapport à la section 4.3.1.

Ce léger contretemps, qui énonce seulement la possibilité de normaliser le terme de preuve, n'est pas spécifique à la normalisation faible ; il se rencontre aussi avec la normalisation forte. Dans les deux cas, nous devons définir une *stratégie* de mise en forme normale qui réduit les termes de preuve en appliquant les règles de la figure 4.3 selon un algorithme précis. Cette stratégie peut être aussi être extraite de la preuve par un lemme d'échappement [121]. La normalisation forte offre la garantie additionnelle que toute stratégie est correcte.

Afin de demander aux preuves de normalisation faible d'extraire une preuve sans coupure, nous pouvons tenter d'extraire une algèbre de Heyting, de la même manière qu'aux sections 4.2.2 et 4.2.4. L'idée la plus directe est de définir l'extraction de contextes de la manière suivante.

Définition 4.3.6 (Extraction faible de contextes). *Soit A une formule. L'extraction $[A]$ est définie comme l'ensemble des contextes $A_1, \cdots, A_n$ tels qu'il existe des variables $\alpha_1, \cdots, \alpha_n$ et un terme de preuve π tels que*

- *$\alpha_1 : A_1, \cdots, \alpha_n : A_n \vdash \pi : A$;*
- *pour toute valuation φ, toute substitution de termes σ et toute substitution de termes de preuve θ tels que $\theta(\alpha_i) \in [\![A_i]\!]_\varphi$, on a $\pi\sigma\theta \in [\![A]\!]_\varphi$.*

Cela permet de construire l'algèbre de Heyting contenant toutes les extractions (sans besoin de clôture par intersections arbitraires), de l'ordonner par la relation d'inclusion et de définir immédiatement des opérateurs, à la façon du lemme 4.2.12.

Définition 4.3.7 (Opérateurs sur l'algèbre des extractions faibles)**.** *On définit les opérateurs suivants sur Ω, l'ensemble des extractions faibles :*
- $\top := [\top]$,
- $\bot := [\bot]$,
- $[A] \wedge [B] := [A \wedge B]$,
- $[A] \vee [B] := [A \vee B]$,
- $[A] \Rightarrow [B] := [A \Rightarrow B]$,
- $\forall\{A[t/x] \mid t \in \mathcal{T}\} := [\forall x A]$,
- $\exists\{A[t/x] \mid t \in \mathcal{T}\} := [\exists x A]$.

Nous démontrons ensuite que les opérateurs que nous venons de définir sont bien les opérateurs correspondant d'une algèbre de Heyting ; par rapport au lemme 4.2.12, qui s'appuyait sur une algèbre déjà construite, nous avons donc procédé à rebours, car nos opérations sont données a priori. Notons aussi que cette algèbre n'est pas nécessairement complète, car nous n'avons pas imposé sa clôture par intersections arbitraires. Mais elle est ce qu'on appelle *définitionnellement complète* [133], ce qui veut dire qu'elle contient juste assez de bornes supérieures et inférieures arbitraires pour interpréter les connecteurs et les quantificateurs.

Nous pouvons ensuite construire le modèle, extrait du modèle construit par la définition 4.2.5 à partir de la préalgèbre $\mathcal{WN}$, des candidats de réductibilité faibles. La réalisation technique est similaire à celle de la définition 4.2.13 et nous permet de conclure.

Théorème 4.3.8. *Si le séquent $\Gamma \vdash A$ a une preuve, alors il en existe une preuve faiblement normalisante.*

Ce n'est pas exactement l'objectif recherché, car nous cherchons à produire des preuves en forme normale. À cette fin, un raffinement de la définition 4.3.6 est nécessaire.

Définition 4.3.9 (Extraction forte de contextes)**.** *Soit A une formule. L'extraction $[A]$ est définie comme l'ensemble des contextes $A_1, \cdots, A_n$ tels qu'il existe des variables $\alpha_1, \cdots, \alpha_n$ et un terme de preuve π* en forme normale *tels que :*
- $\alpha_1 : A_1, \cdots, \alpha_n : A_n \vdash \pi : A$ *;*
- *pour toute valuation φ, toute substitution de termes σ et toute substitution de termes de preuve θ tels que $\theta(\alpha_i) \in [\![A_i]\!]_\varphi$, on a $\pi\sigma\theta \in [\![A]\!]_\varphi$.*

La préalgèbre $\mathcal{WN}$ n'est pas adaptée à cette nouvelle définition ; il faut se limiter au strict minimum, c'est-à-dire au plus petit ensemble $\mathcal{B}$ de candidats de réductibilité, clos par les opérations de la définition 4.2.4, qui forme aussi une préalgèbre de Heyting, pour laquelle on obtient un modèle encore une fois par supercohérence (définition 4.2.5). On démontre aussi la stabilité par réduction des candidats.

Lemme 4.3.10 (Stabilité par réduction)**.** *Soit $E \in \mathcal{B}$. Pour tout $\pi \in E$, si $\pi \rhd^* \pi'$, alors $\pi' \in E$.*

Ceci nous permet de modifier l'énoncé et la preuve du théorème 4.3.8, pour obtenir une preuve en forme normale.

Théorème 4.3.11. *Si le séquent $\Gamma \vdash A$ a une preuve, alors il en existe une preuve qui est en forme normale.*

Le contenu calculatoire de cette preuve n'a pas encore été analysé.

Chapitre 5

Conclusion

5.1 Synthèse

Dans ce manuscrit, nous avons pu voir que les démonstrations usuelles de complétude du calcul des séquents, classique ou intuitionniste, donnaient lieu, après un travail d'épuration et d'amélioration, à des méthodes d'élimination des coupures. Deux types de démonstration de complétude ont été considérées.

