Université Paris Cité
En cotutelle avec Universidad de la República

École doctorale 386
Sciences Mathématiques de Paris Centre
Institut de Recherche en Informatique Fondamentale (IRIF)

PEDECIBA
Facultad de Ingeniería
Instituto de Matemática y Estadística Rafael Laguardia (IMERL)

The ramified analytic hierarchy in second-order logic

Par Félix CASTRO

THÈSE DE DOCTORAT
en vue d'obtenir les grades de
DOCTEUR D'UNIVERSITÉ PARIS CITÉ EN INFORMATIQUE
DOCTOR DE LA UNIVERSIDAD DE LA REPÚBLICA EN MATEMÁTICA

Dirigée par Hugo HERBELIN
Et par Alexandre MIQUEL
Présentée et soutenue publiquement le 17/12/2024

Devant un jury composé de :

Hugo HERBELIN, DR	Université Paris Cité	Directeur de thèse
Alexandre MIQUEL, Prof.	Universidad de la República	Directeur de thèse
Laurent REGNIER, Prof.	Université Aix-Marseille	Rapporteur et Président du Jury
Colin RIBA, MCF-HDR	ENS-Lyon	Rapporteur
Walter FERRER, Prof.	Universidad de la República	Examinateur
Laura FONTANELLA, MCF	Université Paris-Est Créteil	Examinatrice
Guillaume GEOFFROY, MCF	Université Paris Cité	Examinateur
Étienne MIQUEY, MCF	Université Aix-Marseille	Membre invité

Résumé

Titre : La hiérarchie analytique ramifiée en logique du second ordre

Mots-clés : *arithmétique du second ordre, hiérarchie analytique ramifiée, axiome du choix, réalisabilité, arithmétique des types finis, extensionnalité, paramétricité*

Cette thèse s'intéresse dans un premier temps à l'étude de la hiérarchie analytique ramifiée (RAH) en arithmétique du second ordre (**PA2**). La hiérarchie analytique ramifiée a été définie par Kleene en 1960. C'est une adaptation de la notion de constructibilité (introduite par Gödel pour la théorie des ensembles) au cadre de l'arithmétique du second ordre. Les propriétés de cette hiérarchie, en relation avec la théorie de la récursion et l'étude des modèles ensemblistes de **PA2**, ont été étudiées en profondeur. Il semble naturel de formaliser RAH dans **PA2** pour essayer de démontrer qu'ajouter l'axiome du choix ou (une variante de) l'axiome de constructibilité à **PA2** n'apporte pas de contradiction. Cependant, la seule trace écrite d'une telle formalisation semble incorrecte. Dans cette thèse, nous souhaitons travailler sur cette formalisation. Pour cela, nous allons travailler dans une version de l'arithmétique obtenue en enlevant l'axiome d'induction des axiomes de **PA2**. Dans ce système, une nouvelle variante de l'axiome du choix apparaît : nous l'appelons l'axiome de collection, en référence à l'axiome de théorie des ensembles portant le même nom. Cet axiome semble n'avoir jamais été considéré dans le cadre de la logique du second ordre. Nous montrons qu'il a de bonnes propriétés calculatoires : sa contraposée est réalisée par l'identité en réalisabilité classique alors qu'il est lui-même réalisé par l'identité en réalisabilité intuitionniste. De plus, nous en énonçons une forme équivalente qui est interprétée par une traduction négative de la logique classique vers la logique intuitionniste. Enfin, nous montrons qu'une variante de cet axiome est plus faible qu'une variante de l'axiome du choix en logique intuitionniste. Nous travaillons donc dans une théorie sans induction mais contenant l'axiome de collection pour étudier la hiérarchie analytique ramifiée. Nous montrons que c'est un modèle de **PA2** satisfaisant une version forte d'axiome du choix : le principe de l'univers bien ordonné. Il nous semble que l'axiome de collection est nécessaire pour démontrer ce résultat et nous donnons des arguments pour expliquer cette intuition.

Dans une deuxième partie de la thèse, plus courte que la première, nous étudions l'égalité extensionnelle en arithmétique des types finis ($\mathbf{HA}^\omega$). L'arithmétique des types finis est une extension conservative de l'arithmétique de Heyting obtenue en étendant la syntaxe des termes à tout le Système T. Elle parle donc des fonctionnelles de types finis. Alors que l'égalité entre entiers naturels est spécifiée par les axiomes de Peano, comment peut être définie l'égalité entre fonctionnelles ? Différentes réponses à cette question poussent à l'étude de différentes variantes de $\mathbf{HA}^\omega$, telles qu'une variante extensionnelle (**E-**$\mathbf{HA}^\omega$) où deux fonctionnelles sont égales si elles ont le même graphe. Dans cette partie, nous allons montrer comment l'étude d'une famille de relations d'équivalence partielle nous amène à définir une interprétation par paramétricité de l'égalité extensionnelle dans $\mathbf{HA}^\omega$.

Abstract

Title : The ramified analytic hierarchy in second-order logic

Key words: *second-order arithmetic, ramified analytic hierarchy, axiom of choice, realizability, finite type arithmetic, extensionality, parametricity*

In its first part, this thesis focuses on the study of the ramified analytic hierarchy (RAH) in second-order arithmetic (**PA2**). The ramified analytic hierarchy was defined by Kleene in 1960. It is an adaptation of the notion of constructibility (introduced by Gödel for set theory) to the framework of second-order arithmetic. The properties of this hierarchy, in relation to computability and to the study of set-theoretic models of **PA2**, have been studied in depth. It seems natural to formalize RAH in **PA2** in an attempt to demonstrate that adding the axiom of choice or (a variant of) the axiom of constructibility to second-order arithmetic does not bring contradiction. However, the only written trace of such a formalization seems to be incorrect. In this thesis, we want to work on this formalization. To do this, we will work in a version of arithmetic obtained by removing the axiom of induction from the axioms of **PA2**. In this system, a new variant of the axiom of choice appears: we call it the axiom of collection, in reference to the homonymous axiom of set theory. It seems that this axiom has never been considered in the context of second-order logic. We show that it has good computational properties: its contrapositive is realized by the identity in classical realizability, while it is itself realized by the identity in intuitionistic realizability. In addition, we show that it is equivalent to an axiom which is well-behaved with respect to a negative translation from classical logic into intuitionistic logic. Finally, we show that this variant of the axiom of collection is weaker than a variant of the axiom of choice in intuitionistic logic. We therefore work in a theory without induction but containing the axiom of collection. We aim at studying the ramified analytic hierarchy. We show that it is a model of **PA2** satisfying a strong version of the axiom of choice: the principle of the well-ordered universe. It seems that the axiom of collection is necessary to prove this result and we will thoroughly explain this intuition.

In the second part of the thesis, shorter than the first, we study extensional equality in finite type arithmetic. Higher Type Arithmetic ($\mathbf{HA}^\omega$) is a first-order many-sorted theory. It is a conservative extension of Heyting Arithmetic obtained by extending the syntax of terms to all of System T: the objects of interest here are the functionals of higher types. While equality between natural numbers is specified by the axioms of Peano, how can equality between functionals be defined? From this question, different versions of $\mathbf{HA}^\omega$ arise, such as an extensional version ($\mathbf{E\text{-}HA}^\omega$) and an intentional version ($\mathbf{I\text{-}HA}^\omega$). In this work, we will see how the study of partial equivalence relations leads us to design a translation by parametricity from $\mathbf{E\text{-}HA}^\omega$ to $\mathbf{HA}^\omega$.

Resumen

Título : La jerarquía analítica ramificada en lógica de segundo orden

Palabras claves : *aritmética de segundo orden, jerarquía analítica ramificada, axioma de elección, realizabilidad, aritmética de tipo finito, extensionalidad, parametricidad*

En su primera parte, esta tesis se centra en el estudio de la jerarquía analítica ramificada (RAH) en aritmética de segundo orden (**PA2**). La jerarquía analítica ramificada fue definida por Kleene en 1960. Se trata de una adaptación de la noción de constructibilidad (introducida por Gödel para la teoría de conjuntos) al marco de la aritmética de segundo orden. Las propiedades de esta jerarquía, en relación con la computabilidad y con el estudio de los modelos de **PA2**, han sido estudiadas en profundidad. Parece natural formalizar RAH en **PA2** en un intento de demostrar que añadir el axioma de elección o (una variante de) el axioma de constructibilidad a la aritmética de segundo orden no conlleva contradicción. Sin embargo, el único rastro escrito de tal formalización parece ser incorrecto. En esta tesis, queremos trabajar sobre esta formalización. Para ello, trabajaremos en una versión de la aritmética obtenida eliminando el axioma de inducción de los axiomas de **PA2**. En este sistema, aparece una nueva variante del axioma de elección: lo llamamos axioma de colección, en referencia al axioma homónimo de la teoría de conjuntos. Parece que este axioma nunca se ha considerado en el contexto de la lógica de segundo orden. Demostramos que tiene buenas propiedades computacionales: su contraposición se realiza por la identidad en la realizabilidad clásica, mientras que él mismo se realiza por la identidad en la realizabilidad intuicionista. Además, mostramos que es equivalente a un axioma que se comporta bien con respecto a una traducción negativa de la lógica clásica a la lógica intuicionista. Finalmente, mostramos que una variante del axioma de colección es más débil que una variante del axioma de elección en lógica intuicionista. Por tanto, trabajamos en una teoría sin inducción pero que contiene el axioma de colección para estudiar la jerarquía analítica ramificada. Demostramos que es un modelo de **PA2** que satisface una versión fuerte del axioma de elección: el principio del universo bien ordenado. Parece que el axioma de colección es necesario para demostrar este resultado y explicaremos a fondo esta intuición.

En la segunda parte de la tesis, más breve que la primera, estudiamos la igualdad extensional en aritmética de tipo finito ($\mathbf{HA}^\omega$). La aritmética de tipo finito es una teoría de primer orden. Es una extensión conservativa de la Aritmética de Heyting que se obtiene extendiendo la sintaxis de los términos a todo el Sistema T: los objetos de interés aquí son los funcionales de tipos superiores. Mientras que la igualdad entre números naturales está especificada por los axiomas de Peano, cómo puede definirse la igualdad entre funcionales? A partir de esta pregunta, surgen diferentes versiones de $\mathbf{HA}^\omega$, como una versión extensional ($\mathbf{E}$-$\mathbf{HA}^\omega$) y una versión intencional ($\mathbf{I}$-$\mathbf{HA}^\omega$). En este trabajo veremos cómo el estudio de unas relaciones de equivalencia parciales nos lleva a diseñar una traducción por parametricidad de $\mathbf{E}$-$\mathbf{HA}^\omega$ a $\mathbf{HA}^\omega$.

Résumé substantiel

L'axiome du choix, depuis sa formulation par Zermelo en 1904 [53, 6], a joué un rôle important dans de nombreux domaines des mathématiques. Ses conséquences sont vastes et parfois surprenantes, ce qui a donné lieu à des discussions sur son utilisation dans les preuves mathématiques. D'une part, il est largement accepté en mathématiques constructives [4], en particulier car il a un contenu calculatoire clair en logique intuitionniste [49, 33]. D'autre part, bien que son utilisation (en théorie des ensembles) ne conduit pas à des incohérences, son contenu calculatoire (en logique classique) reste inconnu. Dans ce manuscrit, nous aborderons le problème de la consistance relative de l'axiome du choix dans le cadre de la logique du second ordre et nous esquisserons une traduction syntaxique conçue en analysant la hiérarchie analytique ramifiée [26]. Cette traduction pourrait servir à l'étude du contenu calculatoire de l'axiome du choix en logique classique.

En 1938, Gödel a montré la cohérence relative de l'axiome du choix et de l'hypothèse généralisée du continu par rapport aux axiomes de la théorie des ensembles (**ZF**) en utilisant le concept clé d'ensembles constructibles [21]. Sa construction repose sur un opérateur $X \mapsto \mathbf{Def}(X)$ qui fait correspondre chaque ensemble X à l'ensemble $\mathbf{Def}(X)$ des sous-ensembles de X qui sont définissables à l'aide d'une formule du premier ordre avec des paramètres dans X et des quantifications restreintes à X. En itérant (de façon transfinie) cet opérateur, Gödel définit la suite $(\mathrm{L}_\alpha)_{\alpha \in On}$ satisfaisant l'équation

$$\mathrm{L}_\alpha \;:=\; \bigcup_{\beta<\alpha} \mathbf{Def}(\mathrm{L}_\beta)$$

et dont l'union transfinie est appelée l'univers constructible (noté L). Gödel montre ensuite que la classe L est un modèle intérieur de **ZF** qui satisfait non seulement l'axiome du choix mais aussi l'hypothèse généralisée du continu. En fait, ce modèle satisfait l'axiome de constructibilité (noté V = L) qui implique les deux axiomes précédents. Toutes ces constructions (et ces preuves) peuvent être internalisées dans **ZF**, montrant un résultat de cohérence relative entre les systèmes **ZF** + V = L et **ZF** : l'ajout de l'axiome de constructibilité à la théorie des ensembles n'entraîne pas d'incohérences.

Les idées qui sous-tendent la construction de Gödel sont très générales et peuvent être appliquées à d'autres formalismes tels que l'étude des sous-ensembles de ω. Dans ce cadre, Kleene a introduit [26] la hiérarchie analytique ramifiée $(\mathrm{RAH}_\alpha)_{\alpha \in On}$ définie par l'équation

$$\mathrm{RAH}_\alpha \;:=\; \bigcup_{\beta<\alpha} \mathbf{Def^2}(\mathrm{RAH}_\beta)$$

où, pour chaque sous-ensemble X de $\mathcal{P}(\omega)$, $\mathbf{Def^2}(X)$ est le sous-ensemble de $\mathcal{P}(\omega)$ formé par tous les ensembles définissables au second ordre à partir de X, c'est-à-dire les ensembles d'entiers qui peuvent être définis à l'aide d'une formule du second ordre avec des paramètres du second ordre pris dans X et des quantifications du second ordre relativisées à X. Cette suite (transfinie) a été étudiée en profondeur dans le cadre de la théorie des ensembles, en particulier pour ses liens avec la théorie de la récursion [10], et pour ses conséquences dans l'étude des modèles de l'arithmétique du second ordre [2]. En particulier, son union transfinie (notée RAH) est un modèle de l'arithmétique du second ordre [10] qui satisfait l'axiome du choix et, plus généralement, une forme d'axiome de constructibilité.

Il est alors tout à fait naturel de se demander si tout ce travail peut être internalisé dans **PA2**. Cela permettrait de démontrer un résultat de cohérence relative : l'axiome de constructibilité n'est pas contradictoire en arithmétique du second ordre. Il semble que ce résultat n'ait pas été publié avant les travaux de Colson et Grigorieff [15] en 2001. Un résultat similaire a toutefois été publié pour l'analyse d'ordre supérieur (autrement dit, l'arithmétique d'ordre supérieur avec axiome de choix). L'histoire aurait dû s'arrêter là mais, au cours de ma thèse de master, j'ai

trouvé une erreur dans le travail de Colson et de Grigorieff (voir Section 5.5.1), montrant ainsi que la question de la cohérence relative de l'axiome de constructibilité dans **PA2** semble toujours ouverte (voir Open problem 2 page 113). Une autre question qui semble non résolue est de savoir s'il est possible de montrer de façon interne à **PA2** que RAH est un modèle de **PA2** (voir Open problem 3 page 113).

Dans les pages qui suivent, j'essaierai de répondre à ces questions. Bien que je ne vais pas donner une réponse définitive à ces problèmes, je présenterai une manière originale de les aborder. Mon but était de montrer au sein de **PA2** que RAH est un modèle de l'arithmétique du second ordre satisfaisant une variante de l'axiome de constructibilité. Une preuve de ce résultat est écrite dans les travaux de Colson et Griogorief mais elle utilise une propriété sur les bons ordres qui ne semble pas pouvoir être prouvée dans **PA2** (voir Section 5.5.1). Pour surmonter ce problème, on peut prouver un résultat plus faible : la classe RAH est un modèle de **PA2** satisfaisant l'axiome de choix. Mais, il semble que l'axiome du choix dénombrable soit nécessaire pour montrer que RAH satisfait le schéma d'axiomes de compréhension (de la logique du second ordre). Comme alternative, j'introduirai un nouveau schéma d'axiomes en logique du second ordre, inspiré de l'axiome de collection de la théorie des ensembles.

Ce nouveau principe est équivalent à l'axiome du choix dénombrable en présence de l'axiome d'induction. Néanmoins, sans l'axiome d'induction, il semble qu'il ne l'implique pas (mais je n'ai pas pu prouver ce résultat, voir Open problem 1 page 40). L'avantage de cet axiome est qu'il a un contenu calculatoire clair dans les modèles de réalisabilité de Krivine [30] : sa contraposée est réalisée par l'identité ! Dans les pages qui suivent, je vais travailler dans le cadre de la logique du second ordre enrichie de ce nouveau schéma d'axiomes et je vais montrer que RAH est un modèle de **PA2** qui satisfait l'axiome de choix. Je vais également montrer comment exprimer ce résultat sous la forme d'une traduction syntaxique. Le système source de cette traduction capturera la prouvabilité dans RAH et, en particulier, validera la logique classique et l'axiome du choix. Le système cible sera un système de type classique pour la logique du second ordre enrichi du schéma d'axiomes de collection. Comme le système cible peut être équipé d'une sémantique venant de la réalisabilité classique de Krivine, l'étude de cette traduction pourrait conduire à une interprétation calculatoire de l'axiome du choix dans le cadre de la logique classique du second ordre. Cependant, il s'agit là d'un développement ambitieux qui n'a pas encore été réalisé.

Un autre aspect de ce travail est la conception d'une théorie des bons préordres[1] en logique du second ordre enrichie de l'axiome de collection. En fait, la théorie des bons ordres dans **PA2** présente un inconvénient majeur : la classe des bons ordres n'est que bien préordonnée. C'est la principale différence avec les ordinaux en théorie des ensembles : ils ont une représentation canonique et sont donc bien ordonnés. Dans le cas de RAH, cet inconvénient impose l'utilisation de l'axiome du choix dénombrable dans la preuve du principe de réflexion. Un tel problème n'apparaît pas en théorie des ensembles ! Par conséquent, le fait que la classe des bons ordres n'est que bien préordonnée nous donne un indice pour surmonter cette difficulté : il suffit de considérer des bons préordres au lieu des bons ordres. Effectivement, le fait de travailler avec des bons préordres et le schéma d'axiomes de collection nous permettra de contourner l'utilisation de l'axiome du choix. Enfin, les bons préordres n'ont jamais été étudiés dans le cadre de l'arithmétique du second ordre : en présence de l'axiome d'induction, on peut construire un bon ordre à partir de n'importe quel bon préordre. Or, en arithmétique du second ordre sans induction, ce n'est pas le cas ; ce qui justifie cette approche et l'originalité de cette méthode. C'est dans un tel cadre (logique du second ordre enrichie de l'axiome de collection) que nous nous placerons dans la suite de ce manuscrit.

Les principales contributions de cette thèse sont listées ci-dessous.

[1] Dans la littérature, ils sont appelés prewellorderings [41].

1. On définit une nouvelle arithmétique basée sur les arbres binaires pures plutôt que les entiers et on l'utilise durant toute la première partie.

2. On formule un nouveau schéma d'axiomes en arithmétique du second ordre que j'appelle le schéma d'axiomes de collection, en référence à l'axiome portant le même nom en théorie des ensembles. On étudie son comportement par rapport à la réalisabilité de Krivine, à la réalisabilité de Kleene et à une traduction négative. On démontre qu'une variante de cet axiome est plus faible qu'une variante de l'axiome du choix en logique intuitionniste en exhibant un modèle de réalisabilité à la Kleene qui satisfait le premier et réfute le second.

3. On développe une théorie des bons préordres en arithmétique du second ordre sans induction et avec le schéma d'axiomes de collection.

4. On construit et on étudie la hiérarchie analytique ramifiée dans $\mathbf{PA2}^- + \mathbf{Coll}$. En particulier, on montre que c'est un modèle de **PA2** qui satisfait le principe de l'univers bien ordonné.

5. On explique pourquoi l'axiome du choix (ou l'axiome de collection) semble être nécessaire pour démontrer que RAH satisfait le schéma d'axiomes de compréhension.

Enfin, cette thèse est séparée en six chapitres dont les contenus sont décrits ci-dessous.

Le chapitre 1 est consacré à une présentation d'une arithmétique du second ordre originale, dans laquelle les termes du premier ordre représentent les arbres binaires purs. Ce choix atypique permettra de définir directement dans les individus de **PA2** tous les codages nécessaires (codes de Gödel d'une formule etc...) au développement de la hiérarchie analytique ramifiée, sans passer par les complications arithmétiques habituelles.

Le chapitre 2 introduit le schéma d'axiomes de collection en arithmétique du second ordre. Dans ce chapitre, je présenterai également une traduction par relativisation de **PA2** dans $\mathbf{PA2}^-$, montrant ainsi comment interpréter l'induction dans un système de type à la Curry où les quantifications ne sont pas relativisées. Enfin, à la fin de ce chapitre, je présenterai un modèle de réalisabilité classique pour $\mathbf{PA2}^-$ et je montrerai que ce modèle valide le schéma d'axiomes de collection.

Le chapitre 3 présente une étude de l'arithmétique du seconde ordre en logique intuitionniste. En particulier, dans ce chapitre, je présenterai une traduction négative de **PA2** vers **HA2** et je démontrerai que des variantes de l'axiome de collection et de l'axiome du choix sont validées par cette traduction. À la fin de ce chapitre, je présenterai un modèle de réalisabilité intuitionniste pour la théorie $\mathbf{HA2}^-$ et je démontrerai que ce modèle valide l'axiome de collection mais réfute une variante de l'axiome du choix.

Dans le chapitre 4, je présente les outils nécessaires à l'étude de la hiérarchie analytique ramifiée en logique du second ordre. En particulier, je développe la théorie des bons préordres dans $\mathbf{PA2}^- + \mathbf{Coll}$ et je montre comment définir des suites par récurrences transfinies dans ce système.

Le chapitre 5 est consacré à l'étude de la hiérarchie analytique ramifiée en logique du second ordre. Il commence par une internalisation de la notion de satisfaction qui servira immédiatement après à la définition de la suite transfinie $(\mathrm{RAH}_\alpha)_{\alpha\in\mathrm{WPO}}$. On démontre dans la théorie $\mathbf{PA2}^- + \mathbf{Coll}$ que RAH est un modèle de **PA2** satisfaisant le principe de l'univers bien ordonné. En suite, on proposera une étude de la littérature sur l'internalisation de la hiérarchie analytique ramifiée dans **PA2** et, en particulier, on commentera avec détails les travaux de Colson et Grigorieff à ce sujet. Enfin, on s'aventurera à esquisser une traduction entre un λ-calcul source à la Church et un λ-calcul cible à la Curry correspondant à la relativisation à la classe RAH. Le point essentiel de cette traduction repose sur une analyse calculatoire de la

preuve du schéma de réflexion. Cependant, je n'ai pas eu le temps de terminer cette analyse pendant ma thèse. Néanmoins, ce croquis de traduction logique permet déjà de donner une future application à l'étude de RAH : la conception d'un calcul intégrant de façon primitive la notion de constructibilité.

Le chapitre 6 est consacré à l'étude de l'égalité extensionnelle en arithmétique des types finis ($\mathbf{HA}^\omega$). L'arithmétique des types finis est une extension conservative de l'arithmétique de Heyting obtenue en étendant la syntaxe des termes à tout le Système T. Elle parle donc des fonctionnelles de types finis. Alors que l'égalité entre entiers naturels est spécifiée par les axiomes de Peano, comment peut être définie l'égalité entre fonctionnelles ? Différentes réponses à cette question poussent à l'étude de différentes variantes de $\mathbf{HA}^\omega$, telles qu'une variante extensionnelle ($\mathbf{E}$-$\mathbf{HA}^\omega$) où deux fonctionnelles sont égales si elles ont le même graphe. Dans cette partie, nous allons montrer comment l'étude d'une famille de relations d'équivalence partielle nous amène à définir une interprétation par paramétricité de l'égalité extensionnelle dans $\mathbf{HA}^\omega$.

Remerciements

Pour commencer, je souhaite remercier Laurent et Colin d'avoir accepté de rapporter ma thèse. Vos remarques sur mes travaux m'ont été précieuses. Plus généralement, je souhaite remercier tous les membres de mon jury de thèse.

Mes deux directeurs, Hugo et Alexandre, m'ont soutenu à chaque étape de ma thèse. Je me rends compte de la chance que j'ai eu de travailler avec vous. D'abord, sur le plan scientifique, vous avez été des modèles sur lesquels je me suis appuyé. Mais, en plus, vous avez toujours été disponibles pour moi et vous m'avez toujours apporté de l'aide (et de la compréhension) dans les moments plus difficiles de ma vie de thésard.

J'ai également eu la chance d'interagir avec d'autres chercheurs grâce à diverses collaborations. Je souhaite remercier toutes ces personnes avec lesquelles j'ai passé du temps face à un tableau, à faire des maths.

À l'IRIF, j'ai eu la chance de partager cette aventure avec de nombreux autres thésards. Merci pour ces moments passés ensemble.

Faire une (longue) thèse en cotutelle implique beaucoup de démarches administratives. Je souhaite remercier Amina Hariti, Delia Kesner et Lydia Tappa de m'avoir soutenu dans ces démarches. Sans votre efficacité et votre investissement, cette soutenance n'aurait sans doute pas été possible.

Merci aux membres du CSI pour votre appui lors de moments-clés de mon travail de thèse.

Ma famille m'a toujours soutenu dans cette aventure. Que cela doit être difficile de comprendre un thésard depuis l'extérieur ! Merci d'avoir cru en moi pendant toutes ces années.

Pendant une grosse période de ma thèse, j'ai vécu au boulevard de la Vanne. Merci aux colocs (et spécialement à Gromi) pour nos moments de vie commune. Cela me réchauffe le coeur de savoir qu'il y a de la place dans la bibliothèque du salon pour y faire croupir un manuscrit de ma thèse et perpétuer la tradition vanneuse.

À tous mes amis d'enfance ou d'ailleurs, merci ! Je pense en particulier aux copains de l'escalade qui m'ont aidé à me changer les idées.

Arriver dans un pays hispanophone avec un niveau d'espagnol très moyen, un accent à couper au couteau et seulement duolingo en poche, c'est en fait assez facile quand on rencontre les bonnes personnes. Je remercie sincèrement tous les amis que j'ai rencontrés en Uruguay. Damián a eu la patience de s'intéresser à mes recherches depuis le début, merci !

Encore en Uruguay, je souhaite remercier tout le MBC. Merci de m'avoir intégré si rapidement, vous avez été comme une deuxième famille pour moi.

Mais en parlant d'une deuxième famille, j'ai en fait maintenant une partie de ma famille en Uruguay. Antonella, nous avons passé de magnifiques moments de vie ensemble. Mais aussi, un quotidien si agréable : les oeufs brouillés ou la chapizzati du matin devant un dessin-animé, la pizza du dimanche soir, les sorrentinos du week-end... Et ce quotidien a été partagé avec deux êtres extrêmement importants qui m'ont soutenu pendant tous les moments de mon aventure scientifique. Merci à TehaNuma pour le soutien indéfectible, j'ai hâte de vous revoir. À la famille d'Antonella, dès les premiers moments, vous m'avez accepté et vous m'avez fait me sentir à l'aise. Merci à vous !

Cette année, j'enseigne au lycée. Après l'université, c'est un grand changement ! Merci à tous mes collègues de m'avoir aidé dans cette transition. Je pense particulièrement aux collègues de maths qui, en me partageant leurs ressources, m'ont permis de consacrer un peu plus de temps à la fin de ma thèse. À mes élèves des classes 206, 109 et Terminale SpéMaths, je tiens à vous dire que c'est un réel plaisir de partager cette année avec vous. Enseigner a toujours été une passion pour moi, et je m'estime chanceux de pouvoir la réaliser avec vous. J'espère pouvoir vous transmettre mon goût des mathématiques. Je vous souhaite à tous une belle réussite scolaire.

Agradecimientos

Me gustaría empezar dando las gracias a Laurent y Colin por haber aceptado de revisar mi tesis. Sus comentarios sobre mi trabajo han sido inestimables. En términos más generales, quiero dar las gracias a todos los miembros del jurado de mi tesis.

Mis dos supervisores, Hugo y Alexandre, me apoyaron en todas las etapas de mi tesis. Soy consciente de la suerte que he tenido de trabajar con ustedes. En primer lugar, desde un punto de vista científico, fueron modelos para mí. Pero, además, siempre estuvieron ahí para mí y siempre me prestaron ayuda (y comprensión) en los momentos más difíciles de mi vida como estudiante de doctorado.

También he tenido la oportunidad de interactuar con otros investigadores a través de diversas colaboraciones. Me gustaría dar las gracias a todas aquellas personas con las que he pasado tiempo delante de un pizarrón, haciendo matemáticas.

En el IRIF tuve la suerte de compartir esta aventura con muchos otros estudiantes de doctorado. Gracias por el tiempo que pasamos juntos.

Hacer una tesis en cotutela implica muchos trámites administrativos. Me gustaría dar las gracias a Amina Hariti, Delia Kesner y Lydia Tappa por apoyarme durante todo el proceso. Sin su eficacia y compromiso, esta defensa probablemente no habría sido posible.

Me gustaría dar las gracias a los miembros del CSI por su apoyo en momentos claves de mi tesis.

Mi familia siempre me ha apoyado en esta aventura. Qué difícil debe ser entender a un estudiante de doctorado desde fuera. Gracias por creer en mí durante toda mi tesis.

Durante gran parte de mi tesis, viví en el Boulevard de la Vanne. Gracias a los compañeros de piso (y especialmente a Gromi) por el tiempo que pasamos juntos. Me calienta el corazón saber que en la biblioteca del living hay sitio para que se pudra allí un manuscrito de mi tesis y se perpetúe la tradición de la Vanne.

A todos mis amigos de la infancia y de más allá, ¡gracias! Pienso en particular en mis amigos de la escalada, que me han permitido desconectar y distraerme.

Llegar a un país hispanohablante con un nivel de español mas o menos, un acento muy francés y sólo con duolingo en el bolsillo es en realidad bastante fácil cuando conoces a la gente adecuada. Mi más sincero agradecimiento a todos mis amiges de Uruguay. La primera persona que conocí fue Damián, y fue durante mi máster. Tuviste la paciencia por interesarte a mi investigación desde el principio, ¡gracias por esto!

Todavía en Uruguay, me gustaría dar las gracias a todo el MBC. Gracias por integrarme tan rápidamente, fueron como una segunda familia para mí.

Pero hablando de una segunda familia, en realidad incluso tengo una familia en Uruguay. Antonella, hemos pasado momentos maravillosos juntis. Pero también hemos tenido una rutina cotidiana muy linda: huevos revueltos o chapizzati por la mañana delante de un dibujito, pizza el domingo por la noche, sorrentinos el fin de semana... Y esta vida cotidiana la hemos compartido con dos seres importantísimos que me han apoyado durante todo el proceso. Gracias TehaNuma por su apoyo incondicional, estoy deseando volver a verles pronto. A la familia de Antonella, desde el primer momento me aceptaron y me hicieron sentir cómodo con ustedes. Muchas gracias.

Este año doy clases en el liceo. Después de la universidad, ¡es un gran cambio! Gracias a todos mis colegas por ayudarme en esta transición. Pienso en particular en mis colegas de matemáticas que, al compartir sus recursos conmigo, me han permitido dedicar un poco más de tiempo a terminar mi tesis. A mis alumnos de las clases 206, 109 y Terminale SpéMaths, quiero decirles que es un verdadero placer compartir este año con ustedes. Enseñar siempre ha sido una pasión para mí, y me siento afortunado de poder hacerlo con ustedes. Espero poder transmitirles mi gusto por las matemáticas. Les deseo a todos mucho éxito en la escuela.

Contents

Introduction

This thesis is made of two very different parts, and of different sizes, each of them equipped with its own introduction. The first part represents most of the work included in this report. It is concerned with second-order logic and with the ramified analytic hierarchy [26] in this framework. The second part deals with extension of equality in higher type arithmetic [49]. While these two subjects do not look connected, our manner of tackling them advocates for a way of studying proof systems. Many of the results spread in this thesis are described as a translation between two proof systems:

1. An interpretation of the axiom of induction in a proof system for second-order arithmetic without induction is presented as a translation between two typed λ-calculus in Section 2.2.3.

2. An interpretation of classical logic and of the axiom of induction in an intuitionistic proof system for second-order arithmetic without induction is presented as a translation between two typed λ-calculus in Section 3.2.1.

3. A translation interpreting the principle of the well-ordered universe in a proof system for second-order arithmetic without induction is sketched in Section 5.5.3 and remains as a future work.

4. Finally, the whole Chapter 6 is devoted to the design of a translation interpreting extensional equality in a proof system for higher type arithmetic with only equality on the type $\mathbf{N}$.

While these results are not new, the method to obtain them is at least original. Specifically, the first three interpretations translate a λ-calculus à la Church, where the proof terms are equipped with type information, into a λ-calculus à la Curry, where proof terms are raw λ-terms. The type annotations in the source system are consequently translated into programs that convey a computational meaning. These translations, replacing logical information by programs, seem to frequently appear through a process of relativization and our work shows examples of such phenomena.

Apart from these translations, the contributions of this manuscript are listed below.

1. We formulate a new arithmetic based on pure binary trees rather than integers (Chapter 1) and we use it throughout the whole first part.

2. We formulate a new axiom scheme in second-order arithmetic without induction that we call the axiom scheme of collection, inspired by its homonymous axiom from set theory. We study its behavior with respect to Krivine realizability, negative translations and Kleene realizability (Chapter 2 and 3). We strongly suspect that this new axiom scheme is weaker than the axiom scheme of choice (over the individuals) but we were not able to prove it. Thus, we formulate the following open problem.

   Open problem 1. *Is the axiom scheme of collection weaker than the axiom scheme of choice (over the individuals) in* $\mathbf{PA2}^-$*?*

3. We develop a theory of well-preorders in second-order arithmetic without induction and with the axiom scheme of collection (Chapter 4).

4. We construct and study the ramified analytic hierarchy inside $\mathbf{PA2}^- + \mathbf{Coll}$ (Chapter 5). Notably, we show that it is a model of $\mathbf{PA2}$ that satisfies the principle of the well-ordered universe.

5. We found a flaw in a paper of Colson and Grigorieff [15] and we explain why it led to the two following open problems (Section 5.5.1).

   Open problem 2. *Is the theory* **PA2** + V = **S**(RAH) *relatively consistent to the theory* **PA2***?*

   Open problem 3. *Can it be shown in* **PA2** *that* **S**(RAH) *models the axiom scheme of comprehension? Is the principle of reflection provable in* **PA2***?*

   We will explain how a solution of Open Problem 2 could be found by combining the works of Vetulani [51] and Simpson [47]. As for Open Problem 3, this thesis contains a proof of the principle of reflection in the theory $\mathbf{PA2}^- + \mathbf{Coll}$. However, we do not know how to tackle this problem in bare second-order arithmetic.

Last but not least, during my thesis, I participated to researches that I did not include in this manuscript [13, 5].

Part I

The ramified analytic hierarchy in second-order logic

Introduction

The axiom of choice, since it was introduced by Zermelo in 1904 [53, 6], has been important in many different areas of mathematics. Its consequences are wide and, sometimes, surprising, leading to discussions whether it should be accepted and used in mathematical proofs. On one hand, it is widely accepted in constructive mathematics [4] due to the fact that it has a clear computational content in intuitionistic logic [49, 33]. On the other hand, while it has been proven that its use (in classical set theory) does not lead to inconsistencies, the computational content of the full axiom of choice (in classical logic) remains unknown. In this manuscript, we will tackle the problem of its relative consistency in the framework of second-order logic, and we will give a path to the study of its computational content through a syntactic translation designed by the analysis of the ramified analytic hierarchy [26].

In 1938, Gödel showed the relative consistency of the axiom of choice and of the generalized continuum hypothesis with respect to the axioms of set theory (**ZF**) using the key concept of constructible sets [21]. His construction relies on an operator $X \mapsto \mathbf{Def}(X)$ that maps each sets X to the set $\mathbf{Def}(X)$ of the subsets of X that are definable using a first-order formula with parameters in X and quantifications restricted to X. By transfinitely iterating this operator, Gödel defines the sequence $(\mathrm{L}_\alpha)_{\alpha \in On}$ by the equation

$$\mathrm{L}_\alpha \ := \ \bigcup_{\beta<\alpha} \mathbf{Def}(\mathrm{L}_\beta)$$

from which the constructible universe (denoted L) is obtained as its transfinite union. Gödel then shows that the class L is an inner model of **ZF** that satisfies not only the axiom of choice but also the generalized continuum hypothesis. In fact, this model satisfies the axiom of constructibility (denoted $\mathrm{V} = \mathrm{L}$) that implies the former two statements. Notably, all these constructions (and proofs) can be done inside **ZF**, showing a result of relative consistency between the systems **ZF** + $\mathrm{V} = \mathrm{L}$ and **ZF**: adding the axiom of constructibility to set theory does not entail inconsistencies.

The ideas behind the construction of Gödel are very general and can be applied to other formalisms such as the study of subsets of ω. In this framework, Kleene introduced [26] the ramified analytic hierarchy $(\mathrm{RAH}_\alpha)_{\alpha \in On}$ defined by the equation

$$\mathrm{RAH}_\alpha \ := \ \bigcup_{\beta<\alpha} \mathbf{Def^2}(\mathrm{RAH}_\beta)$$

where, for each subset X of $\mathcal{P}(\omega)$, $\mathbf{Def^2}(X)$ is the subset of $\mathcal{P}(\omega)$ formed by all second-order definable sets from X, that is sets of integers that can be defined using a second-order formula with second-order parameters in X and second-order quantifications relativized to X. This sequence has been deeply studied in set theory, specifically its links with recursion theory [10], and its consequences in the realm of models of second-order arithmetic [2]. In particular, its transfinite union (denoted RAH) is a model of second-order arithmetic (**PA2**) that satisfies the axiom of choice and, more generally, a second-order form of the axiom of constructibility.

It is then very natural to ask if all this work can be internalized within **PA2** to obtain a result about the consistency of the axiom of constructibility in second-order arithmetic. However, it seems that this result was not published before the work of Colson and Grigorieff [15] in 2001. Notably, a similar result was published for higher-order analysis[2] in the PhD thesis of Vetulani in 1977 [51]. The story should have ended here but, during my master thesis, I found a flaw in the work of Colson and Grigorieff (see Section 5.5.1), thus showing that the question of the relative consistency of the axiom of constructibility in **PA2** seems to be still open[3] (see Open problem 2 page 113). An other question that looks to be unresolved is whether it is possible to

[2] In other words, higher-order arithmetic with the axiom of choice.

[3] At least, it seems that it is not written in the literature.

show that RAH models **PA2** in the framework of second-order arithmetic (see Open problems 3 page 113).

In the following pages, I will try to finally fill this gap between scientific folklore and scientific knowledge. To be frank, I could not achieve this goal as far as I wanted, but I will present an original way to tackle this problem. My goal was to show within **PA2** that RAH is a model of second-order arithmetic satisfying a second-order variant of the axiom of constructibility. The previously written proof of this fact used a result about well-orders that does not seem to be provable (see Section 5.5.1). To overcome this issue, one can prove the following weaker result: the class RAH satisfies second-order arithmetic enriched with a form of axiom of choice. But, it seems that the axiom of countable choice is necessary to show that RAH models the axiom scheme of comprehension. As an alternative, I will introduce a new axiom scheme for second-order logic, inspired by the axiom of collection of set theory.

This new scheme is equivalent to the axiom of countable choice in presence of the axiom of induction. Nevertheless, without the axiom of induction, it seems that it does not imply it but we were not able to prove it (Open problem 1). The advantage of this axiom is that it has a very clear computational content in Krivine realizability models [30]: its contrapositive is realized by the identity! In the following pages, I will work within second-order logic enriched with this new axiom scheme of collection and I will show that RAH is a model of **PA2** that satisfies the axiom of choice. As a future work, I want to extract a syntactic translation from this result of relative consistency. The source system will capture the provability in RAH and, in particular, will validate classical logic and the full axiom of choice. The target system will be a classical type system for second-order logic enriched with the axiom scheme of collection. Because the latter type system can be designed using Krivine's technique of classical realizability, the study of this translation could lead to a computational interpretation of the axiom of choice in the framework of classical second-order logic. However, it is an ambitious development that still remains as a future work.

Another aspect of this work is the design of a theory of well-preorders[4] [41] inside second-order logic enriched with the axiom scheme of collection. In fact, the theory of well-orders inside **PA2** is not so well-rounded and has one major drawback: the class of well-orders is only well-preordered. This is the main difference with ordinals in set theory, that have a canonical representation, and are thus well-ordered. In the case of RAH, this drawback enforces the use of the axiom of countable choice in the proof of the principle of reflection. Such a problem does not appear in set theory! Therefore, the fact that the class of well-orders is only well-preordered gives us the clue to overcome this difficulty: just consider well-preorders instead of well-orders. Notably, working with well-preorders and the axiom scheme of collection will allow us to circumvent the use of the axiom of choice. Finally, well-preorders were never studied in the framework of second-order arithmetic: in presence of the axiom of induction, one can construct a well-order from any well-preorders. However, in bare second-order logic, it is not the case, thus justifying this approach and the originality of this method[5]. It is in such a framework (second-order logic enriched with the axiom of collection) that we will place ourselves in the remainder of this manuscript.

[4] In the literature, they are called prewellorderings [41] but I will stick with the nomenclature "well-preorders" that I find adequate.

[5] As far as I know, it was never done.

Chapter 1

Second-order logic over the type of pure binary trees

In this chapter, we present a variation of second-order logic based on pure binary trees as individuals (Section 1.1). The set of pure binary trees is generated from the constant symbol 0 and the binary operator $\langle \_, \_ \rangle$. Because the pairing operator is primitive, it is enough to consider unary operations and unary predicates over the individuals. The syntax of **SOL** is equipped with a set of codes of unary functions that generate the primitive recursive operations over the type of binary trees. The meaning of these codes is specified by the axioms of computation (Definition 1.2.1.1) and, as examples, we show how they are used to generate operations over the (encodings of the) data types of natural numbers and of finite lists (Subsection 1.2.2). This is the framework in which we describe second-order arithmetic and various related systems (Section 1.2) as many-sorted first-order theories. In particular, we introduce the theory **PA2** (Subsection 1.2.1) of second-order arithmetic. Various subsystems arise from this theory by restricting the axiom scheme of comprehension to classes of formulas (Subsection 1.2.3). These subsystems, that have their own interest in the world of recursive mathematics [47], will be used in this thesis to take into account which instances of the axiom of comprehension are used in a proof.

1.1 Second-order logic

Second-order logic (**SOL**) is a first-order theory that distinguishes two kinds of objects:

1. The individuals, or first-order objects. In arithmetic, the integers traditionally serve as individuals. Other structures (formulas, recursive functions...) can then be defined by encoding (à la Gödel). In this thesis, we will rather use as individuals the type of pure binary trees, generated from the constructors 0 (zero) and $\langle \_, \_ \rangle$ (pairing binary operator). As a consequence, we will consider a version of second-order logic that comes equipped with function symbols to represent all primitive recursive functions over the type of pure binary trees. We will see that this choice makes possible the design of elegant encodings[1] and the representation of different structures (such as binary relations) inside the "second-order objects". In a system without induction, the primitive aspects of the pairing function and

[1] All the encodings done with pure binary trees can be done with integers by using Cantor's bijection from $\mathbb{N} \times \mathbb{N}$ to $\mathbb{N}$.

its projections will appear necessary to show that these operations behave as needed on non standard individuals.

2. The sets of individuals, or second-order objects. Because the set of individuals are in bijection with $\mathbb{R}$, the second-order objects are sometimes called "reals".

As a consequence, the language of **SOL** distinguishes two kinds of variables: one for individuals[2] (denoted $x, y, z...$) and an other for sets[3] (denoted $X, Y, Z...$).

1.1.1 Syntax of second-order logic

Codes of primitive recursive functions, terms and formulas

We will work with four syntactic categories:

1. the codes of primitive recursive functions
2. the first-order terms
3. the second-order terms
4. the formulas.

Definition 1.1.1.1. Function symbols, first-order terms, second-order terms and formulas of **SOL** are generated by the following grammars:

1. The function symbols of the language of **SOL** contain all the codes of primitive recursive functions over the type of pure binary trees, which are generated by the grammar:

$$\begin{array}{lll} \text{Codes of primitive} & f, g & ::= \quad \mathbf{0} \mid \text{id} \mid \text{fst} \mid \text{snd} \\ \text{recursive functions} & & \mid (f \circ g) \mid \langle f, g \rangle \mid [f|g] \end{array}$$

   The meaning of these codes will be specified by the axioms of computation inside the system **PA2** (Definition 1.2.1.1).

2. The first-order terms of **SOL** are defined by the grammar:

$$\text{Terms} \qquad t, u \quad ::= \quad x \mid 0 \mid \langle t, u \rangle \mid f(t)$$

   where x denotes a first-order variable and f is a code of primitive recursive function.

3. The second-order terms of **SOL** are generated from a set of (unary) second-order variables (denoted $X, Y, Z...$).

4. Formulas of **SOL** are defined by the grammar:

$$\text{Formulas} \qquad \phi, \psi \quad ::= \quad \bot \mid t = u \mid t \in X \mid \phi \Rightarrow \psi \mid \forall x \phi \mid \forall X \phi$$

   where t, u denote first-order terms while X is a second-order variable.

[2] They are the first-order variables.
[3] They are the second-order variables.

Remark 1.1.1.1. We will consider extensions of the language of second-order logic obtained by adding new constant symbols and new (unary) predicate symbols to the one already in **SOL**. These new symbols will be useful when we will study semantics of second-order logic (such as Tarski semantic, Krivine classical realizability and Kleene intuitionistic realizability).

Notation 1.1.1.1. The following notations will be used in the sequel:

1. We consider that the binary operation $\langle \_, \_ \rangle$ is right associative, and we lighten its use as follows:

$$\begin{array}{rcl} \langle t, u, v \rangle & \triangleq & \langle t, \langle u, v \rangle \rangle \\ \langle t, u, v, w \rangle & \triangleq & \langle t, \langle u, \langle v, w \rangle \rangle \rangle \quad (Etc.) \end{array}$$

2. A code of function f can be used as an operator of any positive arities thanks to the following convention:

$$\begin{array}{rcl} f(t, u) & \triangleq & f(\langle t, u \rangle) \\ f(t, u, v) & \triangleq & f(\langle t, u, v \rangle) \quad (Etc.) \end{array}$$

3. In a similar fashion, a second-order variable X can be seen of any positive arities:

$$(t_1, ...t_n) \in X \quad \triangleq \quad \langle t_1, ..., t_n \rangle \in X.$$

   This is why the syntax of **SOL** constructed over pure binary trees only contains unary second-order variable.

4. Finally, the connective $\Rightarrow$ is right associative:

$$\begin{array}{rcl} \phi \Rightarrow \psi \Rightarrow \chi & \triangleq & \phi \Rightarrow (\psi \Rightarrow \chi) \\ \phi \Rightarrow \psi \Rightarrow \chi \Rightarrow \upsilon & \triangleq & \phi \Rightarrow (\psi \Rightarrow (\chi \Rightarrow \upsilon)) \quad (Etc.) \end{array}$$

Definition 1.1.1.2. The other logical connectives and quantifiers are defined by the following shortcuts:

$$\begin{array}{rcl@{\qquad}rcl} \neg\phi & \triangleq & \phi \Rightarrow \bot & \phi \Leftrightarrow \psi & \triangleq & (\phi \Rightarrow \psi) \wedge (\psi \Rightarrow \phi) \\ \phi \wedge \psi & \triangleq & \neg(\phi \Rightarrow \psi \Rightarrow \bot) & \exists x\phi & \triangleq & \neg\forall x \neg\phi \\ \phi \vee \psi & \triangleq & \neg\phi \Rightarrow \neg\psi \Rightarrow \bot & \exists X\phi & \triangleq & \neg\forall X \neg\phi \\ t \neq u & :\triangleq & \neg(t = u) & t \notin X & \triangleq & \neg(t \in X) \end{array}$$

Remark 1.1.1.2. The aforementioned encodings are different from the usual impredicative encodings used in second-order logic. Actually, we use the primitive symbol $\bot$ (and the law of De Morgan verified by classical logic) to encode the other connectives in a first-order fashion. As a result, we can construct arithmetical formulas (i.e formulas without second-order quantifications) using all the logical connectives, the equality symbol and the first-order quantifications.

Definition 1.1.1.3. Given a term t, we write $FV(t)$ the set of its free variables and $t[x := u]$ the term obtained by substituting in t all the occurrences of x by u. Given a formula ϕ, we write $FV^1(\phi)$ the set of its free first-order variables, $FV^2(\phi)$ the set of its free second-order variables and $FV(\phi)$ the set of its free variables. Formulas are manipulated up to α-equivalence and we write $\phi[x := u]$ the capture free substitution of x by u. A closed formula will be called a sentence.

Set of individuals

In the syntax of **SOL**, the only second-order terms are the second-order variables. However, every formula ϕ with a distinguish variable x represents a set of individuals as suggested by the notation:

$$E \triangleq \{x \mid \phi\}.$$

This notation is treated as a binder that binds the variable x. The notions of free variables and first-order substitutions are extended to this new syntactical category:

$$\begin{array}{rcll} FV(\{x \mid \phi\}) & \triangleq & FV(\phi) \setminus \{x\} & \\ \{x \mid \phi\}[y := u] & \triangleq & \{x \mid \phi[y := u]\} & \text{if } x \neq y \text{ and } x \notin FV(u). \end{array}$$

A set $E \triangleq \{x \mid \phi\}$ can be used in a formula as follow:

$$t \in E \quad \triangleq \quad \phi[x := t] \qquad t \notin E \quad \triangleq \quad \neg\phi[x := t].$$

They are treated as second-order constants. We will call formula with parameters a formula in which second-order constants appear. Second-order variables can also be identified with sets: the variable X is associated to the set $\{x \mid x \in X\}$.

A first example of set is the set of pure binary trees

$$\mathbb{B} \triangleq \{z \mid \forall X(0 \in X \Rightarrow \forall x \forall y(x \in X \Rightarrow y \in X \Rightarrow \langle x, y\rangle \in X) \Rightarrow z \in X)\}.$$

In presence of the axiom scheme of induction, one can prove that all individuals are in this set. However, it is not always the case in **SOL**: models may contain more individuals, as for instance models arising from classical realizability.

In the sequel, we will use the following notations:

$$\begin{array}{rcl@{\qquad}rcl} (\forall x \in E)\phi & \triangleq & \forall x(x \in E \Rightarrow \phi) & (\exists x \in E)\phi & \triangleq & \exists x(x \in E \wedge \phi) \\ E = F & \triangleq & \forall x(x \in E \Leftrightarrow x \in F) & E \neq F & \triangleq & \neg(E = F) \\ E \subseteq F & \triangleq & \forall x(x \in E \Rightarrow x \in F) & E \not\subseteq F & \triangleq & \neg(E \subseteq F) \\ \emptyset & \triangleq & \{x \mid \bot\} & E \setminus F & \triangleq & \{x \mid x \in E \wedge x \notin F\} \\ E \cap F & \triangleq & \{x \mid x \in E \wedge x \in F\} & E \cup F & \triangleq & \{x \mid x \in E \vee x \in F\} \quad (Etc.) \end{array}$$

Finally, if t is a first-order term with free variables among $x_1, ..., x_n$, we use the shortcut :

$$\{t \mid \phi\} \quad :\triangleq \quad \{x \mid \exists x_1 ... \exists x_n(x = t \wedge x \in \phi)\}.$$

For instance, the cartesian product of two sets E and F can be written as

$$E \times F :\triangleq \{\langle x, y\rangle \mid x \in E \wedge y \in F\}.$$

A set E can also be seen as encoding a family of sets, using the concept of slices of a set.

Definition 1.1.1.4. The slice of a set E at the individual x is defined as the set

$$E[x] \quad \triangleq \quad \{y \mid \langle x, y\rangle \in E\}.$$

Therefore, a set E can also be seen as a function $x \mapsto E[x]$ from the set of individuals to the reals. This encoding will be extensively used in the Chapter 5 of this thesis, where higher-order structures will be represented as sets.

We now use the notion of sets to define second-order substitution.

Definition 1.1.1.5. Given a formula ϕ, a variable X and a set E, $\phi[X := E]$ is the formula obtained from ϕ by substituting all free occurrences of X by E. This operation extends to set definition:

$$\{x \mid \psi\}[X := E] \triangleq \{x \mid \psi[X := E]\}.$$

1.1.2 Deduction in classical second-order logic

We define the logical system that we will use in this chapter. Our presentation includes symbols for the equality $\_ = \_$ and for the false proposition $\bot$, even though they could be defined from the other connectives (using a second-order encoding). Furthermore, we include the rule of double negation elimination to capture classical logic. These choices yield arithmetical encodings of other connectives and quantifiers.

Natural deduction for classical second-order logic

Definition 1.1.2.1. A context is a set of formulas (denoted $\Gamma, \Theta ...$). Given two contexts Γ and Θ, the context Γ, Θ denotes the union of Γ and Θ. All the operations and notions defined at the level of formulas extend to contexts. For instance:

$$FV(\Gamma) \triangleq \bigcup_{\phi \in \Gamma} FV(\phi).$$

Definition 1.1.2.2. We define the relation $\Gamma \vdash \phi$ ("Γ proves ϕ") as the smallest relation between contexts and formulas generated by the inference rules described in Figure. 1.1 p. 23. In other words, the sequent $\Gamma \vdash \phi$ is provable in **SOL** if it can be derived using the rules of second-order classical natural deduction.

Axiom:	$\dfrac{}{\Gamma \vdash \phi}\ {\scriptstyle \phi \in \Gamma}$	
Implication:	$\dfrac{\Gamma, \phi \vdash \psi}{\Gamma \vdash \phi \Rightarrow \psi}$	$\dfrac{\Gamma \vdash \phi \Rightarrow \psi \quad \Gamma \vdash \phi}{\Gamma \vdash \psi}$
Fst order univ. quant.:	$\dfrac{\Gamma \vdash \phi}{\Gamma \vdash \forall x \phi}\ {\scriptstyle x \notin FV(\Gamma)}$	$\dfrac{\Gamma \vdash \forall x \phi}{\Gamma \vdash \phi[x := t]}$
Snd order univ. quant.:	$\dfrac{\Gamma \vdash \phi}{\Gamma \vdash \forall X \phi}\ {\scriptstyle X \notin FV(\Gamma)}$	$\dfrac{\Gamma \vdash \forall X \phi}{\Gamma \vdash \phi[X := Y]}$
Equality:	$\dfrac{}{\Gamma \vdash \forall x\ x = x}$	$\dfrac{\Gamma \vdash t = u \quad \Gamma \vdash \phi[x := t]}{\Gamma \vdash \phi[x := u]}$
Ex falso quodlibet, double negation elim.:	$\dfrac{\Gamma \vdash \bot}{\Gamma \vdash \phi}$	$\dfrac{\Gamma \vdash \neg\neg\phi}{\Gamma \vdash \phi}$

Figure 1.1: Rules of second-order classical natural deduction

Remark 1.1.2.1. The rule of elimination of the second-order quantifiers is restricted to second-order variables because variables are the only primitive second-order terms in the syntax of **SOL**. Nevertheless, more general rules of elimination can be retrieved in axiomatic theories containing the axioms scheme of comprehension (definition 1.2.1.1 and proposition 1.2.1.2). Actually, as we will see later, these rules depend on the sets that can be shown to exist and therefore, it depends of the ambient theory.

Definition 1.1.2.3. The intuitionistic fragment of **SOL** is obtained by removing the rule of double negation elimination from the rules of second-order classical natural deduction.

Remark 1.1.2.2. To show that a sequent $\Gamma \vdash \phi$ is provable in **SOL**, one should exhibit a proof tree. However, most of the time, we will write a proof in English to convince the reader that $\Gamma \vdash \phi$ (as in the proof of proposition 1.1.2.1).

Admissible and derivable rules

Definition 1.1.2.4. An inference rule is admissible if its conclusion is provable whenever its premises are provable.

Definition 1.1.2.5. An inference rule is derivable if its conclusion is in the smallest relation between contexts and formulas generated from the deduction rules of classical second-order natural deduction extended with its premises as axioms. It means that its conclusion can be derived using the rules of **SOL** and its hypotheses.

Proposition 1.1.2.1. *The rules presented in Figure 1.2 p.24 are admissible. Moreover, they are all derivable except the rule of weakening.*

Proof. The proof that the weakening rule is admissible uses an induction over the proof tree of its premise. It is not derivable: a counter-example can be found by choosing Γ to be empty, Γ' to contain only a tautology and ϕ to be an undecidable formula of **SOL**.

All the other rules are derivable. As an example, we prove the introduction rule of the conjunction. Assume $\Gamma \vdash \phi$ and $\Gamma \vdash \psi$. We need to show $\Gamma \vdash \phi \wedge \psi$, i.e. $\Gamma \vdash \neg(\phi \Rightarrow \psi \Rightarrow \bot)$. But, with $\phi \Rightarrow \psi \Rightarrow \bot$ in the context, the hypothesis allows us to derive $\bot$ (using two times the rule of elimination of implication and the rule of weakening). □

Weakening:		$\dfrac{\Gamma \vdash \phi}{\Gamma' \vdash \phi}\,\Gamma \subseteq \Gamma'$	
Conjunction:	$\dfrac{\Gamma \vdash \phi \quad \Gamma \vdash \psi}{\Gamma \vdash \phi \wedge \psi}$	$\dfrac{\Gamma \vdash \phi \wedge \psi}{\Gamma \vdash \phi}$	$\dfrac{\Gamma \vdash \phi \wedge \psi}{\Gamma \vdash \psi}$
Disjunction:	$\dfrac{\Gamma \vdash \phi}{\Gamma \vdash \phi \vee \psi}$ $\dfrac{\Gamma \vdash \psi}{\Gamma \vdash \phi \vee \psi}$	$\dfrac{\Gamma \vdash \phi \vee \psi \quad \Gamma, \phi \vdash \chi \quad \Gamma, \psi \vdash \chi}{\Gamma \vdash \chi}$	
Fst order exist.:	$\dfrac{\Gamma \vdash \phi[x := t]}{\Gamma \vdash \exists x \phi}$	$\dfrac{\Gamma \vdash \exists x \phi \quad \Gamma, \phi \vdash \psi}{\Gamma \vdash \psi}\, x \notin FV(\Gamma, \psi)$	
Snd order exist.:	$\dfrac{\Gamma \vdash \phi[X := Y]}{\Gamma \vdash \exists X \phi}$	$\dfrac{\Gamma \vdash \exists X \phi \quad \Gamma, \phi \vdash \psi}{\Gamma \vdash \psi}\, X \notin FV(\Gamma, \psi)$	

Figure 1.2: Rules of conjunction, disjunction and existential quantifiers.

Remark 1.1.2.3. Note that the rule of introduction of the second-order existential quantifier is restricted to variables. As for the elimination of the second-order universal quantifier, this rule is parameterized by the ambient theory and will grow more expressive as axioms are added to the theory. However, this version already proves the existence of a set as $\vdash \exists X X = X$ is derivable.

Second-order equality

First-order equality and second-order equality are treated differently. Indeed, the former is primitive while the latter is not. In **SOL**, two sets E_1 and E_2 will be equal if they are extensionnally equal:

$$E_1 = E_2 \triangleq \forall z(z \in E_1 \Leftrightarrow z \in E_2).$$

This binary relation is reflexive and it verifies Leibniz principle saying that equal objects should satisfy the same properties, as shown in the next theorem.

Theorem 1.1.2.1. *For every formula $\phi(\vec{x}, X, \overrightarrow{Y})$ with free variables included in $\vec{x}, X, \overrightarrow{Y}$, if E and V are two extensionally equal sets, then $\phi[X := E]$ is equivalent to $\phi[X := V]$. Formally, this proposition is a scheme of propositions, for all formulas $\phi(\vec{x}, X, \overrightarrow{Y})$ and sets E_1, E_2:*

$$\vdash \forall \vec{x} \forall \overrightarrow{Y}(E_1 = E_2 \Rightarrow \phi[X := E_1] \Rightarrow \phi[X := E_2]).$$

Proof. The proof uses an external induction over the syntax of the formulas. We only treat the cases where X appears freely in ϕ.

1. If ϕ is atomic, it is of the form $t \in X$ and $t \in E_1 \Rightarrow t \in E_2$ immediately follows from $E_1 = E_2$.
2. If $\phi \triangleq \psi \Rightarrow \chi$ or $\phi \triangleq \forall x \psi$, the result follows from the induction hypothesis.
3. If $\phi \triangleq \forall X' \psi$ with X and X' being two distinct variables, the induction hypothesis is:

   $$\forall \vec{x} \forall X' \forall \overrightarrow{Y}(E_1 = E_2 \Rightarrow (\psi[X := E_1] \Rightarrow \psi[X := E_2]).$$

   Now assume $\phi[X := E_1]$ (for fresh variables $\vec{x}, \overrightarrow{Y}$). Eliminating the quantifier over X' with the variable X', we first obtain $\psi[X := E_1]$ and then, by the induction hypothesis, we get $\psi[X := E_2]$. Finally, because X' only appears bound in the context, it can be generalized (using the introduction rule of second-order quantification) to deduce $\forall X' \psi[X := E_2] \triangleq \phi[X := E_2]$.

□

As a consequence, the rules of introduction and of elimination of the second-order equality are derivable in **SOL**:

$$\frac{}{\Gamma \vdash \forall X X = X} \qquad \frac{\Gamma \vdash E_1 = E_2 \quad \Gamma \vdash \phi[X := E_1]}{\Gamma \vdash \phi[X := E_2]}$$

1.1.3 Theories and models

Rather than working in plain **SOL** only, we will be interested in the study of first-order theories in the language of second-order logic (with equality). In the following, we will briefly introduce the notions of theory and model (à la Tarski) before starting to study various different examples of theories.

Theories

Definition 1.1.3.1. A theory $\mathcal{T}$ over an extension of the language of **SOL** is a set of sentences constructed from the syntax of **SOL** potentially extended with predicate symbols and function symbols. We say that a theory proves ϕ in the context Γ and write $\mathcal{T} + \Gamma \vdash \phi$ if there is a finite set $\mathcal{T}_0 \subseteq \mathcal{T}$ such that the sequent $\mathcal{T}_0, \Gamma \vdash \phi$ is derivable from the rules of second-order classical natural deduction. If $\mathcal{T} \vdash \phi$, we say that ϕ is a consequence of $\mathcal{T}$.

Intuitively, when working with a theory $\mathcal{T}$, one assumes that the formulas $\phi \in \mathcal{T}$ are "true": they are the axioms of the theory $\mathcal{T}$. We are interested in the consequences of the theory $\mathcal{T}$.

Definition 1.1.3.2. A theory $\mathcal{T}_1$ is an extension of a theory $\mathcal{T}_2$ if $\mathcal{T}_2 \subseteq \mathcal{T}_1$. Therefore, all consequences of $\mathcal{T}_2$ are also consequences of $\mathcal{T}_1$.

Definition 1.1.3.3. A theory $\mathcal{T}$ is consistent if it does not prove the formula $\bot$, equivalently it is consistent if it does not prove all the formulas of its language.

Inconsistent theories are degenerated and therefore it is an important property of a theory $\mathcal{T}$ to be consistent. However, from Gödel's incompleteness theorem, it is known that the consistency of a theory cannot be proved inside it.

Theorem 1.1.3.1 (Gödel's incompleteness theorem)**.** *If $\mathcal{T}$ is a consistent and recursively axiomatizable theory that contains Peano arithmetic, then the formula Cons($\mathcal{T}$) (stating the consistency of $\mathcal{T}$ in the language of $\mathcal{T}$) is not a consequence of $\mathcal{T}$.*

Therefore, if we prove the consistency of a theory $\mathcal{T}$, we know that we are working in a "stronger" theory than the initial theory. In this case, we proved a result of relative consistency between two theories.

Definition 1.1.3.4. A theory $\mathcal{T}_1$ is relatively consistent to a theory $\mathcal{T}_2$ if the consistency of $\mathcal{T}_2$ implies the consistency of $\mathcal{T}_1$.

Results of relative consistency can be proved using syntactical translations[4] and a weak metatheory such as primitive recursive arithmetic (**PRA**). Moreover, when designing such translations, one can also focus on how the translation acts on proofs. Such concerns, coupled with the Curry-Howard correspondence, led to link consistency proofs to translations of programs [36].

Tarski semantic of first-order logic

Definition 1.1.3.5. A structure $\mathfrak{S}$ of second-order logic over a language $\mathcal{L}$ (potentially extending the language of **SOL**) is given by the data of:

1. a non-empty set $\mathfrak{S}_1$ to interpret individuals
2. a non-empty set $\mathfrak{S}_2$ to interpret second-order objects
3. functions and predicates to interpret all symbols of the language (in particular, including interpretations for $\_ = \_$ and $\_ \in \_$):
   - For each other constant symbol c, an element $[\![c]\!] \in \mathfrak{S}_1$.
   - A function $[\![\langle \_, \_ \rangle]\!] : {\mathfrak{S}_1}^2 \to \mathfrak{S}_1$ to interpret $\langle \_, \_ \rangle$.

[4] In particular, a semantical approach characterized by the construction of a model can (in general) be reformulated as a syntactic translation. For instance, it is the case for all inner models of set theory [23].

- For each unary function symbol f, a function $[\![f]\!] : \mathfrak{S}_1 \to \mathfrak{S}_1$.
- A subset $[\![=]\!] \subseteq \mathfrak{S}_1 \times \mathfrak{S}_1$ to interpret $=$.
- A subset $[\![\in]\!] \subseteq \mathfrak{S}_1 \times \mathfrak{S}_2$ to interpret $\in$.
- For each unary predicate symbol P, an element $[\![P]\!] \in \mathfrak{S}_2$.

Remark 1.1.3.1. Because the logic of **SOL** only quantifies over unary second-order variables, it is enough to consider only two sorts to define the notion of structures for this language.

A structure $\mathfrak{S}$ over a language $\mathcal{L}$ contains all the necessary information to interpret closed terms constructed from the symbols of $\mathcal{L}$, it is done recursively as follow:

$$\begin{array}{rcl} [\![c]\!] & \triangleq & [\![c]\!] \\ [\![\langle t, u\rangle]\!] & \triangleq & [\![\langle \_, \_\rangle]\!]([\![t]\!], [\![u]\!]) \\ [\![f(t)]\!] & \triangleq & [\![f]\!]([\![t]\!]). \end{array}$$

Remark 1.1.3.2. In the interpretation of the symbols of the language, the function $\langle \_, \_\rangle$ and the predicates $\_ = \_$ and $\_ \in \_$ are treated differently from the other symbols of functions and predicates. Indeed, they are the only symbols interpreted as binary relations!

Given a structure $\mathfrak{S}$ over $\mathcal{L}$, we extend $\mathcal{L}$ to a language $\mathcal{L}_{\mathfrak{S}}$ that contains constant symbols $\dot{c}$ and $\dot{C}$ for all elements $c \in \mathfrak{S}_1$ and $C \in \mathfrak{S}_2$. These new constants will be interpreted in $\mathfrak{S}$ by the elements they denote. A formula with parameters in $\mathfrak{S}$ is then a formula of the language $\mathcal{L}_{\mathfrak{S}}$. From these concepts, we define the notion of satisfiability.

Definition 1.1.3.6. We define by external induction over formulas the notion of satisfiability between structures $\mathfrak{S}$ and closed formulas ϕ of the language $\mathcal{L}_{\mathfrak{S}}$ ($\mathfrak{S} \vDash \phi$):

$$\begin{array}{lcll} \mathfrak{S} & \nvDash & \bot & (\bot \text{ is not satisfied}) \\ \mathfrak{S} & \vDash & t = u \quad \text{if} & ([\![t]\!], [\![u]\!]) \in [\![=]\!] \\ \mathfrak{S} & \vDash & t \in P \quad \text{if} & ([\![t]\!], [\![P]\!]) \in [\![\in]\!] \\ \mathfrak{S} & \vDash & \phi \Rightarrow \psi \quad \text{if} & \mathfrak{S} \vDash \phi \text{ implies } \mathfrak{S} \vDash \psi \\ \mathfrak{S} & \vDash & \forall x \phi \quad \text{if} & \text{for all elements } c \in \mathfrak{S}_1, \mathfrak{S} \vDash \phi[x := \dot{c}] \\ \mathfrak{S} & \vDash & \forall X \phi \quad \text{if} & \text{for all elements } C \in \mathfrak{S}_2, \mathfrak{S} \vDash \phi[x := \dot{C}] \end{array}$$

In this definition, the meaning of a symbol (function symbols, predicate symbols but also connectives and quantifiers) is lifted from the syntax to the meta-theory. The interpretation of a symbol in the meta-theory is called its semantic.

Definition 1.1.3.7. A structure $\mathcal{S}$ is a model of a theory $\mathcal{T}$ if it satisfies all sentences ϕ of the theory $\mathcal{T}$.

The following theorems relate syntax and semantic.

Theorem 1.1.3.2 (Soundness)**.** *If a theory $\mathcal{T}$ proves a closed formula ϕ, then ϕ is satisfied by all models of $\mathcal{T}$.*

Theorem 1.1.3.3 (Gödel's completeness theorem)**.** *If a theory is consistent, then it has a model.*

In particular, the former gives a way to show that a formula ϕ is not provable in a theory $\mathcal{T}$: it is enough to find a model of $\mathcal{T} \cup \{\neg\phi\}$!

Remark 1.1.3.3. Although we are studying "second-order" logic in this thesis, we formalized it as a many-sorted first-order language. In particular, techniques and theorems from first-order logic also apply in this framework.

1.2 Variations of arithmetic in second-order logic

After introducing the syntax and the rules of second-order logic, we continue with the study of second-order theories. The arguably most famous theory of second-order logic is second-order arithmetic (**PA2**). However, many variations of **PA2** are of interest. For instance, subsystems of **PA2** are deeply studied in the context of reverse mathematics [47]. Other variations occur in the realm of classical realizability [30] where a second-order arithmetic without the axiom of induction is considered ($\mathbf{PA2}^-$). At last, extensions of **PA2** are also objects of interest. Various axioms of choice can be formulated in **SOL** and were intensively studied in analysis and in realizability. In this thesis, we will introduce the axiom scheme of collection for **SOL**. While it collapses to the axiom of countable choice in the presence of induction, it is of great interest in $\mathbf{PA2}^-$ thanks to its very simple realizability interpretation and to its importance in the use of well-preorders.

1.2.1 Second-order arithmetic (PA2)

The axioms of PA2

Definition 1.2.1.1. The axioms of **PA2** are the universal closures of the following formulas:

Axioms of injectivity and of non confusion:

- $\langle x, y\rangle = \langle x', y'\rangle \Rightarrow (x = x' \wedge y = y')$
- $\langle x, y\rangle \neq 0$

Axioms of computation:

- $\mathbf{0}(x) = 0$
- $\mathrm{id}(x) = x$
- $\mathrm{fst}(0) = 0$
- $\mathrm{snd}(0) = 0$
- $\mathrm{fst}(x, y) = x$
- $\mathrm{snd}(x, y) = y$
- $(f \circ g)(x) = f(g(x))$
- $\langle f, g\rangle(x) = \langle f(x), g(x)\rangle$
- $[f|g](0) = 0$
- $[f|g](x, 0) = f(x)$
- $[f|g](x, y, z) = g(x, y, z, [f|g](x, y), [f|g](x, z))$

where f and g are codes of primitive recursive functions.

Axiom of induction:

- $0 \in X \Rightarrow \forall x \forall y(x \in X \Rightarrow y \in X \Rightarrow \langle x, y\rangle \in X) \Rightarrow \forall x\ x \in X$

Axiom scheme of comprehension:

- $\exists X \forall x(x \in X \Leftrightarrow \phi)$ which may be written $\exists X\ X = \{x \mid \phi\}$

where ϕ is a formula that does not contain the variable X.

Remark 1.2.1.1. The restriction in the axiom scheme of comprehension is actually necessary: the system would be inconsistent otherwise! Actually, taking ϕ to be $\neg(x \in X)$, the axiom would entail $\bot$ as it is a consequence of $0 \in X \Leftrightarrow 0 \notin X$. In fact, it is a more general fact of intuitionistic logic that a formula ϕ cannot be equivalent to its own negation in a consistent system.

The case of the full second-order universal quantifier elimination

Definition 1.2.1.2. The "full" elimination for the second-order universal quantifier and the "full" introduction for the second-order existential quantifier are described by the following inference rules:

$$\frac{\Gamma \vdash \forall X \phi}{\Gamma \vdash \phi[X := E]} \qquad \frac{\Gamma \vdash \phi[X := E]}{\Gamma \vdash \exists X \phi}$$

Proposition 1.2.1.1. *In* **SOL***, the rule of full elimination for the second-order universal quantifier implies the rule of full introduction for the second-order existential quantifier.*

Proof. It follows from the encoding of $\exists X \phi$ as $\neg(\forall X \neg \phi)$. □

Proposition 1.2.1.2. *The scheme of comprehension implies the rule of full elimination of the second-order universal quantifier. Conversely, all the instances of the scheme of comprehension can be derived from this rule (and the other rules of second-order natural deduction).*

Proof. Let $E \triangleq \{x \mid \psi\}$ and assume $\forall X \phi$. We want to prove $\phi[X := E]$. But, the scheme of comprehension entails the existence of a variable Y that is extensionally equal to the set E and therefore Theorem 1.1.2.1 gives $\phi[X := E] \Leftrightarrow \phi[x := Y]$. However, $\phi[X := Y]$ follows from the restricted second-order universal quantifier elimination and we can conclude $\phi[X := E]$.

In the other direction, the scheme of comprehension is a consequence of the rule of full introduction for the second-order existential quantifier. Indeed, since $\phi \Leftrightarrow \phi$ is a tautology, $(\forall x(x \in X \Leftrightarrow \phi))[X := \{x|\phi\}]$ is provable and therefore $\exists X \forall x(x \in X \Leftrightarrow \phi)$ follows. □

The standard model of PA2

Let $\mathbf{B}$ be the set of pure binary trees (in the meta-theory[5]).

The standard model $\mathfrak{M}$ of **PA2**, or full model of **PA2**, is defined as follow:

- The individuals are interpreted in $\mathbf{B}$, i.e $\mathfrak{M}_1 \triangleq \mathbf{B}$.
- The codes of primitive recursive functions are interpreted by the functions they denote.
- The second-order variables are interpreted in the power set of $\mathbf{B}$, i.e $\mathfrak{M}_2 \triangleq \mathcal{P}(\mathbf{B})$.
- The symbols of equality $=$ and of membership $\in$ are interpreted as the equality relation over $\mathbf{B}$ and the membership relation between individuals $b \in \mathbf{B}$ and sets $A \in \mathcal{P}(\mathbf{B})$.

The structure $\mathfrak{M}$ is a model of **PA2** where all the constructs are given their intended meaning. In $\mathfrak{M}$, the reals range over the full power set of $\mathbb{B}$, this is why it is sometimes called the full model of second-order arithmetic.

1.2.2 Expressiveness of the axioms of computations, examples of the finite lists and of the natural numbers

In arithmetic, all the usual finitely generated data structures can be encoded. We now study the example of finite lists and natural numbers.

1. Finite lists are encoded by setting

$$[t_1; ...; t_n] \triangleq \langle t_1, ..., t_n, 0 \rangle$$

(i.e by using the constructor 0 as a symbol of end of list). In **PA2**, thanks to the axiom of induction, all individuals can be seen as lists. However, in a more general framework, only the individuals in the set $\mathbb{B}$ can be seen as lists.

[5] It can be defined as an inductive type with constructors $0 : \mathbf{B}$ and $\langle \_, \_ \rangle : \mathbf{B} \to \mathbf{B} \to \mathbf{B}$.

2. Every natural number n is represented as the following degenerate pure binary tree of height n:

$$\underline{n} \triangleq \langle \underbrace{0, ...\langle 0, 0 \rangle ...}_{n+1} \rangle = \langle \underbrace{0, ...0, 0}_{n+1} \rangle = [\underbrace{0, ..., 0}_{n}].$$

This definition can be made formal by an external recursion over the (meta) natural numbers:

$$\begin{array}{rcl} \underline{0} & \triangleq & 0 \\ \underline{n+1} & \triangleq & \langle 0, \underline{n} \rangle. \end{array}$$

In the rest of the manuscript, we will simply write n to denote the encoding of a natural number in the syntax.

We now focus on giving examples of primitive recursive functions.

Remark 1.2.2.1. After this chapter, we choose to stop writing in the syntax of **SOL** the codes of primitive recursive functions that we use. We will rather give the equations that they satisfy, it will then be clear that they can be expressed in our syntax.

Operations over the type of integers

We define some operations over the natural numbers that we can express in **PA2**:

1. The successor function is defined by $\mathtt{s} \triangleq \langle \mathbf{0}, \mathrm{id} \rangle$. It satisfies :

$$\mathbf{PA2} \vdash \forall n \ \mathtt{s}(n) = \langle 0, n \rangle$$

2. The function `check_nat` that tests if an individual is an integer is defined by:

$$\mathtt{check\_nat} \triangleq [\langle \mathbf{0}, \mathbf{0} \rangle | [\mathrm{id} | \mathbf{0}] \circ \langle \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd}, \mathrm{fst} \circ \mathrm{snd} \rangle] \circ \langle \mathbf{0}, \mathrm{id} \rangle$$

It satisfies:

$$\begin{array}{l} \mathbf{PA2} \vdash \mathtt{check\_nat}(0) = 1 \\ \mathbf{PA2} \vdash \mathtt{check\_nat}(0, x) = \mathtt{check\_nat}(x) \\ \mathbf{PA2} \vdash \mathtt{check\_nat}(\langle y, z \rangle, x) = 0 \end{array}$$

We define the set of natural numbers as $\mathbb{N} \triangleq \{x \mid \mathtt{check\_nat}(x) = 1\} \cap \mathbb{B}$. It is obtained as all the pure binary trees x such that $\mathtt{check\_nat}(x) = 1$. However, because we will not work with the axiom of induction, we need to ensure that the variable x denotes a pure binary tree and that is why the unrelativzed set $\{x \mid \mathtt{check\_nat}(x) = 1\}$ needs to be intersected with $\mathbb{B}$.

Remark 1.2.2.2. The set of "internal" integers and the set of "meta"-integers are both denoted $\mathbb{N}$. The (meta) context should be enough to clarify if we speak of the former or the latter.

Proposition 1.2.2.1. *A principle of induction over natural numbers is provable in* **PA2***:*

$$\mathbf{PA2} \vdash \forall X (0 \in X \Rightarrow (\forall n \in \mathbb{N})(n \in X \Rightarrow \mathtt{s}(n) \in X) \Rightarrow (\forall n \in \mathbb{N}) \, n \in X)$$

Proof. The proof uses induction on the set $\mathbb{B}$ and the properties of the function `check_nat`. □

3. Addition over natural numbers is defined by:

$$\mathtt{add} \triangleq [\mathrm{id}|\langle \mathrm{fst} \circ \mathrm{snd}, \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd}\rangle] \circ \langle \mathrm{snd}, \mathrm{fst}\rangle$$

We use the notation $x + y$ for $\mathtt{add}(x, y)$. The function $\mathtt{add}$ satisfies :

$$\begin{array}{ll} \mathbf{PA2} & \vdash \ \forall n\ 0 + n = n \\ \mathbf{PA2} & \vdash \ \forall n \forall m\ \mathtt{s}(n) + m = \mathtt{s}(n + m) \\ \mathbf{PA2} & \vdash \ (\forall n \in \mathbb{N})(\forall m \in \mathbb{N})\ n + m = m + n \\ \mathbf{PA2} & \vdash \ \forall n \forall m \forall p\ n + (m + p) = (n + m) + p \end{array}$$

Using the function $\mathtt{add}$, we define an order on the set of integers and its associated strict order:

$$n \leq_\omega m \triangleq \exists x(x + n = m) \qquad \text{and} \qquad n <_\omega m \triangleq n \leq_\omega m \wedge n \neq m.$$

We will see in Chapter 4 that it is a well-order on the set $\mathbb{N}$.

Remark 1.2.2.3. Formally, $\leq_\omega$ is defined (by comprehension) to be the set

$$\leq_\omega \triangleq \{\langle n, m\rangle \mid n \in \mathbb{N} \wedge m \in \mathbb{N} \wedge \exists x(x + n = m)\}.$$

4. Multiplication over integers is defined by:

$$\mathtt{mult} \triangleq [\mathbf{0}|\mathtt{add} \circ \langle \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd}, \mathrm{fst}\rangle] \circ \langle \mathrm{snd}, \mathrm{fst}\rangle$$

We use the notation $x \times y$ for $\mathtt{mult}(x, y)$. The function $\mathtt{mult}$ satisfies:

$$\begin{array}{ll} \mathbf{PA2} & \vdash \ \forall n\ 0 \times n = 0 \\ \mathbf{PA2} & \vdash \ (\forall n \in \mathbb{N})(\forall m \in \mathbb{N})\ \mathtt{s}(n) \times m = (n \times m) + m \end{array}$$

5. The sum of the integers between 0 up to n is defined by the function:

$$\Sigma(n) \triangleq [\langle \mathbf{0}, \mathbf{0}\rangle|\mathrm{add} \circ \langle\langle \mathrm{fst}, \langle \mathrm{fst} \circ \mathrm{snd}, \mathrm{fst} \circ \mathrm{snd} \circ \mathrm{snd}\rangle\rangle, \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd}\rangle]$$

It satisfies:

$$\begin{array}{ll} \mathbf{PA2} & \vdash \ \Sigma(0) = 0 \\ \mathbf{PA2} & \vdash \ (\forall n \in \mathbb{N})\ \Sigma(\mathtt{s}(n)) = \mathtt{s}(n) + \Sigma(n) \end{array}$$

6. Cantor's bijection from $\mathbb{N} \times \mathbb{N}$ to $\mathbb{N}$ is defined by:

$$\alpha_2 \triangleq \mathtt{add} \circ \langle \Sigma \circ \mathtt{add} \circ \langle \mathrm{fst}, \mathrm{snd}\rangle, \mathrm{snd}\rangle$$

It satisfies:

$$\begin{array}{ll} \mathbf{PA2} & \vdash (\forall n \in \mathbb{N})(\forall m \in \mathbb{N})\ \alpha_2(n, m) = \Sigma(n + m) + m \\ \mathbf{PA2} & \vdash (\forall n \in \mathbb{N})(\forall m \in \mathbb{N})(\forall n' \in \mathbb{N})(\forall m' \in \mathbb{N}) \\ & \qquad \alpha_2(n, m) = \alpha_2(n', m') \Rightarrow (n = n' \wedge m = m') \\ \mathbf{PA2} & \vdash (\forall n \in \mathbb{N})(\exists m \in \mathbb{N})(\exists p \in \mathbb{N})\ \alpha_2(m, p) = n \end{array}$$

7. We define a bijection $\mathbf{h}$ from the pure binary trees to the integers by:

$$\mathbf{h} \triangleq [\mathbf{0}|\mathtt{add} \circ \langle\langle \mathbf{0}, \mathbf{0}\rangle, \alpha_2 \circ \langle \mathrm{fst} \circ \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd}, \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd}\rangle\rangle] \circ \langle \mathbf{0}, \mathrm{id}\rangle$$

It satisfies:

$$\begin{array}{l}\mathbf{PA2} \vdash \ \mathbf{h}(0) = 0\\ \mathbf{PA2} \vdash \forall x \forall y \ \mathbf{h}(x, y) = 1 + \alpha_2(\mathbf{h}(x), \mathbf{h}(y))\\ \mathbf{PA2} \vdash \ \forall x \forall y (\mathbf{h}(x) = \mathbf{h}(y) \Rightarrow x = y)\\ \mathbf{PA2} \vdash (\forall n \in \mathbb{N}) \exists x \ \mathbf{h}(x) = n\end{array}$$

Without the axiom of induction, the function $\mathbf{h}$ encodes a bijection from the datatype $\mathbb{B}$ to the datatype $\mathbb{N}$.

Operations over finite lists

In the sequel, the following codes of functions are used:

1. The function :: (used later with an infix notation) is just the identity function:

$$:: \triangleq \mathrm{id}.$$

It satisfies:

$$\mathbf{PA2} \vdash \forall x \forall l \ x::l = \langle x, l \rangle$$

2. The function `add` is also an operation of concatenation over the type of finite lists. In this context, we will call it "`concat`". The function `concat` is not commutative but it is associative. In particular, we define the first-order formula $\mathrm{subseq}(l_1, l_2)$ saying that the list l_1 is an initial subsequence of the list l_2:

$$\mathrm{subseq}(l_1, l_2) \triangleq \exists l_3 \ \mathtt{concat}(l_1, l_3) = l_2$$

3. The length of a list is obtained by the function:

$$\mathtt{length} \triangleq [\langle \mathbf{0}, \mathbf{0} \rangle | \mathtt{add} \circ \langle \langle \mathbf{0}, \mathbf{0} \rangle, \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd} \rangle]$$

It satisfies:

$$\begin{array}{l}\mathbf{PA2} \vdash \ \mathtt{length}(0) = 0\\ \mathbf{PA2} \vdash \ \forall x \forall l \ \mathtt{length}(x, l) = 1 + \mathtt{length}(l)\end{array}$$

4. To access the n^{th} element of a list, we define the following functions:

$$\begin{array}{rcl}\mathtt{cutn} & \triangleq & [\mathrm{id} | \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd} \circ \mathrm{snd}]\\ \mathtt{nth} & \triangleq & \mathrm{fst} \circ \mathtt{cutn}.\end{array}$$

They satisfy:

$$\begin{array}{l}\mathbf{PA2} \vdash \ \forall l \ \mathtt{cutn}(l, 0) = l\\ \mathbf{PA2} \vdash \forall l (\forall n \in \mathbb{N}) \ \mathtt{cutn}(l, \mathtt{s}(n)) = \mathrm{snd}(\mathtt{cutn}(l, n))\\ \mathbf{PA2} \vdash \forall l (\forall n \in \mathbb{N}) \ \mathtt{nth}(l, n) = \mathrm{fst}(\mathtt{cutn}(l, n))\end{array}$$

In particular, if the index n is larger than the length of the list, the function `nth` returns 0. Given a set E, the set of finite sequences of elements of E is defined as follow:

$$l \in \mathrm{Seq}(E) \triangleq (\forall n \in \mathbb{N})(n <_\omega \mathtt{length}(l) \Rightarrow \mathtt{nth}(l, n) \in E)$$

1.2.3 Subsystems of PA2 obtained by restrictions of the axiom scheme of comprehension

In all the previous proofs, the scheme of comprehension was only used to ensure the existence of sets with an arithmetical definition. It will be interesting for the remainder of the thesis to take into account which instances of the axioms of comprehension are used in a proof. This is why we introduce subsystems of **PA2** obtained by restricting this axiom scheme.