Le premier type de démonstration se fonde sur la complétion de théories de Henkin. En raffinant le cahier des charges minimal de ces démonstrations, nous sommes arrivés à des méthodes de tableaux et à des preuves de complétude de ceux-ci. Un effort supplémentaire, notamment dans le cas de tableaux avec variables libres, nous a permis de traduire ces preuves de tableaux en preuves du calcul des séquents *sans coupure*, arrivant, par là même, à un théorème d'élimination des coupures en calcul des séquents dont le contenu calculatoire est essentiellement un algorithme exhaustif de recherche de preuve dans la méthode des tableaux, appelée une procédure complète.

Le second type de démonstration est algébrique, pour lequel nous avons pris comme point de départ l'algèbre de prouvabilité de Lindenbaum. Cette construction ne fonctionne pas sans la règle de coupure, mais on peut former des algèbres de *contextes* pour lesquelles la notion de validité est incluse dans celle de prouvabilité sans coupure et, en réalité, égale à celle-ci a posteriori, c'est-à-dire une fois le théorème d'élimination des coupures démontré. Ces démonstrations algébriques s'étendent à un spectre plus large de logiques que les démonstrations qui passent par la méthode des tableaux ; de plus elles sont constructives et permettent de générer des algorithme d'élimination des coupures dans le style de la normalisation par évaluation. Enfin, nous avons vu qu'il est possible, avec ces méthodes, d'aller au-delà des preuves d'admissibilité des coupures et de construire des algèbres à partir de termes de preuve et de séquents.

Toutes ces méthodes s'appliquent de manière générique à la Déduction modulo théorie, sans qu'il y ait besoin de fixer, jusqu'à un certain point, de condition sur le système de réécriture.

Dans le cas de la méthode des tableaux, on peut retarder cette question jusqu'à la toute fin, lorsque l'on doit, finalement, construire un modèle à partir d'une valuation partielle. Nous avons été amenés, en particulier, à développer des procédures systématiques d'exploration de tableaux, indépendantes du système de réécriture concret, qui ont une stratégie de réécriture très proche des stratégies réellement implémentées. Ces procédures ne dépendent donc pas de la possibilité de construire un modèle ou non et, dans le cas où cette dernière étape échouerait, on aurait, au pire, une recherche de preuve incomplète.

Les méthodes algébriques d'élimination des coupures ou de normalisation peuvent, par exemple, déléguer l'étude du système de réécriture à la supercohérence et, en l'absence d'un tel modèle, on obtiendra, soit des termes de preuve qui ne normaliseront pas, soit l'impossibilité d'appliquer l'algorithme de normalisation par évaluation par absence de modèle.

Enfin, nous avons étudié les liens entre logique classique et intuitionniste, et notamment démontré que la supercohérence d'un système de réécriture n'était pas sensible à ce choix de logique.

5.2 Perspectives

Les travaux décrits dans ce manuscrit répondent à certaines questions ; mais comme dans tout travail de recherche, les réponses sont parfois partielles et amènent d'autres questions, plus nombreuses.

Tout d'abord, il est toujours possible, du fait de l'indécidabilité de l'élimination des coupures en Déduction modulo théorie [26], de généraliser les conditions permettant de construire un modèle.

La condition de positivité est probablement la plus intéressante à étudier, car elle permet d'ajouter des types (prédicats) inductifs à la logique. Plutôt que de caractériser les prédicats comme étant positifs ou négatifs, on peut tenter une catégorisation plus fine des instances de prédicats. L'ajout de règles de réécritures positives au système de réécriture permettant l'encodage de la logique d'ordre supérieur de la section 2.4.3, ou à son expression directe de la section 3.2.4 a aussi un grand intérêt. Ici, il faudra naturellement interdire aux règles additionnelles d'interférer avec les règles déjà existantes, notamment en les limitant à des symboles de tête séparés ; c'est la condition d'orthogonalité.

Au niveau de la méthode des tableaux en Déduction modulo théorie, la preuve de complétude d'algorithmes de recherche de preuve plus optimisés reste à faire. En particulier, il est intéressant de ne réécrire qu'à des endroits précis, les littéraux, et de manière agressive, jusqu'à obtenir une formule en forme normale.

Normaliser les atomes demande une étude assez poussée pour la complétude, car des phénomènes d'entrelacement empêchent de fermer les branches immédiatement. Schématiquement, considérons un atome P et une formule non atomique B telle que $P \longrightarrow B$ et que B ne soit pas en forme normale, c'est-à-dire $B \longrightarrow^{+} \downarrow B$. Supposons que $T\ P$ et $F\ B$ apparaissent sur le tableau ; par l'algorithme que nous souhaitons étudier, nous allons donc réécrire $T\ P$ en $T\ \downarrow B$, en allant plus loin que B lui-même ; il n'y a donc pas de contradiction à ce stade.

Il n'est pas évident, et même grossièrement faux, de supposer que les décompositions $\alpha\beta\delta\gamma$ de $T\ \downarrow B$ et de $F\ B$ donnent les mêmes formules. En effet, $\downarrow B$ peut ne contenir aucun atome de B, et si l'on décompose naïvement $F\ B$ avec des γ-règles, alors il est possible de déclencher d'*autres* réécritures sur les littéraux produits. Ces règles peuvent aller encore plus loin que les sous-formules correspondantes de $\downarrow B$, et ce même en présence d'un système de réécriture confluent et terminant. On voit donc que pour les atomes de B, nous sommes revenus au cas de départ. La stratégie de la preuve de complétude doit se reposer sur des instances de γ-règles particulières, et entrelacées là aussi, ce qui est possible en respectant les contraintes de fraîcheur, car les γ-règles sont répétées : instancier, de manière entrelacée, les γ-règles de $T\ \downarrow B$ par les δ-termes de $F\ B$, et respectivement pour $F\ B$ et $T\ \downarrow B$. Dans ce cas, on peut démontrer que les instances d'atomes de $\downarrow B$ sont elles aussi normales, et arriver à nos fins en réécrivant les atomes de B. Ceci doit encore faire l'objet d'une formalisation.

Dans le cas où le système de réécriture n'est pas terminant, mais "productif", en ce sens qu'il n'existe pas de chaîne de réécriture infinie $A_0 \longrightarrow A_1 \longrightarrow \cdots \longrightarrow A_n \longrightarrow \cdots$ qui ne comporte que des atomes $(A_n)_{n \in \mathbb{N}}$, il est suffisant de limiter l'étape de normalisation des littéraux à la première formule non atomique rencontrée, et la démonstration de la complétude ne devrait pas être fondamentalement modifiée, voire s'en retrouver simplifiée. Il s'agit exactement de la stratégie de réécriture implémentée par Zenon Modulo [44].

En Déduction modulo théorie polarisée, la situation est complexe car normaliser les littéraux peut s'avérer une mauvaise idée, par exemple dans le cas de deux règles de réécriture $P \longrightarrow_{+} Q$ et $Q \longrightarrow_{+} R$. Avec le tableau $F\ P, T\ Q$, il ne faut surtout pas normaliser P, car on se retrouve alors avec $F\ R$ et aucun moyen de fermer le tableau, car $T\ Q$ ne peut être réécrite.

Cette limitation ne doit pas être un frein au développement pratique d'une méthode des tableaux en Déduction modulo théorie polarisée, par exemple dans le cadre de Zenon Modulo. C'est une tâche de longue haleine qui, du point de vue de l'implémentation, afficherait l'avantage pratique de pouvoir utiliser la Skolémisation dans les règles de réécriture. La Déduction modulo théorie polarisée a déjà donné d'excellents résultats dans le cadre de la résolution clausale [25].

La théorie sous-jacente demande aussi son lot de nouveautés, dont certaines semblent abordables et d'autres, plus techniques. La notion de modèle demande clairement une adaptation, car l'on souhaiterait avoir, dans les algèbres de Boole et de Heyting, des inégalités du type

$$\begin{array}{l} \text{si } A \longrightarrow_{+}^{*} B, \text{ alors } [\![B]\!] \leq [\![A]\!] \text{ et} \\ \text{si } A \longrightarrow_{-}^{*} B, \text{ alors } [\![A]\!] \leq [\![B]\!], \end{array}$$

afin de correspondre au mieux à la notion de réécriture dissociée à gauche et à droite. Le cas des préalgèbres est plus délicat, puisque la compréhension actuelle de ces structures est que le préordre sur celles-ci est une condition faible, liée à la prouvabilité, tandis que la réécriture doit correspondre à une égalité forte. Ces deux notions sont confondues dans les structures algébriques, par les vertus de l'ordre $\leq$, ce qui nous permet de conjecturer que les deux conditions ci-dessus sont correctes lorsqu'il s'agit des algèbres. Dans le cas des préalgèbres, il sera peut-être nécessaire de se restreindre aux préalgèbres ordonnées et de se servir de la relation d'ordre $\sqsubseteq$ pour énoncer des conditions similaires sur la réécriture.

Bien entendu, le travail esquissé ci-dessus est un simple point de départ. En étant ambitieux, il y a besoin de refaire en Déduction modulo théorie polarisée tout le cheminement qui a été fait dans ce manuscrit, de développer les notions adéquates de candidats de réductibilité, et ainsi de suite. D'autres travaux potentiels, de portée et de difficulté plus mesurées, pourraient appliquer les idées de traduction par double négation, ou encore construire des ponts syntaxiques entre la réécriture polarisée et la réécriture non polarisée plus efficaces que [26].

En ce qui concerne la méthode des tableaux intuitionnistes, quelques ajustements doivent faire l'objet d'une étude, notamment la démonstration de correction syntaxique de la section 3.1.2. Nous pourrons ensuite nous appuyer sur ce résultat pour démontrer la complétude de la méthode des tableaux intuitionniste d'ordre supérieur en Déduction modulo théorie, sur le modèle de [99].

Comme il est souligné à la section 3.2.5, les techniques d'admissibilité des coupures par des méthodes algébriques peuvent s'adapter à d'autres cadres logiques. C'est donc une des directions principales d'extension des techniques que nous avons décrites ; on peut en premier lieu penser à la logique d'ordre supérieur avec contraintes [96, 100].

Un cadre logique, très proche de la Déduction modulo théorie, est la superdéduction [23, 85]. L'adaptation des techniques sémantiques présentées ici ne devrait pas poser de problème.

La superdéduction fait intervenir, au lieu (ou en complément) des règles de réécriture, des règles d'inférence dites *non-logiques*, où chaque règle d'inférence est taillée à la mesure de l'axiome qu'elle remplace. On peut retrouver, en partie, cette idée dans les connecteurs *synthétiques* de la logique linéaire [98, 66], mais surtout dans les travaux de Negri et von Plato [107, 108, 109]. Dans ce dernier cas, l'élimination des coupures syntaxique est connue pour certaines classes de formules – les formules régulières de la logique intuitionniste propositionnelle par exemple. L'adaptation de nos méthodes, ainsi qu'une possible utilisation du focusing [97], pourrait donner une caractérisation de l'élimination des coupures pour une plus grande classe de formules. Pour ce faire, il faudra préalablement étudier les modèles en Déduction modulo théorie polarisée, comme proposé ci-dessus, car la transformation d'axiomes en règle d'inférence de [108] ne concerne que les membres gauche des séquents.

L'étude du contenu calculatoire des preuves d'admissibilité des coupures représente le défi le plus important. On peut citer, dans la liste des travaux à entreprendre, une simplification des preuves et de leur formalisation, un meilleur traitement de la disjonction, qui est le point fort des preuves algébriques et la traçabilité du lien entre la preuve originale et la preuve normalisée. Pour illustrer les difficultés de ce dernier point, mentionnons qu'à la section 4.3.2, nous produisons bien une preuve en forme normale, mais nous avons perdu le lien avec la preuve originelle dans le théorème de correction. Elle a été reflétée au niveau du discours et, pour garder la main sur celle-ci, il nous faut la conserver, d'une manière ou d'une autre, en tant qu'objet qui "justifie" la validité sémantique d'un séquent.