Analytical hierarchy of formulas

Definition 1.2.3.1. We define the classes of formulas $(\Pi^1_n)_{n\in\mathbb{N}}$ and $(\Sigma^1_n)_{n\in\mathbb{N}}$ according to their logical complexity:

1. A formula ϕ is Π^1_0 or Σ^1_0 (or arithmetical) if it is does not contain second-order quantification.
2. A formula ϕ is Π^1_{n+1} if it has the shape $\forall X_1 ... \forall X_k \phi'$ where ϕ' is a Σ^1_n-formula.
3. A formula ϕ is Σ^1_{n+1} if it has the shape $\exists X_1 ... \exists X_k \phi'$ where ϕ' is a Π^1_n-formula.

This categorization is independent from any ambient theory. We reformulate this hierarchy using the notion of logically equivalent formula inside a theory $\mathcal{T}$. As a result, we obtain a hierarchy that is parameterized by a set of closed formulas.

Definition 1.2.3.2. Let $\phi(\vec{x}, \overrightarrow{X})$ and $\psi(\vec{x}, \overrightarrow{X})$ be two formulas and $\mathcal{T}$ a theory.

1. ϕ is logically equivalent to Ψ in $\mathcal{T}$ if $\mathcal{T} \vdash \forall \vec{x} \forall \overrightarrow{X} (\phi \Leftrightarrow \psi)$.
2. ϕ is Π^1_n in $\mathcal{T}$ if it is logically equivalent to a Π^1_n-formula in $\mathcal{T}$.
3. ϕ is Σ^1_n in $\mathcal{T}$ if it is logically equivalent to a Σ^1_n-formula in $\mathcal{T}$.
4. ϕ is Δ^1_n in $\mathcal{T}$ if it is Π^1_n and Σ^1_n in $\mathcal{T}$.

We now introduce subsystems of **PA2** [47].

1. $\mathbf{ACA_0}$ is obtained by restricting the axiom scheme of comprehension to the sets defined with an arithmetical formula (with parameters), i.e $\mathbf{ACA_0}$ is the theory containing the axioms of injectivity, of computation, of induction and the scheme of arithmetical comprehension:

   $$\exists X \forall x (x \in X \Leftrightarrow \phi)$$

   where ϕ is an arithmetical formula that does not contain the variable X.

2. $\mathbf{\Pi^1_n\text{-}CA_0}$ is obtained by restricting the axiom scheme of comprehension to the sets defined with a Π^1_n-formula (with parameters), i.e. $\mathbf{\Pi^1_n\text{-}CA_0}$ is the theory containing the axioms of injectivity, of computation, of induction and the scheme of Π^1_n comprehension:

   $$\exists X \forall x (x \in X \Leftrightarrow \phi)$$

   where ϕ is a Π^1_n-formula that does not contain the variable X.

The acronym $\mathbf{ACA_0}$ stands for arithmetical comprehension axiom while $\mathbf{\Pi^1_n\text{-}CA_0}$ stands for Π^1_n comprehension axiom. The 0 is used to indicate that we work with the axiom of induction (and not the axiom scheme of induction[6]). Therefore, the axiom of induction will be less "powerful" in $\mathbf{ACA_0}$ and in $\mathbf{\Pi^1_n\text{-}CA_0}$ than it is in **PA2**.

Remark 1.2.3.1. $\mathbf{\Pi^1_n\text{-}CA_0}$ implies all the instances of the comprehension scheme for Π^1_n-formula in $\mathbf{\Pi^1_n\text{-}CA_0}$ (see Definition 1.2.3.2). Concretely, if ϕ is equivalent to a Π^1_n-formula in $\mathbf{\Pi^1_n\text{-}CA_0}$, one will be able to show (in the theory $\mathbf{\Pi^1_n\text{-}CA_0}$) that the set defined from ϕ by comprehension exists.

Remark 1.2.3.2. The collection of sets that provably exist in $\mathbf{ACA_0}$ is closed under all boolean operations and more generally under all operations definable by a first-order formula. The same is true for the collection of sets that provably exist in $\mathbf{\Pi^1_n\text{-}CA_0}$. This is a consequence of the fact that parameters are allowed in the formulas indexing the axiom scheme of comprehension.

Proposition 1.2.3.1. $\mathbf{\Pi^1_n\text{-}CA_0}$ *proves the closure*[7] *of the formulas*

$$\exists X(x \in X \Leftrightarrow \phi)$$

where ϕ *does not contain the variable* X *and is either:*

1. *an arithmetical formula*
2. *a* Π^1_k*-formula for* $k \leq n$
3. *a* Σ^1_k*-formula for* $k \leq n$.

Proof. If ϕ if an arithmetical formula or if it is Π^1_k for $k \leq n$, it is logically equivalent to the Π^1_n-formula obtained from ϕ by adding the necessary quantifiers over variables not appearing in it. Note that the added quantifiers do not change the meaning of ϕ as they bind variables that do not appear free in ϕ. All in all, we add quantifiers only to increase the logical complexity of the formula ϕ.

If ϕ is a Σ^1_k formula for $k \leq n$, its negation is logically equivalent to a Π^1_k-formula. Therefore, $\mathbf{\Pi^1_n\text{-}CA_0}$ proves the existence of a set $Z = \{x \mid \neg\phi\}$. The result follows from the closure properties of the definable sets in $\mathbf{\Pi^1_n\text{-}CA_0}$. □

In the following, we will specify when we work in the subsystems $\mathbf{ACA_0}$ and $\mathbf{\Pi^1_1\text{-}CA_0}$. All the proof we did about the functions we previously defined are valid in $\mathbf{ACA_0}$. This extra attention will let us prove results of relativizations to classes of sets that only satisfy some instances of the comprehension axiom. For instance, a class that satisfies the arithmetical comprehension axiom will be closed under all boolean operations.

Remark 1.2.3.3. To be frank, only the closure by boolean operations of some classes will be of interest in the remaining of this manuscript. Paying close attention to the variant of comprehension used in a proof is a choice of design that is not essential for our research.

[6]The axiom scheme of induction contains the closure of the following formulas:

$$\phi[z := 0] \Rightarrow \forall x \forall y(\phi[z := x] \Rightarrow \phi[z := y] \Rightarrow \phi[z := \langle x, y\rangle]) \Rightarrow \forall x\ \phi[z := x]$$

where ϕ is a formula of **SOL**. It implies the axiom of induction and they are equivalent in the presence of the full axiom of comprehension.

[7]The closure of a formula is obtained by quantifying over all its free variables.

1.2.4 Classes of sets and inner models

Classes of sets and relativization

We saw in subsection 1.1.1 that a formula can represent a set of individuals, but it can also represent a class of sets with the notation:

$$\Theta \triangleq \{X \mid \phi\}.$$

Θ is then the class of all sets X that satisfy the formula ϕ. In the syntax, we will use classes as we use sets, which means that we will allow the shortcuts:

$$\begin{array}{rcl@{\qquad}rcl} E \in \{X \mid \phi\} & \triangleq & \phi[X := E] & E \notin \{X \mid \phi\} & \triangleq & \neg\phi[X := E] \\ (\forall X \in \Theta)\phi & \triangleq & \forall X(X \in \Theta \Rightarrow \phi) & (\exists X \in \Theta)\phi & \triangleq & \exists X(X \in \Theta \wedge \phi) \end{array}.$$

Definition 1.2.4.1. The relativization of a formula to the class of sets Θ is defined by external induction over the syntax of formulas:

$$\begin{array}{rcl@{\qquad}rcl} \bot^\Theta & \triangleq & \bot & (t = u)^\Theta & \triangleq & t = u \\ (t \in X)^\Theta & \triangleq & t \in X & (\forall x\phi)^\Theta & \triangleq & \forall x\phi^\Theta \\ (\phi \Rightarrow \psi)^\Theta & \triangleq & \phi^\Theta \Rightarrow \psi^\Theta & (\forall X\phi)^\Theta & \triangleq & (\forall X \in \Theta)\phi^\Theta \end{array}$$

Remark 1.2.4.1. One could have defined $(t \in X)^\Theta \triangleq t \in X \wedge X \in \Theta$. However, because second-order quantifiers are relativized, it is not necessary. Intuitively, free second-order variables in the image of a relativization to a class Θ denote sets of this class. This is formalized in Proposition 1.2.4.1 page 35.

This notion of relativization extends to sets and context:

$$\{x \mid \phi\}^\Theta \triangleq \{x \mid \phi^\Theta\} \qquad \Gamma^\Theta \triangleq \{\phi^\Theta \mid \phi \in \Gamma\}.$$

In particular, we can show by external induction over the syntax of formulas that the operation of relativization commutes with set substitution:

$$(\phi[X := E])^\Theta := \phi^\Theta[X := E^\Theta]$$

Lemma 1.2.4.1. *For all collections of sets Θ and for all formulas $\phi(\vec{x}, \overrightarrow{X})$:*

$$\begin{array}{ll} \vdash \forall\overrightarrow{X}\forall\vec{x}\ (\phi(\vec{x}, \overrightarrow{X}) \Rightarrow \phi^\Theta(\vec{x}, \overrightarrow{X})) & \textit{(if } \phi \textit{ is } \Pi^1_1\textit{)} \\ \vdash \forall\overrightarrow{X}\forall\vec{x}\ (\phi^\Theta(\vec{x}, \overrightarrow{X}) \Rightarrow \phi(\vec{x}, \overrightarrow{X})) & \textit{(if } \phi \textit{ is } \Sigma^1_1\textit{)} \\ \vdash \forall\overrightarrow{X}\forall\vec{x}\ (\phi(\vec{x}, \overrightarrow{X}) \Leftrightarrow \phi^\Theta(\vec{x}, \overrightarrow{X})) & \textit{(if } \phi \textit{ is } \Delta_1\textit{).} \end{array}$$

Proposition 1.2.4.1. *Second-order logic satisfies the following property:*

$$\Gamma \vdash \phi \qquad \textit{implies} \qquad \Gamma^\Theta + \overrightarrow{X} \in \Theta \ \vdash \phi^\Theta$$

where ϕ is a formula with free second-order variables among $\overrightarrow{X}$.

Proof. It is shown by external induction over the proof of second-order classical natural deduction. □

Inner models

We define the notion of an inner model of a theory $\mathcal{T}$, which will be defined to be a non-empty class that satisfies all the axioms of $\mathcal{T}$. We will be especially interested in cases where $\mathcal{T}$ is a variant of **PA2**. The axioms of injectivity, non confusion and confusion are arithmetical and the axiom of induction is Π^1_1. Consequently, Lemma 1.2.4.1 implies that they are satisfied by every class (in a theory that satisfies the axiom of induction). Therefore, while showing that a class models a variant of second-order arithmetic, it is enough to verify that it satisfies the different instances of the axiom scheme of comprehension and the potentially other extra axioms (for instance, choice axioms).

Definition 1.2.4.2. If $\mathcal{T}$ is a theory, ϕ is a closed formula and Θ is a class of sets, we say that the class Θ satisfies the formula ϕ in the theory $\mathcal{T}$ (or ϕ relativizes to Θ in $\mathcal{T}$) if:

$$\mathcal{T} \vdash \phi^{\Theta}.$$

We may omit the ambient theory $\mathcal{T}$ if it is clear from the context.

Definition 1.2.4.3. Let $\mathcal{T}_1$ and $\mathcal{T}_2$ be two theories over the language of **SOL**. We say that a class Θ (given by a closed class definition) models $\mathcal{T}_1$ in $\mathcal{T}_2$ if

1. Θ is provably non-empty in $\mathcal{T}_2$:

$$\mathcal{T}_2 \vdash \exists X(X \in \Theta)$$

2. $\mathcal{T}_2$ proves that Θ satisfies all the axioms of $\mathcal{T}_1$.

In the case where $\mathcal{T}_1$ and $\mathcal{T}_2$ are the same theory $\mathcal{T}$, we will simply say that Θ models $\mathcal{T}$, or that Θ is an inner model for $\mathcal{T}$.

Example 1.2.4.1. A class Θ models **PA2** if

1. **PA2** $\vdash \exists X(X \in \Theta)$
2. **PA2** $\vdash \forall \vec{z}(\forall \overrightarrow{Z} \in \Theta)(\exists X \in \Theta)\forall x(x \in X \Leftrightarrow \phi^{\Theta})$ for every formula ϕ with free variables among $x, \vec{z}, \overrightarrow{Z}$.

This notion will be used to show results of relative consistency:

Proposition 1.2.4.2. *If a class Θ models $\mathcal{T}_1$ in $\mathcal{T}_2$:*

$$\mathcal{T}_1 + \Gamma \vdash \phi \quad \textit{implies} \quad \mathcal{T}_2 + \Gamma^{\Theta} + \overrightarrow{X} \in \Theta \vdash \phi^{\Theta}$$

where ϕ is a formula with free second-order variables among $\overrightarrow{X}$. It implies a result of relative consistency between $\mathcal{T}_1$ and $\mathcal{T}_2$:

$$\mathcal{T}_1 \vdash \bot \quad \textit{implies} \quad \mathcal{T}_2 \vdash \bot.$$

Proof. It is shown by external induction over the proofs of second-order classical natural deduction. The fact that Θ models $\mathcal{T}_1$ inside $\mathcal{T}_2$ is used in the case of the axiom rule. □

Remark 1.2.4.2. It is not the same as saying that the theory $\mathcal{T}_2$ proves the consistency of $\mathcal{T}_1$. To show this result, it would be necessary to internalize in $\mathcal{T}_2$ that there exists a definable class of sets that models $\mathcal{T}_1$.

Proposition 1.2.4.3. *If Θ models a theory $\mathcal{T}$:*

1.	$\mathcal{T}$	$\vdash$	$\forall \vec{x}(\forall \overrightarrow{X} \in \Theta)(\phi \Rightarrow \phi^{\Theta})$	*(if ϕ is Π^1_1 in $\mathcal{T}$)*
2.	$\mathcal{T}$	$\vdash$	$\forall \vec{x}(\forall \overrightarrow{X} \in \Theta)(\phi^{\Theta} \Rightarrow \phi)$	*(if ϕ is Σ^1_1 in $\mathcal{T}$)*
3.	$\mathcal{T}$	$\vdash$	$\forall \vec{x}(\forall \overrightarrow{X} \in \Theta)(\phi^{\Theta} \Leftrightarrow \phi)$	*(if ϕ is Δ^1_1 in $\mathcal{T}$)*

We say that

1. *Π^1_1-formulas are downward absolute*
2. *Σ^1_1-formulas are upward absolute*
3. *Δ^1_1-formulas are absolute.*

Proof. We show 1. as an example. Let ψ be a Π^1_1 formula logically equivalent to ϕ in $\mathcal{T}$:

$$\mathcal{T} \vdash \forall \vec{x} \forall \overrightarrow{X} (\phi \Leftrightarrow \psi)$$

From ϕ, we can deduce ψ and then ψ^{Θ} because ψ is a Π^1_1-formula. On the other other hand, we deduce $\mathcal{T} \vdash (\phi \Leftrightarrow \psi)^{\Theta}$ as Θ models $\mathcal{T}$. Therefore, we conclude ϕ^{Θ}. $\square$

Chapter 2

The axiom scheme of collection

The axiom of induction included in the theory **PA2** does not have a computational interpretation inside classical realizability. Therefore, we introduce the system $\mathbf{PA2}^-$ (Subsection 2.1) obtained from **PA2** by removing this axiom. Without induction, a new variant of choice principle appears, the axiom scheme of collection which is strongly inspired from the homonymous axiom of set theory. This axiom scheme will be of crucial importance in Chapter 5 when dealing with well-preorders. However, while we suspect that it is weaker than the axiom of choice, we were not yet able to prove it (see Open problem 1).

After a logical study of second-order arithmetic, we will continue with a computational study. In the context of the Curry-Howard correspondence between proofs and programs, we design two type systems $\boldsymbol{\lambda}\mathbf{PA2}$ and $\boldsymbol{\lambda}\mathbf{PA2}^-$ that respectively capture the theory **PA2** and $\mathbf{PA2}^-$ (Section 2.2). Taking advantage of these two systems, we implement an interpretation of $\boldsymbol{\lambda}\mathbf{PA2}$ into $\boldsymbol{\lambda}\mathbf{PA2}^-$ by translating the former in the latter (Subsection 2.2.3). This interpretation is done by relativization of the first-order quantifiers to the set of Dedekind trees[1]. Finally, we will describe a realizability model for the system $\mathbf{PA2}^-$. In particular, we will show that the axiom scheme of collection is realized and has a very simple computational content (Section 2.3).

2.1 The system $\mathbf{PA2}^-$, axiom scheme of collection, axiom scheme of choice

2.1.1 The system $\mathbf{PA2}^-$

The axiom of induction of second-order arithmetic ensures that all individuals are standard in the sense that they belong to the set $\mathbb{B}$. Indeed

$$\mathbf{\Pi^1_1\text{-}CA_0} \vdash \forall x(x \in \mathbb{B})$$

is proved by an induction over the variable x. However, the axiom of induction does not have good computational property: it cannot be realized! This was our first motivation to work in a system without induction. In a second step, we will see how relativization can be used to retrieve the axiom of induction (in Subsection 2.2.3) ; showing that working in a system without induction is in fact not a restriction.

Definition 2.1.1.1. The system $\mathbf{PA2}^-$ is obtained from **PA2** by removing from its axioms the axiom of induction.

[1] The analogous of Dedekind numerals for the data type of pure binary trees.

Remark 2.1.1.1. Similarly, we consider the systems $\mathbf{ACA_0^-}$ and $\mathbf{\Pi^1_n\text{-}CA_0^-}$ obtained by removing the axiom of induction of the corresponding theory.

In Section 2.2, we show a proof of relative consistency between **PA2** and $\mathbf{PA2^-}$.

2.1.2 Axiom scheme of collection

Without the axiom of induction, the range of first-order quantifications increases to a wider domain of discourse. As a consequence, equivalent formulas in presence of induction can have different meanings without it. For instance, a new variant of choice principles appears in $\mathbf{PA2^-}$, the axiom scheme of collection (**Coll**), obtained as a reformulation of the homonymous axiom scheme of set theory in the framework of second-order logic.

Definition 2.1.2.1. The axiom scheme of collection (**Coll**) is defined as the closure of the formulas

$$\forall x \exists Y \phi(x, Y) \Rightarrow \exists Z \forall x \exists y \phi(x, Z[y])$$

where $\phi(x, Y)$ is a formula of **SOL** (that may contain other free variables).

Recall that the notation $E[x]$ for a set E and an individual x denotes the slice of E at x (Definition 1.1.1.4):

$$E[x] \triangleq \{y \mid \langle x, y\rangle \in E\}.$$

A set seen with this angle encodes a family of sets indexed by the individuals. Therefore, the axiom scheme of collection says that from every total relation between individuals and reals (encoded by a formula $\phi(x, Y)$), one can find a definable family of sets encoded by Z that meets, for every x, the collection $\{Y \mid \phi(x, Y)\}$. Intuitively, the family Z is a "bound" for the image of the previous relation[2].

This scheme, that has a very simple computational interpretation, will be used to work with well-preorders in the theory $\mathbf{PA2^-} + \mathbf{Coll}$. It will be specifically used to prove the Principle of Reflection (Theorem 5.4.1.1). This is the alternative we chose rather than working in a system with an axiom of choice.

Remark 2.1.2.1. Recall that the axiom scheme of collection in set theory is stated as

$$\forall A((\forall x \in A)\exists y \phi(x, y) \Rightarrow \exists z(\forall x \in A)(\exists y \in z)\phi(x, y))$$

where $\phi(x, y)$ is a formula with parameters of the language of **ZF**. In set theory, the intuition behind the axiom scheme of collection is that the image of a relation over a set can be bound. Consequently, the two principles of collection have very similar meanings.

However, the denotation of the collection principle of second-order logic in a set-theoretic model of $\mathbf{PA2^-}$ is very different. Namely, for a fixed set A used to interpret the individuals, the interpretation of the collection principle in the model $(A, \mathcal{P}(A))$ is

$$\begin{aligned}(\forall B : A \to \mathcal{P}(\mathcal{P}(A)))((\forall a \in A)B(a) \neq \emptyset \Rightarrow \\ (\exists B' : A \to \mathcal{P}(\mathcal{P}(A)))(\forall a \in A)(B'(a) \subseteq B(a) \wedge B'(a) \neq \emptyset \wedge |B'(a)| < |A|)).\end{aligned}$$

where, in that specific case, by $|B'(a)| < |A|$, we mean that there is a surjection from A onto $B'(a)$. All in all, it means that, from a family of non-empty subsets of $\mathcal{P}(A)$ indexed by A, we can extract a family of size bound by A. Consequently, the interpretation of the collection principle of **SOL** in set theory is not an instance of the collection principle of **ZF**.

[2] I said "intuitively" because it would be the case only if the relation $\phi(x, Y)$ is functional or if Z also satisfies $\forall x \forall Y(\phi(x, Y) \Rightarrow \exists y\, Y = Z[y])$.

2.1.3 Axiom scheme of choice

In the presence of induction, the axiom scheme of collection collapses to the axiom scheme of choice over the individuals ($\mathbf{AC}_\iota$).

Definition 2.1.3.1. The axiom scheme of choice over the individuals ($\mathbf{AC}_\iota$) is defined as the closure of the formulas

$$\forall x \exists Y \phi(x, Y) \Rightarrow \exists Z \forall x \phi(x, Z[x])$$

where $\phi(x, Y)$ is a formula of **SOL** (that may contain other free variables).

This scheme says that a function $x \mapsto Z[x]$ can be extracted from any total relation (encoded by a formula with parameters $\phi(x, Y)$).

Proposition 2.1.3.1. *The axiom scheme of collection and the axiom scheme of choice over the individuals are equivalent in* **PA2**. *It means that, in the system* $\mathbf{PA2} + \mathbf{Coll}$, *one can derive all the instances of* $\mathbf{AC}_\iota$ *and, reciprocally, all the instances of* **Coll** *are consequences of* $\mathbf{PA2} + \mathbf{AC}_\iota$.

The proof relies on the fact that the set of individuals can be well-ordered[3] in **PA2**.

Remark 2.1.3.1. The interpretation of this axiom of choice in a set-theoretic model of $\mathbf{PA2}^-$ is a variant of the axiom of choice. For a set $\mathcal{I}$ used to interpret the individuals, it says that the product of a non-empty family of subsets of $\mathcal{P}(\mathcal{I})$ indexed by $\mathcal{I}$ is non-empty. If the individuals are interpreted by a countable set, it is an instance of the axiom of countable choice.

2.1.4 Links between the axiom scheme of collection and the axiom scheme of choice in $\mathbf{PA2}^-$

We saw that the axiom scheme of collection and the axiom scheme of choice (over the individuals) are equivalent in **PA2**. However, is it still the case in $\mathbf{PA2}^-$? It is easily seen that $\mathbf{AC}_\iota$ implies **Coll**. But what about the reverse direction? We believe that the axiom of collection is weaker. However, we were not able to prove it. As a consolation prize, we will give in this thesis various arguments in favor of this hypothesis.

1. The axiom scheme of collection is realized in almost all models of classical realizability[4] (see Section 2.3). However, the axiom scheme of choice over the individuals needs an extra instruction[5] in the programming language to be realized [30].

2. In Chapter 3, we formulate the axiom schemes $\mathbf{Coll}_{\mathbf{Dom}}$ and $\mathbf{AC}_{\mathbf{Dom}}$, which are classically equivalent to the two previous ones. However, they are both well-behaved with respect to negative translations [17], as we show in Section 3.2.2. But it appears that in second-order intuitionistic arithmetic, the axiom scheme $\mathbf{Coll}_{\mathbf{Dom}}$ is weaker than the axiom scheme $\mathbf{AC}_{\mathbf{Dom}}$! We show in Section 3.3 that Kleene realizability for $\mathbf{HA2}^-$ interprets the former and refutes the latter.

Open problem 1. *Is the axiom scheme of collection weaker than the axiom scheme of choice over the individuals in* $\mathbf{PA2}^-$*?*

[3] We will prove in the next chapter that the set of individuals is indeed well-ordered in **PA2**.

[4] Precisely, it is realized in all models constructed from a realizability algebra with at most countable instructions.

[5] Specifically, an instruction `quote` used to internalize in the programming language an injection from the instructions to the natural numbers.

Remark 2.1.4.1. We tried to find a set-theoretic model that refutes the axiom of choice and validates the axiom of collection. However, we encountered difficulties as it seems that the set-theoretic interpretation of the axiom scheme of collection is a new principle. It remains as a future work to study in depth this new set-theoretic principle.

2.2 Type systems for PA2 and PA2$^-$

We presented various theories in the language of **SOL**, and notably the theories **PA2** and **PA2**$^-$. We now design two type systems that capture these theories, in the spirit of the Curry-Howard correspondence between proofs and programs. Concretely, we will work with two different typed λ-calculus:

1. A Church style λ-calculus **λPA2** capturing the axioms of **PA2** and the deduction rules of **SOL**. As in every Church style λ-calculus, the proof terms of **λPA2** contain all the necessary information to retrieve the derivations, and notably all the introduction and eliminations of first and second-order quantifiers.

2. A Curry style λ-calculus **λPA2**$^-$ in the spirit of **AF2** [27] and of classical realizability, which captures the theory **PA2**$^-$. The principal advantage of this calculus is that it is equipped with a well understood semantic coming from classical realizability models. The architecture of these models will be presented in the end of this chapter (Section 2.3).

The main difference between these two calculi is that the axiom of induction is derivable in the former but not in the latter. In **λPA2**$^-$, this difference can be resolved by relativizing all first-order quantifiers to the set $\mathbb{B}$ of pure binary trees. We will formulate this process as a syntactical translation from **λPA2** into **λPA2**$^-$. It will give a proof of relative consistency between the theories **PA2** and **PA2**$^-$.

2.2.1 The system λPA2

The system **λPA2** is a typed λ-calculus à la Church: it is the Curry-Howard counter part of the theory **PA2** (equipped with the rules of **SOL**).

Definition 2.2.1.1. The syntax and the rules of the system **λPA2** are presented in the Figure 2.1 page 42.

In the system **λPA2**:

1. First-order equality $t = u$ is primitive. It is introduced with the proof term $\mathrm{refl}(t)$ and eliminated with the proof term $\mathrm{peel}(M, E, M')$. The same is true for the formula $\bot$, it has no rule of introduction and it is eliminated with the proof term $\mathrm{efq}(M, \phi)$.

2. The rewriting rules on top of the codes of (primitive recursive) functions incorporate computations inside the deductions. In particular, they are used to derive the formula:

$$\forall x \forall y \forall x' \forall y (\langle x, y\rangle = \langle x', y'\rangle \Rightarrow x = x' \wedge y = y').$$

3. The predicate $\mathrm{null}(t)$ equipped with its rewriting rules is used to derive the formula:

$$\forall x \forall y \langle x, y\rangle \neq 0.$$

Syntax

Codes of functions	f, g	::=	$\mathbf{0} \mid \mathrm{id} \mid \mathrm{fst} \mid \mathrm{snd} \mid (f \circ g) \mid \langle f, g\rangle \mid [f\|g]$
Terms	t, u	::=	$x \mid 0 \mid \langle t, u\rangle \mid f(t)$
Formulas	ϕ, ψ	::=	$\bot \mid t = u \mid \mathrm{null}(t) \mid t \in X \mid \phi \Rightarrow \psi$ $\mid \forall x\phi \mid \forall X\phi$
Sets	E	::=	$\{x \mid \phi\}$
Proof terms	M, N	::=	$\xi \mid \mathrm{efq}(M, \phi)$ $\mid \mathrm{oc}(\phi, \psi) \mid \mathrm{ind}(E, M, M', t)$ $\mid \mathrm{refl}(t) \mid \mathrm{peel}(M, E, M')$ $\mid \lambda\xi : \phi.M \mid MN$ $\mid \lambda x.M \mid Mt$ $\mid \lambda X.M \mid ME$
Contexts	Γ	::=	$\emptyset \mid \Gamma, \xi : \phi$

Congruence on terms and formulas

Rules generating the congruence $t \simeq u$ (on the terms) and $\phi \simeq \psi$ (on the formulas):

$$\begin{array}{rcl@{\qquad}rcl}
\mathbf{0}(t) & \succ & 0 & [f|g](0) & \succ & 0 \\
\mathrm{id}(t) & \succ & t & [f|g](t, 0) & \succ & f(t) \\
(f \circ g)(t) & \succ & f(g(t)) & [f|g](t, u, v) & \succ & g(t, u, v, [f|g](t, u), [f|g](t, v)) \\
\langle f, g\rangle(t) & \succ & \langle f(t), g(t)\rangle & \mathrm{null}(0) & \succ & \bot \Rightarrow \bot \\
 & & & \mathrm{null}(t, y) & \succ & \bot
\end{array}$$

Typing rules

$$\frac{}{\Gamma \vdash \xi : \phi}\,(\xi:\phi)\in\Gamma \qquad \frac{\Gamma \vdash M : \phi}{\Gamma \vdash M : \phi'}\,(\phi \simeq \phi')$$

$$\frac{\Gamma \vdash M : \bot}{\Gamma \vdash \mathrm{efq}(M, \phi) : \phi} \qquad \frac{}{\Gamma \vdash \mathrm{oc}(\phi, \psi) : ((\phi \Rightarrow \psi) \Rightarrow \phi) \Rightarrow \phi}$$

$$\frac{\Gamma \vdash M : 0 \in E \quad \Gamma \vdash M' : \forall x \forall y (x \in E \Rightarrow y \in E \Rightarrow \langle x, y\rangle \in E)}{\Gamma \vdash \mathrm{ind}(E, M, M', t) : t \in E}$$

$$\frac{}{\Gamma \vdash \mathrm{refl}(t) : t = t} \qquad \frac{\Gamma \vdash M : t = u \quad \Gamma \vdash M' : t \in E}{\Gamma \vdash \mathrm{peel}(M, E, M') : u \in E}$$

$$\frac{\Gamma, \xi : \phi \vdash M : \psi}{\Gamma \vdash \lambda\xi : \phi.M : \phi \Rightarrow \psi} \qquad \frac{\Gamma \vdash M : \phi \Rightarrow \psi \quad \Gamma \vdash N : \phi}{\Gamma \vdash MN : \psi}$$

$$\frac{\Gamma \vdash M : \phi}{\Gamma \vdash \lambda x.M : \forall x\phi}\,x \notin FV(\Gamma) \qquad \frac{\Gamma \vdash M : \forall x\phi}{\Gamma \vdash Mt : \phi[x := t]}$$

$$\frac{\Gamma \vdash M : \phi}{\Gamma \vdash \Lambda X.M : \forall X\phi}\,X \notin FV(\Gamma) \qquad \frac{\Gamma \vdash M : \forall X\phi}{\Gamma \vdash ME : \phi[X := E]}$$

Figure 2.1: The proof system **λPA2**

4. The first-order universal quantifier $\forall x\phi$ is implicitly relativized: it represents a quantification over the set of pure binary trees. As a consequence, the system is equipped with a proof term $\mathrm{ind}(E, M, M', t)$, enabling the use of induction inside **λPA2**.

5. The missing logical connectives are encoded as in **SOL**. For instance:

$$\begin{array}{ll}
\text{If } \Gamma \vdash M : \phi \text{ and } \Gamma \vdash M' : \psi \text{ then } & \Gamma \vdash \lambda\xi : \phi \Rightarrow \neg\psi.\xi M M' : \phi \wedge \psi \\
\text{If } \Gamma \vdash M : \phi \wedge \psi \text{ then } & \Gamma \vdash \mathcal{CC}(\phi, \bot)\lambda\xi : \neg\phi.\mathrm{efq}(M\lambda\eta : \phi.\lambda\mu : \psi.\xi\eta, \phi) : \phi \\
\text{If } \Gamma \vdash M : \phi \wedge \psi \text{ then } & \Gamma \vdash \mathcal{CC}(\psi, \bot)\lambda\xi : \neg\psi.\mathrm{efq}(M\lambda\eta : \phi.\lambda\mu : \psi.\xi\mu, \psi) : \psi \qquad \text{(etc...)}
\end{array}$$

Theorem 2.2.1.1. *Every axiom of* **PA2** *is provable in* **λPA2**.

Proof. We exhibit the proof terms proving the axioms of **PA2**.

- The axiom of injectivity is proved by:

$$\lambda x\lambda y\lambda x'\lambda y'.\lambda\xi : \langle x, y\rangle = \langle x', y'\rangle.\lambda\eta : x = x' \Rightarrow \neg y = y'.$$
$$\eta\ \mathrm{peel}(\xi, \{z|\ x = \mathrm{fst}(z)\}, \mathrm{refl}(x))\ \mathrm{peel}(\xi, \{z \mid y = \mathrm{snd}(z)\}, \mathrm{refl}(y))$$

  Recall that the axiom of injectivity is $\forall x\forall y\forall x'\forall y'(\langle x, y\rangle = \langle x', y'\rangle \Rightarrow (x = x' \wedge y = y'))$. By unfolding the definition of the conjunction, we see that the previous λ-term needs to denote a proof of the formula $\forall x\forall y\forall x'\forall y'(\langle x, y\rangle = \langle x', y'\rangle \Rightarrow (x = x' \Rightarrow \neg y = y') \Rightarrow \bot)$.

  It is obtained by combining the proof variable $\xi : \langle x, y\rangle = \langle x', y'\rangle$, the instruction peel and the proof variable $\eta : x = x' \Rightarrow \neg y = y'$.

- The axiom of non confusion is proved by:

$$\lambda x\lambda y.\lambda\xi : \langle x, y\rangle = 0.\mathrm{peel}(\mathrm{sym}\langle x, y\rangle 0\xi, \{z \mid \mathrm{null}(z)\}, \lambda\eta : \bot.\eta)$$

  where

$$\mathrm{sym} \triangleq \lambda x\lambda y.\lambda\xi : x = y.\mathrm{peel}(\xi, \{z \mid z = x\}, \mathrm{refl}(x)) : \forall x\forall y.(x = y \Rightarrow y = x)$$

- All the axioms of computation are proved by the term $\lambda x.\mathrm{refl}(x)$.

- The axiom of induction is proved by:

$$\lambda X.\lambda\xi : 0 \in X.\lambda\eta : \forall x\forall y(x \in X \Rightarrow y \in X \Rightarrow \langle x, y\rangle \in X).\lambda x.\mathrm{ind}(X, \xi, \eta, x)$$

- Finally, because the "full" elimination rule of the second-order universal quantifier is included in this system, all the instances of the axiom scheme of comprehension are derivable (as in Proposition 1.2.1.2). In particular, there is a proof term attesting the provability of each instance of the axiom scheme of comprehension.

□

The converse is also true, every formula in the syntax of **SOL** provable in **λPA2** is provable in **PA2**.

Theorem 2.2.1.2. *If* $\vdash M : \phi$ *and* ϕ *does not contain the predicate* $\mathrm{null}(t)$ *then* $\mathbf{PA2} \vdash \phi$.

In fact, this result can be made more precise. If $\Gamma \vdash M : \phi$ then **PA2** proves the formula obtained from ϕ by replacing all occurrences of the predicate $\mathrm{null}(t)$ by the formula $t = 0$.

2.2.2 The proof system $\boldsymbol{\lambda}$**PA2**$^-$

The system $\boldsymbol{\lambda}$**PA2**$^-$ is a typed λ-calculus à la Curry: it is the Curry-Howard counterpart of the theory **PA2**$^-$ (with the rules of **SOL**).

Definition 2.2.2.1. The syntax and the rules of the system $\boldsymbol{\lambda}$**PA2**$^-$ are presented in the Figure 2.2 page 45.

In the system $\boldsymbol{\lambda}$**PA2**$^-$:

1. The formulas $\bot$ and $t = u$ are not primitive anymore. They are defined using second-order encodings:

$$\bot \triangleq \forall X(0 \in X) \qquad\qquad t = u \triangleq \forall X(t \in X \Rightarrow u \in X)$$

   However, the predicate $\text{null}(t)$ remains primitive.

2. The other logical connectives are obtained by replacing $\bot$ with $\forall X(0 \in X)$ in the encodings done for **SOL**. These encodings are still different from the usual impredicative encodings. For instance, for all proof terms M of type ϕ and N of type ψ, we write $[M, N]$ the proof term of type $\phi \wedge \psi$:

$$[M, N] \triangleq \lambda\xi.\xi MN \qquad \text{for } \xi \notin (FV(M) \cup FV(N)).$$

   If $\Gamma \vdash M : \phi$ and $\Gamma \vdash N : \psi$, then:

$$\Gamma \vdash [M, N] : \phi \wedge \psi$$

   The associated eliminators π_1, π_2 are:

$$\begin{array}{rcl} \pi_1 & \triangleq & \lambda\xi.\mathfrak{cc}\lambda\eta.\xi\lambda\mu\lambda\chi.\eta\mu \\ \pi_2 & \triangleq & \lambda\xi.\mathfrak{cc}\lambda\eta.\xi\lambda\chi\lambda\mu.\eta\mu \end{array}$$

   They satisfy the following typing judgements:

$$\begin{array}{rcl} \vdash \pi_1 & : & \forall X\forall Y(X \wedge Y \Rightarrow X) \\ \vdash \pi_2 & : & \forall X\forall Y(X \wedge Y \Rightarrow Y) \end{array}$$

3. The meaning of the formula $\forall x\phi$ changes: it is now an unrelativized quantification. To interpret the system $\boldsymbol{\lambda}$**PA2** in the system $\boldsymbol{\lambda}$**PA2**$^-$, we will relativize all first-order quantifiers to the predicate $\mathbb{B}$.

4. The language of proof terms of $\boldsymbol{\lambda}$**PA2**$^-$ is obtained from the pure λ-calculus by adding the constant $\mathfrak{cc}$ (call/cc).

Notation 2.2.2.1. In a term $\lambda\xi.M$, if the variable ξ does not appear in M, we may replace it by the symbol _ and write $\lambda\xi.M$ as $\lambda\_.M$. For instance, with this notation, $\pi_1 \triangleq \lambda\xi.\mathfrak{cc}\lambda\eta.\xi\lambda\mu\lambda\_.\eta\mu$.

Notation 2.2.2.2. We may write $\lambda\xi_1, ..., \xi_n.M$ to denote the λ-term $\lambda\xi_1...\lambda\xi_n.M$.

Theorem 2.2.2.1. *Every axiom of* **PA2**$^-$ *is provable in* $\boldsymbol{\lambda}$**PA2**$^-$ *(when interpreted using the encodings specific to* $\boldsymbol{\lambda}$**PA2**$^-$*).*

As for $\boldsymbol{\lambda}$**PA2** and **PA2**, the converse is also true.

Theorem 2.2.2.2. *If* $\vdash M : \phi$ *in the system* $\boldsymbol{\lambda}$**PA2**$^-$ *then* **PA2**$^-$ *proves the formula obtained from* ϕ *by replacing all occurrences of the predicate* $\text{null}(t)$ *by the formula* $t = 0$.

Syntax

Codes of functions	f, g	$::=$	$\mathbf{0} \mid \mathrm{id} \mid \mathrm{fst} \mid \mathrm{snd} \mid (f \circ g) \mid \langle f, g\rangle \mid [f\vert g]$
Terms	t, u	$::=$	$x \mid 0 \mid \langle t, u\rangle \mid f(t)$
Formulas	ϕ, ψ	$::=$	$\mathrm{null}(t) \mid t \in X \mid \phi \Rightarrow \psi \mid \forall x\phi \mid \forall X\phi$
Sets	E	$::=$	$\{x \mid \phi\}$
Proof terms	M, N	$::=$	$\xi \mid \infty \mid \lambda\xi.M \mid MN$
Contexts	Γ	$::=$	$\emptyset \mid \Gamma, \xi : \phi$

Congruence on terms and formulas

Same as the rules in **λPA2**, with $\bot \triangleq \forall Z(0 \in Z)$

Typing rules

$$\frac{}{\Gamma \vdash \xi : \phi}\,(\xi{:}\phi)\in\Gamma \qquad \frac{\Gamma \vdash M : \phi}{\Gamma \vdash M : \phi'}\,(\phi \simeq \phi')$$

$$\frac{\Gamma, \xi : \phi \vdash M : \psi}{\Gamma \vdash \lambda\xi.M : \phi \Rightarrow \psi} \qquad \frac{\Gamma \vdash M : \phi \Rightarrow \psi \quad \Gamma \vdash N : \phi}{\Gamma \vdash MN : \psi}$$

$$\frac{\Gamma \vdash M : \phi}{\Gamma \vdash M : \forall x\phi}\,x \notin FV(\Gamma) \qquad \frac{\Gamma \vdash M : \forall x\phi}{\Gamma \vdash M : \phi[x := t]}$$

$$\frac{\Gamma \vdash M : \phi}{\Gamma \vdash M : \forall X\phi}\,X \notin FV(\Gamma) \qquad \frac{\Gamma \vdash M : \forall X\phi}{\Gamma \vdash M : \phi[X := E]}$$

$$\frac{}{\Gamma \vdash \infty : ((\phi \Rightarrow \psi) \Rightarrow \phi) \Rightarrow \phi}$$

Figure 2.2: The proof system **λPA2⁻**

2.2.3 A translation by relativization

We will now describe a translation from **λPA2** into **λPA2⁻** done by relativizing first-order quantifiers to the set $\mathbb{B}$. To design this translation, we will:

1. Translate formulas ϕ and contexts Γ of **λPA2** into formulas $\phi^{\mathbb{B}}$ and contexts $\Gamma^{\mathbb{B}}$ of **λPA2⁻**.
2. Translate first-order terms t into proof terms t^* of **λPA2⁻** (of type $t \in \mathbb{B}$).
3. Translate proof terms M of **λPA2** into proof terms M^* of **λPA2⁻**.

Translation of formulas, contexts and sets

We associate to every formula ϕ in the language of **λPA2** a formula $\phi^{\mathbb{B}}$ in the language of **λPA2⁻**:

$$\begin{array}{rcl}(t \in X)^{\mathbb{B}} & \triangleq & t \in X \\ (\mathrm{null}(t))^{\mathbb{B}} & \triangleq & \mathrm{null}(t) \\ (\bot)^{\mathbb{B}} & \triangleq & \forall Z(0 \in Z) \\ (t = u)^{\mathbb{B}} & \triangleq & \forall Z(t \in Z \Rightarrow u \in Z) \\ (\phi \Rightarrow \psi)^{\mathbb{B}} & \triangleq & \phi^{\mathbb{B}} \Rightarrow \psi^{\mathbb{B}} \\ (\forall x \phi)^{\mathbb{B}} & \triangleq & (\forall x \in \mathbb{B})\phi^{\mathbb{B}} \\ (\forall X \phi)^{\mathbb{B}} & \triangleq & \forall X \phi^{\mathbb{B}}\end{array}$$

This translation replaces the primitive formula $\bot$ and $t = u$ by their second-order corresponding encodings (in $\boldsymbol{\lambda}\mathbf{PA2}^-$) and relativizes the first-order quantifiers. It commutes with all the remaining connectives. As a consequence, it also commutes with the defined connectives. For instance, the translation of $\phi \wedge \psi$ is $\phi^{\mathbb{B}} \wedge \psi^{\mathbb{B}}$.