Être en mesure de manipuler la preuve de départ en tant que justification sémantique de la validité du séquent est aussi un impératif pour mieux comprendre le contenu calculatoire des preuves d'admissibilité des coupures qui passent par la méthode des tableaux. Mentionnons, à ce sujet, l'analyse d'Avigad [6]. Une autre direction est celle suivie par Herbelin et Ilik [73], qui analysent, en utilisant des opérateurs de contrôle, le contenu calculatoire des démonstrations de complétude à la Henkin, directement, mais qui, pour l'instant, introduisent des coupures dans la preuve générée. Au contraire, la complétion à la Henkin de la section 2.1.5 se fait sans coupure et le théorème de complétude associé génère des preuves sans coupure, ce qui est une piste de travail supplémentaire.

Toujours dans l'étude de la normalisation des preuves, un travail similaire à la définition des algèbres de séquents en calcul des séquents de la section 4.2.4 manque au calcul des séquents intuitionniste modulo théorie.

Un autre angle d'attaque, qui permettrait une comparaison avec les travaux de normalisation par évaluation à travers les (preuves de complétude par les) structures de Kripke, serait d'utiliser les traductions algèbre de Heyting – structures de Kripke [133], voire les généralisations de celles-ci aux faisceaux et aux sites. Notons que la notion associée de recouvrement apparaît aussi dans [6].

La plus grande partie de ce travail concerne la logique du premier ordre, à laquelle on rajoute de l'expressivité par les règles de réécriture, ce qui permet de viser (et d'atteindre) la logique d'ordre supérieur, ce qui apporte le polymorphisme. Les outils de démonstration automatique s'appuient sur de tels formalismes. Mais depuis quelques années, le développement de Dedukti [16, 119] et du cadre logique associé [36, 118] a apporté les types dépendants (de termes), ce qui permet une interaction entre réécriture et logique bien plus riche [15, 14]. L'outil Dedukti est fonctionnel et passe très bien à l'échelle en tant que noyau vérificateur de type, mais de nombreux outils demandent à être construits autour de ce noyau, afin d'en faire plus qu'un back-end. L'ajout de fonctionnalités vient aussi avec la nécessité d'enrichir le cadre logique, par exemple pour gérer la réécriture avec des symboles associatifs et commutatifs, ce qui pose autant de défis pour les années à venir, qui vont de l'implémentation aux modèles du calcul et à la démonstration de sa terminaison.

Bibliographie

[1] L. Allali and O. Hermant. Semantic a-translations and super-consistency entail classical cut elimination. In K. McMillan, A. Middeldorp, and A. Voronkov, editors, *LPAR*, volume 8312 of *LNCS ARCoSS*, pages 407–422. Springer, 2013.

[2] T. Altenkirch, M. Hofmann, and T. Streicher. Categorical reconstruction of a reduction free normalization proof. In D. Pitt, D. E. Rydeheard, and P. Johnstone, editors, *Category Theory and Computer Science*, LNCS 953, pages 182–199. Springer, 1995.

[3] J.-M. Andreoli. Logic programming with focusing proofs in linear logic. *J. Log. Comput.*, 2(3) :297–347, 1992.

[4] P. B. Andrews. Resolution in type theory. *The Journal of Symbolic Logic*, 36(3) :414–432, September 1971.

[5] Z. M. Ariola, P. Downen, H. Herbelin, K. Nakata, and A. Saurin. Classical call-by-need sequent calculi : The unity of semantic artifacts. In T. Schrijvers and P. Thiemann, editors, *Functional and Logic Programming - 11th International Symposium, FLOPS 2012, Kobe, Japan, May 23-25, 2012. Proceedings*, volume 7294 of *Lecture Notes in Computer Science*, pages 32–46. Springer, 2012.

[6] J. Avigad. Algebraic proofs of cut elimination. *Journal of Logic and Algebraic Programming*, 49(1-2) :15–30, 2001.

[7] M. Baaz and C. G. Fermüller. Non-elementary Speedups between Different Versions of Tableaux. In P. Baumgartner, R. Hähnle, and J. Posegga, editors, *TABLEAUX'95*, volume 918 of *LNCS (LNAI)*, pages 217–230, St.~Goar, 1995. Springer.

[8] H. Barendregt. *The Lambda Calculus : Its Syntax and Semantics.* Studies in Logic and the Foundations of Mathematics. Elsevier Science, 1985.

[9] J. Barwise. *Handbook of Mathematical Logic.* Studies in Logic and the Foundations of Mathematics. Elsevier Science, 1982.

[10] U. Berger, M. Eberl, and H. Schwichtenberg. Normalisation by evaluation. In B. Möller and J. V. Tucker, editors, *Prospects for Hardware Foundations, ESPRIT Working Group 8533, NADA - New Hardware Design Methods, Survey Chapters*, volume 1546 of *Lecture Notes in Computer Science*, pages 117–137. Springer, 1998.

[11] P. Bernays. Review : Oiva ketonen, untersuchungen zum prädikätenkalkul. *Journal of Symbolic Logic*, 10 :127–130, 1945.

[12] E. W. Beth. Semantic entailment and provability in logic. In J. Largeault, editor, *Logique Mathématique, Textes.* Armand Colin, 1972.

[13] W. Bibel. *Automated theorem proving.* Artificial intelligence. F. Vieweg und Sohn, Braunschweig, 1982.

[14] F. Blanqui, J. Jouannaud, and A. Rubio. The computability path ordering : The end of a quest. In M. Kaminski and S. Martini, editors, *Computer Science Logic, 22nd International Workshop, CSL 2008, 17th Annual Conference of the EACSL, Bertinoro, Italy, September 16-19, 2008. Proceedings*, volume 5213 of *Lecture Notes in Computer Science*, pages 1–14. Springer, 2008.

[15] F. Blanqui, J.-P. Jouannaud, and M. Okada. The calculus of algebraic constructions. In P. Narendran and M. Rusinowitch, editors, *Rewriting Techniques and Applications*, volume 1631 of *LNCS*. Springer, 1999.

[16] M. Boespflug, Q. Carbonneaux, and O. Hermant. The $\lambda\Pi$-calculus modulo as a universal proof language. In *In Second Workshop on Proof Exchange for Theorem Proving (PxTP)*, volume 878, pages 28–43. CEUR-WS.org, 2012.

[17] R. Bonichon. TaMeD : A tableau method for deduction modulo. *Automated Reasoning : Second International Joint Conference, IJCAR 2004, July 4-8, 2004*, 2004.

[18] R. Bonichon and O. Hermant. On constructive cut admissibility in deduction modulo. In T. Altenkirch and C. McBride, editors, *TYPES for proofs and programs*, volume 4502 of *Lecture Notes in Computer Science*, pages 33–47. Springer, 2006.

[19] R. Bonichon and O. Hermant. A semantic completeness proof for tableaux modulo. In M. Hermann and A. Voronkov, editors, *LPAR 2006*, volume 4246 of *Lecture Notes in Computer Science*, pages 167–181, Phom Penh, Cambodia, November 2006. Springer-Verlag.

[20] R. Bonichon and O. Hermant. Taming tamed. unpublished, 2008.

[21] R. Bonichon and O. Hermant. A syntactic soundness proof for free-variable tableaux with on-the-fly skolemization. unpublished, 2016.

[22] M. Boudard and O. Hermant. Polarizing double-negation translations. In K. McMillan, A. Middeldorp, and A. Voronkov, editors, *LPAR*, volume 8312 of *LNCS ARCoSS*, pages 182–197. Springer, 2013.

[23] P. Brauner, C. Houtmann, and C. Kirchner. Principles of superdeduction. In *Proceedings of the 22nd IEEE Symposium on Logic in Computer Science (LICS 2007), 10-12 July 2007, Wroclaw, Poland,*, pages 41–50. IEEE Computer Society, 2007.

[24] A. Brunel, O. Hermant, and C. Houtmann. Orthogonality and boolean algebras for deduction modulo. In C.-H. L. Ong, editor, *TLCA*, volume 6690 of *Lecture Notes in Computer Science*, pages 76–90. Springer, 2011.

[25] G. Burel. Experimenting with deduction modulo. In N. Bjørner and V. Sofronie-Stokkermans, editors, *CADE*, volume 6803 of *Lecture Notes in Computer Science*, pages 162–176. Springer, 2011.

[26] G. Burel and C. Kirchner. Regaining cut admissibility in deduction modulo using abstract completion. *Information and Computation*, 208(2) :140–164, 2010.

[27] G. Bury, D. Delahaye, D. Doligez, P. Halmagrand, and O. Hermant. Automated deduction in the B set theory using typed proof search and deduction modulo. In A. Fehnker, A. McIver, G. Sutcliffe, and A. Voronkov, editors, *20th International Conferences on Logic for Programming, Artificial Intelligence and Reasoning - Short Presentations, LPAR 2015, Suva, Fiji, November 24–28, 2015*, volume 35 of *EPiC Series in Computing*, pages 42–58. EasyChair, 2015.

[28] D. Cantone and M. Nicolosi Asmundo. A Further and Effective Liberalization of the delta-Rule in Free Variable Semantic Tableaux. In *Selected Papers from Automated Deduction in Classical and Non-Classical Logics*, pages 109–125, London, UK, UK, 2000. Springer-Verlag.

[29] Z. Chihani, D. Ilik, and D. Miller. Classical polarizations yield double-negation translations. Technical report, Inria Saclay, Aug. 2016.

[30] A. Church. A note on the Entscheidungsproblem. *The Journal of Symbolic Logic*, 1 :40–41, 1936.

[31] C. Coquand. From semantics to rules : A machine-assisted analysis. In *CSL*, LNCS 832, pages 91–105. Springer, 1993.

[32] T. Coquand and J. Gallier. A proof of strong normalization for the theory of constructions using a kripke-like interpretation. In *Preliminary Proceedings 1st Intl. Workshop on Logical Frameworks*, 1990.

[33] R. Cori and D. Lascar. *Logique mathématique*. Masson, 1993.

[34] D. Cousineau. *Models and Proof Normalization*. PhD thesis, École Polytechnique, 2009.

[35] D. Cousineau. On completeness of reducibility candidates as a semantics of strong normalization. *Logical Methods in Computer Science*, 8(1), 2012.

[36] D. Cousineau and G. Dowek. Embedding pure type systems in the lambda-pi-calculus modulo. In S. R. D. Rocca, editor, *Typed Lambda Calculi and Applications, 8th International Conference, TLCA 2007*, volume 4583 of *Lecture Notes in Computer Science*, pages 102–117. Springer, 2007.

[37] D. Cousineau and O. Hermant. A semantic proof that reducibility candidates entail cut elimination. In A. Tiwari, editor, *RTA*, volume 15 of *LIPIcs*, pages 133–148. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2012.

[38] M. Crabbé. Non-normalisation de ZF. Manuscript, 1974.

[39] D. Cubric, P. Dybjer, and P. J. Scott. Normalization and the yoneda embedding. *Mathematical Structures in Computer Science*, 8(2) :153–192, 1998.

[40] H. B. Curry. The permutability of rules in the classical inferential calculus. *The Journal of Symbolic Logic*, 17(4) :245–248, 1952.

[41] O. Danvy. Type-directed partial evaluation. In J. Hatcliff, T. Æ. Mogensen, and P. Thiemann, editors, *Partial Evaluation - Practice and Theory, DIKU 1998 International Summer School, Copenhagen, Denmark, June 29 - July 10, 1998*, volume 1706 of *Lecture Notes in Computer Science*, pages 367–411. Springer, 1998.