Lemma 2.2.3.1. *If $\phi \simeq \phi'$ in $\boldsymbol{\lambda}\mathbf{PA2}$ then $\phi^{\mathbb{B}} \simeq \phi'^{\mathbb{B}}$ in $\boldsymbol{\lambda}\mathbf{PA2}^-$.*

The translation $\phi \mapsto \phi^{\mathbb{B}}$ extends to sets and contexts:

$$\{x \mid \phi\}^{\mathbb{B}} \triangleq \{x \mid \phi^{\mathbb{B}}\} \quad \text{and} \quad (\xi_1 : \phi_1, ..., \xi_n : \phi_n)^{\mathbb{B}} \triangleq \xi_1 : \phi_1^{\mathbb{B}}, ..., \xi_n : \phi_n^{\mathbb{B}}$$

Translation of first-order terms

Because the syntax of proof terms of $\boldsymbol{\lambda}\mathbf{PA2}$ includes first-order terms and that the syntax of $\boldsymbol{\lambda}\mathbf{PA2}^-$ does not, we need to translate first-order terms into proof terms of $\boldsymbol{\lambda}\mathbf{PA2}^-$. A first-order term t will be interpreted as a proof t^* of the formula $t \in \mathbb{B}$. This will be done by external induction over the syntax of first-order terms. The set $\mathbb{B}$ of pure binary trees will be of crucial importance in this translation. We recall its definition:

$$x \in \mathbb{B} \triangleq \forall X(0 \in X \Rightarrow \forall y \forall z(y \in X \Rightarrow z \in X \Rightarrow \langle y, z\rangle \in X) \Rightarrow x \in X).$$

- We will start by translating each code of primitive recursive functions into a λ-term. We introduce the following notation that will be used in this definition. For each proof terms M and N in $\boldsymbol{\lambda}\mathbf{PA2}^-$:

$$\begin{array}{rcll}\langle M, N\rangle & \triangleq & \lambda\xi\lambda\eta.\eta(M\xi\eta)(N\xi\eta) & \text{for } \xi, \eta \notin (FV(M) \cup FV(N)) \\ 0 & \triangleq & \lambda\xi\lambda\eta.\xi & \end{array}$$

  This is in fact the standard Church encoding of binary trees in λ-calculus, applied to the proof terms M and N. In particular, if $M : t \in \mathbb{B}$ and $N : u \in \mathbb{B}$, then $\langle M, N\rangle : \langle t, u\rangle \in \mathbb{B}$. Note also that $0 : 0 \in \mathbb{B}$. This is why we overload the notations used for first-order terms. This overloading is possible because no first-order terms appear in the proof terms of $\boldsymbol{\lambda}\mathbf{PA2}^-$ and, therefore, no confusion is possible. In particular $\langle \_, \_\rangle$ is right associative, it means:

$$\begin{array}{rcll}\langle M, N, P\rangle & \triangleq & \langle M, \langle N, P\rangle\rangle & \\ \langle M, N, P, Q\rangle & \triangleq & \langle M, \langle N, \langle P, Q\rangle\rangle\rangle & \text{(etc...)}\end{array}$$

  We associate a closed proof term f^* (in the language of $\boldsymbol{\lambda}\mathbf{PA2}^-$) of type

$$\forall x(x \in \mathbb{B} \Rightarrow f(x) \in \mathbb{B})$$

  to each code of primitive recursive functions f:

$$
\begin{array}{rcl}
0^* & \triangleq & \lambda\_.0 \\
\mathrm{id}^* & \triangleq & \lambda\xi.\xi \\
\mathrm{fst}^* & \triangleq & \lambda\xi.\pi_1(\xi[0,0]\lambda\xi_1\lambda\xi_2.[\pi_2\xi_1,\langle\pi_2\xi_1,\pi_2\xi_2\rangle]) \\
\mathrm{snd}^* & \triangleq & \lambda\xi.\pi_1(\xi[0,0]\lambda\xi_1\lambda\xi_2.[\pi_2\xi_2,\langle\pi_2\xi_1,\pi_2\xi_2\rangle]) \\
(f\circ g)^* & \triangleq & \lambda\xi.f^*(g^*\xi) \\
\langle f,g\rangle^* & \triangleq & \lambda\xi.\pi_1(\xi[\langle f^*0,g^*0\rangle,0]\lambda\xi_1\lambda\xi_2.[\langle f^*\langle\pi_2\xi_1,\pi_2\xi_2\rangle,g^*\langle\pi_2\xi_1,\pi_2\xi_2\rangle\rangle,\langle\pi_2\xi_1,\pi_2\xi_2\rangle]) \\
[\mathrm{f}|\mathrm{g}]^* & \triangleq & \lambda\xi.\pi_1(\xi[0,0] \\
\multicolumn{3}{r}{\lambda\nu\lambda\mu.[\pi_1((\pi_2\mu)[f^*(\pi_2\nu),0]\lambda\mu_1\lambda\mu_2.[g^*\langle\pi_2\nu,\pi_2\mu_1,\pi_2\mu_2,\pi_1\mu_1,\pi_1\mu_2\rangle,\langle\pi_2\mu_1,\pi_2\mu_2\rangle]),\langle\pi_2\nu,\pi_2\mu\rangle])}
\end{array}
$$

This interpretation uses the following trick that we explain for the interpretation of the function fst. Let x be a variable and assume $x \in \mathbb{B}$. We want to prove that $\mathrm{fst}(x) \in \mathbb{B}$ and we can be tempted to eliminate the second-order variable X with the set $\{x \mid \mathrm{fst}(x) \in \mathbb{B}\}$. However, it won't be enough as the induction hypothesis given by this choice is not strong enough. We should rather use the set $\{x \mid \mathrm{fst}(x) \in \mathbb{B} \wedge x \in \mathbb{B}\}$. This is why proof terms for introducing and eliminating conjunction appears in the translation of the codes of functions.

Lemma 2.2.3.2. *For each symbol of primitive recursive functions f:*

$$\vdash f^* : \forall x(x \in \mathbb{B} \Rightarrow f(x) \in \mathbb{B}) \qquad \textit{(in } \boldsymbol{\lambda}\mathbf{PA2}^-\textit{)}.$$

- Using the previous interpretation of codes of functions, we can now translate first-order terms into proof terms of $\boldsymbol{\lambda}\mathbf{PA2}^-$. We associate a new variable ξ_x to each first-order variable x. We extend this translation to finite list of first-order variables: each list of variables $\vec{x} \triangleq (x_1,...,x_p)$ is associated with the context $\Xi_{\vec{x}}$ defined by

$$\Xi_{\vec{x}} \triangleq \xi_{x_1} : x_1 \in \mathbb{B},...,\xi_{x_p} : x_p \in \mathbb{B}.$$

For each term t with free variables among $(x_1,...,x_p)$, we construct a proof term t^* (in the system $\boldsymbol{\lambda}\mathbf{PA2}^-$) with free variables among $\xi_{x_1}...,\xi_{x_p}$:

$$
\begin{array}{rcl}
x^* & \triangleq & \xi_x \\
0^* & \triangleq & 0 \\
\langle t,u\rangle^* & \triangleq & \langle t^*,u^*\rangle \\
f(t) & \triangleq & f^*t^*
\end{array}
$$

Proposition 2.2.3.1. *For every first-order term t with free variables in $\vec{x}$:*

$$\Xi_{\vec{x}} \vdash t^* : t \in \mathbb{B} \qquad \textit{(in } \boldsymbol{\lambda}\mathbf{PA2}^-\textit{)}$$

Translation of proof terms

Finally, we can translate proof terms of the system $\boldsymbol{\lambda}\mathbf{PA2}$ into proof terms of the system $\boldsymbol{\lambda}\mathbf{PA2}^-$. In this translation, we use the following term:

$$\mathrm{ind}^* \triangleq \lambda\xi_1\lambda\xi_2,\eta.\pi_1(\eta[\xi_1,0]\lambda\eta_1\lambda\eta_2.[\xi_2(\pi_2\eta_1)(\pi_2\eta_2)(\pi_1\eta_1)(\pi_1\eta_2),\langle\eta_1,\eta_2\rangle])$$

that performs an induction to show the formula $\phi[z := t]$ when

- $\xi_1 : \phi[x := 0]$

- $\xi_2 : \forall x \forall y (\phi[z := x] \Rightarrow \phi[z := y] \Rightarrow \phi[z := \langle x, y \rangle])$
- $\eta : t \in \mathbb{B}$.

To each proof term M (of the system $\boldsymbol{\lambda}$**PA2**) with free variables among $\vec{\xi}, \vec{x}, \overrightarrow{X}$, we associate a proof term M^* (of the system $\boldsymbol{\lambda}$**PA2**$^-$) with free variables among $\vec{\xi}, \Xi_{\vec{x}}$:

$$\begin{array}{rcl}
\xi^* & \triangleq & \xi \\
\mathrm{efq}(M, \phi)^* & \triangleq & M* \\
\mathrm{\propto}(\phi, \psi)^* & \triangleq & \mathrm{\propto} \\
\mathrm{ind}(E, M, N, t)^* & \triangleq & \mathrm{ind}^* M^* N^* t^* \\
\mathrm{refl}(t)^* & \triangleq & \lambda \xi . \xi \\
\mathrm{peel}(M, E, N)^* & \triangleq & M^* N^* \\
(\lambda \xi : \phi . M)^* & \triangleq & \lambda \xi . M^* \\
(MN)^* & \triangleq & M^* N^* \\
(\lambda x . M)^* & \triangleq & \lambda \xi_x . M^* \\
(Mt)^* & \triangleq & M^* t^* \\
(\Lambda X . M)^* & \triangleq & M^* \\
(ME)^* & \triangleq & M^*
\end{array}$$

Note that λ-abstractions over first-order variables are replaced by λ-abstractions over proof variables and that λ-abstractions over second-order variables are erased. During the translation, all the "logical part" of the proof term M is erased or replaced by a program (when a first-order term t is interpreted as a λ-term t^*).

Proposition 2.2.3.2 (Soundness)**.** *If* $\Gamma \vdash M : \phi$ *(in* $\boldsymbol{\lambda}$**PA2***), then:*

$$\Gamma^{\mathbb{B}}, \Xi_{\vec{x}} \vdash M^* : \phi^{\mathbb{B}} \qquad \textit{(in } \boldsymbol{\lambda}\mathbf{PA2}^-\textit{)}$$

where $\vec{x}$ *are the first-order variables that appear freely in* M*.*

Corollary 2.2.3.1. *The theory* **PA2** *is relatively consistent to the theory* **PA2**$^-$*.*

Remark 2.2.3.1. This proof of relative consistency can be adapted to a proof of relative consistency between the systems $\mathbf{\Pi^1_n}$**-CA**$_\mathbf{0}$ and $\mathbf{\Pi^1_n}$**-CA**$^-_\mathbf{0}$ for $n \in \mathbb{N}$.

2.2.4 Related works and future work

Related works

The ideas behind the proof systems $\boldsymbol{\lambda}$**PA2** and $\boldsymbol{\lambda}$**PA2**$^-$ are folklore.

1. Representing the axiom scheme of induction as an inference rule can be seen in many other proof systems as for instance in Martin-Löf Type Theory [33].

2. The idea to use the predicate $\mathrm{null}(t)$ to derive the forth axiom of Peano already appears in the work of Miquel [35].

3. The terminology peel that we use to denote the eliminator of equality is similar to the one used in some presentations of type theory [42], however our own motivation to use it is to emphasize that Leibniz principle can be recovered by doing an external induction on the formulas or, more graphically, by peeling out the syntax. A showcase for this statement is given in the Chapter 6 of this thesis.

4. The constant $\mathfrak{cc}$ denotes the control construct `call/cc` from the programming language Scheme. It was discovered by Griffin [20] that it can be used to give a computational content to Peirce law ; thus allowing to extend the Curry-Howard isomorphism between proofs and programs to classical logic.

5. The congruence on terms and formulas, incorporated on top of the proof systems and used through the conversion rule, allow to prove all the axioms of computation. Adding rewriting rules on top of a type system is very common [33]. For instance, it is intensively used in classical realizability to ease the computations [36].

Finally, interpreting induction though relativization is a trick frequently used in classical realizability [30]. However, this translation allows us to highlight the importance of relativization: with this process, the first-order parts of a term in $\boldsymbol{\lambda}$**PA2** are given a computational meaning in $\boldsymbol{\lambda}$**PA2**$^-$. For instance, a code of recursive functions is interpreted as a λ-term of type $\mathbb{B} \to \mathbb{B}$ computing the function denoted by the code.

Future work

The proof systems presented here lack of computational rules. I made the choice to let all the computational studies of the translations presented in this thesis as future works. Nevertheless, it would be interesting to add an operational semantic on top of the proof systems and to study if the translation respects it. The encoding of $\mathbb{B}$ in $\boldsymbol{\lambda}$**PA2**$^-$ defines what would be called a data type in System F. Results about normal forms in data types of System F are obtained through (intuitionistic) realizability [27]. However, the systems $\boldsymbol{\lambda}$**PA2** and $\boldsymbol{\lambda}$**PA2**$^-$ are classical type systems, that can be equipped with a call-by-name reduction and studied through classical realizability. Data types in this framework have also been investigated [28] and we could certainly apply similar techniques to the study of the computational content of the previous translation.

2.3 Realizability models for $\boldsymbol{\lambda}$PA2$^-$

The system $\boldsymbol{\lambda}$**PA2**$^-$ can be equipped with various semantics. We present here a classical realizability model for this system, as described in the work of Krivine [30]. We claim no originality in this section as all these definitions are folklore[6].

2.3.1 The λ_c-calculus

Syntax

The λ_c-calculus is composed of three distinct types of objects:

1. The terms (M,N...) that extend the proof terms of $\boldsymbol{\lambda}$**PA2**$^-$ with continuations ($\mathbf{k}_\pi$) to capture the operational semantic of $\mathfrak{cc}$ and classical logic. In classical realizability, proof terms can be seen as defending the validity of a formula against opponents: the stacks.

2. The stacks (π, π_1...) are finite list of closed terms that terminate with a stack constant (taken from a set Π_0). They act as environments in which terms are evaluated (in a call-by-name fashion). They can be seen as attacking the validity of a formula and this is why they are sometimes called counter-proof. The interactions between terms and stacks form processes.

[6] Moreover, this section follows a presentation of realizability models done by Alexandre Miquel in an unpublished manuscript.

3. The processes $(p, q...)$ are pairs of closed terms and stacks (denoted $M \star \pi$). They constitute the abstract machine in which a program t is evaluated in an environment π following a call-by-name operational semantic.

Definition 2.3.1.1. Let Π_0 be a countable set of stack constants. Terms, stacks and processes are generated from the following grammar:

Terms	M, N	$::=$	ξ	$\mid$	$\lambda\xi.t$	$\mid$	MN	$\mid$	cc	$\mid$ $\mathbf{k}_\pi$
Stacks	π	$::=$	α	$\mid$	$M \cdot \pi$					($\alpha \in \Pi_0$, M closed)
Processes	p, q	$::=$	$M \star \pi$							(M closed)

Stacks and processes do not contain free variables. The set of closed terms is denoted Λ, the set of stacks is denoted Π and the set of processes is denoted $\Lambda \star \Pi$.

The subset of terms coming from $\boldsymbol{\lambda}\mathbf{PA2}^-$ has a special role in classical realizability, they are called the proof-like terms.

Definition 2.3.1.2. A proof-like term is a term that does not contain a continuation symbol $\mathbf{k}_\pi$.

They will be used when defining the notion of validity in a realizability model.

Operational semantic

Processes are evaluated following a weak head reduction strategy. We consider the following reduction rules on processes:

(Push)	$MN \star \pi$	$\succ_1$	$M \star N \cdot \pi$
(Grab)	$(\lambda x.M) \star N \cdot \pi$	$\succ_1$	$M[x := u] \star \pi$
(Save)	$\mathsf{cc} \star M \cdot \pi$	$\succ_1$	$M \star \mathbf{k}_\pi \cdot \pi$
(Restore)	$\mathbf{k}_\pi \star M \cdot \pi'$	$\succ_1$	$M \star \pi$

These rules are deterministic and encode the weak head reduction of λ-calculus extended with the instruction cc and continuations. The relation of evaluation $p \succ p'$ is defined as the reflexive-transitive closure of $\succ_1$.

2.3.2 Realizability models

In a realizability model, the interpretation of a formula is done in two steps via orthogonality:

1. a set of stacks (closed by anti-reduction) and dubbed "falsity value" is associated to each closed formula (with parameters)

2. a set of closed terms dubbed "truth value" is associated to each closed formula and is computed as the orthogonal of its falsity value.

This interpretation is based on a distinguished set of processes $\perp\!\!\!\perp$ closed by anti-reduction and called the pole of the realizability model.

Definition 2.3.2.1. A pole $\perp\!\!\!\perp$ is a set of processes closed by anti-evaluation:

$$\text{if } p \succ p' \text{ and } p' \in \perp\!\!\!\perp \text{ then } p \in \perp\!\!\!\perp.$$

A choice of pole induces a relation of orthogonality between closed terms and stacks. A (closed) term M is orthogonal to a stack π (denoted $M \perp\!\!\!\perp \pi$) if $M \star \pi \in \perp\!\!\!\perp$. This relation induces an operator $\_^{\perp\!\!\!\perp} : \mathcal{P}(\Pi) \to \mathcal{P}(\Lambda)$ that maps a set of stacks S to a set of closed terms

$$S^{\perp\!\!\!\perp} \triangleq \{M \mid \forall \pi \in S \ (M \perp\!\!\!\perp \pi)\}.$$

Note that such a set $S^{\perp\!\!\!\perp}$ is always closed by anti-evaluation. This operation is crucial for the interpretation of formulas.

Proposition 2.3.2.1. *The operation $S \mapsto S^{\perp\!\!\!\perp}$ is contravariant, meaning that*

$$\text{if } S_1 \subseteq S_2, \text{ then } S_2^{\perp\!\!\!\perp} \subseteq S_1^{\perp\!\!\!\perp}.$$

Remark 2.3.2.1. A realizability model $\mathcal{M}$ is parameterized by a choice of pole. In fact, the framework of classical realizability is extremely modular. The λ_c-calculus and its operational semantic can also be seen as parameters of the realizability model. However, in this thesis, we only consider the version of the λ_c-calculus introduced before.

Interpretation of formulas

In the context of realizability, we consider formulas with parameters, that we define as generated from the language of (the types of) $\boldsymbol{\lambda}\mathbf{PA2}^-$ extended with constant symbols $\dot{b}$ for every pure binary tree $b \in \mathbf{B}$ and predicate symbols $\dot{P}$ for every function $P : \mathbf{B} \to \mathcal{P}(\Pi)$. In a realizability model, the interpretation $t \mapsto [\![t]\!]$ of first-order terms is done as in the standard model of **PA2** (Subsection 1.2.1).

Remark 2.3.2.2. Because the individuals range over the set $\mathbf{B}$, the constants $\dot{b}$ are in fact already definable in the syntax in the sense that for every $b \in \mathbf{B}$, there exist a term $\underline{b}$ such that $[\![\underline{b}]\!] = b$. We still make the distinction between syntactic and semantic trees because it helps to clarify the presentation.

Definition 2.3.2.2. Given a choice of pole $\perp\!\!\!\perp$, the falsity value $\|\phi\|_{\perp\!\!\!\perp}$ and the truth value $|\phi|_{\perp\!\!\!\perp}$ of a closed formula with parameters ϕ are defined by simultaneous induction. The falsity value is defined as follow:

$$\begin{array}{rcl}
\|\mathrm{null}(t)\| & \triangleq & \begin{cases} \{M \cdot \pi \mid (M \star \pi) \in \perp\!\!\!\perp\} & \text{if } [\![t]\!] = 0 \\ \Pi & \text{otherwise} \end{cases} \\
\|t \in \dot{P}\| & \triangleq & P([\![t]\!]) \\
\|\phi \Rightarrow \psi\| & \triangleq & \{M \cdot \pi \mid M \in |\phi|, \pi \in \|\psi\|\} \\
\|\forall x \phi\| & \triangleq & \bigcup_{b \in \mathbf{B}} \|\phi[x := \dot{b}]\| \\
\|\forall X \phi\| & \triangleq & \bigcup_{P : \mathbb{B} \to \mathcal{P}(\Pi)} \|\phi[X := \dot{P}]\|
\end{array}$$

and the truth value $|\phi|$ is obtained by orthogonality:

$$|\phi| \quad \triangleq \quad \|\phi\|^{\perp\!\!\!\perp}.$$

Note that this definition is parameterized by a choice of pole $\perp\!\!\!\perp$. We will leave the pole implicit in the notations if it does not add confusion. As in Tarski semantic, it suffices to consider unary predicate to interpret second-order variables. Because the operation $S \mapsto S^{\perp\!\!\!\perp}$ is contravariant, the interpretation of the implication $\Rightarrow$ is contravariant on its left argument and covariant on its right argument.

The notion of validity

A term M is said to

- realize a formula ϕ for the pole $\perp\!\!\!\perp$ (denoted $M \Vdash_{\perp\!\!\!\perp} \phi$ and abbreviated $M \Vdash \phi$) if $M \in |\phi|$

- universally realize ϕ if $M \Vdash_{\perp\!\!\!\perp} \phi$ for every choice of pole.

If the pole $\perp\!\!\!\perp$ is not empty, every formula is realized.

Proposition 2.3.2.2. *Assume $M \star \pi$ is a process in the pole $\perp\!\!\!\perp$. The term $k_\pi M$ realizes all formulas ϕ.*

Proof. Let $\pi' \in \|\phi\|$. The process $k_\pi M \star \pi'$ reduces to $k_\pi \star M \cdot \pi'$ and then to $M \star \pi$. But $M \star \pi \in \perp\!\!\!\perp$ and because it is closed by anti-reduction, $k_\pi M \star \pi' \in \perp\!\!\!\perp$. □

Therefore, being realized is not a consistent manner to define the validity of a formula. This is where the notion of proof-like terms intervene.

Definition 2.3.2.3. A formula ϕ is valid in $\mathcal{M}$ if it is realized by a proof-like term.

We will see that a realizability model induces a theory and that the consistency of this induced theory can be ensured by a property of the pole.

2.3.3 Soundness and induced theory

Soundness

Classical realizability succeeds in interpreting the rules of second-order classical logic and the axioms of $\mathbf{PA2^-}$. This claim is formalized in the soundness theorem (or adequacy theorem) that we will now state. As a tool to state this theorem, we introduce the notion of valuations.

Definition 2.3.3.1. A valuation ρ is a function mapping

- every first-order variable x to a binary tree $\rho(x) \in \mathbf{B}$
- every second-order variable X to a function $\rho(X) : \mathbf{B} \to \mathcal{P}(\mathbf{B})$.

Given a formula ϕ and a valuation ρ, we write $\phi[\rho]$ for the closed formula with parameters obtained by replacing in the formula ϕ:

- all occurrences of a first-order variable x by the constant symbol associated to the binary tree $\rho(x)$
- all occurrences of a second-order variable X by the predicate symbol associated to the function $\rho(X)$.

Theorem 2.3.3.1 (Soundness)**.** *If a typing judgement $\xi_1 : \phi_1, ..., \xi_k : \phi_k \vdash M : \phi$ is derivable in the system $\boldsymbol{\lambda}\mathbf{PA2^-}$, then for all valuations ρ and for all terms $N_1, ..., N_k$ such that $N_1 \Vdash \phi_1[\rho], ..., N_k \Vdash \phi_k[\rho]$, we have: $M[\xi_1 := N_1, ..., \xi_k := N_k] \Vdash \phi[\rho]$.*

Theory induced by a realizability model and consistency

A realizability model $\mathcal{M}$ induces a theory containing all the formulas realized by a proof-like term. Moreover, the soundness theorem implies that this theory is closed by the rules of deduction of second-order natural deduction. Therefore, a solution to ensure that it is a consistent theory is to check that the formula

$$\begin{aligned} \bot &\triangleq \bigcup_{P:\mathbf{B}\to\mathcal{P}(\Pi)} P(0) \\ &= \Pi \end{aligned}$$

is not realized by a proof-like term.

Proposition 2.3.3.1. *The realizability model $\mathcal{M}$ is coherent if and only if for every proof-like term M, there exists a stack π such that the process $M \star \pi$ is not in the pole $\perp\!\!\!\perp$.*

Example 2.3.3.1. Taking $\perp\!\!\!\perp \triangleq \emptyset$ gives the standard model of **PA2**: in this case, a formula is valid if and only if it is satisfied by the standard model. Non trivial examples of consistent realizability models are presented in the work of Krivine [31].

2.4 Realizing the axiom scheme of collection

We saw that classical realizability interprets the system $\mathbf{PA2}^-$. Even more, we will show that it also interprets the axiom scheme of collection, and gives it a very simple computational content: the identity $\lambda\xi.\xi$ realizes it's contraposition! As a tool to show this result, we construct a family of falsity function $P_\phi : \mathbf{B} \to \mathcal{P}(\Pi)$ that implements an operator allowing to reduce second-order quantification to first-order (unrelativized) quantification. The construction of this operator uses the axiom of choice in the meta-theory.

The falsity functions P_ϕ

Let $x \mapsto \pi_x$ be a surjection from the set of binary trees $\mathbf{B}$ to the set of stacks Π. Such a function exists because $\mathbf{B}$ and Π are both countable. For every formula $\phi(x, X)$, using the axiom of choice in the meta-theory, we consider a family $\{F_b\}_{(b\in\mathbf{B})}$ defined as follow:

- if $b = \langle b_1, b_2\rangle$, and if the set of falsity functions F such that $\pi_{b_2} \in \|\phi(\dot{b_1}, \dot{F})\|$ is non-empty, then F_b is chosen in this set
- F_b is the function $x \mapsto \emptyset$ otherwise.

We now define the falsity function P_ϕ.

$$P_\phi(b'') \triangleq \begin{cases} F_b(b') & \text{if } b'' = \langle b, b'\rangle \\ \emptyset & \text{otherwise} \end{cases}$$

The function P_ϕ is better understood when used with the notion of slices introduced in Definition 1.1.1.4:

$$\|\dot{b'} \in \dot{P_\phi}[\dot{b}]\| = \|\langle \dot{b}, \dot{b'}\rangle \in \dot{P_\phi}\| = P_\phi(\langle b, b'\rangle) = F_b(b') = \|\dot{b'} \in \dot{F_b}\|.$$

Therefore, the slice of $\dot{P_\phi}$ at the individual $\dot{b}$ is interpreted as $\dot{F_b}$. It will be of interest when $b \triangleq \langle b_1, b_2\rangle$ and when b_2 is considered as the stack π_{b_2}. In the light of this encoding, the falsity function $P_\phi(x)$ will be used to replace a quantification over second-order variables by a quantification over first-order variables. As a first step, we prove the following lemma.

Lemma 2.4.0.1. *For all binary tree $b_1 \in \mathbf{B}$:*

$$\|\forall Y \phi(\dot{b_1}, Y)\| \subseteq \|\forall y \phi(\dot{b_1}, \dot{P_\phi}[y])\|.$$

Proof. Let $\pi = \pi_{b_2}$ for some tree b_2. If $\pi \in \|\forall Y \phi(\dot{b_1}, Y)\| = \bigcup_{F:\mathbf{B}\to\mathcal{P}(\Pi)} \|\phi(\dot{b_1}, \dot{F})\|$, there is a falsity function F_0 such that $\pi_{b_2} \in \|\phi(\dot{b_1}, \dot{F_0})\|$.

But the slice $P_\phi[\langle b_1, b_2\rangle]$ of P_ϕ is interpreted as $F_{\langle b_1, b_2\rangle}$. Therefore, by definition, it is interpreted as some falsity function F_1 such that

$$\pi_{b_2} \in \|\phi(\dot{b_1}, \dot{F_1})\| \subseteq \|\forall y \phi(\dot{b_1}, \dot{P_\phi}[y])\|.$$

□

Proposition 2.4.0.1. *The formula* $\forall x(\forall y \phi(x, P_\phi[y]) \Rightarrow \forall Y \phi(x, Y))$ *is universally realized.*

Proof. For all individuals b, $\|\forall Y \phi(\dot{b}, Y)\| \subseteq \|\forall y \phi(\dot{b}, \dot{P_\phi}[y])\|$. Therefore $\lambda\xi.\xi$ realizes the formula $\forall x(\forall y \phi(x, P_\phi[y]) \Rightarrow \forall Y \phi(x, Y))$. □

The contraposition of this formula is maybe more meaningful:

$$\forall x(\exists Y \phi(x, Y) \Rightarrow \exists y \phi(x, \dot{P_\phi}[y]))$$

and is also realizable (as a consequence of the Adequacy theorem 2.3.3.1). It means that the family $\{P_\phi[b]\}_{b \in \mathbf{B}}$ represents a sequence of potential witnesses for the (higher-order) predicate $\{X \mid \phi(\dot{b}, X)\}$.

Remark 2.4.0.1. We did this construction for a formula with one distinguished first-order variable but it would have been (a little) easier to do it for a formula of the shape $\phi(X)$. However, this is the former that we will use to realize the axiom scheme of collection.

Realizing the axiom scheme of collection

Fixing a formula $\phi(x, Y)$, we will rather realize the equivalent formula

$$\forall Z \exists x \forall y \phi(x, Z[y]) \Rightarrow \exists x \forall Y \phi(x, Y).$$

It is in fact realized by $\lambda\xi.\xi$. We show that

$$\|\exists x \forall Y \phi(x, Y)\| \subseteq \|\forall Z \exists x \forall y \phi(x, Z[y])\|.$$

Unfolding the encoding of the existential quantifiers, the problem reduces[7] to

$$\|\neg\forall x \neg\forall Y \phi(x, Y)\| \subseteq \|\forall Z \neg\forall x \neg\forall y \phi(x, Z[y])\|.$$

It is enough to show

$$\|\neg\forall x \neg\forall Y \phi(x, Y)\| \subseteq \|\neg\forall x \neg\forall y \phi(x, \dot{P_\phi}[y])\|.$$

Because the interpretation of the implication is contravariant in its first argument[8], the problem again reduces to

$$\|\forall x \neg\forall y \phi(x, \dot{P_\phi}[y])\| \subseteq \|\forall x \neg\forall Y \phi(x, Y)\|.$$

Finally, for all individuals b, the assertion

$$\|\neg\forall y \phi(\dot{b}, \dot{P_\phi}[y])\| \subseteq \|\neg\forall Y \phi(\dot{b}, Y)\|$$

follows from the contravariance of the interpretation of the negation coupled with Lemma 2.4.0.1.

Theorem 2.4.0.1. *For every formula with parameters* $\phi(x, X)$*, the formula*

$$\forall Z \exists x \forall y \phi(x, Z[y]) \Rightarrow \exists x \forall Y \phi(x, Y)$$

is universally realized by $\lambda\xi.\xi$*.*

Corollary 2.4.0.1. *Every instance of the axiom scheme of collection is universally realized.*

[7] Or expands!

[8] And because the negation is encoded as an implication.

2.4.1 Related work

Classical realizability models of second-order logic were introduced by Krivine [30] where he showed that both the axioms of countable choice and of dependent choice are realized. Concretely, he showed that these principles are realized using an instruction `quote` internalizing in the calculus the injection from the stacks into the integers. In a second step, Krivine extended classical realizability to non-extensional set theory $\mathbf{ZF}_\varepsilon$ [31] and showed that, again, the axioms of countable choice and dependent choice can be realized. In this framework, different versions of axioms of choice were studied. For instance, Fontanella and Geoffroy showed that weak forms of Zorn's lemma can be realized [16]. The full axiom of choice was even showed to be realized by Krivine [32]. However, it seems that its computational content is still unclear and remains to be studied.

Finally, we were strongly inspired by the proof that the axiom of non-extensional choice is realized [31]. The functions P_ϕ, employed as a variant of Hilbert's epsilon for second-order logic, are also used when proving that this principle is realized.

Chapter 3

A glimpse at the intuitionistic world

Intuitionistic logic is obtained from classical logic by removing the law of excluded-middle from its set of rules. In this chapter, we will study variants of second-order arithmetic in this setting. We will start by presenting a minimal syntax that will be used to describe second-order arithmetic in the intuitionistic framework (Section 3.1), relying on second-order encodings to define the missing connectives.

We will then describe a negative translation from the proof system **λPA2** (for second-order classical arithmetic with induction) into the proof system $\boldsymbol{\lambda}\mathbf{HA2}^{-}$ (for second-order intuitionistic arithmetic without induction) in Section 3.2.1. This translation extends the one presented in Section 2.2.3 by relativizing the second-order quantifiers to the class of "stable" predicates and, thus, interpreting second-order variables in the source calculus by proofs that they denote a stable predicate in the target calculus. Hoping that the axiom scheme of collection and the axiom scheme of choice will behave differently with respect to negative translations, we will study their interpretations in Section 3.2.2. However, both behaved similarly: they admit classically equivalent formulations that are validated by the negative translation. This part was deeply inspired by the work of Friedman about negative translations of set theory [17].

All in all, the study of negative translations failed to separate the axiom scheme of collection and the axiom scheme of choice. However, a glimmer of hope appears in the study of intuitionistic realizability models of second-order arithmetic (Section 3.3). We will exhibit a model separating variants of these schemes: it will validate the former and refute the latter! Nevertheless, we will not be able to lift this result to second-order classical arithmetic: it would have been possible only if we managed to separate their negative translations.

3.1 Arithmetic in second-order intuitionistic logic

3.1.1 Intuitionistic second-order logic and second-order encodings

In this part, we present a version of second-order logic with only membership as atomic formula and with only implication and universal quantifiers as connectives. Therefore, the syntax of the formulas is the following

$$\phi, \psi \quad ::= \quad t \in X \mid \phi \Rightarrow \psi \mid \forall x \phi \mid \forall X \phi$$

The other connectives are defined using a second-order encoding.

Definition 3.1.1.1. We define the following encodings in second-order logic where Z is a fresh variable not contained in ϕ and ψ:

$$\begin{array}{rcl@{\qquad}rcl} \bot & \triangleq & \forall Z\,(0 \in Z) & t = u & \triangleq & \forall Z\,(t \in Z \Rightarrow u \in Z) \\ \phi \wedge \psi & \triangleq & \forall Z((\phi \Rightarrow \psi \Rightarrow 0 \in Z) \Rightarrow 0 \in Z) & \phi \vee \psi & \triangleq & \forall Z((\phi \Rightarrow 0 \in Z) \Rightarrow (\psi \Rightarrow 0 \in Z) \Rightarrow 0 \in Z) \\ \exists x \phi(x) & \triangleq & \forall Z(\forall x(\phi(x) \Rightarrow 0 \in Z) \Rightarrow 0 \in Z) & \exists X \phi(X) & \triangleq & \forall Z(\forall x(\phi(x) \Rightarrow 0 \in Z) \Rightarrow 0 \in Z) \end{array}$$

It is folklore[1] [19] that these encodings satisfy the rules of the encoded connectives (described for instance in Figure 1.2 page 24).

Remark 3.1.1.1. Note however that they use the full strength of the axiom scheme of comprehension (or, in other words, the full strength of second-order quantification). This is why an other encoding is used in the classical setting. Concretely, having a boolean encoding of the other connectives allows us to prove closure results about the subsystems of **PA2** defined by restriction of the axiom scheme of comprehension.

Definition 3.1.1.2. The rules of natural deduction for second-order intuitionistic logic are the one presented in Figure 1.1 page 23 where the rules for the equality, the ex-falso quodlibet and the elimination of the double negation are removed. We will write $\Gamma \vdash_I \phi$ when this sequent is provable with the rules of intuitionistic logic.

Definition 3.1.1.3. The different variations of arithmetic in the framework of intuitionistic logic are defined exactly as in the classical framework, but all the connectives used in the axioms are now considered as defined with second-order encodings. We will denote **HA2** (resp. $\mathbf{HA2}^-$) for the analogous of **PA2** (resp. $\mathbf{PA2}^-$) in intuitionistic logic.

A type system for $\mathbf{HA2}^-$: the system $\boldsymbol{\lambda}\mathbf{HA2}^-$

We will now describe a type system for the theory $\mathbf{HA2}^-$ formulated in the previous syntax. The system $\boldsymbol{\lambda}\mathbf{HA2}^-$ is a typed λ-calculus à la Curry: it is the Curry-Howard counter part of the theory $\mathbf{HA2}^-$,it is obtained from $\boldsymbol{\lambda}\mathbf{PA2}^-$ by removing the term $\mathfrak{cc}$. This system is an extension of System F [19] obtained by adding first-order terms in the language of types.

Definition 3.1.1.4. The syntax and the rules of the system $\boldsymbol{\lambda}\mathbf{HA2}^-$ are presented in the Figure 3.1 page 58.

Because this system is equipped with an unrestricted elimination rule for second-order quantification, the second-order encodings described in Definition 3.1.1.1 take their intended meaning.

Proposition 3.1.1.1. *The rules presented in Figure 3.2 p.59 are derivable.*

Notation 3.1.1.1. In particular, we introduce notations[2] to use the second-order encodings of the conjunctions and disjunctions.

- The proof term capturing the introduction rule of the conjunction is unchanged:

$$[M, N] \quad \triangleq \quad \lambda\xi.\xi MN \qquad \text{for } \xi \notin (FV(M) \cup FV(N)).$$

- The proof terms capturing the elimination rules of the conjunction are now defined to be:

$$\pi_1 \triangleq \lambda\xi.\xi\lambda\eta_1\lambda\eta_2.\eta_1 \qquad \pi_2 \triangleq \lambda\xi.\xi\lambda\eta_1\lambda\eta_2.\eta_2.$$

[1] Modulo the fact that $\phi \triangleq 0 \in \{z \mid \phi\}$ for $z \notin FV(\phi)$.
[2] In fact, we will overload them in the case of the conjunction.

Syntax

Codes of functions	f, g	::=	$\mathbf{0} \mid \mathrm{id} \mid \mathrm{fst} \mid \mathrm{snd} \mid (f \circ g) \mid \langle f, g\rangle \mid [f\vert g]$
Terms	t, u	::=	$x \mid 0 \mid \langle t, u\rangle \mid f(t)$
Formulas	ϕ, ψ	::=	$\mathrm{null}(t) \mid t \in X \mid \phi \Rightarrow \psi \mid \forall x\phi \mid \forall X\phi$
Sets	E	::=	$\{x \mid \phi\}$
Proof terms	M, N	::=	$\xi \mid \lambda\xi.t \mid MN$
Contexts	Γ	::=	$\emptyset \mid \Gamma, \xi : \phi$

Congruence on terms and formulas

Same as the rules in $\boldsymbol{\lambda}\mathbf{PA2}^-$, see Figure 2.2 page 45

Typing rules

$$\frac{}{\Gamma \vdash_I \xi : \phi}\,(\xi:\phi)\in\Gamma \qquad \frac{\Gamma \vdash_I M : \phi}{\Gamma \vdash_I M : \phi'}\,(\phi \simeq \phi')$$

$$\frac{\Gamma, \xi : \phi \vdash_I M : \psi}{\Gamma \vdash_I \lambda\xi.M : \phi \Rightarrow \psi} \qquad \frac{\Gamma \vdash_I M : \phi \Rightarrow \psi \quad \Gamma \vdash_I N : \phi}{\Gamma \vdash_I MN : \psi}$$

$$\frac{\Gamma \vdash_I M : \phi}{\Gamma \vdash_I M : \forall x\phi}\,x \notin FV(\Gamma) \qquad \frac{\Gamma \vdash_I M : \forall x\phi}{\Gamma \vdash_I M : \phi[x := t]}$$

$$\frac{\Gamma \vdash_I M : \phi}{\Gamma \vdash_I M : \forall X\phi}\,X \notin FV(\Gamma) \qquad \frac{\Gamma \vdash_I M : \forall X\phi}{\Gamma \vdash_I M : \phi[X := E]}$$

Figure 3.1: The proof system $\boldsymbol{\lambda}\mathbf{HA2}^-$

- The proofs term capturing the introduction rules of the disjunction are defined as follow:

$$\mathrm{inl} \triangleq \lambda\xi\lambda\xi_1\lambda\xi_2.\xi_1\xi \qquad \mathrm{inr} \triangleq \lambda\xi\lambda\xi_1\lambda\xi_2.\xi_2\xi.$$

- The proof term capturing the elimination rule of the disjunction is defined as follow:

$$\mathrm{destruct}^{\vee}_{\xi} M \text{ in } N_1(\xi) \mid N_2(\xi) \quad \triangleq \quad M(\lambda\xi.N_1(\xi))(\lambda\xi.N_2(\xi)) \ .$$

As for $\boldsymbol{\lambda}\mathbf{PA2}^-$ and $\mathbf{PA2}^-$ (see Theorem 2.2.2.1), the type system $\boldsymbol{\lambda}\mathbf{HA2}^-$ interprets $\mathbf{HA2}^-$.

Theorem 3.1.1.1. *Every axiom of* $\mathbf{HA2}^-$ *is provable in* $\boldsymbol{\lambda}\mathbf{HA2}^-$.

Theorem 3.1.1.2. *If* $\vdash_I M : \phi$ *then* $\mathbf{HA2}^-$ *proves the formula obtained from* ϕ *by replacing all occurrences of the predicate* $\mathrm{null}(t)$ *by the formula* $t = 0$.

3.2 Negative translations

In this section, we will study two negative translations.

$$\frac{\Gamma \vdash_I M : \bot}{\Gamma \vdash_I M : \phi}$$

$$\frac{}{\Gamma \vdash_I \lambda\xi.\xi : \forall x\, x = x} \qquad \frac{\Gamma \vdash_I M : t = u \quad \Gamma \vdash_I N : \phi[x := t]}{\Gamma \vdash_I MN : \phi[x := u]}$$

$$\frac{\Gamma \vdash_I M : \phi \quad \Gamma \vdash_I N : \psi}{\Gamma \vdash_I \lambda\xi.\xi M N : \phi \wedge \psi} \qquad \frac{\Gamma \vdash_I M : \phi \wedge \psi}{\Gamma \vdash_I M\lambda\xi_1\lambda\xi_2.\xi_1 : \phi} \qquad \frac{\Gamma \vdash_I M : \phi \wedge \psi}{\Gamma \vdash_I M\lambda\xi_1\lambda\xi_2.\xi_2 : \psi}$$

$$\frac{\Gamma \vdash_I M : \phi}{\Gamma \vdash_I \lambda\xi_1.\lambda\xi_2.\xi_1 M : \phi \vee \psi} \qquad \frac{\Gamma \vdash_I M : \psi}{\Gamma \vdash_I \lambda\xi_1.\lambda\xi_2.\xi_2 M : \phi \vee \psi}$$

$$\frac{\Gamma \vdash_I M : \phi \vee \psi \quad \Gamma, \xi_1 : \phi \vdash_I N_1 : \chi \quad \Gamma, \xi_2 : \psi \vdash_I N_2 : \chi}{\Gamma \vdash_I M(\lambda\xi_1.N_1)(\lambda\xi_2.N_2) : \chi}$$

$$\frac{\Gamma \vdash_I M : \phi[x := t]}{\Gamma \vdash_I \lambda\xi.\xi M : \exists x \phi} \qquad \frac{\Gamma \vdash_I M : \exists x \phi \quad \Gamma, \xi : \phi \vdash_I N : \psi}{\Gamma \vdash_I M\lambda\xi.N : \psi}\, {\scriptstyle x \notin FV(\Gamma,\psi)}$$

$$\frac{\Gamma \vdash_I M : \phi[X := Y]}{\Gamma \vdash_I \lambda\xi.\xi M : \exists X \phi} \qquad \frac{\Gamma \vdash_I M : \exists X \phi \quad \Gamma, \xi : \phi \vdash_I N : \psi}{\Gamma \vdash_I M\lambda\xi.N : \psi}\, {\scriptstyle X \notin FV(\Gamma,\psi)}$$

Figure 3.2: Rules of defined connectives.

1. The first one will be fully described as a translation between type systems and will show a result of relative consistency between **PA2** and $\mathbf{HA2}^-$.

2. The second one will be described only at the level of formulas. It will be an interpretation of $\mathbf{PA2}^-$ inside $\mathbf{HA2}^-$. The most important difference with the first one is that it does not interpret induction. We will then show that both the axiom of choice and the axiom of collection behave similarly with respect to this translation. Notably, we will show how it interprets classically equivalent reformulations of these schemes.