[42] D. Delahaye, D. Doligez, F. Gilbert, P. Halmagrand, and O. Hermant. Proof certification in zenon modulo : When achilles uses deduction modulo to outrun the tortoise with shorter steps. In *IWIL, 10th International Workshop on the Implementation of Logics*, 2013.

[43] D. Delahaye, D. Doligez, F. Gilbert, P. Halmagrand, and O. Hermant. Proof compression and certification in zenon modulo. In *Third Workshop of the Amadeus Project on Proof Compression*, 2013.

[44] D. Delahaye, D. Doligez, F. Gilbert, P. Halmagrand, and O. Hermant. Zenon modulo : When achilles outruns the tortoise using deduction modulo. In K. McMillan, A. Middledorp, and A. Voronkov, editors, *LPAR*, volume 8312 of *LNCS ARCoSS*, pages 274–290. Springer, 2013.

[45] G. Dowek. Truth values algebras and normalization. In T. Altenkirch and C. McBride, editors, *TYPES for proofs and programs*, volume 4502 of *Lecture Notes in Computer Science*, pages 110–124. Springer, 2006.

[46] G. Dowek. Polarized deduction modulo. In *IFIP Theoretical Computer Science*, 2010.

[47] G. Dowek, T. Hardin, and C. Kirchner. Hol-lambda-sigma : an intentional first-order expression of higher-order logic. *Mathematical Structures in Computer Science*, 11 :1–25, 2001.

[48] G. Dowek, T. Hardin, and C. Kirchner. Binding logic : Proofs and models. In M. Baaz and A. Voronkov, editors, *LPAR*, volume 2514 of *Lecture Notes in Computer Science*, pages 130–144. Springer, 2002.

[49] G. Dowek, T. Hardin, and C. Kirchner. Theorem proving modulo. *Journal of Automated Reasoning*, 31 :33–72, 2003.

[50] G. Dowek and O. Hermant. A simple proof that super-consistency implies cut elimination. In F. Baader, editor, *RTA*, volume 4533 of *Lecture Notes in Computer Science*, pages 93–106. Springer, 2007.

[51] G. Dowek and O. Hermant. A simple proof that super-consistency implies cut elimination. *Notre-Dame Journal of Formal Logic*, 53(4) :439–456, 2012.

[52] G. Dowek and B. Werner. Proof normalization modulo. *The Journal of Symbolic Logic*, 68(4) :1289–1316, December 2003.

[53] G. Dowek and B. Werner. Arithmetic as a theory modulo. In J. Giesl, editor, *RTA*, volume 3467 of *Lecture Notes in Computer Science*, pages 423–437. Springer, 2005.

[54] A. G. Dragalin. *Mathematical Intuitionism : Introduction to Proof Theory.* Translations of mathematical monographs. American Mathematical Society, Providence, Rhode Island, 1988.

[55] M. A. Dummett. *Elements of Intuitionism.* Oxford logic guides. Clarendon Press, 2000.

[56] R. Dyckhoff. Contraction-free sequent calculi for intuitionistic logic. *J. Symb. Log.*, 57(3) :795–807, 1992.

[57] R. Dyckhoff and S. Negri. Admissibility of structural rules for contraction-free systems of intuitionistic logic. *J. Symb. Log.*, 65(4) :1499–1518, 2000.

[58] Y. G. Egon Börger, Erich Grädel. *The Classical Decision Problem.* Springer-Verlag Berlin Heidelberg, 1997.

[59] M. Fitting. *First Order Logic and Automated Theorem Proving.* Springer-Verlag, 2nd edition, 1996.

[60] G. Frege. *Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinenDenkens.* Verlag von Louis Nebert, Halle, 1879.

[61] H. Friedman. Classically and intuitionistically provably recursive functions. In G. H. Müller and D. S. Scott, editors, *Higher Set Theory*, volume 669 of *Lecture Notes in Mathematics*, pages 21–27. Springer Berlin Heidelberg, 1978.

[62] G. Gentzen. Untersuchungen über das logische Schliessen. *Mathematische Zeitschrift*, 39 :176–210, 405–431, 1934.

[63] G. Gentzen. Über das Verhaltnis zwischen intuitionistischer une klassischer Logik. *Archiv für matematische Logik und Grundlagenforschung*, 16 :119–132, 1974.

[64] F. Gilbert. A lightweight double-negation translation. In A. Fehnker, A. McIver, G. Sutcliffe, and A. Voronkov, editors, *20th International Conferences on Logic for Programming, Artificial Intelligence and Reasoning - Short Presentations, LPAR 2015, Suva, Fiji, November 24-28, 2015.*, volume 35 of *EPiC Series in Computing*, pages 81–93. EasyChair, 2015.

[65] G. Gilbert and O. Hermant. Normalisation by completeness with heyting algebras. In M. Davis, A. Fehnker, A. McIver, and A. Voronkov, editors, *Logic for Programming, Artificial Intelligence, and Reasoning - 20th International Conference, LPAR-20 2015, Suva, Fiji, November 24-28, 2015, Proceedings*, volume 9450 of *Lecture Notes in Computer Science*, pages 469–482. Springer, 2015.

[66] J.-Y. Girard. Linear logic. *Theoretical Computer Science*, 50 :1–102, 1987.

[67] J.-Y. Girard, P. Taylor, and Y. Lafont. *Proofs and Types.* Cambridge University Press, New York, NY, USA, 1989.

[68] K. Gödel. *Über die Vollstandigkeit des Logikkalkuls.* PhD thesis, Vienna, 1929.

[69] K. Gödel. Zur intuitionistischen arithmetik und zahlentheorie. *Ergebnisse eines mathematischen Kolloquiums*, 4 :34–38, 1933.

[70] R. Hähnle and P. Schmitt. The liberalized δ-rule in free variable semantic tableaux. *Journal of Automated Reasoning*, 13(2) :211–221, 1994.

[71] P. Halmagrand. *Automated Deduction and Proof Certification for the B Method.* PhD thesis, Conservatoire National des Arts et Métiers, 1996.

[72] H. Herbelin. *Séquents qu'on calcule.* PhD thesis, Université Paris VII, Janvier 1995.

[73] H. Herbelin and D. Ilik. An analysis of the constructive content of henkin's proof of gödel's completeness theorem. Available on the authors' webpage, 2016.

[74] H. Herbelin and G. Lee. Forcing-based cut-elimination for gentzen-style intuitionistic sequent calculus. In H. Ono, M. Kanazawa, and R. J. G. B. de Queiroz, editors, *Logic, Language, Information and Computation, 16th International Workshop, WoLLIC 2009, Tokyo, Japan, June 21-24, 2009. Proceedings*, volume 5514 of *Lecture Notes in Computer Science*, pages 209–217. Springer, 2009.

[75] O. Hermant. A model-based cut elimination proof. *2nd St-Petersburg Days of Logic and Computability*, 2003.

[76] O. Hermant. Completeness of cut-free sequent calculus modulo. Available on the author's web page, 2004.

[77] O. Hermant. *Méthodes Sémantiques en Déduction Modulo.* PhD thesis, Université Paris 7 – Denis Diderot, 2005.

[78] O. Hermant. Semantic cut elimination in the intuitionistic sequent calculus. In P. Urzyczyn, editor, *Typed Lambda-Calculi and Applications*, volume 3461 of *LNCS*, pages 221–233, Nara, Japan, 2005. Springer.

[79] O. Hermant. Resolution is cut-free. *Journal of Automated Reasoning*, 44(3) :245–276, 2010.

[80] O. Hermant and J. Lipton. A constructive semantic approach to cut elimination in type theories with axioms. In M. Kaminski and S. Martini, editors, *CSL*, volume 5213 of *Lecture Notes in Computer Science*, pages 169–183. Springer, 2008.

[81] O. Hermant and J. Lipton. Cut elimination in the Intuitionistic Theory of Types with axioms and rewriting cuts, constructively. In C. E. Benzmuller, C. E. Brown, J. Siekmann, and R. Statman, editors, *Festschrift in Honor of Peter B. Andrews on His 70th Birthday*, Studies in Logic and the Foundations of Mathematics, pages 101–134. IFCoLog, 2008.

[82] O. Hermant and J. Lipton. Completeness and cut-elimination in the Intuitionistic Theory of Types - Part 2. *J. Log. Comput.*, 20(2) :597–602, 2010.

[83] O. Hermant and M. Okada. Cut elimination in higher-order intuitionistic linear logic. unpublished, 2007.

[84] W. Hodges. *A Shorter Model Theory.* Cambridge University Press, 1997.

[85] C. Houtmann. *Representation and interaction of proofs in superdeduction modulo.* Theses, Université Henri Poincaré - Nancy I, Mar. 2010.

[86] D. Ilik. *Constructive Completeness Proofs and Delimited Control.* PhD thesis, Ecole polytechnique, Oct. 2010.

[87] D. Ilik, G. Lee, and H. Herbelin. Kripke models for classical logic. *Ann. Pure Appl. Logic*, 161(11) :1367–1378, 2010.

[88] S. Kanger. *Provability in Logic.* Almquist and Wicksell, Stockholm, 1957.

[89] O. Ketonen. *Untersuchungen zum Prädikatenkalkül.* PhD thesis, 1944.

[90] S. C. Kleene. *Introduction to Metamathematics.* North-Holland, 1952.

[91] S. C. Kleene. Permutability of inferences in Gentzen's calculi LK and LJ. *Memoirs of the American Mathematical Society*, 10 :1–26, 27–68, 1952.

[92] S. C. Kleene. *Mathematical logic.* Wiley, 1967. Republished in 2002 by Dover.

[93] A. Kolmogorov. On the principle of the excluded middle. *Mat. Sb.*, 32 :646–667, 1925.

[94] J.-L. Krivine. Une preuve formelle et intuitionniste du théorème de complétude de la logique classique. *The Bulletin of Symbolic Logic*, 2 :405–421, 1996.

[95] J.-L. Krivine. *Realisability in Classical Logic.* Course Notes, Marseille-Luminy, 2004.

[96] J. Leach and S. Nieva. A higher-order logic programming language with constraints. In H. Kuchen and K. Ueda, editors, *FLOPS*, volume 2024 of *Lecture Notes in Computer Science*, pages 108–122. Springer, 2001.

[97] C. Liang and D. Miller. Focusing and polarization in linear, intuitionistic, and classical logics. *Theor. Comput. Sci.*, 410(46) :4747–4768, 2009.

[98] C. Liang and D. Miller. On subexponentials, synthetic connectives, and multi-level delimited control. In M. Davis, A. Fehnker, A. McIver, and A. Voronkov, editors, *Logic for Programming, Artificial Intelligence, and Reasoning - 20th International Conference, LPAR-20 2015, Suva, Fiji, November 24-28, 2015, Proceedings*, volume 9450 of *Lecture Notes in Computer Science*, pages 297–312. Springer, 2015.

[99] J. Lipton and M. De Marco. Completeness and cut elimination in Church's intuitionistic theory of types. *Journal of Logic and Computation*, 15 :821–854, December 2005.

[100] J. Lipton and S. Nieva. Higher-order logic programming languages with constraints : A semantics. In S. R. D. Rocca, editor, *TLCA*, volume 4583 of *Lecture Notes in Computer Science*, pages 272–289. Springer, 2007.

[101] S. Maehara. Eine Darstellung der intuitionistischen Logik in der klassischen. *Nagoya Math. J.*, 7 :45–64, 1954.

[102] S. Maehara. Lattice-valued representation of the cut-elimination theorem. *Tsukuba Journal of Mathematics*, 15(2) :509–521, 1991.