3.2.1 Negative translation of λPA2 into λHA2⁻

In this section, we design a negative translation from **λPA2** into $\boldsymbol{\lambda}\mathbf{HA2}^-$. This translation will extend the translation defined in Section 2.2.3 by relativizing the second-order quantifiers to the class of stable predicates defined as

$$\mathrm{Sbl} \triangleq \{X \mid (\forall x \in \mathbb{B})(\neg\neg x \in X \Rightarrow x \in X)\}$$

. The steps to design this translation are described below.

1. Formulas ϕ and contexts Γ of **λPA2** are translated into formulas $\mathbf{G}_{\mathbb{B}}(\phi)$ and contexts $\mathbf{G}_{\mathbb{B}}(\Gamma)$ of $\boldsymbol{\lambda}\mathbf{HA2}^-$. This translation extends the translation $\phi \mapsto \phi^{\mathbb{B}}$ by relativizing the second-orders quantifiers to the class of stable predicates.

2. First-order terms t are again translated into proof terms t^* of $\boldsymbol{\lambda}\mathbf{HA2}^-$ (of type $t \in \mathbb{B}$). This part of the translation does not change.

3. Formulas ϕ are translated into proof terms ϕ^* (of type $\neg\neg\mathbf{G}_{\mathbb{B}}(\phi) \Rightarrow \mathbf{G}_{\mathbb{B}}(\phi)$).

4. Proof terms M of **λPA2** are translated into proof terms $\mathbf{G}_{\mathbb{B}}(M)$ of $\boldsymbol{\lambda}\mathbf{HA2}^{-}$. This translation extends the translation $M \mapsto M^*$ (defined in Section 2.2.3) by interpreting the second-order logical annotations contained in M by computational machinery in $\mathbf{G}_{\mathbb{B}}(M)$. Notably, it interprets the proof term $\mathbf{cc}(\phi, \psi)$ by a term whose free variables are generated from the free first-order and second-order variables of ϕ.

Translation of formulas, contexts and sets

We associate to every formula ϕ in the language of **λPA2** a formula $\mathbf{G}_{\mathbb{B}}(\phi)$ in the language of $\boldsymbol{\lambda}\mathbf{HA2}^{-}$. This translation differs from the translation $\phi \mapsto \phi^{\mathbb{B}}$ in only one case:

$$\mathbf{G}_{\mathbb{B}}(\forall X \phi) \triangleq \forall X(\mathrm{Sbl}(X) \Rightarrow \mathbf{G}_{\mathbb{B}}(\phi)).$$

All in all, this translation replaces the primitive formula $\bot$ and $t = u$ by their second-order encodings (in $\boldsymbol{\lambda}\mathbf{HA2}^{-}$), relativizes the first-order quantifiers to the set $\mathbb{B}$ and relativizes the second-order quantifiers to the class of stable predicates. In particular, the atomic formula $t \in X$ is translated into itself, without the use of double negation in front of it!

Lemma 3.2.1.1. *If $\phi \simeq \phi'$ in* **λPA2** *then $\mathbf{G}_{\mathbb{B}}(\phi) \simeq \mathbf{G}_{\mathbb{B}}(\phi')$ in* $\boldsymbol{\lambda}\mathbf{HA2}^{-}$.

The translation $\phi \mapsto \mathbf{G}_{\mathbb{B}}(\phi)$ extends to sets and contexts:

$$\mathbf{G}_{\mathbb{B}}(\{x \mid \phi\}) \triangleq \{x \mid \mathbf{G}_{\mathbb{B}}(\phi)\} \quad \text{and} \quad (\xi_1 : \phi_1, ..., \xi_n : \phi_n)^{\mathbb{B}} \triangleq \xi_1 : \mathbf{G}_{\mathbb{B}}(\phi_1), ..., \xi_n : \mathbf{G}_{\mathbb{B}}(\phi_n)$$

Translation of first-order terms

This part does not change: it is done as in Section 2.2.3.

Translation of formulas into proof terms

Lemma 3.2.1.2. *We define the following terms that will be used in the translation.*

$$\begin{array}{rcl} \mathrm{Func}_{\neg\neg} & \triangleq & \lambda\xi\lambda\eta\lambda\chi.\eta(\lambda\mu.\chi(\xi\mu)) \\ \mathrm{eq}_{\langle \_,\_ \rangle} & \triangleq & \lambda\xi\lambda\eta.\xi(\eta\lambda\chi.\chi) \\ \mathrm{eq}_{\mathrm{fst}} & \triangleq & \lambda\xi.\xi(\lambda\chi.\chi) \\ \mathrm{eq}_{\mathrm{snd}} & \triangleq & \lambda\xi.\xi(\lambda\chi.\chi) \end{array}$$

They have the following properties

$$\begin{array}{rcl} \vdash_I \mathrm{Func}_{\neg\neg} & : & \forall X \forall Y((0 \in X \Rightarrow 0 \in Y) \Rightarrow \neg\neg 0 \in X \Rightarrow \neg\neg 0 \in Y) \\ \vdash_I \mathrm{eq}_{\langle \_,\_ \rangle} & : & \forall x_1 \forall x_2 \forall y_1 \forall y_2 (x_1 = y_1 \Rightarrow x_2 = y_2 \Rightarrow \langle x_1, x_2 \rangle = \langle y_1, y_2 \rangle) \\ \vdash_I \mathrm{eq}_{\mathrm{fst}} & : & \forall x_1 \forall x_2 \forall y_1 \forall y_2 (\langle x_1, x_2 \rangle = \langle y_1, y_2 \rangle \Rightarrow x_1 = y_1) \\ \vdash_I \mathrm{eq}_{\mathrm{snd}} & : & \forall x_1 \forall x_2 \forall y_1 \forall y_2 (\langle x_1, x_2 \rangle = \langle y_1, y_2 \rangle \Rightarrow x_2 = y_2) \end{array}$$

We associate a new proof variable ξ_X to each second-order variable X. From each finite list of second-order variables $\vec{X} \triangleq (X_1, ..., Xp)$, we associate a context $\Xi_{\vec{X}}$ defined by

$$\Xi_{\vec{X}} \triangleq \xi_{X_1} : \mathrm{Sbl}(X_1), ..., \xi_{X_p} : \mathrm{Sbl}(X_p).$$

For each formula ϕ with free variables among $(x_1, ..., x_k, X_1, ..., X_p)$, we construct a proof term ϕ^* (in the system $\boldsymbol{\lambda}\mathbf{HA2}^{-}$) with free variables among $\xi_{x_1} ..., \xi_{x_k}, \xi_{X_1} ..., \xi_{X_p}$:

$$
\begin{array}{rcl}
\bot^* & \triangleq & \lambda\xi.\xi(\lambda\eta.\eta) \\
(t=u)^* & \triangleq & t^*(u^*(\lambda\xi\lambda\_.\_)(\lambda\_\lambda\_\lambda\xi.\xi(\lambda\eta.\eta\lambda\_.\_))) \\
 & & (\lambda\xi_{x_1}\lambda\xi_{x_2}.u^*(\lambda\xi.\xi(\lambda\eta.\eta(\lambda\_.\_)(\lambda\_.\_)))(\lambda\_\lambda\_.\lambda\xi.\mathrm{eq}_{\langle\_,\_\rangle}(\xi_{x_1}(\mathrm{Func}_{\neg\neg}\mathrm{eq}_{\mathrm{fst}}\xi))(\xi_{x_2}(\mathrm{Func}_{\neg\neg}\mathrm{eq}_{\mathrm{snd}}\xi)))) \\
\mathrm{null}(t)^* & \triangleq & t^*(\lambda\xi\lambda\eta.\eta)(\lambda\_\lambda\_\lambda\xi.\xi(\lambda\eta.\eta)) \\
(t\in X)^* & \triangleq & \xi_X t^* \\
(\phi\Rightarrow\psi)^* & \triangleq & \lambda\xi.\lambda\eta.\psi^*(\lambda\chi.\xi(\lambda\mu.\chi(\mu\eta))) \\
(\forall x\phi)^* & \triangleq & \lambda\xi\lambda\xi_x.\phi^*(\lambda\eta.\xi(\lambda\chi.\eta(\chi\xi_x))) \\
(\forall X\phi)^* & \triangleq & \lambda\xi\lambda\xi_X.\phi^*(\lambda\eta.\xi(\lambda\chi.\eta(\chi\xi_X)))
\end{array}
$$

Proposition 3.2.1.1. *For every formula ϕ with free first-order variables among $\vec{x}$ and with free second-order variables among $\vec{X}$:*

$$\Xi_{\vec{x}}, \Xi_{\vec{X}} \vdash \phi^* : \neg\neg\mathbf{G}_{\mathbb{B}}(\phi) \Rightarrow \mathbf{G}_{\mathbb{B}}(\phi) \qquad \textit{(in } \boldsymbol{\lambda}\mathbf{HA2}^-\textit{)}$$

Proof. This proof is done by induction on the syntax of formulas of $\boldsymbol{\lambda}\mathbf{PA2}$. □

Translation of proof terms

Finally, we can translate proof terms of the system $\boldsymbol{\lambda}\mathbf{PA2}$ into proof terms of the system $\boldsymbol{\lambda}\mathbf{HA2}^-$.

To each proof term M (of the system $\boldsymbol{\lambda}\mathbf{PA2}$) with free variables among $\vec{\xi}, \vec{x}, \overrightarrow{X}$, we associate a proof term $\mathbf{G}_{\mathbb{B}}(M)$ (of the system $\boldsymbol{\lambda}\mathbf{HA2}^-$) with free variables among $\vec{\xi}, \Xi_{\vec{x}}, \Xi_{\vec{X}}$. It differs from the previous translation in the following cases:

$$
\begin{array}{rcl}
\mathbf{G}_{\mathbb{B}}(\mathrm{cc}(\phi,\psi)) & \triangleq & \lambda\xi.\phi^*(\lambda\eta.\eta(\xi\lambda\chi.\eta\chi)) \\
\mathbf{G}_{\mathbb{B}}(\Lambda X.M) & \triangleq & \lambda\xi_X.\mathbf{G}_{\mathbb{B}}(M) \\
\mathbf{G}_{\mathbb{B}}(M\{x\mid\phi\}) & \triangleq & \mathbf{G}_{\mathbb{B}}(M)\lambda\xi_x.\phi^*
\end{array}
$$

During the translation, all the "logical part" of the proof term M is erased or replaced by a program: λ-abstractions over first-order and second-order variables are replaced by λ-abstractions over proof variables.

Proposition 3.2.1.2 (Soundness)**.** *If $\Gamma \vdash M : \phi$ (in $\boldsymbol{\lambda}\mathbf{PA2}$), then:*

$$\mathbf{G}_{\mathbb{B}}(\Gamma), \Xi_{\vec{x}}, \Xi_{\vec{X}} \vdash_I \mathbf{G}_{\mathbb{B}}(M) : \mathbf{G}_{\mathbb{B}}(\phi) \qquad \textit{(in } \boldsymbol{\lambda}\mathbf{HA2}^-\textit{)}$$

where $\vec{x}, \vec{X}$ are the (first-order and second-order) variables that appear freely in M.

Corollary 3.2.1.1. *The theory* $\mathbf{PA2}$ *is relatively consistent to the theory* $\mathbf{HA2}^-$.

On the originality of this translation

The previous result is not new [49]. Nor is the idea that classical logic can be interpreted in a model of intuitionistic impredicative logic by restricting the set of "truth values" to stable one. For instance, it is used in topos theory to show that every topos induces a boolean topos [24].

However, up to the author knowledge, describing a syntactic translation that implements, by relativization, classical logic within an impredicative intuitionistic system is, at least, uncommon.

3.2.2 Interpreting collection and choice through negative translation

We are now interested in interpreting the axiom scheme of collection and the axiom scheme of choice through a negative translation. We formulate an other negative translation, more suited to the study of these schemes.

An interpretation of PA2$^-$ inside HA2$^-$

In this section, we design a negative translation interpreting **PA2**$^-$ inside **HA2**$^-$. We recall that the formulas of **PA2**$^-$ are constructed using the syntax presented in Definition 1.1.1.1 and that the axioms of **PA2**$^-$ are the axioms of injectivity, of non confusion, of computation and the axiom scheme of comprehension presented in Definition 1.2.1.1. Finally recall that in **PA2**$^-$ the other connectives $(\wedge, \vee, \exists)$ are defined using a classical encoding presented in Definition 1.1.1.2.

Definition 3.2.2.1. We design a translation $\phi \mapsto \mathbf{G}(\phi)$ from formulas of **PA2**$^-$ into formulas of **HA2**$^-$ as follow:

$$\begin{array}{rcl}
\mathbf{G}(\bot) & \triangleq & \forall Z(0 \in Z) \qquad (= \bot) \\
\mathbf{G}(t = u) & \triangleq & \neg\neg\forall Z(t \in Z \Rightarrow u \in Z) \\
\mathbf{G}(t \in X) & \triangleq & \neg\neg\, t \in X \\
\mathbf{G}(\phi \Rightarrow \psi) & \triangleq & \mathbf{G}(\phi) \Rightarrow \mathbf{G}(\psi) \\
\mathbf{G}(\forall x \phi) & \triangleq & \forall x \mathbf{G}(\phi) \\
\mathbf{G}(\forall X \phi) & \triangleq & \forall X \mathbf{G}(\phi)
\end{array}$$

This translation differs from the translation $\mathbf{G}_{\mathbb{B}}(\_)$ in three manners:

1. The first-order quantifiers are not relativized. In particular, induction is not provable in the image of the translation (and neither in the source).
2. As a consequence, the equality is not decidable and double negations needs to be placed in front of its second-order encoding.
3. The second-order quantifiers are not relativized. It implies that double negations needs to be put in front of the atomic formulas of the shape $t \in X$.

We fix the convention that if the symbol of a connective not in the syntax of **HA2**$^-$ appears outside of a $\mathbf{G}(\_)$, it denotes the second-order encoding of this connective. For instance, $\bot \triangleq \forall Z(0 \in Z)$. On the other hand, the translation of classically encoded connectives in **PA2**$^-$ induces extra connectives in **HA2**$^-$.

Definition 3.2.2.2. We introduce theses new connectives in **HA2**$^-$:

$$\begin{array}{rcl}
\phi \vee^{\mathrm{c}} \psi & \triangleq & (\phi \Rightarrow \bot) \Rightarrow (\psi \Rightarrow \bot) \Rightarrow \bot \\
\psi \wedge^{\mathrm{c}} \psi & \triangleq & (\phi \Rightarrow \psi \Rightarrow \bot) \Rightarrow \bot \\
\exists^{\mathrm{c}} x \phi & \triangleq & \neg\forall x \neg\phi \\
\exists^{\mathrm{c}} X \phi & \triangleq & \neg\forall X \neg\phi.
\end{array}$$

These new connectives, coming from the negative translations of encoded connectives in **PA2**$^-$, are "weaker" than their intuitionistic counterpart in **HA2**$^-$. It is shown by the following proposition.

Proposition 3.2.2.1. **HA2**$^-$ *proves that*

$$\begin{array}{l}
\forall X \forall Y((0 \in X \vee 0 \in Y) \Rightarrow (0 \in X \vee^{\mathrm{c}} 0 \in Y)) \\
\forall X \forall Y((0 \in X \wedge 0 \in Y) \Rightarrow (0 \in X \wedge^{\mathrm{c}} 0 \in Y)) \\
\forall X(\exists x(x \in X) \Rightarrow \exists^{\mathrm{c}} x(x \in X)) \\
(\exists X \phi) \Rightarrow (\exists^{\mathrm{c}} X \phi) \qquad (\textit{for all formulas } \phi).
\end{array}$$

The last technical details needed in this section are described in the following lemma.

Lemma 3.2.2.1. *These formulas are provable in* $\mathbf{HA2^-}$*:*

$$\forall X(0 \in X \Rightarrow \neg\neg 0 \in X)$$
$$\forall X \forall Y((0 \in X \Rightarrow 0 \in Y) \Rightarrow \neg\neg 0 \in X \Rightarrow \neg\neg 0 \in Y$$
$$\neg\neg \mathbf{G}(\phi) \Rightarrow \mathbf{G}(\phi) \qquad (\textit{for all formulas } \phi).$$

Proposition 3.2.2.2. *If* $\mathbf{PA2^-}$ *proves a formula* ϕ *then* $\mathbf{HA2^-}$ *proves* $\mathbf{G}(\phi)$*.*

Proof. This result is very similar to the one obtained in the last section about the translation $\mathbf{G}_{\mathbb{B}}(\_)$. We show that $\mathbf{G}(\_)$ succeeds in interpreting all the axioms of $\mathbf{PA2^-}$ in $\mathbf{HA2^-}$. In other words, we show that $\mathbf{HA2^-}$ proves the translation of all the axioms of $\mathbf{PA2^-}$.

1. All the axioms of computation and the axiom of non confusion are validated because $\phi \Rightarrow \neg\neg\phi$ is provable for every formula ϕ.
2. The axiom of injectivity is validated because of the functorality of $\phi \mapsto \neg\neg\phi$ and the closure by double negation of formulas of the shape $\mathbf{G}(\phi)$.
3. For the axiom scheme of comprehension, it is enough to prove $\exists X \forall x(\neg\neg x \in X \Leftrightarrow \mathbf{G}(\phi))$. It is a consequence of the axiom scheme of comprehension used with the formula $\mathbf{G}(\phi)$.

□

Interpretation of the axiom scheme of collection

It appears that our formulation of the axiom of collection is not suited to be interpreted via the translation $\phi \mapsto \mathbf{G}(\phi)$. Concretely, we did not succeed to prove that the axiom of collection implies its negative translation[3]. Taking inspiration from the work of Friedman about his interpretation of classical set theory within intuitionistic set theory [17], we formulate a classically equivalent form of this axiom scheme.

Definition 3.2.2.3. The axiom scheme of collection with domain ($\mathbf{Coll_{Dom}}$) is defined as the closure of the formulas

$$\forall D((\forall x \in D)\exists Y \phi(x, Y) \Rightarrow \exists Z(\forall x \in D)\exists y \phi(x, Z[y])).$$

Proposition 3.2.2.3. *1. The axiom scheme of collection and the axiom scheme of collection with domain are equivalent in* $\mathbf{PA2^-}$*.*

2. The axiom scheme of collection with domain implies the axiom scheme of collection in $\mathbf{HA2^-}$*.*

As a tool to show that $\mathbf{Coll_{Dom}}$ is interpreted by the negative translation, we introduce yet an other form of axiom scheme of collection.

Definition 3.2.2.4. A variant of the axiom scheme of collection with domain ($\mathbf{Coll'_{Dom}}$) is defined as the closure of the formulas

$$\forall D \exists Z(\forall x \in D)\forall Y(x \in D \Rightarrow \phi(x, Y) \Rightarrow \exists y \phi(x, Z[y]))$$

Lemma 3.2.2.2. *The axiom scheme* $\mathbf{Coll_{Dom}}$ *and the axiom scheme* $\mathbf{Coll'_{Dom}}$ *are equivalent in* $\mathbf{HA2^-}$*.*

[3]My intuition here is that this formulation of the axiom of collection lacks a "tiny bit" of classical logic. I think that this absence prevents it from implying its negative translation.

Proof. We prove $\mathbf{Coll}'_{\mathbf{Dom}}$ for a formula $\phi(x, Y)$ using $\mathbf{Coll}_{\mathbf{Dom}}$. Let D be a domain (given by $\mathbf{Coll}'_{\mathbf{Dom}}$). We apply $\mathbf{Coll}_{\mathbf{Dom}}$ with the formula $\phi(x, Y)$ and the domain $D'(x) \triangleq D(x) \wedge \exists Y \phi(x, Y)$. Then, $(\forall x \in D')\exists Y \phi(x, Y)$ is provable and therefore $\mathbf{Coll}_{\mathbf{Dom}}$ implies $\exists Z(\forall x \in D')\exists y \phi(x, Z[y]))$. The set Z_0 obtained by eliminating this existential is the witness allowing us to prove $\mathbf{Coll}'_{\mathbf{Dom}}$ for the formula ϕ and the domain D. □

Notation 3.2.2.1. If $\mathcal{A}$ is a set of formulas, $\mathbf{G}(\mathcal{A})$ denotes the set of formulas obtained by applying $\_ \mapsto \mathbf{G}(\_)$ to all the formulas in $\mathcal{A}$.

Lemma 3.2.2.3. $\mathbf{HA2}^-$ *proves that the axiom scheme* $\mathbf{Coll}'_{\mathbf{Dom}}$ *implies its negative translation* $\mathbf{G}(\mathbf{Coll}'_{\mathbf{Dom}})$.

Proof. The scheme $\mathbf{G}(\mathbf{Coll}'_{\mathbf{Dom}})$ contains the formulas of the shape

$$\forall D \exists^{\mathrm{c}} Z \forall x (\neg\neg x \in D \Rightarrow \forall Y (\neg\neg x \in D \Rightarrow \mathbf{G}(\phi)(x, Y, \overrightarrow{X}) \Rightarrow \exists^{\mathrm{c}} y \mathbf{G}(\phi(x, Z[y], \overrightarrow{X}))))$$

But because the second-order encoded existential $\exists$ implies the "classical" existential $\exists^{\mathrm{c}}$, these formulas are consequences of the scheme $\mathbf{Coll}'_{\mathbf{Dom}}$. □

Theorem 3.2.2.1. *The axiom scheme* $\mathbf{Coll}_{\mathbf{Dom}}$ *implies its negative translation* $\mathbf{G}(\mathbf{Coll}_{\mathbf{Dom}})$ *in* $\mathbf{HA2}^-$.

Proof. The proof is a chain of implication in $\mathbf{HA2}^-$:

$$\mathbf{Coll}_{\mathbf{Dom}} \Rightarrow \mathbf{Coll}'_{\mathbf{Dom}} \Rightarrow \mathbf{G}(\mathbf{Coll}'_{\mathbf{Dom}}) \Rightarrow \mathbf{G}(\mathbf{Coll}_{\mathbf{Dom}}).$$

□

Interpretation of the axiom scheme of choice

In the previous work, we see that we never used the difference[4] between the axiom scheme of collection and the axiom scheme of choice. Consequently, this result scales to the axiom scheme of choice.

Definition 3.2.2.5. The axiom scheme of choice with domain ($\mathbf{AC}_{\mathbf{Dom}}$) is defined as the closure of the formulas

$$\forall D((\forall x \in D)\exists Y \phi(x, Y) \Rightarrow \exists Z(\forall x \in D)\phi(x, Z[x])).$$

Theorem 3.2.2.2. *The axiom scheme* $\mathbf{AC}_{\mathbf{Dom}}$ *implies its negative translation* $\mathbf{G}(\mathbf{AC}_{\mathbf{Dom}})$ *in* $\mathbf{HA2}^-$.

Related works and comments about this interpretation

The previous interpretations are deeply inspired by the negative translation of classical set theory into intuitionistic set theory done by Friedman [17]. The idea to reformulate the axiom scheme of collection as a principle using a domain came from the study of his work. All the remaining proofs are a direct adaptation of his proofs in the set-theoretic framework within second-order logic.

In higher type arithmetic ($\mathbf{HA}^\omega$), the negative translation of the axiom of choice is stronger than the axiom of choice [7]. In fact, in higher type arithmetic, $\mathbf{G}(\mathbf{AC}_\iota)$ implies the axiom scheme of comprehension, enabling the use of full impredicativity in the theory $\mathbf{HA}^\omega + \mathbf{G}(\mathbf{AC}_\iota)$.

[4] Namely, the extra first-order existential quantifier that appears in the axiom scheme of collection.

On the other hand, $\mathbf{HA}^\omega + \mathbf{AC}_\iota$ is conservative over $\mathbf{HA}^\omega$. As a consequence, $\mathbf{AC}_\iota$ does not imply $\mathbf{G}(\mathbf{AC}_\iota)$ in $\mathbf{HA}^\omega$.

Nevertheless, the proof of Friedman that the axiom of collection is interpreted by negative translations scales also to the axiom of choice in second-order logic ; showing that both of these principles are self-interpreted by negative translations. The study of choice principles in $\mathbf{HA2}^-$ is therefore very different than their study inside $\mathbf{HA}^\omega$.

3.3 Intuitionistic realizability

In this part, we will present a computational interpretation of $\mathbf{HA2}^-$ using the tools of intuitionistic realizability, originally developed by Kleene to interpret Heyting arithmetic [25]. An intuitionistic realizability model of **HA2** is already presented in the work of McCarty [34] where first-order quantifiers are interpreted as relativized quantifiers (over the domain $\mathbb{N}$ of natural numbers) and second-order quantifiers are interpreted uniformly[5]. The model presented here will interpret both first-order quantifiers and second-order quantifiers in an uniform fashion. Because of this choice, we will not be able to interpret the axiom of induction, but it is not a restriction as this previous principle can be validated in $\mathbf{HA2}^-$ by the use of relativization. This realizability model will allow us to distinguish between the principles $\mathbf{Coll}_{\mathbf{Dom}}$ and $\mathbf{AC}_{\mathbf{Dom}}$: it will satisfy the former and not the latter. Therefore, we will obtain the result that in $\mathbf{HA2}^-$ the principle $\mathbf{Coll}_{\mathbf{Dom}}$ is strictly weaker than the principle $\mathbf{AC}_{\mathbf{Dom}}$. Finally, as we did for Krivine realizability, we will present here an external version of intuitionistic realizability. This choice is motivated because the main goal of this model is to distinguish between the two aforementioned principles, and working internally[6] is not needed to achieve this goal.

3.3.1 Preliminaries

Digression: the choice of a partial combinatory algebra

Kleene realizability is based on a chosen model of computations, called a partial combinatory algebra [50], and used to interpret the formulas as sets of programs (taken from this specific choice of model of computations). Originally, Kleene formulated his computational interpretation of arithmetic using codes of partial recursive functions (called Kleene first algebra). However, the study of their properties is cumbersome, although it was extensively studied in the literature [43]. In this thesis, we will choose as model of computations[7] the set Λ of closed terms of pure λ-calculus.

[5] In realizability, there is two ways of interpreting quantifiers.

1. Universal quantifiers can be interpreted as dependent products. This was done by Kleene in his interpretation of Heyting arithmetic and it is crucial to interpret the axiom scheme of induction.
2. Universal quantifiers can be interpreted uniformly, as intersection types. This is done in Krivine realizability and that is what we will do in our intuitionistic realizability model.

These differences are pointed out in the work of Miquel about implicative algebras [37]. This explanation is strongly inspired by one of his presentation about intuitionistic realizability done in a seminar at Montevideo in 2021.

[6] Achieving an internal interpretation is necessary to obtained more refined result about the theory at stake [49, 48].

[7] In fact, we will not formally use a partial combinatory algebra as we will not considered the term up to β-equivalence. Formulas will be interpreted as sets of (closed) λ-terms that are closed by anti-evaluation.

Pure λ-calculus

We will use untyped λ-calculus to interpret the formulas of $\mathbf{HA2}^-$. We recall very briefly its syntax and reduction rules, and refer to the literature for more details [3]. It is generated from the following grammar:

$$M, N \quad ::= \quad \xi \mid \lambda\xi.t \mid MN$$

This calculus will be enough to interpret all the connectives of second-order logic, notably because we will treat all the quantifiers uniformly (as intersection types, or subtyping). As usual, this calculus is considered up to α-equivalence. We consider the following reduction rule on terms

$$(\lambda\xi.M)\, N \quad \succ \quad M[\xi ::= N] \ .$$

from which we generate reduction and congruence (or β-equivalence)

$$M \quad \rightsquigarrow \quad N \qquad \text{and} \qquad M \quad \cong \quad N$$

as respectively the least reflexive, transitive and closed by congruence relation containing $\succ$ and the least closed by congruence equivalence relation containing $\succ$.

Notation 3.3.1.1. In this section, we will write Λ for the set of all closed terms of pure λ-calculus.

The set of truth values

In intuitionistic realizability, a closed formula ϕ will be interpreted as a set $|\phi|$ of closed λ-terms. The set $|\phi|$ is seen as the evidences that the formula ϕ is true. This is the reason why the set $\mathcal{P}(\Lambda)$ is sometimes called the set of truth values (while the set $|\phi|$ is called the truth value of ϕ). If λ-terms are considered up to β-equivalence, this set can be equipped with a structure of implicative algebra [37]. The fundamental operation to turn it into an implicative structure is Kleene implication defined as

$$A \to B \triangleq \{M \mid \forall N \in A, MN \in B\}.$$

3.3.2 Interpreting $\lambda\mathbf{HA2}^-$

Interpretation of formulas

As in the context of classical realizability (in the Section 2.3.2), we will interpret formulas with parameters, that we define as generated from the language of $\boldsymbol{\lambda}\mathbf{HA2}^-$ extended with constant symbols $\dot{b}$ for every pure binary tree $b \in \mathbf{B}$ and predicate symbols $\dot{P}$ for every function $P : \mathbf{B} \to \mathcal{P}(\Lambda)$. The interpretation $t \mapsto [\![t]\!]$ of first-order terms is done as in the standard model of **PA2** (and as in classical realizability).

Definition 3.3.2.1. The interpretation $|\phi|$ of a closed formula with parameters is defined as a set of closed λ-terms. This definition is done by induction over the syntax as follow:

$$\begin{array}{rcl}
|t \in \dot{P}| & \triangleq & \{M \mid \exists N \in P([\![t]\!]), M \rightsquigarrow N\} \\
|\mathrm{null}(t)| & \triangleq & \begin{cases} \Lambda & \text{if } [\![t]\!] = 0 \\ \emptyset & \text{otherwise} \end{cases} \\
|\phi \Rightarrow \psi| & \triangleq & |\phi| \to |\psi| \qquad (= \{M \mid \forall N \in |\phi|, MN \in |\psi|\}) \\
|\forall x \phi| & \triangleq & \bigcap_{b \in \mathbf{B}} |\phi[x := \dot{b}]| \\
|\forall X \phi| & \triangleq & \bigcap_{P : \mathbf{B} \to \mathcal{P}(\Lambda)} |\phi[X := \dot{P}]|
\end{array}$$

A formula ϕ is realized by a λ-term M if $M \in |\phi|$, in this case we will write $M \Vdash \phi$.

Notation 3.3.2.1. If $P : \mathbf{B} \to \mathcal{P}(\Lambda)$, we write $\hat{P} = \{b \in \mathbf{B} \mid P(b) \neq \emptyset\}$.

Lemma 3.3.2.1 (Closure by anti-evaluation)**.** *Let ϕ be a closed formula with parameter. If $M \rightsquigarrow N$ and $N \in |\phi|$, then $M \in |\phi|$.*

This crucial lemma will allow us to prove the soundness theorem, in a very similar fashion as in classical realizability (see Theorem 2.3.3.1).

Soundness

Intuitionistic realizability succeeds in interpreting the rules of second-order classical logic and the axioms of $\mathbf{PA2^-}$. This claim is formalized in the soundness theorem that we will now state. We adapt the notion of valuations to the context of intuitionistic realizability.

Definition 3.3.2.2. A valuation ρ is a function mapping

- every first-order variable x to a binary tree $\rho(x) \in \mathbf{B}$
- every second-order variable X to a function $\rho(X) : \mathbf{B} \to \mathcal{P}(\Lambda)$.

Given a formula ϕ and a valuation ρ, we write $\phi[\rho]$ for the closed formula with parameters obtained by replacing in the formula ϕ:

- all occurrences of a first-order variable x by the constant symbol associated to the binary tree $\rho(x)$
- all occurrences of a second-order variable X by the predicate symbol associated to the function $\rho(X)$.

Theorem 3.3.2.1 (Soundness)**.** *If a typing judgement $\xi_1 : \phi_1, ..., \xi_k : \phi_k \vdash_I M : \phi$ is derivable in the system $\boldsymbol{\lambda}\mathbf{HA2^-}$, then for all valuations ρ and for all terms $N_1, ..., N_k$ such that $N_1 \Vdash \phi_1[\rho], ..., N_k \Vdash \phi_k[\rho]$, we have: $M[\xi_1 := N_1, ..., \xi_k := N_k] \Vdash \phi[\rho]$.*

As a consequence, intuitionistic realizability provides a model for $\mathbf{HA2^-}$. Moreover, this model is consistent because the formula $\bot$ is not realized. We will use this model to show that the scheme $\mathbf{Coll_{Dom}}$ is weaker than the scheme $\mathbf{AC_{Dom}}$.

Extending the syntax: simplify the interpretation of the existential quantifiers

The goal of this section is to motivate an extension of the syntax by two quantifiers interpreted in the model by joins (or union types) and used to replace the second-order encodings of the existential quantifiers. This trick will be useful while studying the computational contents of formulas using existential quantifiers, such as the axiom of collection with domain and the axiom of choice with domain.

Example 3.3.2.1. Let $\phi(x)$ be a formula with one free variable x. We will study the truth value of $\exists x \phi$. It is computed as follow

$$\begin{array}{rcl} |\exists x \phi(x)| & \triangleq & |\forall Z(\forall x(\phi(x) \Rightarrow 0 \in Z) \Rightarrow 0 \in Z)| \\ & = & \bigcap\limits_{P:\mathcal{P}(\Lambda)} (\bigcap\limits_{b:\mathbf{B}} (|\phi[x := \dot{b}]| \to P) \to P) \\ & = & \bigcap\limits_{P:\mathcal{P}(\Lambda)} \{M \mid \forall N \in \bigcap\limits_{b:\mathbf{B}} (|\phi[x := \dot{b}]| \to P), MN \in P\}. \end{array}$$

It is equivalent to the truth value $\bigcup\limits_{b \in \mathbf{B}} |\phi[x := \dot{b}]|$. The meaning of this is twofold:

1. $|\exists x\phi(x)| \to \bigcup_{b\in\mathbf{B}} |\phi[x := \dot{b}]|$ is realized by $\lambda\xi.\xi\lambda\eta.\eta$.

2. $\bigcup_{b\in\mathbf{B}} |\phi[x := \dot{b}]| \to |\exists x\phi(x)|$ is realized by $\lambda\xi.\lambda\eta.\eta\xi$.

It advocates that, in the model of intuitionistic realizability, the existential connectives can be successfully replaced by joins.

Definition 3.3.2.3. We extend the syntax of $\boldsymbol{\lambda}\mathbf{HA2}^{-}$ with two new quantifiers $\bigsqcup_x \phi(x)$ and $\bigsqcup_X \phi(X)$ that are interpreted as follow:

$$\begin{array}{rcl} |\bigsqcup_x \phi(x)| & \triangleq & \bigcup_{b\in\mathbf{B}} |\phi[x := \dot{b}]| \\ |\bigsqcup_X \phi(X)| & \triangleq & \bigcup_{P:\mathbf{B}\to\mathcal{P}(\Lambda)} |\phi[X := \dot{P}]|. \end{array}$$

Proposition 3.3.2.1. *For all formulas $\phi(x)$ with one free variable x and $\psi(X)$ with one free variable X, the following formulas are realized:*

$$\begin{array}{ll} \exists x\phi(x) \Rightarrow \bigsqcup_x \phi(x) & \bigsqcup_x \phi(x) \Rightarrow \exists x\phi(x) \\ \exists X\psi(X) \Rightarrow \bigsqcup_X \psi(X) & \bigsqcup_X \psi(X) \Rightarrow \exists X\psi(X). \end{array}$$

3.3.3 The axiom scheme of collection with domain is realized

We show that the axiom scheme $\mathbf{Coll_{Dom}}$ is realized. Recall that it is defined as the closure of the formulas

$$\forall D((\forall x \in D)\exists Y \phi(x, Y) \Rightarrow \exists Z(\forall x \in D)\exists y \phi(x, Z[y])).$$

As discussed before, we will rather be interested in a formula of the following shape

$$\forall D((\forall x \in D)\bigsqcup_Y \phi(x, Y) \Rightarrow \bigsqcup_Z (\forall x \in D)\bigsqcup_y \phi(x, Z[y])).$$

The trick to realize this formula is very similar to the one used in classical realizability to realize the axiom scheme of collection. Fixing a function $P_D : \mathbf{B} \to \mathcal{P}(\Lambda)$, we show that

$$|(\forall x \in \dot{P_D})\bigsqcup_Y \phi(x, Y)| \subseteq |\bigsqcup_Z (\forall x \in \dot{P_D})\bigsqcup_y \phi(x, Z[y])|.$$

It relies on a surjection $b \mapsto M_b$ from the set $\mathbf{B}$ of binary trees into the set Λ of closed λ-terms. Note that such a surjection exists because both of these sets are countable. We will also need an inverse $M \mapsto \ulcorner M \urcorner$ of this surjection. Using the axiom of choice in the meta-theory, we define a family of functions $(F_{M,b})_{\{M\in\Lambda, b\in\mathbf{B}\}} \in (\mathcal{P}(\Lambda)^{\mathbf{B}})^{\Lambda\times\mathbf{B}}$ indexed by a closed λ-term and a binary tree as follow:

1. If $M \Vdash \bigsqcup_Y \phi(\dot{b}, Y)$, $F_{M,b}$ is chosen to be a function $P : \mathbf{B} \to \mathcal{P}(\Lambda)$ such that $M \Vdash \phi(\dot{b}, \dot{P})$.

2. Otherwise, $F_{M,b}$ is the function $b \mapsto \emptyset$.

We can now define the function $P_Z : \mathbf{B} \to \mathcal{P}(\Lambda)$ that will be used to interpret the variable Z as

$$P_Z(b) \triangleq \begin{cases} F_{M_{b_1},b_2}(b_3) & \text{if } b = \langle\langle b_1, b_2\rangle, b_3\rangle \\ \emptyset & \text{otherwise} \end{cases}$$

Lemma 3.3.3.1. *For all functions $P_D : \mathbf{B} \to \mathcal{P}(\Lambda)$:*

$$|(\forall x \in \dot{P_D})\bigsqcup_Y \phi(x, Y)| \subseteq |(\forall x \in \dot{P_D})\bigsqcup_y \phi(x, \dot{P_Z}[y])|.$$

Proof. Assume $M \Vdash (\forall x \in \dot{P_D})\bigsqcup_Y \phi(x, Y)$ and let $b \in \mathbf{B}$. If $N \Vdash \dot{b} \in \dot{P_D}$, then $MN \Vdash \bigsqcup_Y \phi(\dot{b}, Y)$ and by definition of P_Z, we have $MN \Vdash \phi(\dot{b}, \dot{P_Z}[\langle \ulcorner MN \urcorner, \dot{b} \rangle])$. □

Theorem 3.3.3.1. *All the instances of the axiom scheme* $\mathbf{Coll_{Dom}}$ *are realized.*

Proof. They are all realized by $\lambda\xi.\xi$. □

3.3.4 The axiom scheme of choice with domain is not realized

The goal of this section is to show that the scheme $\mathbf{AC_{Dom}}$ is not realized. Recall that it is defined as the closure of the formulas

$$\forall D((\forall x \in D)\exists Y \phi(x, Y) \Rightarrow \exists Z(\forall x \in D)\phi(x, Z[x])).$$

As discussed before, we will rather be interested in a formula of the following shape

$$\forall D((\forall x \in D)\bigsqcup_Y \phi(x, Y) \Rightarrow \bigsqcup_Z (\forall x \in D)\phi(x, Z[x])).$$

To ease the process, we will first work in an extension of $\mathbf{HA2^-}$ with a third-order relation symbol. Then we will discuss how to adapt this work to the language of $\mathbf{HA2^-}$.

$\mathbf{AC_{Dom}}$ is not realized in an extension of $\mathbf{HA2^-}$

We fix two functions $f, g : \mathbf{B} \to \Lambda$ satisfying

$$\begin{array}{ll} f(0) = \lambda\xi\lambda\eta.\xi & g(0) = \lambda\xi\lambda\eta.\eta \\ f(1) = \lambda\xi\lambda\eta.\xi & g(1) = \lambda\xi\lambda\eta.\xi \\ f(2) = \lambda\xi\lambda\eta.\eta & g(2) = \lambda\xi\lambda\eta.\eta \\ f(b) = g(b) = \lambda\xi\lambda\eta.\xi & \text{(in all the other cases)} \end{array}$$

The point of these functions is that:

$$\begin{array}{c} f(0) \ncong g(0) \\ f(1) \cong g(1) \cong f(0) \\ f(2) \cong g(2) \cong g(0). \end{array}$$

We will show that a realizer of $\mathbf{AC_{Dom}}$ will imply the existence of a λ-term that reduces to $f(0)$ and to $g(0)$, thus breaking Church-Rosser theorem [3] and leading to a contradiction.

Now, we define $P_D : \mathbb{B} \to \mathcal{P}(\Lambda)$ as $P_D(b) \triangleq \{f(b), g(b)\}$.

Consider a third-order relation $\mathcal{R}(x, X)$ with the following interpretation

$$|\mathcal{R}(\dot{b}, \dot{P})| \triangleq \begin{cases} \{M \mid M \rightsquigarrow f(b)\} & \text{if } \hat{P} = \{\ulcorner f(b) \urcorner\} \\ \{M \mid M \rightsquigarrow g(b)\} & \text{if } \hat{P} = \{\ulcorner g(b) \urcorner\} \\ \emptyset & \text{otherwise} \end{cases}$$

where $\hat{P} = \{b \in \mathbf{B} \mid P(b) \neq \emptyset\}$.

Lemma 3.3.4.1. *The formula* $(\forall x \in \dot{P}_D)\bigsqcup_Y \mathcal{R}(x, Y)$ *is realized by* $\lambda\xi.\xi$.

Lemma 3.3.4.2. *The formula* $\bigsqcup_Z(\forall x \in \dot{P}_D)\mathcal{R}(x, Z[x])$ *is not realized.*

Proof. Assume $M \Vdash (\forall x \in \dot{P}_D)\mathcal{R}(x, \dot{P}_Z[x])$ for some function $P_Z : \mathbf{B} \to \mathcal{P}(\Lambda)$.

1. For $x = 0$, $M\lambda\xi\lambda\eta.\xi \cong M\lambda\xi\lambda\eta.\eta$ because both of these terms are in $|\mathcal{R}(\dot{0}, \dot{P}_Z[0])|$.
2. For $x = 1$, $M\lambda\xi\lambda\eta.\xi \cong \lambda\xi\lambda\eta.\xi$ because $M\lambda\xi\lambda\eta.\xi \in \mathcal{R}(1, \dot{P}_Z[1])$.
3. Similarly, for $x = 2$, $M\lambda\xi\lambda\eta.\eta \cong \lambda\xi\lambda\eta.\eta$.

It follows that $\lambda\xi\lambda\eta.\xi \cong \lambda\xi\lambda\eta.\eta$ which is a contradiction. □

Theorem 3.3.4.1. *The formula* $\forall D((\forall x \in D)\bigsqcup_Y \mathcal{R}(x, Y) \Rightarrow \bigsqcup_Z(\forall x \in D)\mathcal{R}(x, Z[x]))$ *is not realized.*

$\mathbf{AC_{Dom}}$ is not realized in $\mathbf{HA2^-}$

We need to adapt this result to the syntax of $\mathbf{HA2^-}$. In other words, we need to replace the third-order predicate $\mathcal{R}$ by a third-order predicate definable in second-order logic. We define the formula

$$\phi(x, Y) \quad \triangleq \quad (x \in \dot{F}_c \wedge Y = \dot{F}_l[x]) \vee (x \in \dot{G}_c \wedge Y = \dot{G}_l[x])$$

where

$$\begin{array}{rcl} F_c(b) & \triangleq & \{f(b)\} \\ G_c(b) & \triangleq & \{g(b)\} \\ F_l(b) & \triangleq & \begin{cases} \Lambda & \text{if } b = \langle b_1, \ulcorner f(b_1) \urcorner \rangle \\ \emptyset & \text{otherwise} \end{cases} \\ G_l(b) & \triangleq & \begin{cases} \Lambda & \text{if } b = \langle b_1, \ulcorner g(b_1) \urcorner \rangle \\ \emptyset & \text{otherwise} \end{cases} \end{array}$$

The intuition is that F_c will convey a computational meaning, forcing a realizer of $F_c(b)$ to reduce to $f(b)$ and F_l convey a logical meaning, forcing that $\dot{P} = \dot{F}_l[b]$ is realized if and only if $\hat{P} = \{\ulcorner f(b) \urcorner\}$.

Lemma 3.3.4.3. *The formula* $\bigsqcup_Z(\forall x \in \dot{P}_D)\phi(x, Z[x]) \Rightarrow \bigsqcup_Z(\forall x \in \dot{P}_D)\mathcal{R}(x, Z[x])$ *is realized.*

Proof. It is realized by $\lambda\xi.\lambda\eta.\mathrm{destruct}^{\vee}_{\chi}(\xi\eta)\,\mathrm{in}\,\pi_1\chi \mid \pi_1\chi$. □

Corollary 3.3.4.1. *The formula* $\bigsqcup_Z(\forall x \in \dot{P}_D)\phi(x, Z[x])$ *is not realized.*

Lemma 3.3.4.4. *The formula* $(\forall x \in \dot{P}_D)\bigsqcup_Y \phi(x, Y)$ *is realized.*

Proof. It is realized by $\lambda\xi.\xi(\mathrm{inl}[\xi, [\lambda\eta.\eta, \lambda\eta.\eta]])(\mathrm{inr}[\xi, [\lambda\eta.\eta, \lambda\eta.\eta]])$. □

Theorem 3.3.4.2. *The axiom scheme* $\mathbf{AC_{Dom}}$ *is not realized.*

Proof. If all the instances of $\mathbf{AC_{Dom}}$ were realized, the formula $\bigsqcup_Z(\forall x \in \dot{P}_D)\phi(x, Z[x])$ would be realized. □

The final result

Finally, we are able to conclude that in $\mathbf{HA2^-}$ the axiom scheme $\mathbf{AC_{Dom}}$ is strictly stronger than the axiom scheme $\mathbf{Coll_{Dom}}$.

Theorem 3.3.4.3. *The theory* $\mathbf{HA2^- + Coll_{Dom}}$ *does not imply* $\mathbf{AC_{Dom}}$.

However, we cannot lift this result to $\mathbf{PA2^-}$. Our original idea was to show that $\mathbf{G(Coll_{Dom})}$ is weaker than $\mathbf{G(AC_{Dom})}$ in $\mathbf{HA2^-}$. Using the soundness of the negative translation, it would imply that $\mathbf{Coll}$ is weaker than $\mathbf{AC}_\iota$ in $\mathbf{PA2^-}$. We were not able to show this result.

Chapter 4

Well-preorders in second-order logic

This chapter acts as a preliminary to the study of the ramified analytic hierarchy (in Chapter 5). It is of crucial importance to encode various structures in the language of second-order logic. It can be cumbersome but it is done in Section 4.1. Notably, the notion of slices, introduced in Definition 1.1.1.4 and recalled in this chapter, will be fundamental for the rest of this work.

However, adapting constructibility from set theory into arithmetical frameworks requires important adjustments. The most important difference is that it is not possible to work with ordinals in second-order arithmetic. The lack of this tool leads to various complications. In this work, to overcome some of these problems, we replace ordinals by well-preorders and we develop a theory of well-preorders inside $\mathbf{PA2}^- + \mathbf{Coll}$ (Section 4.2). In particular, it means that we will work without the axiom of induction. As far as I know, all this chapter is totally new: well-preorders were never studied in the framework of second-order logic. The reason why they were not studied is straightforward! In presence of the axiom of induction (and in a classical logic), this notion is of no interest: because the individuals can be shown to be well-ordered, a well-order can be extracted from every well-preorder.

While this research seems new, it is merely a very direct adaptation of all the results already known about well-orders. However, well-preorders are better behaved than well-orders in $\mathbf{PA2}^-$. The best example of this statement is the following: the supremum of a family of well-preorders (represented as slices of a set) can always be constructed (Theorem 4.2.2.1). Up to my knowledge, there is no machinery available in $\mathbf{PA2}^-$ to define the supremum of a family of well-orders. The existence of supremum for well-preorders, coupled with the axiom scheme of collection, will lead to the construction of well-preorders defined as the result of iterating a compatible functional relation along ω (Theorem 4.2.4.1). This tool will be decisive in the study of the ramified analytic hierarchy and, precisely, to show the principle of reflection (Theorem 5.4.1.1).

In set theory, well-preorders were studied and generally, in the literature, they are called prewellorderings [41, 46, 40].

4.1 Encoding in second-order logic

4.1.1 Relations, functions, families of sets and functional relations

We define the notion of relations in **SOL**. This fundamental concept will be used to represent different kind of objects in the syntax of **SOL**, as for instance functions and families of sets. However, a function from reals to reals cannot be represented by a relation. This kind of function, called functional relations will be represented by formulas with two distinguished second-order variables.

Relations

A binary relation $\mathcal{R}$ is a set consisting of pairs of elements. The syntax of **SOL** naturally allows us to represent these objects, a set $\mathcal{R}$ will be a binary relation if it only contains elements in the image of $\langle \_, \_ \rangle$. Therefore, a set $\mathcal{R}$ is a relation if it satisfies the formula:

$$\mathrm{Rel}(\mathcal{R}) \triangleq (\forall x \in \mathcal{R})\, x = \langle \mathrm{fst}(x), \mathrm{snd}(x) \rangle.$$

We write $x\mathcal{R}y$ if the individual $\langle x, y \rangle$ is in the set $\mathcal{R}$, i.e.:

$$x\mathcal{R}y \triangleq \langle x, y \rangle \in \mathcal{R}.$$

The definition of a relation makes it possible to prove a principle of extensionality for binary relations. If $\mathcal{R}_1$ and $\mathcal{R}_2$ are two equal binary relations, then they are also equal as set:

$$\mathbf{ACA_0^-} + \mathrm{Rel}(\mathcal{R}_1) + \mathrm{Rel}(\mathcal{R}_2) \ \vdash \ \forall x \forall y (x\mathcal{R}_1 y \Leftrightarrow x\mathcal{R}_2 y) \Leftrightarrow \mathcal{R}_1 = \mathcal{R}_2$$

We write:

1. $E_{\mathcal{R}}$ the field of the relation $\mathcal{R}$, i.e. the set

$$E_{\mathcal{R}} \triangleq \{x \mid \exists y (x\mathcal{R}y \vee y\mathcal{R}x)\}.$$

2. $\mathrm{Dom}_{\mathcal{R}}$ the domain of the relation $\mathcal{R}$, i.e. the set

$$\mathrm{Dom}_{\mathcal{R}} \triangleq \{x \mid \exists y \ x\mathcal{R}y\}.$$

3. $\mathrm{Im}_{\mathcal{R}}$ the image of the relation $\mathcal{R}$, i.e. the set

$$\mathrm{Im}_{\mathcal{R}} \triangleq \{y \mid \exists x \ x\mathcal{R}y\}.$$

4. $\mathcal{R}|X$ the restriction of the relation $\mathcal{R}$ to the set X:

$$\mathcal{R}|X \triangleq \{\langle x, y \rangle \mid x \in X \wedge y \in X \wedge x\mathcal{R}y\}.$$

5. $\mathcal{R}|_X$ the relation obtained by restricting the domain of the relation $\mathcal{R}$, i.e.:

$$\mathcal{R}|_X \triangleq \{\langle x, y \rangle \mid x \in X \wedge x\mathcal{R}y\}.$$

6. $\mathcal{R}_2 \circ \mathcal{R}_1$ the composition of the relation $\mathcal{R}_1$ followed by $\mathcal{R}_2$, i.e.

$$(\mathcal{R}_2 \circ \mathcal{R}_1) \triangleq \{\langle x, y \rangle \mid \exists z \, (x\mathcal{R}_1 z \wedge z\mathcal{R}_2 y)\}.$$

7. $\mathcal{R}^{-1}$ the inverse of the relation $\mathcal{R}$, i.e.

$$\mathcal{R}^{-1} \triangleq \{\langle x, y \rangle \mid \langle y, x \rangle \in \mathcal{R}\}.$$

Functions

A natural example of binary relations is a function, it is a relation such that every element of its domain is paired with exactly one individual. We say that the set F is a function if it satisfies the formula:

$$\mathrm{Func}(F) \triangleq \mathrm{Rel}(F) \wedge \forall x \forall y \forall y' (xFy \Rightarrow xFy' \Rightarrow y = y').$$

For every $x \in \mathrm{Dom}_F$, we write $F(x)$ the unique individual y such that xFy. Formally, if $\phi(x)$ is a formula with parameters:

$$\phi(F(x)) \triangleq \exists y (xFy \wedge \phi[x := y]).$$

where the variable y is not free in ϕ. The shortcut $\phi(F(x))$ denotes then a formula of **SOL**. We use capital letters $(F, G, H...)$ to denote functions. The composition of two functions is again a function:

$$\mathbf{ACA_0^-} + \mathrm{Func}(F) + \mathrm{Func}(G) \vdash \mathrm{Func}(G \circ F)$$

Families of sets

A relation $\mathcal{U}$ naturally encodes a family of sets indexed by the individuals. In other words, it naturally encodes a function from the individuals to the reals using the concept of slices introduced in Definition 1.1.1.4. The individual x is associated to the set

$$\mathcal{U}[x] \triangleq \{y \mid x\mathcal{U}y\}$$

of individuals related to x in the relation $\mathcal{U}$. We write $\mathbf{S}(\mathcal{U})$ for the (definable) class of sets containing all the slices of $\mathcal{U}$:

$$\mathbf{S}(\mathcal{U}) \triangleq \{X \mid \exists x \ X = \mathcal{U}[x]\}.$$

Note that in the presence of the axiom of induction, this class is intuitively countable. Two different sets can encode the same class. For instance, the sets $\{\langle 0, 1\rangle\}$ and $\{\langle 1, 1\rangle\}$ both encode the class $\{\{1\}\}$.

Because we will abundantly use this encoding, we introduce a specific nomenclature for the use of relations in this context: we will speak of families of sets, and we will use the notation $\mathrm{Fam}(\mathcal{U})$ (defined to be a macro for $\mathrm{Rel}(\mathcal{U})$) to denote a set that encodes a family. Calligraphic capital letters $(\mathcal{U}, \mathcal{V}...)$ will range over families of sets.

Given a set X, we will consider the union $\bigcup_{x \in X} \mathcal{U}[x]$ of all the sets in the class $\mathbf{S}(\mathcal{U}|_X)$, defined as

$$\bigcup_{x \in X} \mathcal{U}[x] \triangleq \{y \mid (\exists x \in X)(y \in \mathcal{U}[x])\}.$$

Functional relations

Functions from sets into sets form an other kind of relations. However, such a function is an object oh third-order that cannot be encoded in a set. This is why we use formulas to represent functional relations. We say that the formula $\mathcal{H}(X_1, ..., X_k, Y)$ with $k + 1$ free second-order variables is a functional relation of arity k if:

$$\mathrm{Func}_{\mathcal{H}} \triangleq \forall X_1 ... \forall X_k \ [\exists Y \ \mathcal{H}(X_1, ...X_k, Y) \Rightarrow \exists! Y \mathcal{H}(X_1, ..., X_k, Y)].$$

In the following, we will only study total functional relations. Concretely, a functional relation is total if it satisfies

$$\text{FuncTot}_{\mathcal{H}} \triangleq \forall X_1 ... \forall X_k \exists! Y \ \mathcal{H}(X_1, ..., X_k, Y)$$

However, it is not a restriction! It is always possible to extend a functional relation into a total functional relation. Actually, if $\mathcal{H}(\overrightarrow{X}, Y)$ is a functional relation, it is enough to consider:

$$\mathcal{H}'(\overrightarrow{X}, Y) \triangleq \mathcal{H}(\overrightarrow{X}, Y) \vee (\forall Z \ \neg \mathcal{H}(\overrightarrow{X}, Z) \wedge Y = \emptyset).$$

4.1.2 Preorders

We study the concept of preorders in the framework of second-order logic.

Definitions and examples

We define the following propertied that can be satisfied by a binary relation $\mathcal{R}$.

- $\mathcal{R}$ is reflexive if $\text{Refl}(\mathcal{R}) \triangleq (\forall x \in E_{\mathcal{R}}) \, x\mathcal{R}x$.
- $\mathcal{R}$ is transitive if $\text{Trans}(\mathcal{R}) \triangleq (\forall x \in E_{\mathcal{R}})(\forall y \in E_{\mathcal{R}})(\forall z \in E_{\mathcal{R}})(x\mathcal{R}y \Rightarrow y\mathcal{R}z \Rightarrow x\mathcal{R}z)$.
- $\mathcal{R}$ is antisymmetric if $\text{ASym}(\mathcal{R}) \triangleq (\forall x \in E_{\mathcal{R}})(\forall y \in E_{\mathcal{R}})(x\mathcal{R}y \Rightarrow y\mathcal{R}x \Rightarrow x = y)$.
- $\mathcal{R}$ is total if $\text{Tot}(\mathcal{R}) \triangleq (\forall x \in E_{\mathcal{R}})(\forall y \in E_{\mathcal{R}})(x\mathcal{R}y \vee y\mathcal{R}x)$.
- $\mathcal{R}$ is a preorder if $\text{PreOrd}(\mathcal{R}) \triangleq \text{Rel}(\mathcal{R}) \wedge \text{Refl}(\mathcal{R}) \wedge \text{Trans}(\mathcal{R})$.
- $\mathcal{R}$ is a partial order if $\text{POrd}(\mathcal{R}) \triangleq \text{PreOrd}(\mathcal{R}) \wedge \text{ASym}(\mathcal{R})$.
- $\mathcal{R}$ is a total order if $\text{TotOrd}(\mathcal{R}) \triangleq \text{PreOrd}(\mathcal{R}) \wedge \text{Tot}(\mathcal{R})$.

Therefore, a preorder is a reflexive and transitive relation. We use greek letters $(\alpha, \beta, \gamma ...)$ to denote preorders and we write $x \alpha y$ but also $x \leq_\alpha y$ to say that x is in relation with y in the preorder α.

From a preorder α, we can generate an equivalence relation $\cong_\alpha$ on its domain E_α defined by

$$x \cong_\alpha y \triangleq x \leq_\alpha y \wedge y \leq_\alpha x.$$

If α is a partial order, then $\cong_\alpha$ collapses to the relation of equality on its domain. But, in general, it is not the case. We will also consider the strict preorder associated to a preorder α and we use the notation

$$x <_\alpha y \triangleq x \leq_\alpha y \wedge x \ncong_\alpha y.$$

Example 4.1.2.1. As a first example, we construct (using the tools developed in 1.2.2) a total order on $\mathbb{N}$ and we use it to define a total order on $\mathbb{B}$.

1. We equip the set $\mathbb{N}$ with the relation:

$$\omega \triangleq \{\langle x, y \rangle \mid x \in \mathbb{N} \wedge y \in \mathbb{N} \wedge (\exists z \in \mathbb{N}) \ z + x = y\}.$$

   The property of the addition ensures that ω is a total order on $\mathbb{N}$:

$$\mathbf{ACA_0^-} \ \vdash \ \omega \in \text{TotOrd} \wedge E_\omega = \mathbb{N}$$

2. Thanks to the bijection $\mathbf{h}$ between $\mathbb{B}$ and $\mathbb{N}$, we define a total order on $\mathbb{B}$ by:

$$\omega_b \triangleq \{\langle x, y\rangle \mid \mathbf{h}(x) \leq_\omega \mathbf{h}(y)\}.$$

It satisfies $(\forall x \in \mathbb{B})(\forall y \in \mathbb{B})(x \leq_{\omega_b} \langle x, y\rangle \wedge y \leq_{\omega_b} \langle x, y\rangle)$.

The definitions of ω and ω_b are arithmetical (with the set $\mathbb{N}$ as a parameter).

A preorder α is either empty, or has a maximal element, or is not empty and does not have a maximal element. These three cases are mutually exclusive. Formally, we consider the formulas $\mathrm{Succ}(\alpha)$ and $\mathrm{Lim}(\alpha)$:

$$\mathrm{Succ}(\alpha) \triangleq (\exists x \in E_\alpha)(\forall y \in E_\alpha)\ x \not<_\alpha y$$
$$\mathrm{Lim}(\alpha) \triangleq E_\alpha \neq \emptyset \wedge \alpha \notin \mathrm{Succ}.$$

These formulas respectively express that α is successor (meaning that α has a maximal element) and that α is limit. As said before, a preorder is either successor, or empty or limit and these three cases are mutually exclusive:

$$\mathbf{ACA_0^-} + \alpha \in \mathrm{PreOrd} \vdash (\alpha = \emptyset \vee \alpha \in \mathrm{Lim} \vee \alpha \in \mathrm{Succ})$$
$$\mathbf{ACA_0^-} + \alpha \in \mathrm{PreOrd} \vdash \neg(\alpha = \emptyset \wedge \alpha \in \mathrm{Lim}) \wedge \neg(\alpha = \emptyset \wedge \alpha \in \mathrm{Succ}) \wedge \neg(\alpha \in \mathrm{Lim} \wedge \alpha \in \mathrm{Succ})$$

For instance, the relation $\{\langle 0, 0\rangle\}$ is successor while ω is limit. However, these notions will be used only in the case of total preorders.

Definition 4.1.2.1. If α is a preorder, we write $\mathrm{M}_\alpha \triangleq \{x \mid x \in E_\alpha \wedge (\forall y \in E_\alpha)\ x \not<_\alpha y\}$ the set of its maximal elements.

If α is total, then all its maximal elements are equivalent up to $\cong_\alpha$.

Initial segments, chains of preorders

The operation of restriction is well-behaved on the class of preorders: the restriction of a preorder is again a preorder. For instance, from a preorder α and an individual i in its field, we define the preorders:

$$\alpha_{<i} \triangleq \alpha|\{x \mid x <_\alpha i\} \qquad \text{and} \qquad \alpha_{\leq i} \triangleq \alpha|\{x \mid x \leq_\alpha i\}.$$

These relations are downward closed, it means that if $x \in \alpha_{<i}$ (resp. $x \in \alpha_{\leq i}$) and $y \leq x$ then $x \in \alpha_{<i}$ (resp. $y \in \alpha_{\leq i}$). We call initial segment of α a relation satisfying this property. Formally, a preorder β is an initial segment of α if the following formula is satisfied:

$$\mathrm{InitSeg}(\beta, \alpha) \triangleq \beta \subseteq \alpha \wedge (\forall x \in E_\beta)\forall y(y \leq_\alpha x \Rightarrow y \in E_\beta).$$

If besides $\beta \neq \alpha$, we say that β is a proper initial segment of α (notation: $\beta \subseteq^{s.i.} \alpha$).

A limit order α is the union of its proper initial segments of the shape $\alpha_{<i}$ (for $i \in E_\alpha$). As a tool to formalize this property, we introduce the notion of α-chain of relations.

Definition 4.1.2.2. We say that a set $\mathcal{U}$ is an α-chain of preorders if it is an increasing family of preorders indexed by E_α:

$$\mathrm{Chain}_\alpha(\mathcal{U}) \triangleq \mathrm{Fam}(\mathcal{U}) \wedge (\forall x \in E_\alpha)(\forall y \in E_\alpha)(\mathrm{PreOrd}(\mathcal{U}[x]) \wedge x \leq_\alpha y \Rightarrow \mathrm{InitSeg}(\mathcal{U}[x],\mathcal{U}[y])).$$

If α is a total order, the union of the preorders in an α-chain is again a preorder:

$$\mathbf{ACA_0^-} + \mathrm{TotOrd}(\alpha) + \mathrm{Chain}_\alpha(\mathcal{U}) \vdash \mathrm{PreOrd}(\textstyle\bigcup_{i \in E_\alpha} \mathcal{U}[i])$$

While this concept can be ill-behaved when α is not a total order (in general, the union of an α-chain is not an order), it is however enough to show the following proposition.

Proposition 4.1.2.1. *A limit preorder α is equal to the union of its proper initial segments of the shape $\alpha_{<i}$. Formally, for a preorder α we define the α-chain $\mathcal{S}_\alpha \triangleq \{\langle x, y\rangle \mid x \in E_\alpha \wedge y \in \alpha_{<x}\}$ and we show:*

$$\mathbf{ACA_0^-} + \mathrm{PreOrd}(\alpha) + \mathrm{Lim}(\alpha) \ \vdash \ \alpha = \bigcup_{i \in E_\alpha} S_\alpha[i]$$

As a consequence, we have for instance

$$\omega = \bigcup_{n \in E_\omega} \mathcal{S}_\omega[n]$$

This example will allow us to show that ω is in fact a well-order.

Operations on preorders

We introduce the operations over preorders that are used in the rest of this thesis.

Proposition 4.1.2.2. *Let α and β be two preorders.*

1. *The Cartesian product $\alpha \otimes \beta$ is defined by:*

   $$\alpha \otimes \beta \triangleq \{\langle\langle x_1, x_2\rangle, \langle y_1, y_2\rangle\rangle \mid \\ x_1 \in E_\alpha \wedge y_1 \in E_\alpha \wedge x_2 \in E_\beta \wedge y_2 \in E_\beta \wedge (x_1 <_\alpha y_1 \vee (x_1 \cong_\alpha y_1 \wedge x_2 \leq_\beta y_2))\}.$$

   $\mathbf{ACA_0^-}$ *shows that $\alpha \otimes \beta$ is a preorder on $E_\alpha \times E_\beta$.*

2. *The disjoint union $\alpha + \beta$ is defined by:*

   $$\alpha + \beta \triangleq \{\langle\langle 0, x\rangle, \langle 0, y\rangle\rangle \mid x \leq_\alpha y\} \ \cup \ \{\langle\langle 1, x\rangle, \langle 1, y\rangle\rangle \mid x \leq_\beta y\} \\ \cup \ \{\langle\langle 0, x\rangle, \langle 1, y\rangle\rangle \mid x \in E_\alpha \wedge y \in E_\beta\}.$$

   In $\alpha+\beta$, the preorder β is put "above" α: the elements of β are greater than all the elements of α. $\mathbf{ACA_0^-}$ *shows that $\alpha + \beta$ is a preorder on $\{0\} \times E_\alpha \cup \{1\} \times E_\beta$.*

3. *The sum $\alpha \uplus \beta$ is defined by:*

   $$\alpha \uplus \beta \triangleq \alpha \cup \beta \cup \{\langle x, y\rangle \mid x \in E_\alpha \wedge y \in E_\beta\}.$$

   This operation concatenates two relations without changing their fields, it will only be applied to disjoint relations. $\mathbf{ACA_0^-}$ *shows that $\alpha \uplus \beta$ is a preorder on $E_\alpha \cup E_\beta$.*

4. *Given a preorder α, we define the relation α^ω on the set of finite sequences of elements in E_α (see Subsection 1.2.2):*

   $$\alpha^\omega \triangleq \{\langle x, y\rangle \mid x \in \mathrm{Seq}(E_\alpha) \wedge y \in \mathrm{Seq}(E_\alpha) \wedge \\ [\mathtt{length}(x) <_\omega \mathtt{length}(y) \vee (\exists n \in \mathbb{N})(\mathtt{length}(x) = \mathtt{length}(y) \wedge n <_\omega \mathtt{length}(x) \wedge \\ \mathtt{nth}(x, n) <_\alpha \mathtt{nth}(y, n) \wedge (\forall m \in E_{\omega_{<n}})\, \mathtt{nth}(x, m) \cong_\alpha \mathtt{nth}(y, m)) \vee \\ (\mathtt{length}(x) = \mathtt{length}(y) \wedge (\forall m \in E_{\omega_{<\mathtt{length}(x)}})\mathtt{nth}(x, m) \cong_\alpha \mathtt{nth}(y, m))]\}.$$

   Two sequences x and y are compared as follow:

   - *If the length of x is strictly smaller than the length of y, then $x \leq_{\alpha^\omega} y$.*

- *If x and y have the same length and if there exists an index n_0 such that $\texttt{nth}(x, n_0) \not\cong_\alpha \texttt{nth}(y, n_0)$, we consider the first such index n and, in that case, $\texttt{nth}(x, n) <_\alpha \texttt{nth}(y, n)$ implies $x \leq_{\alpha^\omega} y$.*
- *The last possibility is the case where x and y are two lists which elements are pairwise isomorphic in α. In that case, the lists x and y are isomorphic in α^ω.*

$\mathbf{ACA_0^-}$ *shows that α^ω is a preorder on* $\mathrm{Seq}(E_\alpha)$.

Morphisms of preorders

A relation F of domain included in E_α and of image included in E_β is a morphism of preorders if it is an increasing function from E_α to E_β up to $\cong_\alpha$ and $\cong_\beta$:

$$\begin{array}{rcl} \mathrm{Morph}(F, \alpha, \beta) & \triangleq & \mathrm{Dom}_F \subseteq E_\alpha \wedge (\forall x \in E_\alpha)\exists x' \, (x \cong_\alpha x' \wedge x' \in \mathrm{Dom}_F) \wedge \\ & & \forall x_1 \forall x_2 \forall y_1 \forall y_2 (x_1 \leq_\alpha x_2 \Rightarrow x_1 F y_1 \Rightarrow x_2 F y_2 \Rightarrow y_1 \leq_\beta y_2) \end{array}$$

If α and β are partial orders, then F is an increasing function between α and β. Surjections, embeddings and isomorphisms between two preorders α and β are defined as expected (up to $\cong_\alpha$ and $\cong_\beta$):

$$\begin{array}{rcl} \mathrm{Surj}(F, \alpha, \beta) & \triangleq & \mathrm{Morph}(F, \alpha, \beta) \wedge (\forall y \in E_\beta)\exists x \exists y' (y \cong_\beta y' \wedge x F y') \\ \mathrm{Emb}(F, \alpha, \beta) & \triangleq & \mathrm{Morph}(F, \alpha, \beta) \wedge \forall x_1 \forall x_2 \forall y_1 \forall y_2 \, (y_1 \leq_\beta y_2 \Rightarrow x_1 F y_1 \Rightarrow x_2 F y_2 \Rightarrow x_1 \leq_\alpha x_2) \\ \mathrm{Isom}(F, \alpha, \beta) & \triangleq & \mathrm{Emb}(F, \alpha, \beta) \wedge \mathrm{Surj}(F, \alpha, \beta) \end{array}$$

We use the notion of embedding and not of injection in the definition of an isomorphism because the preorders need not to be total on their domain.

Example 4.1.2.2. The set $\{\langle x, \mathbf{h}(x)\rangle \mid x \in \mathbb{B}\}$ defines an isomorphism between ω and ω_b.

The composition of two morphisms $F : \alpha \to \beta$ and $G : \beta \to \gamma$ is obtained as the composition of F and G up to $\cong_\beta$:

$$(G \circ_\beta F) \quad \triangleq \quad \{\langle x, y\rangle \mid \exists z \exists z' (x F z \wedge z \cong_\beta z' \wedge z' G y)\}.$$

The notions of morphisms, embeddings and isomorphisms are stable by composition. The inverse of an isomorphism is an isomorphism. All these facts are provable in $\mathbf{ACA_0^-}$.

4.2 Well-preorders

We focus our interest on a special case of preorders: the well-preorders. The study of these relations will mostly be done inside $\mathbf{\Pi^1_1}$-$\mathbf{CA_0^-}$.

4.2.1 Definitions, examples and first properties

Definition

A well-preorder α is a preorder satisfying an extra Π^1_1-condition:

$$\mathrm{WPO}(\alpha) \quad \triangleq \quad \mathrm{PreOrd}(\alpha) \wedge (\forall X \subseteq E_\alpha)(X \neq \emptyset \Rightarrow (\exists m \in X)(\forall x \in X) \, m \leq_\alpha x)$$

It means that all subsets of E_α have an α-smallest element. In particular, this formula is downward absolute but not upward absolute (see Proposition 1.2.4.3). A well-preorder is always total. If a well-preorder α is antisymmetric, it is called a well-order (denoted $\mathrm{WO}(\alpha)$).

Example 4.2.1.1. For all $n \in \mathbb{N}$, the relation $\omega_{<n}$ is a well-order:

$$\mathbf{\Pi^1_1\text{-}CA^-_0} \vdash (\forall n \in \mathbb{N})(\mathrm{WO}(\omega_{<n}))$$

This result is shown by an induction on $\mathbb{N}$ (see Subsection 1.2.2).

Generalization of the results about preorders

We generalize the results about preorders that we obtain in the previous section.

Proposition 4.2.1.1. *Given two well-preorders α and β, the following properties are provable in $\mathbf{ACA^-_0}$.*

1. *Every proper initial segment of α is of the shape $\alpha_{<i}$ (for some $i \in E_\alpha$).*
2. *The restriction of a well-preorder is a well-preorder.*
3. *$\alpha \otimes \beta$ is a well-preorder.*
4. *$\alpha + \beta$ is a well-preorder.*
5. *$\alpha \uplus \beta$ is a well-preorder.*
6. *α^ω is well-preorder.*
7. *If γ if a total order and $\mathcal{U}$ is a γ-chain[1] of well-preorders, then $\bigcup_{i \in E_\gamma} \mathcal{U}[i]$ is a well-preorder.*

These propositions remain true if well-preorders are replaced by well-orders.

The characterization of ω as the union of its proper initial segments (which are all well-orders) and the last point of the previous proposition imply in $\mathbf{\Pi^1_1\text{-}CA^-_0}$ that ω is a well-order. We then immediately deduce that ω_b also is a well-order.

We finish this subsection by noting that it is valid in $\mathbf{ACA^-_0}$ to reason by induction on a well-preorder:

$$\mathbf{ACA^-_0} + \mathrm{WPO}(\alpha) \vdash \forall X((\forall x \in E_\alpha)((\forall y \in E_{\alpha_{<x}})y \in X \Rightarrow x \in X) \Rightarrow (\forall x \in E_\alpha)\ x \in X)$$

4.2.2 Comparing well-preorders: almost a well-preorder on the class of well-preorders

The goal of this subsection is to define a total preorder on the class of well-preorders. It is achieved by constructing a formula with two free second-order variables that implements this relation. We will see that this preorder will almost be a well-preorder but not exactly, it won't be able to distinguish two isomorphic well-preorders! Therefore, the minimality property won't be satisfied for all classes, but only for the classes that do not discriminate isomorphic well-preorders.

[1] The definition of an α-chain is given in Definition 4.1.2.2.

Constructing the preorder

We define the following formulas:

$$\alpha \cong_{\mathrm{WPO}} \beta \triangleq \exists F\ \mathrm{Isom}(F, \alpha, \beta)$$
$$\alpha \preccurlyeq_{\mathrm{WPO}} \beta \triangleq \exists F \exists \gamma(\mathrm{InitSeg}(\gamma, \beta) \wedge \mathrm{Isom}(F, \alpha, \gamma))$$
$$\alpha \prec_{\mathrm{WPO}} \beta \triangleq \exists F(\exists i \in E_\beta)\mathrm{Isom}(F, \alpha, \beta_{<i}).$$

We note that $\preccurlyeq_{\mathrm{WPO}}$ is reflexive (because the identity is an isomorphism) and transitive (because composing two isomorphisms gives an isomorphism). We now show that $\prec_{\mathrm{WPO}}$ is indeed the strict order associated to the total (on the class of well-preorders) and antisymmetric (up to $\cong_{\mathrm{WPO}}$) order $\preccurlyeq_{\mathrm{WPO}}$.

$$(a)\ \mathbf{ACA_0^-} + \mathrm{WPO}(\alpha) + \mathrm{WPO}(\beta)\ \vdash\ \alpha \prec_{\mathrm{WPO}} \beta \Leftrightarrow (\alpha \preccurlyeq_{\mathrm{WPO}} \beta \wedge \neg(\alpha \cong_{\mathrm{WPO}} \beta))$$
$$(b)\ \mathbf{\Pi^1_1\text{-}CA_0^-} + \mathrm{WPO}(\alpha) + \mathrm{WPO}(\beta)\ \vdash\ \alpha \prec_{\mathrm{WPO}} \beta \vee \beta \prec_{\mathrm{WPO}} \alpha \vee \alpha \cong_{\mathrm{WPO}} \beta$$
$$(c)\ \mathbf{\Pi^1_1\text{-}CA_0^-} + \mathrm{WPO}(\alpha) + \mathrm{WPO}(\beta)\ \vdash\ \alpha \preccurlyeq_{\mathrm{WPO}} \beta \Rightarrow \beta \preccurlyeq_{\mathrm{WPO}} \alpha \Rightarrow \alpha \cong_{\mathrm{WPO}} \beta$$

For the statement (a), the reverse implication is a consequence of the characterization of the initial segments of a well-preorder. The converse implication uses the following lemma.

Lemma 4.2.2.1. *Let α be a well-preorder and F be an embedding between initial segments of α. For every $x \in \mathrm{Dom}_F$, $x \leq_\alpha F(x)$:*

$$\mathbf{ACA_0^-} + \mathrm{WPO}(\alpha) + \mathrm{InitSeg}(\beta, \alpha) + \mathrm{InitSeg}(\gamma, \alpha) + \mathrm{Emb}(F, \beta, \gamma)\ \vdash\ (\forall x \in E_\beta)\ x \leq_\alpha F(x)$$

Especially, there is no isomorphism F between α and a proper initial segment of α. Indeed, a proper initial segment of α has the shape $\alpha_{<i}$ for some $i \in E_\alpha$. But, the existence of an isomorphism between α and $\alpha_{<i}$ contradicts the former lemma (because such an isomorphism satisfies $F(i) <_\alpha i$). Therefore, if $\alpha \prec_{\mathrm{WPO}} \beta$ then $\neg(\alpha \cong_{\mathrm{WPO}} \beta)$. Finally, the totality and the anti-symmetry (up to $\cong_{\mathrm{WPO}}$) of $\preccurlyeq_{\mathrm{WPO}}$ follow from the next lemma.

Lemma 4.2.2.2. *Let α and β be two well-preorders.*

1. $\mathbf{ACA_0^-}$ *implies that if $\alpha_{<i} \cong_{\mathrm{WPO}} \alpha_{<j}$ for i, j in the field of α, then $i \cong_\alpha j$.*
2. $\mathbf{\Pi^1_1\text{-}CA_0^-}$ *implies that α is isomorphic to a proper initial segment of β or β is isomorphic to a proper initial segment of α or α and β are isomorphic (and these three cases are mutually exclusive).*

Almost a well-preorder on the class of well-preorders

The relation $\preccurlyeq_{\mathrm{WPO}}$ defines a well-preorder up to the equivalence relation $\cong_{\mathrm{WPO}}$ on the class of well-preorders. This property is captured as follow. We say that a formula $\Xi(X_1, ..., X_n)$ is compatible for well-preorders if it does not discriminate isomorphic well-preorders:

$$\mathrm{Compat}_\Xi \triangleq \forall\alpha_1 ... \forall\alpha_n \forall\beta_1 ... \forall\beta_n((\alpha_1 \cong_{\mathrm{WPO}} \beta_1 \wedge ... \wedge \alpha_n \cong_{\mathrm{WPO}} \beta_n) \Rightarrow \Xi(\alpha_1, ..., \alpha_n) \Rightarrow \Xi(\beta_1, ..., \beta_n)).$$

A compatible formula $\Xi(X)$ with one free variable X represents a class of well-preorders $\{\gamma \mid \Xi(\gamma)\}$ stable by isomorphisms. If such a class is non-empty, it contains a minimal element β for the preorder $\preccurlyeq_{\mathrm{WPO}}$.

Proposition 4.2.2.1. *If $\Xi(X)$ is a compatible class containing at least one well-preorder, then there exists a well-preorder which is minimal for $\preccurlyeq_{\mathrm{WPO}}$ in the class $\Xi(X)$.*

$\mathbf{PA2}^- + \mathrm{Compat}_\Xi + \mathrm{WPO}(\alpha) + \Xi(\alpha) \vdash \exists\beta(\mathrm{WPO}(\beta) \wedge \Xi(\beta) \wedge \forall\gamma(\mathrm{WPO}(\gamma) \Rightarrow \Xi(\gamma) \Rightarrow \beta \preccurlyeq_{\mathrm{WPO}} \gamma))$

Proof. Let $i \in E_\alpha$ be a minimal element of α satisfying the property $\Xi(\alpha_{<i})$. We claim that $\alpha_{<i}$ is minimal in $\{\gamma \mid \Xi(\gamma)\}$. Let γ be a well-preorder in this class. If $\gamma \prec_{\mathrm{WPO}} \alpha_{<i}$, there exists $j <_\alpha i$ such that $\gamma \cong_{\mathrm{WPO}} \alpha_{<j}$. But then $\Xi(\alpha_j)$ by compatibility of $\Xi(X)$, which contradicts the minimality of i. □

Remark 4.2.2.1. Here, the instance of the comprehension scheme used depends on the formula $\Xi(X)$. Indeed, it is used to show the existence of the set $\{x \mid x \in E_\alpha \wedge \Xi(\alpha_{<x})\}$. This is hidden in the sentence "Let $i \in E_\alpha$ be a minimal element...".

An application: supremum of a family of well-preorders

Let $\mathcal{U}$ be a family of sets containing only well-preorders. We define the supremum of this family as follow:

$$\sup(\mathcal{U}) \triangleq \{\langle\langle x', x\rangle, \langle y', y\rangle\rangle \mid x \in E_{\mathcal{U}[x']} \wedge y \in E_{\mathcal{U}[y']} \wedge \mathcal{U}[x']_{<x} \preccurlyeq_{\mathrm{WPO}} \mathcal{U}[y']_{<y}\}$$

Let's explain this construction. The field of $\sup(\mathcal{U})$ is exactly the field of the family $\mathcal{U}$: all elements are tagged with the index of the well-preorders they come from. Therefore, even if the fields of the well-preorders in $\mathcal{U}$ are not disjoint, elements are not overwritten in $E_{\sup(\mathcal{U})}$. The properties of the preorder $\preccurlyeq_{\mathrm{WPO}}$ is then intensively used to design the well-preorder on this field.

Theorem 4.2.2.1. *If $\mathcal{U}$ is a family of sets containing only well-preorders, then $\mathbf{\Pi^1_1}$-$\mathbf{CA^-_0}$ proves the following properties.*

1. *The set $\sup(\mathcal{U})$ exists and is a well-preorder.*
2. *All well-preorders $\mathcal{U}[x']$ in $\mathbf{S}(\mathcal{U})$ are smaller than $\sup(\mathcal{U})$.*
3. *All initial segments of $\sup(\mathcal{U})$ are smaller than some well-preorders $\mathcal{U}[x']$ in $\mathbf{S}(\mathcal{U})$.*

Proof. 1. First, $\mathbf{\Pi^1_1}$-$\mathbf{CA^-_0}$ proves the existence of $\sup(\mathcal{U})$ and it is a preorder because $\preccurlyeq_{\mathrm{WPO}}$ is one. Now let X be a set intersecting the field of $\sup(\mathcal{U})$. Consider the Π^1_1-formula $\Theta(Y) \triangleq \exists x \exists x'(\langle x', x\rangle \in X \cap E_{\sup(\mathcal{U})} \wedge Y \cong_{\mathrm{WPO}} \mathcal{U}[x']_{<x})$. Then, $\Theta(Y)$ is a compatible and non-empty collection of well-preorders. Let β be a minimal well-preorder in this collection. In particular, $\beta \cong_{\mathrm{WPO}} \mathcal{U}[x']_{<x}$ for some $\leq_{\sup(\mathcal{U})}$-minimal element $\langle x', x\rangle \in X \cap E_{\sup(\mathcal{U})}$.

2. For $x' \in \mathrm{Dom}(\mathcal{U})$, the function $\mathcal{I}^{x'}(x) = \langle x', x\rangle$ defines an isomorphism from $\mathcal{U}[x']$ into an initial segment of $\sup(\mathcal{U})$.

3. If $\langle x', x\rangle \in \mathrm{Dom}(\sup(\mathcal{U}))$, the function $\mathcal{I}(\langle x', y\rangle) = y$ defines an isomorphism from $\sup(\mathcal{U})_{\leq\langle x', x\rangle}$ into an initial segment of $\mathcal{U}[x']$.

□

Remark 4.2.2.2. This construction wouldn't work for well-orders. In fact, up to the author knowledge, there is no machinery in $\mathbf{PA2}^-$ to define the supremum of an arbitrary family of well-orders (and to show that it is still a well-order). However, it is possible in the presence of the axiom scheme of countable choice (or if the family is indexed by a well-order).

4.2.3 Definitions by transfinite recursion

We will now show how definitions by transfinite recursion are done inside $\mathbf{PA2}^-$ (see Theorem 4.2.3.1). This section is a direct adaptation of the work of Colson and Grigorieff [15]. The only (minor) change is that definitions by transfinite recursion are made over well-preorders (rather than well-orders). But before tackling this challenge, let's give an intuition by going rapidly through this process in set theory.

An intuition: transfinite recursion in set theory

This part is inspired by the book of Krivine about set theory [30]. In set theory, it is possible to define functions by transfinite recursion over ordinals [29, 23]. Given a total functional relation $\mathcal{H}$, for every ordinal α, there exists a unique function f "$\mathcal{H}$-inductive" of domain α, i.e. satisfying

$$(\forall \beta \in \alpha) f(\beta) = \mathcal{H}(f|_\beta).$$

This construction strongly relies on ordinals, that are used as canonical representatives for classes of equivalent well-orders. Moreover, elements of an ordinal are again ordinals, and therefore the typing of the function f is not an issue. However, ordinals are not available in the framework of second-order arithmetic. A solution to bypass this problem is to parameterize the definition of being $\mathcal{H}$-inductive by a well-preorder α.

In second-order arithmetic, rather than constructing a function, we will construct a family of sets[2]. This is why we introduce the concept of a $(\alpha, \mathcal{H})$-inductive family, defined as a family $\mathcal{U}$ satisfying

$$(\forall i \in E_\alpha) \mathcal{U}[i] = \mathcal{H}(\alpha_{<i}, \mathcal{U}|_{E_{\alpha_{<i}}}).$$

In this context, the symbol $\mathcal{H}$ denotes a functional relation with two arguments:

1. An initial segment $\alpha_{<i}$ of α.
2. A family of sets $\mathcal{U}$ gathering all the sets constructed on the previous steps of the transfinite recursion.

Theorem 4.2.3.1 will show the existence of a unique $(\alpha, \mathcal{H})$-inductive family. In particular, the transfinite recursion is done on an arbitrary well-preorder and not only for a canonical choice of well-preorders. In general, transfinite recursions done on two isomorphic well-preorders give different results.

Definition and lemmas

We define the notion of $(\alpha, \mathcal{H})$-inductive families.

Definition 4.2.3.1. Let $\mathcal{H}$ be a total binary functional relation and α a preorder. A family $\mathcal{U}$ is $(\alpha, \mathcal{H})$-inductive if its domain is included in E_α and if for all $i \in E_\alpha$, the set $\mathcal{U}[i]$ is the image of $(\alpha_{<\mathrm{i}}, \mathcal{U}|_{\mathrm{E}_{\alpha_{<\mathrm{i}}}})$ by $\mathcal{H}$:

$$\mathrm{Ind}_{(\alpha,\mathcal{H})}(\mathcal{U}) \triangleq \mathcal{U} \in \mathrm{Fam} \wedge \mathrm{Dom}_{\mathcal{U}} \subseteq E_\alpha \wedge (\forall i \in E_\alpha) \mathcal{H}(\alpha_{<i}, \mathcal{U}|_{E_{\alpha_{<i}}}, \mathcal{U}[i])$$

The proof of the main result of this section (i.e. of the transfinite recursion theorem) is done in two steps:

1. We first show that there is at most one $(\alpha, \mathcal{H})$-inductive family in Proposition 4.2.3.1.

[2]Remember that it is in fact a function from the individuals to the reals.