[103] The Coq development team. *The Coq proof assistant reference manual.* PiR2, Inria, 2016.

[104] J. Mitchell. *Foundations for Programming Languages.* MIT Press, Cambridge, Massachusetts, 1996.

[105] J. Mitchell and E. Moggi. Kripke-style models for typed lambda calculus. *Annals of Pure and Applied Logic*, 51 :99–124, 1991.

[106] A. Naibo. *Le statut dynamique des axiomes. Des preuves aux modèles.* PhD thesis, Université Paris 1, 2013.

[107] S. Negri. Sequent calculus proof theory of intuitionistic apartness and order relations. *Arch. Math. Log.*, 38(8) :521–547, 1999.

[108] S. Negri and J. von Plato. Cut elimination in the presence of axioms. *Bulletin of Symbolic Logic*, 4(4) :418–435, 1998.

[109] S. Negri and J. von Plato. Proof systems for lattice theory. *Mathematical Structures in Computer Science*, 14(4) :507–526, 2004.

[110] A. Nerode and R. A. Shore. *Logic for Applications.* Springer, 1993.

[111] W. Nutt, P. Réty, and G. Smolka. Basic narrowing revisited. *J. Symb. Comput.*, 7(3/4) :295–317, 1989.

[112] M. Okada. Phase semantic cut-elimination and normalization proofs of first- and higher-order linear logic. *Theoretical Computer Science*, 227 :333–396, 1999.

[113] M. Okada. A uniform semantic proof for cut-elimination and completeness of various first and higher order logics. *Theoretical Computer Science*, 281 :471–498, 2002.

[114] D. Prawitz. Completeness and hauptsatz for second order logic. *Theoria*, 33 :246–258, 1964.

[115] D. Prawitz. Hauptsatz for higher order logic. *The Journal of Symbolic Logic*, 33(3) :452–457, September 1968.

[116] H. Rasiowa and R. Sikorski. *The mathematics of metamathematics.* PWN, Polish Scientific Publishers, Warszawa, 1963.

[117] B. Russell. *The Principles of Mathematics.* Cambridge University Press, 1903.

[118] R. Saillard. Rewriting modulo βin the $\lambda\Pi$-calculus modulo. In I. Cervesato and K. Chaudhuri, editors, *Proceedings Tenth International Workshop on Logical Frameworks and Meta Languages : Theory and Practice, LFMTP 2015, Berlin, Germany, 1 August 2015.*, volume 185 of *EPTCS*, pages 87–101, 2015.

[119] R. Saillard. *Typechecking in the $\lambda\Pi$-Calculus Modulo : Theory and Practice.* Thèse de doctorat, Ecole Nationale Supérieure des Mines de Paris, September 2015.

[120] J. E. Santo, R. Matthes, and L. Pinto. Continuation-passing style and strong normalisation for intuitionistic sequent calculi. *Logical Methods in Computer Science*, 5(2), 2009.

[121] C. Schürmann and J. Sarnat. Structural logical relations. In *Proceedings of the Twenty-Third Annual IEEE Symposium on Logic in Computer Science, LICS 2008, 24-27 June 2008, Pittsburgh, PA, USA*, pages 69–80. IEEE Computer Society, 2008.

[122] K. Schütte. Schlussweisen kalküle der prädikatenlogik. *Mathematische Annalen*, 122 :47–65, 1950.

[123] K. Schütte. Syntactical and semantical properties of simple type theory. *The Journal of Symbolic Logic*, 25(4) :305–326, December 1960.

[124] K. Schütte. *Proof Theory.* Grundlehren der matematischen Wissenschaften. Springer-Verlag, 1977.

[125] R. M. Smullyan. *First-Order Logic.* Springer-Verlag, Berlin Heidelberg New York, 1968.

[126] J. Stuber. A model-based completeness proof of extended narrowing and resolution. In *First International Joint Conference on Automated Reasoning (IJCAR-2001)*, volume 2083 of *LNCS*, pages 195–210. Springer, June 2001.

[127] M. E. Szabo, editor. *Collected Papers of Gerhard Gentzen.* Studies in Logic and the Foundation of Mathematics. North Holland Publishing Company, 1969.

[128] W. W. Tait. A nonconstructive proof for Gentzen's Hauptsatz for second-order predicate logic. *Bulletin of the American Mathematical Society*, 72 :980–983, 1966.

[129] M.-o. Takahashi. A proof of cut-elimination theorem in simple type-theory. *Journal of the Mathematical Society of Japan*, 19(4) :399–410, 1967.

[130] G. Takeuti. *Proof Theory.* North Holland, 2nd edition, 1987.

[131] The Agda Team. The Agda Wiki.

[132] A. S. Troelstra. *Lectures on Linear Logic*, volume 30 of *CSLI Lecture Notes.* Center for the Study of Language and Information, Stanford, CA, USA, 1992.

[133] A. S. Troelstra and D. van Dalen. *Constructivism in Mathematics, An Introduction.* North-Holland, 1988.

[134] A. Turing. On computable numbers, with an application to the Entscheidungsproblem. *Proceedings of the London Mathematical Society*, 42 :230–265, 1937. Correction in vol. 43, pp. 544-546.

[135] C. Urban. *Classical Logic and Computation.* PhD thesis, University of Cambridge, October 2000.

[136] C. Urban. Strong normalisation for a gentzen-like cut-elimination procedure. In *Proceedings of TLCA*, pages 415–430, 2001.

[137] J. Van Heijenoort. *From Frege to Gödel : A Source Book in Mathematical Logic, 1879-1931.* Source books in the history of the sciences. Harvard University Press, 1967.

[138] W. Veldman. An intuitionistic completeness theorem for intuitionistic predicate logic. *Journal of Symbolic Logic*, 41 :159–166, 1976.

[139] A. Waaler and L. Wallen. *Handbook of Tableau Methods*, chapter Tableaux for Intutionistic Logics, pages 255–296. Kluwer Academic Publishers, 1999.

Index