2. We construct a $(\alpha, \mathcal{H})$-inductive family (Theorem 4.2.3.1).

Proposition 4.2.3.1. *With the same notations as in the previous definition, if α is a well-preorder, then there exists at most one $(\alpha, \mathcal{H})$-inductive family:*

$$\mathbf{ACA_0^-} + \mathrm{WPO}(\alpha) + \mathrm{FuncTot}_{\mathcal{H}} + \mathrm{Ind}_{(\alpha,\mathcal{H})}(\mathcal{U}) + \mathrm{Ind}_{(\alpha,\mathcal{H})}(\mathcal{V}) \ \vdash \ \mathcal{U} = \mathcal{V}$$

Proof. Because $Dom_{\mathcal{U}} \subseteq E_\alpha$ and $Dom_{\mathcal{V}} \subseteq E_\alpha$, it is enough to show $(\forall i \in E_\alpha)\mathcal{U}[i] = \mathcal{V}[i]$. It is done by induction over α. Assume $(\forall j \in E_{\alpha_{<i}})\mathcal{U}[j] = \mathcal{V}[j]$ for some $i \in E_\alpha$. In particular, $\mathcal{U}|_{E_{\alpha_{<i}}} = \mathcal{V}|_{E_{\alpha_{<i}}}$. But, by Theorem 1.1.2.1 of extensionality in **SOL**, $\mathcal{H}(\alpha_{<i}, \mathcal{U}|_{E_{\alpha_{<i}}}, \mathcal{U}[i])$ implies $\mathcal{H}(\alpha_{<i}, \mathcal{V}|_{E_{\alpha_{<i}}}, \mathcal{U}[i])$. Finally, because $\mathcal{H}$ is functional, $\mathcal{U}[i] = \mathcal{V}[i]$. □

The proof of Theorem 4.2.3.1 uses the two following lemmas.

Lemma 4.2.3.1. *The restriction of a $(\alpha, \mathcal{H})$-inductive family to the field of an initial segment β of α is a $(\beta, \mathcal{H})$-inductive family.*

$$\mathbf{ACA_0^-} + \mathrm{FuncTot}_{\mathcal{H}} + \mathrm{PreOrd}(\alpha) + \mathrm{Ind}_{(\alpha,\mathcal{H})}(\mathcal{U}) + \mathrm{InitSeg}(\beta, \alpha) \ \vdash \ \mathrm{Ind}_{(\beta,\mathcal{H})}(\mathcal{U}|_{E_\beta})$$

Lemma 4.2.3.2. *If there exists a unique $(\alpha_{<i}, \mathcal{H})$-inductive family for every initial segment $\alpha_{<i}$ of α, then there exists a $(\alpha, \mathcal{H})$-inductive family.*

$$\mathbf{PA2^-} + \mathrm{FuncTot}_{\mathcal{H}} + \mathrm{PreOrd}(\alpha) \ \vdash (\forall i \in E_\alpha)\exists!\mathcal{V}\, \mathrm{Ind}_{(\alpha_{<i},\mathcal{H})}(\mathcal{V}) \Rightarrow \exists\mathcal{U}\, \mathrm{Ind}_{(\alpha,\mathcal{H})}(\mathcal{U})$$

Proof. We reason by case on whether the preorder α is empty, successor or limit. The hypothesis of uniqueness is only used for the limit case, as is the comprehension axiom. □

Recursion on well-preorders in PA2$^-$

Theorem 4.2.3.1. *Let $\mathcal{H}$ be a binary functional relation. For every well-preorder α, there is exactly one $(\alpha, \mathcal{H})$-inductive family. Moreover, $\mathcal{H}$ induces a unary functional relation over the class of well-preorders.*

1. $\mathbf{PA2^-} + \mathrm{FuncTot}_{\mathcal{H}} + \mathrm{WPO}(\alpha) \ \vdash \exists!\mathcal{U}\, \mathrm{Ind}_{(\alpha,\mathcal{H})}(\mathcal{U})$
2. $\mathbf{PA2^-} + \mathrm{FuncTot}_{\mathcal{H}} + \mathrm{WPO}(\alpha) \ \vdash \exists!\mathcal{U}\exists!R(\mathrm{Ind}_{(\alpha,\mathcal{H})}(\mathcal{U}) \wedge \mathcal{H}(\alpha, \mathcal{U}, R))$

Proof. 1. The uniqueness is a consequence of Proposition 4.2.3.1. It remains to show the existence. It will follow from Lemma 4.2.3.2. Therefore, we need to construct a $(\alpha_{<i}, \mathcal{H})$-inductive family for every $i \in E_\alpha$. We will show that the set $X \triangleq \{i \mid i \in E_\alpha \wedge \exists\mathcal{U}\ \mathrm{Ind}_{(\alpha_{<i},\mathcal{H})}(\mathcal{U})\}$ is equal to E_α.

- We first note that X is an initial segment of α: this is a consequence of Lemma 4.2.3.1.
- Now, we reason classically. If $X \neq E_\alpha$, let $i \in E_\alpha$ be a minimal element of $E_\alpha \setminus X$. But, $X = E_{\alpha_{<i}}$ and then, by Lemma 4.2.3.2, $i \in X$. This is a contradiction and we conclude that $X = E_\alpha$.

2. The uniqueness of $\mathcal{U}$ is a consequence of the first part of this theorem and the uniqueness of $\mathcal{R}$ is because $\mathcal{H}$ is functional. □

The previous results are valid in **PA2$^-$** because we used instances of the axiom scheme of comprehension where $\mathcal{H}$ appears and this formula can have arbitrary complexity. If $\mathcal{H}$ is an arithmetical formula, $\mathbf{\Pi^1_1}$-$\mathbf{CA_0^-}$ is enough to show the last theorem.

Transfinite recursion is a tool that we will abundantly use in the following. We introduce notations to use recursively defined objects.

Notation 4.2.3.1. Given a well-preorder α, an individual $i \in E_\alpha$ and a binary functional relation $\mathcal{H}$, we write:

1. $\mathcal{H}^\alpha$ for the unique family satisfying the formula $\mathrm{Ind}_{(\alpha,\mathcal{H})}(X)$. In particular:
$$\mathcal{H}^\alpha|_{E_{\alpha_{<i}}} = \mathcal{H}^{\alpha_{<i}}$$
and, therefore, $i \cong_\alpha j$ implies $\mathcal{H}^\alpha[i] = \mathcal{H}^\alpha[j]$.
2. $\mathcal{H}_\alpha$ for the unique set satisfying the formula $\exists \mathcal{U}(\mathrm{Ind}_{(\alpha,\mathcal{H})}(\mathcal{U}) \wedge \mathcal{H}(\alpha,\mathcal{U},X))$.

Reasoning about recursively defined objects

We give two tools to reason about recursively defined objects.

Proposition 4.2.3.2. *We can use induction to reason about recursively defined objects and to prove a property of the shape $\phi(\alpha,\mathcal{H}^\alpha,\mathcal{H}_\alpha)$ for a formula ϕ and a binary functional relation $\mathcal{H}$. Formally, the following formula* $\mathrm{Inductive}_{(\phi,\mathcal{H})}$ *says that the formula ϕ is "inductive":*

$$\mathrm{Inductive}_{(\phi,\mathcal{H})} \triangleq (\forall \alpha \in \mathrm{WPO})\forall \mathcal{U}\forall R(\mathcal{H}(\alpha,\mathcal{U},R) \Rightarrow (\forall i \in E_\alpha)\phi(\alpha_{<i},\mathcal{U}|_{E_{\alpha_{<i}}},\mathcal{U}[i]) \Rightarrow \phi(\alpha,\mathcal{U},R)).$$

Then, for every inductive property $\phi(X,Y,Z)$ and every well-preorder α, $\phi(\alpha,\mathcal{H}^\alpha,\mathcal{H}_\alpha)$ is provable:

$$\mathbf{PA2}^- + \mathrm{FuncTot}_\mathcal{H} + \mathrm{Inductive}_{(\phi,\mathcal{H})} + \mathrm{WPO}(\alpha) \ \vdash \ \phi(\alpha,\mathcal{H}^\alpha,\mathcal{H}_\alpha)$$

We give a variant of the previous proposition to relate objects recursively defined on isomorphic well-preorders.

Proposition 4.2.3.3. *We can use induction to reason simultaneously on two objects recursively defined on isomorphic well-preorders. Let's note:*

$$\mathrm{IndIsom}_{(\Phi,\mathcal{H})} \triangleq (\forall \alpha \in \mathrm{WPO})(\forall \beta \in \mathrm{WPO})\forall \mathcal{F}\forall \mathcal{U}\forall R\forall \mathcal{U}'\forall R'$$
$$(\mathrm{Isom}(\mathcal{F},\alpha,\beta) \Rightarrow \mathcal{H}(\alpha,\mathcal{U},R) \Rightarrow \mathcal{H}(\beta,\mathcal{U}',R') \Rightarrow (\forall i \in E_\alpha)(\forall j \in E_\beta)(i\mathcal{F}j \Rightarrow \phi(\alpha,\mathcal{U}[i],\mathcal{U}'[j]) \Rightarrow \phi(\alpha,R,R')))$$

Then, for every well-preorder α and β and for every property $\phi(X,Y,Z)$:

$$\mathbf{PA2} + \mathrm{FuncTot}_\mathcal{H} + \mathrm{IndIsom}_{(\Phi,\mathcal{H})} + \mathrm{WPO}(\alpha) + \mathrm{WPO}(\beta) + \alpha \cong_{\mathrm{WPO}} \beta \ \vdash \ \phi(\alpha,\mathcal{H}_\alpha,\mathcal{H}_\beta)$$

When proving properties about recursively defined objects, we will sometimes start a proof with "We reason by induction on...". It means that we are using one of these two propositions.

Finally, Propositions 4.2.3.2 and 4.2.3.3 can be adapted to do a simultaneous induction on two recursively defined objects. When such a technique is used in a proof, we will start it by "We reason by simultaneous induction on...".

4.2.4 A variation: iterating relations over ω

In the following, we will need to prove the existence of a well-preorder constructed by recursion over a relation that is functional only up to $\cong_{\mathrm{WPO}}$. However, the machinery presented before cannot be used in such a case. We explain here the case where the recursion is made on the well-order ω. It will use the axiom scheme of collection in an essential manner.

Let $\mathcal{F}(X,Y)$ be a formula that describes a total compatible functional (up to $\cong_{\mathrm{WPO}}$) relation on the class of well-preorders. Intuitively, for a well-preorder α, we want to show the existence of a well-preorder isomorphic to $\sup_{n\in\mathbb{N}}\mathcal{F}^n(\alpha)$. It is formalized in Theorem 4.2.4.1. But before stating this theorem, we need to introduce a new notation to represent finite iterations of a relation (over well-preorders).

Definition 4.2.4.1. Given a binary relation $\mathcal{F}$ over well-preorders, the formula $\mathrm{Iter}_{\mathcal{F}}(n, \alpha, \beta)$ says that the result of iterating n times $\mathcal{F}$ from α gives the result β:

$$\mathrm{Iter}_{\mathcal{F}}(n, \alpha, \beta) \triangleq n \in \mathbb{N} \wedge \exists \mathcal{U}(\mathcal{U}[0] \cong_{\mathrm{WPO}} \alpha \wedge \mathcal{U}[n] \cong_{\mathrm{WPO}} \beta \wedge (\forall i < n)\mathcal{U}[i]\mathcal{F}\mathcal{U}[i+1]).$$

For a fixed integer $n, \mathrm{Iter}^n_{\mathcal{F}}(\alpha, \beta)$ will denote the relation $\mathrm{Iter}_{\mathcal{F}}(n, \alpha, \beta)$.

A binary relation $\mathcal{F}$ over well-preorders is said to be a (total) functional up to $\cong_{\mathrm{WPO}}$ if

$$\mathrm{Func}^{\cong_{\mathrm{WPO}}}_{\mathcal{F}} \triangleq (\forall \alpha \in \mathrm{WPO})(\exists \beta \in \mathrm{WPO})(\alpha \mathcal{F} \beta \wedge (\forall \beta' \in \mathrm{WPO})\, \alpha \mathcal{F} \beta' \Rightarrow \beta \cong_{\mathrm{WPO}} \beta').$$

Lemma 4.2.4.1. *If $\mathcal{F}$ is compatible and functional up to $\cong_{\mathrm{WPO}}$, then, for all integer n, the formula $\mathrm{Iter}^n_{\mathcal{F}}$ is also functional up to $\cong_{\mathrm{WPO}}$.*

$$\mathbf{PA2}^- + \mathrm{Compat}_{\mathcal{F}} + \mathrm{Func}^{\cong_{\mathrm{WPO}}}_{\mathcal{F}} \vdash (\forall n \in \mathbb{N})(\mathrm{Compat}_{\mathrm{Iter}^n} \wedge \mathrm{Func}^{\cong_{\mathrm{WPO}}}_{\mathrm{Iter}^n})$$

Proof. By induction on ω. □

Remark 4.2.4.1. A total functional relation $\mathcal{F}_0$ which is compatible can always be made functional up to $\cong_{\mathrm{WPO}}$ by defining $\mathcal{F}$ as

$$\mathcal{F}(\alpha, \beta) \triangleq \mathcal{F}_0(\alpha, \beta) \wedge (\forall \gamma \prec \beta) \neg \mathcal{F}_0(\alpha, \gamma).$$

Then, the work done in Subsection 4.2.2 about the relation $\preccurlyeq_{\mathrm{WPO}}$ implies that $\mathcal{F}$ is a total compatible functional relation up to $\cong_{\mathrm{WPO}}$. Indeed, because $\mathcal{F}_0$ is compatible, for each α, we can chose the minimal well-preorders β such that $\mathcal{F}_0(\alpha, \beta)$.

Theorem 4.2.4.1. *Assume $\mathcal{F}(X, Y)$ is a formula that describes a compatible and functional relation up to $\cong_{\mathrm{WPO}}$, then, for all well-preorder α, the axiom scheme of collection implies the existence of a family of well-preorders $\mathcal{V}$ such that*

1. $(\forall n \in \mathbb{N})\exists i\, \mathrm{Iter}^n_{\mathcal{F}}(\alpha, \mathcal{V}[i])$
2. $(\forall i \in \mathrm{Dom}_{\mathcal{V}})(\exists n \in \mathbb{N})\mathrm{Iter}^n_{\mathcal{F}}(\alpha, \mathcal{V}[i])$

Formally:

$$\mathbf{PA2}^- + \mathbf{Coll} + \alpha \in \mathrm{WPO} \vdash (\exists \mathcal{V} \in \mathrm{Fam})((\forall n \in \mathbb{N})\exists i\, \mathrm{Iter}^n_{\mathcal{F}}(\alpha, \mathcal{V}[i]) \wedge (\forall i \in \mathrm{Dom}_{\mathcal{V}})(\exists n \in \mathbb{N})\mathrm{Iter}^n_{\mathcal{F}}(\alpha, \mathcal{V}[i]))$$

In particular, $\mathcal{V}$ is a family of well-preorders and $\sup(\mathcal{V})$ exists: it is the result of iterating the functional $\mathcal{F}$ along ω. We will also say that the family $\mathcal{V}$ is obtained by iteration along ω.

Proof. Let $\alpha \in \mathrm{WPO}$ and define

$$\mathrm{Iter}^{\alpha}(n, \beta) \triangleq (n \in \mathbb{N} \wedge \mathrm{Iter}_{\mathcal{F}}(n, \alpha, \beta)) \vee (n \notin \mathbb{N} \wedge \beta = \emptyset)$$

Using induction on $\mathbb{N}$, we can show $\forall n \exists \beta \mathrm{Iter}^{\alpha}(n, \beta)$.

It follows that the axiom scheme of collection implies $\exists \mathcal{U} \forall n \exists y \mathrm{Iter}^{\alpha}(n, \mathcal{U}[y])$. Finally, the family $\mathcal{V}$ defined by

$$\mathcal{V} \triangleq \{\langle x, y\rangle \mid (\exists n \in \mathbb{N})\mathrm{Iter}^{\alpha}(n, \mathcal{U}[x]) \wedge \langle x, y\rangle \in \mathcal{U}\}$$

satisfies the two desired properties. □

This proof uses the axiom scheme of collection. It is one of the only place where it is used[3]. However, Theorem 4.2.4.1 is a key ingredient for the proof of the reflection principle (see Subsection 5.4.1).

Remark 4.2.4.2. This theorem can be adapted to a relation $\mathcal{F}(n, \alpha, \beta)$ that also depends on an integer $n \in \mathbb{N}$.

[3]Its other use being in the proof of the reflection principle (Theorem 5.4.1.1).

4.2.5 Well-preorders on classes of sets

We studied well-preorders on individuals. We continue with the study of well-preorders on classes of sets. Recall that a class of sets is just a formula $\Theta(X)$ with one distinguished variable. An order on such a class will be represented by a formula $\leq_\Theta (X, Y)$ with two distinguished free second-order variables.

Definition 4.2.5.1. With the notation introduced above, $\leq_\Theta$ is a well-preorder on Θ if:

1. It is a preorder, i.e.

$$\mathrm{PreOrd}_{(\Theta,\leq_\Theta)} \triangleq (\forall X \in \Theta)(\forall Y \in \Theta)(\forall Z \in \Theta)(X \leq_\Theta X \wedge (X \leq_\Theta Y \Rightarrow Y \leq_\Theta Z \Rightarrow X \leq_\Theta Z)).$$

2. For every class $\{X \mid \phi(X)\}$ that intersects Θ, there is a $\leq_\Theta$-minimal set in their intersection. Formally, for every formula $\phi(X)$ with parameters:

$$\mathrm{WO}_{(\Theta,\leq_\Theta,\phi)} \triangleq (\exists M \in \Theta)\phi[X := M] \Rightarrow (\exists M \in \Theta)(\phi[X := M] \wedge (\forall Y \in \Theta)(\phi[X := Y] \Rightarrow M \leq_\Theta Y)).$$

If all these formulas are satisfied, then $\leq_\Theta$ is a well-preorder on Θ. The (meta-theoretic) set of all theses formulas is written $\mathrm{WPO}_{(\Theta,\leq_\Theta)}$. If on top of that it is antisymmetric, i.e.

$$(\forall X \in \Theta)(\forall Y \in \Theta)(X \leq_\Theta Y \Rightarrow Y \leq_\Theta X \Rightarrow X = Y)$$

then we say that it is a well-order and we write $\mathrm{WO}_{(\Theta,\leq_\Theta)}$ for the set of formulas obtained by adding the latter to the set $\mathrm{WPO}_{(\Theta,\leq_\Theta)}$.

Well-orders on a family of sets

Let $\mathcal{U}$ be a family, a well-order on $\mathbf{S}(\mathcal{U})$ can be extracted from a well-preorder α if:

1. The field of α contains an index for every set in $\mathbf{S}(\mathcal{U})$.
2. Two isomorphic elements $x \cong_\alpha y$ are indexes of the same set in $\mathbf{S}(\mathcal{U})$, i.e. $\mathcal{U}[x] = \mathcal{U}[y]$.

The key is to compare the sets through their α-minimal indexes.

Proposition 4.2.5.1. *With the hypothesis described above, α induces a well order $\leq_\alpha^{\mathcal{U}}$ on $\mathbf{S}(\mathcal{U})$ defined as follow*

$$X \leq_\alpha^{\mathcal{U}} Y \triangleq (\exists p \in E_\alpha)(\exists q \in E_\alpha) (X = \mathcal{U}[p] \wedge Y = \mathcal{U}[q] \wedge p \leq_\alpha q \wedge \forall x((X = \mathcal{U}[x] \Rightarrow p \leq_\alpha x) \wedge (Y = \mathcal{U}[x] \Rightarrow q \leq_\alpha x))).$$

Concretely, for all such $\mathcal{U}$ and α, $\mathbf{PA2}^-$ proves $\mathrm{WO}_{(\mathbf{S}(\mathcal{U}),\leq_\alpha^{\mathcal{U}})}$.

Proof. The assumptions are necessary to prove that it is a total order. The existence of minimal sets is deduced from the fact that α is a well-preorder. □

Chapter 5

The ramified analytic hierarchy

The goal of this chapter is to introduce and study the ramified analytic hierarchy (RAH) [26, 15] which, in **PA2**, has the same role as the constructible universe (L) in **ZF** [21, 29]. We recall that in **ZF** the hierarchy of the constructible sets is the transfinite sequence $(\mathrm{L}_\alpha)_{\alpha\in On}$ indexed by the ordinals and satisfying the equation

$$\mathrm{L}_\alpha = \bigcup_{\beta<\alpha} \mathbf{Def}(\mathrm{L}_\beta)$$

where **Def** is the set-theoretic operator that maps each set X to the set $\mathbf{Def}(X)$ of its constructible subsets.

In a similar fashion, in **PA2**, we want to define RAH as the transfinite sequence RAH_α satisfying the equation

$$RAH_\alpha = \bigcup_{\beta<\alpha} \mathbf{Def^2}(\mathrm{RAH}_\beta).$$

The formalization and the study of the operator $\mathbf{Def^2}$ is done in Section 5.1. We will start by encoding the syntax of $\mathbf{PA2}^-$ in the individuals and, with these encodings, we will internalize the notion of satisfiability. Concretely, for each family of sets $\mathcal{U}$ (represented by a set), we will construct a set $G_\mathcal{U}$ containing the codes of the closed formulas with second-order parameters in $\mathbf{S}(\mathcal{U})$ satisfied by $\mathcal{U}$ (when $\mathcal{U}$ is seen as a structure).

Finally, we will be able to construct the hierarchy RAH by iterating the operator $\mathbf{Def^2}$ over the class of the well-preorders and to show that it is a model of **PA2** satisfying the principle of the well-ordered universe (Section 5.2). However, we won't be able to show that RAH satisfies the axiom of constructibility. In the Section 5.5, we discuss in depth this problem and we compare our work to what have been done in the literature about the internalization of RAH in second-order arithmetic. Notably, we explain how we find a flaw in a paper [15] and how we think that the problem of proving in **PA2** that RAH satisfies the axiom of constructibility is still open.

As a final remark, we mention that this part was strongly inspired by the work of Colson and Grigorieff [15] (see Section 5.5.2). We rearranged their work to make it fit in our framework and we explain how we were able to give a partial solution to the flaw of their research.

5.1 Internalization of the notion of satisfiability

This section is divided in four parts:

1. We start by an internalization of the syntax of $\mathbf{PA2}^-$. To this end, we intensively use the programming made in the Section 1.2.2.

2. Building on this work, we internalize inside of $\mathbf{PA2}^-$ the notion of the satisfiability of a formula relativized to standard individuals for the first-order part and to a class of sets (encoded by the slices of a family of sets as explained in Paragraph 4.1.1) for the second-order part. Specifically, for every family of sets $\mathcal{U}$, the goal is to show the existence of a set $\mathrm{Sat}(\mathcal{U})$ such that for all closed formulas ϕ:

$$(\phi^{\mathbb{B}})^{\mathbf{S}(\mathcal{U})} \Leftrightarrow \ulcorner\phi\urcorner \in \mathrm{Sat}(\mathcal{U}).$$

   It is done in Theorem 5.1.2.2.

3. This work will allow us to consider families of sets as structures for the language of **PA2**. We will then see how to express that such a structure models a theory $\mathcal{T}$ (represented by a definable set of codes of formulas). In particular, these encoded structures will always satisfy the axiom of induction as the first-order part of a formula will be relativized to the set $\mathbb{B}$.

4. Finally, we will define the operator $\mathbf{Def^2}$ that sends a family of set $\mathcal{U}$ to the family $\mathbf{Def^2}(\mathcal{U})$ whose slices are exactly the sets definable by a formula relativized to $\mathbb{B}$ for the first-order part and to $\mathbf{S}(\mathcal{U})$ for the second-order part.

5.1.1 Internalization of the syntax

The goal of this subsection is to develop tools to interpret the language of $\mathbf{PA2}^-$ inside $\mathbf{PA2}^-$. We will successively define the encodings (à la Gödel) of a code of functions, a term and a formula. We will then define the notion of valuations, allowing us to consider formulas with parameters. Finally, we will show how to compute the denotation of the encoding of a term with parameters. In this chapter, we strongly use all the programming done in Section 1.2.2. In particular, we will not write the codes of the primitive functions that we use in this subsection. We will only give the equations that they satisfy (as advertised in Remark 1.2.2.1).

Gödel encodings

We suppose given

1. A bijection $x \mapsto \sharp x$ from the set of first-order variables into the set $\mathbb{N}$.

2. A bijection $X \mapsto \sharp X$ from the set of second-order variables into the set $\mathbb{N}$.

Definition 5.1.1.1. A code of primitive recursive functions f (of the syntax of $\mathbf{PA2}^-$) is mapped to its Gödel encoding $\ulcorner f\urcorner$ defined by the following clauses.

$$\begin{array}{rcl@{\qquad}rcl}
\ulcorner\mathbf{0}\urcorner & \triangleq & 0 & & & \\
\ulcorner\mathrm{id}\urcorner & \triangleq & 1 & \ulcorner(f \circ g)\urcorner & \triangleq & \langle 1, \ulcorner f\urcorner, \ulcorner g\urcorner\rangle \\
\ulcorner\mathrm{fst}\urcorner & \triangleq & 2 & \ulcorner\langle g, f\rangle\urcorner & \triangleq & \langle 2, \ulcorner f\urcorner, \ulcorner g\urcorner\rangle \\
\ulcorner\mathrm{snd}\urcorner & \triangleq & 3 & \ulcorner(f[f|g]\urcorner & \triangleq & \langle 3, \ulcorner f\urcorner, \ulcorner g\urcorner\rangle
\end{array}$$

We also consider a primitive recursive function `check_func` testing if its entry is the Gödel encoding of a primitive recursive function, i.e. satisfying

$$\begin{aligned}\mathbf{ACA_0^-} \vdash\ (\forall x \in \mathbb{B})(\texttt{check\_func}(x) = 1\ \Leftrightarrow\ & \\ & x = 0 \vee x = 1 \vee x = 2 \vee x = 3 \vee \\ & \exists n \exists y \exists z (x = \langle n, y, z\rangle \wedge (n = 1 \vee n = 2 \vee n = 3) \wedge \\ & \qquad \texttt{check\_func}(y) = 1 \wedge \texttt{check\_func}(z) = 1))\end{aligned}$$

Definition 5.1.1.2. A term t (of the syntax of $\mathbf{PA2^-}$) is mapped to its Gödel encoding $\ulcorner t \urcorner$ defined by the following clauses.

$$\begin{array}{rclcrcl} \ulcorner 0 \urcorner & \triangleq & 0 & & \ulcorner \langle t, u\rangle \urcorner & \triangleq & \langle 1, \ulcorner t \urcorner, \ulcorner u \urcorner\rangle \\ \ulcorner x \urcorner & \triangleq & \langle 0, \sharp x\rangle & & \ulcorner f(t) \urcorner & \triangleq & \langle 2, \ulcorner f \urcorner, \ulcorner t \urcorner\rangle \end{array}$$

We also consider a primitive recursive function `check_term` testing if its entry is the Gödel encoding of a term, i.e. satisfying

$$\begin{aligned}\mathbf{ACA_0^-} \vdash\ (\forall x \in \mathbb{B})\,(\texttt{check\_term}(x) = 1\ \Leftrightarrow\ & \\ & x = 0 \vee \\ & \exists y (x = \langle 0, y\rangle \wedge \texttt{check\_nat}(y) = 1) \vee \\ & \exists y \exists z (x = \langle 1, y, z\rangle \wedge \texttt{check\_term}(y) = 1 \wedge \texttt{check\_term}(z) = 1) \vee \\ & \exists y \exists z (x = \langle 2, y, z\rangle \wedge \texttt{check\_func}(y) = 1 \wedge \texttt{check\_term}(z) = 1))\end{aligned}$$

We can then define the set containing all the encodings of terms as $\texttt{Term} \triangleq \{x \mid x \in \mathbb{B} \wedge \texttt{check\_term}(x) = 1\}$.

Definition 5.1.1.3. A formula ϕ (of the syntax of $\mathbf{PA2^-}$) is mapped to its Gödel encoding $\ulcorner \phi \urcorner$ defined by the following clauses.

$$\begin{array}{rclcrcl} \ulcorner \bot \urcorner & \triangleq & 0 & & \ulcorner \phi \Rightarrow \psi \urcorner & \triangleq & \langle 2, \ulcorner \phi \urcorner, \ulcorner \psi \urcorner\rangle \\ \ulcorner t = u \urcorner & \triangleq & \langle 0, \ulcorner t \urcorner, \ulcorner u \urcorner\rangle & & \ulcorner \forall x \phi \urcorner & \triangleq & \langle 3, \sharp x, \ulcorner \phi \urcorner\rangle \\ \ulcorner t \in X \urcorner & \triangleq & \langle 1, \ulcorner t \urcorner, \sharp X\rangle & & \ulcorner \forall X \phi \urcorner & \triangleq & \langle 4, \sharp X, \ulcorner \phi \urcorner\rangle \end{array}$$

We also consider a primitive recursive function `check_form` testing if its entry is the Gödel encoding of some formulas, i.e. satisfying

$$\begin{aligned}\mathbf{ACA_0^-} \vdash (\forall x \in \mathbb{B})(\texttt{check\_form}(x) = 1\ \Leftrightarrow\ & \\ & x = 0 \vee \\ & \exists y \exists z (x = \langle 0, y, z\rangle \wedge \texttt{check\_term}(y) = 1 \wedge \texttt{check\_term}(z) = 1) \vee \\ & \exists y \exists z (x = \langle 1, y, z\rangle \wedge \texttt{check\_term}(y) = 1 \wedge \texttt{check\_nat}(z) = 1) \vee \\ & \exists y \exists z (x = \langle 2, y, z\rangle \wedge \texttt{check\_form}(y) = 1 \wedge \texttt{check\_form}(z) = 1) \vee \\ & \exists y \exists z (x = \langle 3, y, z\rangle \wedge \texttt{check\_nat}(y) = 1 \wedge \texttt{check\_form}(z) = 1) \vee \\ & \exists y \exists z (x = \langle 4, y, z\rangle \wedge \texttt{check\_nat}(y) = 1 \wedge \texttt{check\_form}(z) = 1))\end{aligned}$$

We can then define the set containing all the encodings of formulas as $\texttt{Form} \triangleq \{x \mid x \in \mathbb{B} \wedge \texttt{check\_form}(x) = 1\}$.

Definition 5.1.1.4. We define in the same way a primitive recursive function `check_arith` that checks if its entry is the Gödel encoding of some arithmetical formulas (i.e. without second-order quantifiers) and the set $\texttt{FormArith} \triangleq \{x \mid x \in \mathbb{B} \wedge \texttt{check\_arith}(x) = 1\}$.

Notation 5.1.1.1. Given $n \in \mathbb{N}, f \in \texttt{Form}, g \in \texttt{Form}, t \in \texttt{Term}, u \in \texttt{Term}$, the following notations are used in the sequel.

$$\begin{array}{rcl@{\qquad}rcl} \ulcorner \bot \urcorner & \triangleq & 0 & \ulcorner f \Rightarrow g \urcorner & \triangleq & \langle 2, f, g \rangle \\ \ulcorner t = u \urcorner & \triangleq & \langle 0, t, u \rangle & \ulcorner \forall^1 n f \urcorner & \triangleq & \langle 3, n, f \rangle \\ \ulcorner t \in n \urcorner & \triangleq & \langle 1, t, n \rangle & \ulcorner \forall^2 n f \urcorner & \triangleq & \langle 4, n, f \rangle. \end{array}$$

The notion of free variables can also be internalized and, in particular, we define a primitive recursive function $\texttt{FV}^2(n, f)$ that returns 1 if the second-order variable of code n appears freely in f and that returns 0 otherwise. It allows us to define the set $\texttt{FV}^2 \triangleq \{\langle f, n\rangle \mid f \in \texttt{Form} \wedge n \in \mathbb{N} \wedge \texttt{FV}^2(n, f) = 1\}$ whose slice at $\ulcorner \phi \urcorner$ is the set containing the encodings of all the free second-order variables of ϕ.

Valuations and denotation of encodings

Definition 5.1.1.5. A valuation is a finite list that associates individuals to natural number:

$$\rho \triangleq [\langle n_1, \nu_1\rangle, ..., \langle n_k, \nu_k\rangle].$$

It is a finite list of pairs of the shape $\langle n, \nu\rangle$ where n is an integer (denoting the encoding of a variable) and where ν is a value which is associated to n in the list ρ.

In the sequel, valuations will be used to interpret free first-order variables but also, when coupled with the concept of families of sets, they will be used to interpret free second-order variables. When valuations are used to interpret free first-order variables, they will only assign pure binary trees to first-order variables. It will allow us to recover the axiom of induction in the structures that we will consider. A first-order valuation ρ should then belong to $\mathbb{B}$ while a second-order valuation can range over all individuals[1]. We introduce new notations for the set of first-order valuations and the set of second-order valuations:

$$\texttt{Val}^1 \triangleq \mathbb{B} \qquad \text{and} \qquad \texttt{Val}^2 \triangleq \{\rho \mid \rho = \rho\}.$$

By convention, we treat the list ρ as a stack satisfying the FILO principle (First In Last Out). In the list $\langle\langle n, \nu\rangle, \rho'\rangle$, the first association $\langle n, \nu\rangle$ hides all the other affectations of the integer n in ρ'. Therefore, the function $\texttt{find}(\rho, n)$ that extracts the value associated to an integer n in ρ satisfies the equations:

$$\begin{array}{l} \mathbf{ACA_0^-} \vdash \forall n\, \texttt{find}(0, n) = 0 \\ \mathbf{ACA_0^-} \vdash (\forall n \in \mathbb{N})\forall \nu \forall \rho\, \texttt{find}(\langle n, \nu\rangle :: \rho, n) = \nu \\ \mathbf{ACA_0^-} \vdash (\forall n \in \mathbb{N})(\forall m \in \mathbb{N})\forall \nu \forall \rho (m \neq n \Rightarrow \texttt{find}(\langle m, \nu\rangle :: \rho, n) = \texttt{find}(\rho, n)) \end{array}$$

We note that:

1. The function `find` always returns the most recent affectation, i.e. the leftmost one in the valuation ρ.
2. When no value is associated to the integer n, the function `find` returns 0.
3. The previous equations are satisfied by `find` even if the individuals ν and ρ are not in the set $\mathbb{B}$ (of pure binary trees).

[1] Because a second-order valuation can range over all individuals, the set of second-order valuations will contain all the individuals.

We now define a formula $\mathrm{EvalTerm}(\ulcorner t \urcorner, \rho, y)$ allowing us to compute the denotation of the encoding of a term t with first-order parameters in ρ. To achieve this goal, we need to overcome the following difficulty, explained in the next remark.

Remark 5.1.1.1. There is no primitive recursive function `eval_func` that is universal for primitive recursion, in the sense that $\texttt{eval\_func}(\ulcorner f \urcorner, x) = f(x)$ for all primitive recursive functions f. This is a consequence of a standard diagonal argument: if such a function existed then we could define a primitive recursive function g such that $g(f, x) = \texttt{eval\_func}(f, \langle x, x \rangle) + 1$ and conclude that

$$g(\ulcorner g \urcorner, \ulcorner g \urcorner) = \texttt{eval\_func}(\ulcorner g \urcorner, \langle \ulcorner g \urcorner, \ulcorner g \urcorner \rangle) + 1 = g(\ulcorner g \urcorner, \ulcorner g \urcorner) + 1.$$

However, we can construct a primitive recursive function $\texttt{evaln\_func}(n, f, x)$ simulating n steps of the computation of the function encoded by f on the entry x, i.e. satisfying the equations:

$$\texttt{evaln\_func}(n, \ulcorner f \urcorner, x) = \begin{cases} \mathtt{s}(f(x)) & \text{if } n \in \mathbb{N} \text{ and the computation takes less than } n \text{ steps} \\ 0 & \text{otherwise.} \end{cases}$$

We can then define the formula

$$\mathrm{EvalFunc}(f, x, y) \triangleq (\exists n \in \mathbb{N})\ \texttt{evaln\_func}(n, f, x) = \mathtt{s}(y).$$

For all primitive recursive function f, this formula satisfies:

$$\mathbf{ACA_0^-} \ \vdash\ (\forall x \in \mathbb{B}) \forall y (\mathrm{EvalFunc}(\ulcorner f \urcorner, x, y) \Leftrightarrow y = f(x)).$$

Note that EvalFunc can only compute a function on a pure binary tree (indeed, induction on x is used to prove the last formula). Therefore, in the following, the computation of a term should always be a pure binary tree. This is why first-order valuations interpret variables by pure binary trees.

Using the function `evaln_func`, we implement a primitive recursive function $\texttt{evaln\_term}(n, t, \rho)$ simulating n steps of the computation of the value of the term encoded by t with parameters described by the valuation ρ. We, can then define the formula

$$\mathrm{EvalTerm}(t, \rho, y) \triangleq (\exists n \in \mathbb{N})\ \texttt{evaln\_term}(n, t, \rho) = \mathtt{s}(y).$$

This formula has the following properties.

$\mathbf{ACA_0^-} \vdash \forall y (\forall \rho \in \mathtt{Val}^1)(\mathrm{EvalTerm}(\ulcorner t \urcorner, \rho, y) \Rightarrow y \in \mathbb{B})$

$\mathbf{ACA_0^-} \ \vdash\ \forall y (\mathrm{EvalTerm}(\ulcorner 0 \urcorner, \rho, y) \Leftrightarrow y = 0)$

$\mathbf{ACA_0^-} \ \vdash\ \forall y \forall \rho (\mathrm{EvalTerm}(\ulcorner x \urcorner, \rho, y) \Leftrightarrow y = \texttt{find}(\rho, \sharp x))$

$\mathbf{ACA_0^-} \ \vdash\ \forall y (\forall \rho \in \mathtt{Val}^1)(\mathrm{EvalTerm}(\ulcorner \langle t, u \rangle \urcorner, \rho, y) \Leftrightarrow \exists y_1 \exists y_2 (y = \langle y_1, y_2 \rangle \wedge \mathrm{EvalTerm}(\ulcorner t \urcorner, \rho, y_1) \wedge \mathrm{EvalTerm}(\ulcorner u \urcorner, \rho, y_2)))$

$\mathbf{ACA_0^-} \ \vdash\ \forall y (\forall \rho \in \mathtt{Val}^1)(\mathrm{EvalTerm}(\ulcorner f(t) \urcorner, \rho, y) \Leftrightarrow \exists y' (\mathrm{EvalTerm}(\ulcorner t \urcorner, \rho, y') \wedge \mathrm{EvalFunc}(\ulcorner f \urcorner, y', y)))$

Note that the first formula is a necessary lemma to prove the last one. Finally, we summarize these results in the following scheme of propositions.

Proposition 5.1.1.1. *For each term $t(x_1, ..., x_p)$ of* $\mathbf{PA2^-}$*:*

1. $\mathbf{ACA_0^-} \ \vdash\ (\forall \rho \in \mathtt{Val}^1)\ \mathrm{EvalTerm}(\ulcorner t \urcorner, \rho, t[x_1 := \texttt{find}(\rho, \sharp x_1); ...; x_p := \texttt{find}(\rho, \sharp x_p)])$
2. $\mathbf{ACA_0^-} \ \vdash\ \forall v (\forall \rho \in \mathtt{Val}^1)(\mathrm{EvalTerm}(\ulcorner t \urcorner, \rho, v) \Rightarrow v = t[x_1 := \texttt{find}(\rho, \sharp x_1); ...; x_p := \texttt{find}(\rho, \sharp x_p)])$

5.1.2 Internalization of satisfiability

Intuition

Every class Θ induces a structure $\Theta_{\mathcal{M}}$ of the language of **SOL**:

1. The first-order part of $\Theta_{\mathcal{M}}$ is the term model of $\mathbf{PA2^-}$, i.e. the set of closed terms (with parameters in $\mathbb{B}$) of $\mathbf{PA2^-}$ quotiented by the equivalence relation

$$t \cong u \quad \text{if and only if} \quad \mathbf{PA2^-} \vdash t = u.$$

   In particular, it is a standard model as this quotient set is isomorphic to $\mathbb{B}$ (we will see that each structure $\Theta_{\mathcal{M}}$ satisfies the axiom of induction).

2. The second-order part is defined as all the sets included in the class Θ.

A formula is then inductively interpreted in the structure $\Theta_{\mathcal{M}}$. If Θ is encoded by a family of sets $\mathcal{U}$, it is possible to internalize this process inside $\mathbf{PA2^-}$. The formula $\mathrm{SatGraph}(\mathcal{U}, G)$ (Definition 5.1.2.2) expresses that the set G contains all triples $\langle \ulcorner \phi \urcorner, \rho_1, \rho_2 \rangle$ such that

$$\text{“ } \mathbf{S}(\mathcal{U})_{\mathcal{M}} \text{ satisfies the formula } \phi[\vec{x} := \mathtt{find}(\rho_1, \sharp\vec{x}), \overrightarrow{X} := \mathcal{U}[\mathtt{find}(\rho_2, \sharp\overrightarrow{X})]]\text{”}.$$

Especially:

1. Under the valuation ρ_1, the code of a first-order variable $\sharp x$ is interpreted as the individual $\mathtt{find}(\rho_1, \sharp x)$ (which is a pure binary tree).

2. Under the valuation ρ_2, the code of a second-order variable $\sharp X$ is interpreted as the slice $\mathcal{U}[\mathtt{find}(\rho_2, \sharp X)]$. Consequently, valuations are also used to interpret second-order variables. Because the index of a set in $\mathbf{S}(\mathcal{U})$ is not necessarily a pure binary tree[2], second-order valuations are not restricted to pure binary trees. All in all, this process will replace second-order quantifiers by first-order quantifiers.

The first-order part of a structure $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$ will be isomorphic to $\mathbb{B}$. Therefore, the structure $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$ will satisfy a formula ϕ if and only if its relativization to the set $\mathbb{B}$ (for the first-order part) and to the class $\mathbf{S}(\mathcal{U})$ (for the second-order part) is satisfied. This is the motivation for the following definition that introduces a notation for these two successive relativizations.

Definition 5.1.2.1. For a class of sets Θ, we introduce the operation of relativization to the structure $\Theta_{\mathcal{M}}$ obtained by composing the operations of relativization to the set $\mathbb{B}$ and to the class Θ:

$$\phi^{\Theta_{\mathcal{M}}} \triangleq (\phi^{\mathbb{B}})^{\Theta}.$$

In particular, if a class is always used as a structure[3], the index $\mathcal{M}$ will be dropped.

[2] The use of the axiom scheme of collection forbids us to only consider sets in $\mathbf{S}(\mathcal{U})$ indexed by a standard individual. Indeed, families of sets constructed with this axiom scheme may contain sets only indexed by individuals which are not in the set $\mathbb{B}$.

[3] For instance, it is the case for RAH_α and RAH.

The graph of satisfiability

Definition 5.1.2.2. The following formula, with two distinguished second-order variables denoting a family $\mathcal{U}$ and a set G, expresses that the set G is the graph of the relation of satisfiability in the class $\mathbf{S}(\mathcal{U})$:

$$
\begin{array}{l}
\mathrm{SatGraph}(\mathcal{U}, G) \triangleq \mathcal{U} \in \mathrm{Fam} \quad \wedge \\
\forall x (x \in G \Rightarrow (\exists f \in \mathtt{Form})(\exists \rho_1 \in \mathtt{Val}^1)(\exists \rho_2 \in \mathtt{Val}^2)\ x = \langle f, \rho_1, \rho_2 \rangle) \quad \wedge \\
(\forall n \in \mathbb{N})(\forall t \in \mathtt{Term})(\forall u \in \mathtt{Term})(\forall f \in \mathtt{Form})(\forall g \in \mathtt{Form})(\forall \rho_1 \in \mathtt{Val}^1)(\forall \rho_2 \in \mathtt{Val}^2) \\
(\langle \ulcorner \bot \urcorner, \rho_1, \rho_2 \rangle \notin G \quad \wedge \\
(\langle t^{\ulcorner} = \urcorner u, \rho_1, \rho_2 \rangle \in G \Leftrightarrow \exists x (\mathrm{EvalTerm}(t, \rho_1, x) \wedge \mathrm{EvalTerm}(u, \rho_1, x))) \quad \wedge \\
(\langle t^{\ulcorner} \in \urcorner n, \rho_1, \rho_2 \rangle \in G \Leftrightarrow\ \exists v (\mathrm{EvalTerm}(t, \rho_1, v) \wedge v \in \mathcal{U}[\mathtt{find}(\rho_2, n)])) \quad \wedge \\
(\langle f^{\ulcorner} \Rightarrow \urcorner g, \rho_1, \rho_2 \rangle \in G \Leftrightarrow (\langle f, \rho_1, \rho_2 \rangle \in G \Rightarrow \langle g, \rho_1, \rho_2 \rangle \in G)) \quad \wedge \\
(\langle \ulcorner \forall^1 \urcorner n f, \rho_1, \rho_2 \rangle \in G \Leftrightarrow (\forall x \in \mathbb{B})\ \langle f, \langle \langle n, x \rangle, \rho_1 \rangle, \rho_2 \rangle \in G) \quad \wedge \\
(\langle \ulcorner \forall^2 \urcorner n f, \rho_1, \rho_2 \rangle \in G \Leftrightarrow \forall x\ \langle f, \rho_1, \langle \langle n, x \rangle, \rho_2 \rangle \rangle \in G))
\end{array}
$$

In the formula $\mathrm{SatGraph}(\mathcal{U}, G)$, a code of formulas is lifted to denote a formula of $\mathbf{PA2}^-$. For instance, the code of an implication is interpreted as an implication. All in all, the formula $\mathrm{SatGraph}(\mathcal{U}, G)$ implements an internalization of the notion of satisfiability à la Tarski (see Definition 1.1.3.6).

For every family $\mathcal{U}$, there is a unique set $G_{\mathcal{U}}$ satisfying the formula $\mathrm{SatGraph}(\mathcal{U}, G_{\mathcal{U}})$. The set $G_{\mathcal{U}}$ is defined by transfinite recursion over a well-order $\gamma^{\mathtt{F}}$ on the set of formulas.

Definition 5.1.2.3. The order $\gamma^{\mathtt{F}}$ is defined as the restriction of ω_b to the set of formulas:

$$\gamma^{\mathtt{F}} \triangleq \omega_b | \mathtt{Form} \quad \text{(where } \omega_b \text{ is the well-order on the set } \mathbb{B} \text{ defined in Example 4.1.2.1).}$$

In particular, the relation $\gamma^{\mathtt{F}}$ is a well-order.

The fundamental property of $\gamma^{\mathtt{F}}$ is that it respects the order of the subformulas. In other words, if ϕ is a subformula of ψ, then $\ulcorner \phi \urcorner \leq_{\gamma^{\mathtt{F}}} \ulcorner \psi \urcorner$.

Theorem 5.1.2.1. *Given a family $\mathcal{U}$, there is a unique set $G_{\mathcal{U}}$ such that* $\mathrm{SatGraph}(\mathcal{U}, G_{\mathcal{U}})$.

1. $\mathbf{ACA}_{\mathbf{0}}^- + \mathrm{Fam}(\mathcal{U}) + \mathrm{SatGraph}(\mathcal{U}, G) + \mathrm{SatGraph}(\mathcal{U}, G') \ \vdash\ G = G'$
2. $\mathbf{\Pi}^{\mathbf{1}}_{\mathbf{1}}\text{-}\mathbf{CA}_{\mathbf{0}}^- + \mathrm{Fam}(\mathcal{U}) \ \vdash\ \exists G\ \mathrm{SatGraph}(\mathcal{U}, G)$

Proof. 1. It is enough to show:

$$(\forall f \in \mathtt{Form})(\forall \rho_1 \in \mathtt{Val}^1)(\forall \rho_2 \in \mathtt{Val}^2)(\langle f, \rho_1, \rho_2 \rangle \in G \Leftrightarrow \langle f, \rho_1, \rho_2 \rangle \in G').$$

It is done by induction over $\gamma^{\mathtt{F}}$.

2. We recursively define a family $G_{\mathcal{U}}$ such that $\mathrm{SatGraph}(\mathcal{U}, G_{\mathcal{U}})$. We consider the formula $\mathcal{H}(X_1, X_2, Y, \mathcal{U}, \mathbb{B})$ with parameters $\mathcal{U}$ and $\mathbb{B}$:

$$
\begin{array}{l}
(\forall n \in \mathbb{N})(\forall t \in \mathtt{Term})(\forall u \in \mathtt{Term})(\forall f \in \mathtt{Form})(\forall g \in \mathtt{Form}) \\
[\neg (X_1 \subseteq^{s.i.} \gamma^{\mathtt{F}}) \Rightarrow Y = \emptyset \quad \wedge \\
X_1 = \gamma^{\mathtt{F}}_{<^{\ulcorner} \bot^{\urcorner}} \Rightarrow Y = \emptyset \quad \wedge \\
X_1 = \gamma^{\mathtt{F}}_{<t^{\ulcorner} = \urcorner u} \Rightarrow \\
\quad \mathrm{Y} = \{\langle \rho_1, \rho_2 \rangle \mid \rho_1 \in \mathtt{Val}^1 \wedge \rho_2 \in \mathtt{Val}^2 \wedge \exists \mathrm{x} (\mathrm{EvalTerm}(\mathrm{t}, \rho_1, \mathrm{x}) \wedge \mathrm{EvalTerm}(\mathrm{u}, \rho_1, \mathrm{x}))\} \quad \wedge \\
X_1 = \gamma^{\mathtt{F}}_{t^{\ulcorner} \in \urcorner n} \Rightarrow \\
\quad \mathrm{Y} = \{\langle \rho_1, \rho_2 \rangle \mid \rho_1 \in \mathtt{Val}^1 \wedge \rho_2 \in \mathtt{Val}^2 \wedge \exists \mathrm{x} (\mathrm{EvalTerm}(\mathrm{t}, \rho_1, \mathrm{x}) \wedge \mathrm{x} \in \mathcal{U}[\mathtt{find}(\rho_2, \mathrm{n})])\} \quad \wedge \\
X_1 = \gamma^{\mathtt{F}}_{<f^{\ulcorner} \Rightarrow \urcorner g} \Rightarrow Y = \{\langle \rho_1, \rho_2 \rangle \mid \rho_1 \in \mathtt{Val}^1 \wedge \rho_2 \in \mathtt{Val}^2 \wedge (\langle \rho_1, \rho_2 \rangle \in X_2[f] \Rightarrow \langle \rho_1, \rho_2 \rangle \in X_2[g])\} \quad \wedge \\
X_1 = \gamma^{\mathtt{F}}_{\ulcorner \forall^1 \urcorner n f} \Rightarrow Y = \{\langle \rho_1, \rho_2 \rangle \mid \rho_1 \in \mathtt{Val}^1 \wedge \rho_2 \in \mathtt{Val}^2 \wedge (\forall x \in \mathbb{B}) \langle \langle n, x \rangle, \rho_1 \rangle, \rho_2 \rangle \in X_2[f]\} \quad \wedge \\
X_1 = \gamma^{\mathtt{F}}_{<\ulcorner \forall^2 \urcorner n f} \Rightarrow Y = \{\langle \rho_1, \rho_2 \rangle \mid \rho_1 \in \mathtt{Val}^1 \wedge \rho_2 \in \mathtt{Val}^2 \wedge \forall x\ \langle \rho_1, \langle n, x \rangle, \rho_2 \rangle \rangle \in X_2[f]\}]
\end{array}
$$

The formula $\mathcal{H}$ denotes a total functional relation with two arguments. Therefore, Theorem 4.2.3.1 implies the existence of a unique family $\mathcal{V}$ which is $(\gamma^{\mathtt{F}}, \mathcal{H})$-inductive. Intuitively, for a formula ϕ, the slice $\mathcal{V}[\ulcorner\phi\urcorner]$ contains all the valuations ρ_1 and ρ_2 making "$\phi[\rho_1][\rho_2]$" satisfied by the structure $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$.

The family $\mathcal{V}$ has the following property: $\mathrm{Dom}_{\mathcal{V}} \subseteq \mathrm{E}_{\gamma^{\mathtt{F}}} \wedge (\forall \mathrm{f} \in \mathtt{Form})\ \mathcal{H}(\gamma^{\mathtt{F}}_{<\mathrm{f}}, \mathcal{V}|_{\mathrm{E}_{\gamma^{\mathtt{F}}_{<\mathrm{f}}}}, \mathcal{V}[\mathrm{f}])$.

We can then show by induction on $\gamma^{\mathtt{F}}$ that $\mathrm{SatGraph}(\mathcal{U}, \mathcal{V})$.

Finally, the formula $\mathcal{H}$ is arithmetical, with only $\mathcal{U}$ and $\mathbb{B}$ as parameters. Indeed, all the sets $\mathtt{Form}, \mathtt{Term}, \mathbb{N}, \gamma^{\mathtt{F}}, \mathrm{Val}^1, \mathrm{Val}^2$ can be rewritten using an arithmetical definition (with $\mathbb{B}$ as parameter). Therefore, all this previous reasoning can be done inside $\mathbf{\Pi^1_1}\text{-}\mathbf{CA}^-_{\mathbf{0}}$.

□

As a consequence, the formula $\mathrm{SatGraph}(\mathcal{U}, X)$ describes a functional relation, mapping a family $\mathcal{U}$ to the only set $G_{\mathcal{U}}$ satisfying $\mathrm{SatGraph}(\mathcal{U}, G_{\mathcal{U}})$. This functional relation will also be written $\mathrm{Sat} : \mathrm{Fam} \to \mathrm{Fam}$. Therefore, $\mathrm{Sat}(\mathcal{U})$ denotes $G_{\mathcal{U}}$. In other words, $\mathrm{Sat}(\mathcal{U})$ denotes the only set $G_{\mathcal{U}}$ such that $\mathrm{SatGraph}(\mathcal{U}, G_{\mathcal{U}})$. This set can be defined with the Π^1_1-formula:

$$x \in \mathrm{Sat}(\mathcal{U}) \triangleq \forall G(\mathrm{SatGraph}(\mathcal{U}, G) \Rightarrow x \in G).$$

However, it also has a Σ^1_1-definition:

$$\mathbf{\Pi^1_1}\text{-}\mathbf{CA}^-_{\mathbf{0}} + \mathcal{U} \in \mathrm{Fam} \vdash \forall x(x \in \mathrm{Sat}(\mathcal{U}) \Leftrightarrow \exists G(\mathrm{SatGraph}(\mathcal{U}, G) \wedge x \in G))$$

Consequently, $\mathrm{Sat}(\mathcal{U})$ is a Δ^1_1-set.

Internalization of satisfiability

The next theorem shows that we successfully internalized the notion of satisfiability (inside a structure encoded by a family of sets).

Theorem 5.1.2.2. *Given a family $\mathcal{U}$, for every formula $\phi(x_1, ..., x_p, X_1, ..., X_q)$ of free variables among $x_1, ...x_p, X_1, ..., X_q$ and without parameters:*

$$\begin{aligned}\mathbf{\Pi^1_1}\text{-}\mathbf{CA}^-_{\mathbf{0}} + \mathrm{Fam}(\mathcal{U}) \vdash\ & (\forall \rho_1 \in \mathtt{Val}^1)(\forall \rho_2 \in \mathtt{Val}^2) \\ & (\langle \ulcorner\phi\urcorner, \rho_1, \rho_2\rangle \in \mathrm{Sat}(\mathcal{U}) \Leftrightarrow \\ & \phi^{\mathbf{S}(U)_{\mathcal{M}}}[\mathrm{x}_1 := \mathtt{find}(\rho_1, \sharp \mathrm{x}_1); ...; \mathrm{x}_\mathrm{p} := \mathtt{find}(\rho_1, \sharp \mathrm{x}_\mathrm{p}); \mathrm{X}_1 := \mathcal{U}[\mathtt{find}(\rho_2, \sharp \mathrm{X}_1)]; ...; \mathrm{X}_\mathrm{q} := \mathcal{U}[\mathtt{find}(\rho_2, \sharp \mathrm{X}_\mathrm{q})]])\end{aligned}$$

In particular, if ϕ is a closed formula:

$$\mathbf{\Pi^1_1}\text{-}\mathbf{CA_0} + \mathcal{U} \in \mathrm{Fam} \vdash \langle \ulcorner\phi\urcorner, [\ ], [\ ]\rangle \in \mathrm{Sat}(\mathcal{U}) \Leftrightarrow \phi^{\mathbf{S}(\mathcal{U})_{\mathcal{M}}}$$

Proof. The proof is done by external induction on the formula ϕ:

1. Case $\phi \triangleq \bot$. It follows from $\langle \ulcorner\bot\urcorner, \rho_1, \rho_2\rangle \notin \mathrm{Sat}(\mathcal{U})$.
2. Case $\phi \triangleq t = u$. It follows from Proposition 5.1.1.1 relating codes of terms and computations: for a term $v(x_1, ..., x_p)$, the only individual that satisfies $\mathrm{EvalTerm}(t, \rho, v)$ is $v[x_1 := \mathtt{find}(\rho, \sharp x_1), ..., x_p := \mathtt{find}(\rho, \sharp x_p)]$.
3. Case $\phi \triangleq t \in X$. Again, it follows from Proposition 5.1.1.1 and from the interpretation of the variable X by $\mathcal{U}[\mathtt{find}(\rho_2, \sharp X])$.
4. Case $\phi \triangleq \phi_1 \Rightarrow \phi_2$. It is deduced from the induction hypothesis over ϕ_1 and ϕ_2.

5. Case $\phi \triangleq \forall x\psi$. Recall that

$$\langle \ulcorner \forall x\psi \urcorner, \rho_1, \rho_2\rangle \in \mathrm{Sat}(\mathcal{U}) \text{ if and only if } (\forall y \in \mathbb{B})\langle \ulcorner \psi \urcorner, \langle \sharp x, y\rangle :: \rho_1, \rho_2\rangle \in \mathrm{Sat}(\mathcal{U}).$$

If $y \in \mathbb{B}$, then $\langle \sharp x, y\rangle :: \rho_1 \in \mathtt{Val}^1$. The relativization of $\forall x\psi$ to the structure $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$ is

$$(\forall x\psi)^{\mathbf{S}(\mathcal{U})_{\mathcal{M}}} \triangleq (\forall x \in \mathbb{B})\psi^{\mathbf{S}(\mathcal{U})_{\mathcal{M}}}.$$

Thus, it is possible to conclude from the induction hypothesis.

6. Case $\phi \triangleq \forall X\psi$. Recall that

$$\langle \ulcorner \forall X\psi \urcorner, \rho_1, \rho_2\rangle \in \mathrm{Sat}(\mathcal{U}) \text{ if and only if } \forall z\langle \ulcorner \psi \urcorner, \rho_1, \langle \sharp X, z\rangle :: \rho_2\rangle \in \mathrm{Sat}(\mathcal{U}).$$

We consider the relativization of $\forall X\psi$ to the structure $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$:

$$\begin{aligned}(\forall X\psi)^{\mathbf{S}(\mathcal{U})_{\mathcal{M}}} &\triangleq \forall X(X \in \mathbf{S}(\mathcal{U}) \Rightarrow \psi^{\mathbf{S}(\mathcal{U})_{\mathcal{M}}})\\ &\Leftrightarrow \forall X\forall z(X = \mathcal{U}[z] \Rightarrow \psi^{\mathbf{S}(\mathcal{U})_{\mathcal{M}}})\\ &\Leftrightarrow \forall z\psi^{\mathbf{S}(\mathcal{U})_{\mathcal{M}}}[X := \mathcal{U}[z]].\end{aligned}$$

For all second-order valuations $\rho_2 \in \mathtt{Val}^2, \mathtt{find}(\langle \sharp X, z\rangle :: \rho_2, \sharp X) = z$. From these remarks, the induction hypothesis allows us to conclude.

□

5.1.3 Families as structures of the language of PA2$^-$

Internalization of the notion of models of a theory

A family $\mathcal{U}$ encodes a class of sets that can be used as a structure of the language of **SOL**. In particular, the notion of satisfiability for the structure $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$ was internalized in the last section. Therefore, we can then internally describe when such a structure models a theory.

Definition 5.1.3.1. Let $\mathcal{F}$ be a set of codes of formulas. The structure $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$ models $\mathcal{F}$ if

$$\mathrm{Mod}_{\mathcal{F}}(\mathcal{U}) \triangleq (\forall f \in \mathcal{F})(\forall \rho_1 \in \mathtt{Val}^1)(\forall \rho_2 \in \mathtt{Val}^2)\langle f, \rho_1, \rho_2\rangle \in \mathrm{Sat}(\mathcal{U}).$$

In this case, we may simply say that the family $\mathcal{U}$ models $\mathcal{F}$.

In particular, we can express the fact that a family $\mathcal{U}$ (seen as a structure) models subsystems of second-order arithmetic. We note that:

1. The codes of the first-order axioms of **PA2** are always in $\mathrm{Sat}(\mathcal{U})$.
2. Because first-order quantifiers are relativized to $\mathbb{B}$ when a formula is interpreted in $\mathbf{S}(U)_{\mathcal{M}}$, the code of the axiom of induction is in $\mathrm{Sat}(\mathcal{U})$.

Consequently, a structure $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$ always models the axioms of injectivity, of non confusion, of computation and of induction. It remains to deal with the axiom scheme of comprehension. We study the cases of full comprehension and of arithmetical comprehension. First, we can

express these schemes as sets described by a formula. For a formula ϕ, the axiom scheme of comprehension is written as follow:

$$\begin{aligned}\exists X\forall x(x \in X \Leftrightarrow \Phi) &\triangleq \neg\forall X\neg\forall x((x \in X \Rightarrow \phi) \wedge (\phi \Rightarrow x \in X)) \\ &\triangleq \forall X\Big(\forall x(((x \in X \Rightarrow \Phi) \Rightarrow (\Phi \Rightarrow x \in X) \Rightarrow \bot) \Rightarrow \bot) \Rightarrow \bot\Big) \Rightarrow \bot\end{aligned}$$

We fix a first-order variable x of code $\sharp x$ and a second-order variable X of code $\sharp X$ and we introduce a term $\mathrm{c}(f)$ that represents a code for the previous formula:

$$\mathrm{c}(f) \triangleq \langle 2, \langle 4, \sharp X, \langle 2, \langle 3, \sharp x, \langle 2, \langle 2, \langle 2, \langle 1, \sharp x, \sharp X\rangle, f\rangle, \langle 2, \langle 2, f, \langle 1, \sharp x, \sharp X\rangle\rangle, 0\rangle, 0\rangle, 0\rangle, 0\rangle$$

The point of this computation is to show that the axiom scheme of comprehension is indeed recursive[4]. Finally, we consider the sets:

$$\begin{aligned}\ulcorner\mathrm{CA}\urcorner &\triangleq \{\mathrm{c}(f) \mid f \in \mathtt{Form} \wedge \sharp X \notin \mathtt{FV}^2[f]\} \\ \ulcorner\mathrm{ACA}\urcorner &\triangleq \{\mathrm{c}(f) \mid f \in \mathtt{FormArith} \wedge \sharp X \notin \mathtt{FV}^2[f]\}\end{aligned}$$

We say that $\mathcal{U}$

1. models **PA2** if $\mathrm{Mod}_{\ulcorner\mathrm{CA}\urcorner}(\mathcal{U})$
2. models $\mathbf{ACA_0}$ if $\mathrm{Mod}_{\ulcorner\mathrm{ACA}\urcorner}(\mathcal{U})$.

In particular, if $\mathcal{U}$ models $\mathbf{ACA_0}$, $\mathbf{S}(\mathcal{U})$ is stable under all boolean operations over sets.

An application: Tarski's theorem

We finish this section with a formalization of Tarski's Theorem saying that there is no first-order definition of a truth predicate for arithmetical formulas. As a consequence, we will deduce that the logical complexity of the formula $x \in \mathrm{Sat}(\mathcal{U})$ cannot be arithmetical and, therefore, that the definition we use has the least possible logical complexity (because it is Δ^1_1 in $\mathbf{\Pi^1_1}\text{-}\mathbf{CA_0^-}$).

Theorem 5.1.3.1 (Tarski)**.** *If $\mathbf{\Pi^1_1}\text{-}\mathbf{CA_0}$ is consistent, the formula $x \in \mathrm{Sat}(\mathcal{U})$ is not equivalent to an arithmetical formula (in* **PA2***).*

Proof. We work in $\mathbf{\Pi^1_1}\text{-}\mathbf{CA_0}$ and, consequently, with the axiom of induction. Note that Theorem 5.1.2.2 is still valid. Assume $x \in \mathrm{Sat}(\mathcal{U})$ is equivalent to an arithmetical formula $\psi(x,\mathcal{U})$ (without parameters), i.e.:

$$\forall x\forall\mathcal{U}(x \in \mathrm{Sat}(\mathcal{U}) \Leftrightarrow \psi(x,\mathcal{U})).$$

We use as structure the set[5] $\emptyset \triangleq \{x \mid \bot\}$ and consider the formula $\phi(x) \triangleq \psi[\mathcal{U} := \emptyset]$ that characterizes the satisfiability in the structure $\emptyset_{\mathcal{M}}$. Consider the formula $\mathrm{Diag}(y) \triangleq \neg\phi(\langle y, [\langle\sharp x, y\rangle], []\rangle)$. Then, for the code $\ulcorner\mathrm{Diag}\urcorner \in \mathbb{B}$ of the formula $\mathrm{Diag}(y)$:

$$\begin{aligned}\neg\phi(\langle\ulcorner\mathrm{Diag}\urcorner, [\langle\sharp x, \ulcorner Diag\urcorner\rangle], []\rangle) \triangleq \mathrm{Diag}[y := \ulcorner\mathrm{Diag}\urcorner] &\Leftrightarrow \mathrm{Diag}^{\mathbf{S}(\emptyset)_{\mathcal{M}}}[y := \ulcorner\mathrm{Diag}\urcorner] \\ &\Leftrightarrow \langle\ulcorner\mathrm{Diag}\urcorner, [\langle\sharp x, \ulcorner Diag\urcorner\rangle], []\rangle \in \mathrm{Sat}(\emptyset) \\ &\Leftrightarrow \phi(\langle\ulcorner\mathrm{Diag}\urcorner, [\langle\sharp x, \ulcorner Diag\urcorner\rangle], []\rangle)\end{aligned}$$

where

[4] I mean that the codes of all the formulas in this scheme can be enumerated by a (primitive) recursive function.

[5] Because the second-order part of the structure won't be used, any sets will work. However, it needs to be defined by an arithmetical formula so that it can be replaced by its definition without increasing the logical complexity of ϕ. Also note that any arithmetical sets can be used in ψ. For instance, the set $\mathbb{B}$ is arithmetical in presence of the axiom of induction and therefore can be used in ψ.

1. The first equivalence is because the formula $\mathrm{Diag}(x)$ is arithmetical and therefore the relativization to the structure $\mathbf{S}(\emptyset)$ simplifies: $\mathrm{Diag}^{\mathbf{S}(\emptyset)_{\mathcal{M}}}(x) \triangleq \mathrm{Diag}^{\mathbb{B}}(x) \Leftrightarrow \mathrm{Diag}(x)$ in presence of the axiom of induction.

2. The second equivalence is a consequence of Theorem 5.1.2.2.

3. The last equivalence is the hypothesis that $\phi(x)$ is a definition of the set $\mathrm{Sat}(\mathcal{U})$.

This leads to an inconsistency (as shown in Remark 1.2.1.1). □

Remark 5.1.3.1. The last proof is done in $\mathbf{\Pi^1_1}$-$\mathbf{CA_0}$ as the axiom of induction is used. This is due to the fact that the structures $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$ have a standard first-order part. In particular, for every arithmetical formula ϕ, $\mathbf{S}(\mathcal{U})_{\mathcal{M}} \Vdash \phi$ if and only if $\phi^{\mathbb{B}}$. However, the hypothesis "$\mathbf{\Pi^1_1}$-$\mathbf{CA_0}$ is coherent" is not stronger than the hypothesis "$\mathbf{\Pi^1_1}$-$\mathbf{CA_0^-}$ is coherent" because we prove a result of relative consistency between these theories (see Remark 2.2.3.1).

5.1.4 The operator $\mathbf{Def^2}$

Operator of definability

We define an operator on the families of sets that sends a family $\mathcal{U}$ into the family $\mathbf{Def^2}(\mathcal{U})$ of sets definable in $\mathcal{U}$ by a second-order formula.

Definition 5.1.4.1. Given a family $\mathcal{U}$, we define the family $\mathbf{Def^2}(\mathcal{U})$ as follow:

$$\mathbf{Def^2}(\mathcal{U}) \triangleq \{\langle\langle f, \rho_1, \rho_2\rangle, x\rangle \mid \langle f, \langle 0, x\rangle{::}\rho_1, \rho_2\rangle \in Sat(\mathcal{U})\}.$$

The sets in the slices of $\mathbf{Def^2}(\mathcal{U})$ are the one definable in $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$ by a second-order formula. An index of a set X in $\mathbf{Def^2}(\mathcal{U})$ is a definition of X in $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$. Precisely, it is an individual of the form $\langle f, \rho_1, \rho_2\rangle$ satisfying:

$$\forall x(x \in X \Leftrightarrow \langle f, \langle\langle 0, x\rangle, \rho_1\rangle, \rho_2\rangle \in \mathrm{Sat}(\mathcal{U})).$$

Therefore, using Theorem 5.1.2.2, to show that a set is in $\mathbf{S}(\mathbf{Def^2}(\mathcal{U}))$, it is enough to find a definition of this set that relativizes to $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$.

Remark 5.1.4.1. Note that the meta-variable of code 0 plays a particular role. Say that it is the variable z, i.e. $\sharp z = 0$. Then, to show that a set is in $\mathbf{S}(\mathbf{Def^2}(\mathcal{U}))$, we need to show that it can be defined as $\{z \mid z \in \mathbb{B} \wedge \phi(z)^{\mathbf{S}(\mathcal{U})}\}$.

Example 5.1.4.1. For all families $\mathcal{U}$, the set of pure binary trees ($\mathbb{B}$) is in the slices of $\mathbf{Def^2}(\mathcal{U})$. A definition of this set that relativizes to $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$ is $\{z \mid z = z\}$. It means that the individuals of $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$ are exactly the pure binary trees (defined as the elements of the set $\mathbb{B}$).

Properties of this operator

We show some properties of this operators that we will use in our study of the Ramified Analytic Hierarchy.

Lemma 5.1.4.1. *The class* $\mathbf{S}(\mathcal{U})$ *is included in* $\mathbf{S}(\mathbf{Def^2}(\mathcal{U}))$.

$$\mathbf{\Pi^1_1}\text{-}\mathbf{CA_0^-} + \mathrm{Fam}(\mathcal{U}) \vdash \forall X(X \in \mathbf{S}(\mathcal{U}) \Rightarrow X \in \mathbf{S}(\mathbf{Def^2}(\mathcal{U}))).$$

Proof. Let v be such that $X = \mathcal{U}[v]$. Then $X = \mathbf{Def^2}(\mathcal{U})[\langle\langle 0, 0\rangle^{\ulcorner} \in \ulcorner 0, [\,], [\langle 0, v\rangle]\rangle]$. □

We will now show that the operator $\mathbf{Def^2}(X)$ is well-defined up to the equality over families of sets (see Proposition 5.1.4.1). The proof of this fact uses the following technical lemma.

Lemma 5.1.4.2. *If two families $\mathcal{U}$ and $\mathcal{V}$ contain the same sets, if f is a code of a formula and if ρ_2 (seen as a valuation for $\mathbf{S}(\mathcal{U})$) and ρ_2' (seen as a valuation for $\mathbf{S}(\mathcal{V})$) assigns the same set to the free second-order variables of f, then $\mathbf{Def^2}(\mathcal{U})[\langle f, \rho_1, \rho_2\rangle] = \mathbf{Def^2}(\mathcal{V})[\langle f, \rho_1, \rho_2'\rangle]$ for every first-order valuation ρ_1.*

$$\mathbf{\Pi^1_1\text{-}CA^-_0} + \mathrm{Fam}(\mathcal{U}) \vdash \mathbf{S}(\mathcal{U}) = \mathbf{S}(\mathcal{V}) \Rightarrow (\forall f \in \mathtt{Form})(\forall \rho_1 \in \mathtt{Val}^1)(\forall \rho_2 \in \mathtt{Val}^2)(\forall \rho_2' \in \mathtt{Val}^2)$$
$$(\forall n \in \mathtt{FV}^2[f])(\mathcal{U}[\mathtt{find}(\rho_2, n)] = \mathcal{V}[\mathtt{find}(\rho_2', n)]) \Rightarrow \mathbf{Def^2}(\mathcal{U})[\langle f, \rho_1, \rho_2'\rangle] = \mathbf{Def^2}(\mathcal{V})[\langle f, \rho_1, \rho_2'\rangle]$$

Proof. By induction over the well-order $\gamma^{\mathtt{F}}$ (i.e. by internal induction over the set of formulas).

1. The case $f \triangleq t^{\ulcorner} \in \urcorner \sharp X$ uses the hypothesis $\mathcal{U}[\mathtt{find}(\rho_2, \sharp X)] = \mathcal{V}[\mathtt{find}(\rho_2', \sharp X)]$.
2. The case $f \triangleq \ulcorner \forall^2 \urcorner \sharp X g$ uses the hypothesis $\mathbf{S}(\mathcal{U}) = \mathbf{S}(\mathcal{V})$.

□

Proposition 5.1.4.1. *The operator $\mathbf{Def^2}(X)$ is well-defined: if two families $\mathcal{U}$ and $\mathcal{V}$ contain the same sets, then $\mathbf{S}(\mathbf{Def^2}(\mathcal{U}))$ and $\mathbf{S}(\mathbf{Def^2}(\mathcal{V}))$ also contain the same sets.*

$$\mathbf{\Pi^1_1\text{-}CA^-_0} + \mathrm{Fam}(\mathcal{U}) + \mathrm{Fam}(\mathcal{V}) \vdash \mathbf{S}(\mathcal{U}) = \mathbf{S}(\mathcal{V}) \Rightarrow \mathbf{S}(\mathbf{Def^2}(\mathcal{U})) = \mathbf{S}(\mathbf{Def^2}(\mathcal{V}))$$

Proof. We show $\mathbf{S}(\mathbf{Def^2}(\mathcal{U}) \subseteq \mathbf{S}(\mathbf{Def^2}(\mathcal{V}))$. Assume $X = \mathbf{Def^2}(\mathcal{U})[\langle f, \rho_1, \rho_2\rangle]$. Define a valuation ρ_2' such that:

$$(\forall n \in \mathtt{FV}^2(f))\ \mathcal{U}[\mathtt{find}(\rho_2, n)] = \mathcal{V}[\mathtt{find}(\rho_2', n)].$$

We can then conclude using the previous point. □

Families in the image of the operator $X \mapsto \mathbf{Def^2}(X)$ satisfies "closures properties". In particular, they are closed under all arithmetical definitions and, therefore, they (internally) model the axiom scheme of arithmetical comprehension.

Lemma 5.1.4.3. *For every family $\mathcal{U}$, the structure $\mathbf{S}(\mathbf{Def^2}(\mathcal{U}))_{\mathcal{M}}$ models the axiom scheme of arithmetical comprehension (viewed in a theory with induction).*

$$\mathbf{\Pi^1_1\text{-}CA^-_0} + \mathrm{Fam}(\mathcal{U}) \vdash \mathrm{Mod}_{\ulcorner \mathrm{ACA} \urcorner}(\mathbf{Def^2}(\mathcal{U}))$$

Proof. An arithmetical definition relativizes to $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$. Therefore, the structure $\mathbf{S}(\mathbf{Def^2}(\mathcal{U}))_{\mathcal{M}}$ contains all the arithmetical sets with second-order parameters taken from its slices. □

Lemma 5.1.4.4. *A family $\mathcal{U}$ models $\ulcorner \mathrm{CA} \urcorner$ if and only if it is a fixed-point for the operator $\mathbf{Def^2}(x)$.*

$$\mathbf{\Pi^1_1\text{-}CA^-_0} + \mathrm{Fam}(\mathcal{U}) \vdash \mathrm{Mod}_{\ulcorner \mathrm{CA} \urcorner}(\mathcal{U}) \Leftrightarrow \mathbf{S}(\mathcal{U}) = \mathbf{S}(\mathbf{Def^2}(\mathcal{U}))$$

Proof. 1. Assume that $\mathcal{U}$ models $\ulcorner \mathrm{CA} \urcorner$ and $X = \mathbf{Def^2}(\mathcal{U})[\langle f, \rho_1, \rho_2\rangle]$, i.e.:

$$\forall x\ (x \in X \Leftrightarrow \langle f, \langle\langle 0, x\rangle, \rho_1\rangle, \rho_2\rangle \in \mathrm{Sat}(\mathcal{U})).$$

Because $\mathbf{S}(\mathcal{U})_{\mathcal{M}}$ models $\ulcorner \mathrm{CA} \urcorner$, it satisfies $\langle \mathrm{c}(f), \rho_1, \rho_2\rangle \in \mathrm{Sat}(\mathcal{U})$ and it implies that $X \in \mathbf{S}(\mathcal{U})$.

2. If $\mathcal{U}$ is a fixed-point for the operator $X \mapsto \mathbf{Def^2}(X)$, the family $\mathbf{S}(\mathcal{U})$ is closed under all set definitions and, therefore, satisfies the axiom scheme of comprehension.

□

5.2 The ramified analytic hierarchy

This section is devoted to the construction of the ramified analytic hierarchy. Recall that the ramified analytic hierarchy is the transfinite sequence RAH_α indexed by well-preorders and satisfying

$$\mathrm{RAH}_\alpha = \bigcup_{\beta<\alpha} \mathbf{Def^2}(\mathrm{RAH}_\beta).$$

We interpret each stage of this sequence as a family of sets and we will study the sets contained in $\mathbf{S}(\mathrm{RAH}_\alpha)$. The ramified analytic hierarchy is then the union of the classes $\mathbf{S}(\mathrm{RAH}_\alpha)$ (Part 5.2.2).

We define the sequence $(\mathrm{RAH}_\alpha)_{\alpha\in\mathrm{WPO}}$ in the Subsection 5.2.1. In particular, we do not define it as the transfinite union of the operator $\mathbf{Def^2}$. We use a definition that highlights a precise characterization of the elements of RAH_α (Subsection 5.2.3). This characterization will be used to define a well-order on the class $\mathbf{S}(\mathrm{RAH})$ (see Section 5.3)

5.2.1 A definition of RAH

We define the following total functional relation on the class of well-preorders.

$$\begin{array}{l}
\mathrm{RAH}(X_1, X_2, Y) \triangleq \\
\quad X_1 \notin \mathrm{PreOrd} \Rightarrow Y = \emptyset \quad \wedge \\
\quad X_1 = \emptyset \Rightarrow Y = \emptyset \quad \wedge \\
\quad (X_1 \in \mathrm{PreOrd} \wedge X_1 \in \mathrm{Succ}) \Rightarrow \\
\qquad Y = \bigcup_{m\in \mathrm{M}_{X_1}} X_2[m] \cup \{\langle\langle m, f, \rho_1, \rho_2\rangle, x\rangle \mid \langle\langle f, \rho_1, \rho_2\rangle, x\rangle \in \mathbf{Def^2}(X_2[m])\} \quad \wedge \\
\quad (X_1 \in \mathrm{PreOrd} \wedge X_1 \in \mathrm{Lim}) \Rightarrow Y = \bigcup_{i\in E_{X_1}} X_2[i]
\end{array}$$

The definition is made by case, in particular:

1. If X_1 is a successor, Y is obtained as an union indexed by the set of the maximal elements of X_1 (denoted M_{X_1}, see Definition 4.1.2.1). Note that if m and m' are maximal elements of α and if X_2 is constructed by induction along the formula RAH, then $X_2[m]$ and $X_2[m']$ will in fact be equal. However, this formulation of $\mathrm{RAH}(X_1, X_2, Y)$ allows us to show directly that this formula denotes a total functional relation.
2. If X_1 is limit, Y is obtained as the union of all the previously constructed objects.
3. In all the other cases, Y is empty.

RAH is a total functional relation with two arguments. Theorem 4.2.3.1 implies that

$$\mathbf{PA2^-} + \mathrm{WPO}(\alpha) \vdash \exists!\mathcal{U}\exists!R(\mathrm{Ind}_{(\alpha,\mathrm{RAH})}(\mathcal{U}) \wedge \mathrm{RAH}(\alpha, \mathcal{U}, R))$$

Definition 5.2.1.1. We follow the notations introduced for the $(\alpha, \mathcal{H})$-inductive families (see Notation 4.2.3.1). In particular, for all well-preorders α, RAH_α is the only set satisfying $\exists\mathcal{U}\exists X\ \mathrm{RAH}(\alpha, \mathcal{U}, X)$. In particular:

$$x \in RAH_\alpha \triangleq \exists\mathcal{U}\exists R(\mathrm{Ind}_{(\alpha,\mathrm{RAH})}(\mathcal{U}) \wedge \mathrm{RAH}(\alpha, \mathcal{U}, R) \wedge x \in R).$$

Remark 5.2.1.1. The existential quantifiers in the formula RAH_α can be replaced by universal quantifiers:

$$\mathbf{PA2^-} + \mathrm{WPO}(\alpha) \vdash \forall x(x \in \mathrm{RAH}_\alpha \Leftrightarrow \forall\mathcal{U}\forall R(\mathrm{Ind}_{(\alpha,\mathrm{RAH})}(\mathcal{U}) \Rightarrow \mathrm{RAH}(\alpha, \mathcal{U}, R) \Rightarrow x \in R))$$

The definition of RAH_α can be seen to be Δ^1_1 (but we will not use this fact).

Remark 5.2.1.2. In set theory, the set L_α representing the level α of the constructible universe can be defined as the following union:

$$\mathrm{L}_\alpha = \bigcup_{\beta<\alpha} \mathbf{Def}(\mathrm{L}_\beta).$$

Such a direct definition is not possible in $\mathbf{PA2}^-$. The set RAH_α needs to be defined as a disjoint union of the previously constructed objects. This is the reason why the elements of RAH_α are indexed by the field of the well-preorder α. Our definition of the functional $\mathrm{RAH}(X_1, X_2, Y)$ conveys this intuition (although it may look cumbersome).

5.2.2 RAH_α as a family and RAH as a class

The set RAH_α encodes a family (i.e. $\mathrm{Fam}(\mathrm{RAH}_\alpha)$ is provable) because it contains only elements of the shape $\langle x, y\rangle$. Given a well-preorder α, the formula $X \in \mathbf{S}(\mathrm{RAH}_\alpha)$ expresses that X is in the class encoded by RAH_α:

$$\begin{aligned} X \in \mathbf{S}(\mathrm{RAH}_\alpha) &\triangleq \exists s \forall x (x \in X \Leftrightarrow x \in \mathrm{RAH}_\alpha[s]) \\ &\Leftrightarrow \exists \mathcal{U} \exists R \exists s \forall x (\mathrm{Ind}_{\alpha,\mathrm{RAH}}(\mathcal{U}) \wedge \mathrm{RAH}(\alpha, \mathcal{U}, R) \wedge (x \in X \Leftrightarrow x \in R[s])) \\ &\Leftrightarrow \forall \mathcal{U} \forall R \exists s \forall x (\mathrm{Ind}_{\alpha,\mathrm{RAH}}(\mathcal{U}) \Rightarrow \mathrm{RAH}(\alpha, \mathcal{U}, R) \Rightarrow (x \in X \Leftrightarrow x \in R[s])) \end{aligned}$$

We can then consider the class obtained by the (meta-theoretic) union of all the classes $\mathbf{S}(\mathrm{RAH}_\alpha)$. For this purpose, we introduce the new notation "$X \in \mathbf{S}(\mathrm{RAH})$" to express that the set X is in some $\mathbf{S}(\mathrm{RAH}_\alpha)$:

$$\begin{aligned} X \in \mathbf{S}(\mathrm{RAH}) &\triangleq (\exists \alpha \in \mathrm{WPO})\ X \in \mathbf{S}(\mathrm{RAH}_\alpha) \\ &\Leftrightarrow \exists \alpha \exists \mathcal{U} \exists R \exists s \forall x (\alpha \in \mathrm{WPO} \wedge \mathrm{Ind}_{\alpha,\mathrm{RAH}}(\mathcal{U}) \wedge \mathrm{RAH}(\alpha, \mathcal{U}, R) \wedge (x \in X \Leftrightarrow x \in R[s])) \\ &\Leftrightarrow \exists \alpha \forall \mathcal{U} \forall R \exists s \forall x (\alpha \in \mathrm{WPO} \wedge (\mathrm{Ind}_{\alpha,\mathrm{RAH}}(\mathcal{U}) \Rightarrow \mathrm{RAH}(\alpha, \mathcal{U}, R) \Rightarrow (x \in X \Leftrightarrow x \in R[s])))) \end{aligned}$$

Remark 5.2.2.1. RAH is not a family of sets and, consequently, the notation $\mathbf{S}(\_)$ (as we defined it in Part 4.1.1) should not apply. However, we overload it to convey the intuition that $\mathbf{S}(\mathrm{RAH})$ is obtained as the (meta-theoretic) union of all the families of the form $\mathbf{S}(\mathrm{RAH}_\alpha)$:

$$\mathbf{S}(\mathrm{RAH}) = \bigcup_{\alpha \in \mathrm{WPO}} \mathbf{S}(\mathrm{RAH}_\alpha).$$

The sequence RAH is increasing only for the relation of being an initial segments:

$$\mathrm{RAH}_\alpha \subseteq \mathrm{RAH}_\beta \qquad \text{if } \alpha \subseteq^{s.i.} \beta.$$

However, when viewed as a sequence of families, it is increasing for $\preccurlyeq_{\mathrm{WPO}}$ and it is well-defined up to the equivalence relation $\cong_{\mathrm{WPO}}$, meaning that two isomorphic well-preorders are mapped to two families containing the same sets. A special case of this result is that if $i \cong_\alpha j$, then $\mathbf{S}(\mathrm{RAH}_{\alpha_{<i}}) = \mathbf{S}(\mathrm{RAH}_{\alpha_{<j}})$. This result is proved before the more general one by an induction over α.

Lemma 5.2.2.1. *If α is a well-preorder and if $i \cong_\alpha j$, then $\mathbf{S}(\mathrm{RAH}_{\alpha_{<i}}) = \mathbf{S}(\mathrm{RAH}_{\alpha_{<j}})$.*

Proposition 5.2.2.1. *If α and β are two isomorphic well-preorders, then $\mathbf{S}(\mathrm{RAH}_\alpha) = \mathbf{S}(\mathrm{RAH}_\beta)$.*

Proof. We use Proposition 4.2.3.3 and reason by simultaneous induction over α and β. Fixing an isomorphism F between α and β, assume $(\forall i \in E_\alpha)(\forall j \in E_\beta)(i \mathcal{F} j \Rightarrow \mathbf{S}(RAH_{\alpha_{<i}}) = \mathbf{S}(RAH_{\beta_{<j}}))$. We reason by cases whether α is empty, limit or successor.

1. Case: α is empty. Both families are $\{\emptyset\}$.

2. Case: α is successor. In this case, we show that both classes are equal to $\mathbf{Def^2}(\mathrm{RAH}_{\alpha_{<m}})$ for all $m \in \mathrm{M}_\alpha$.

   (a) First, if m, m' are both in M_α, then $m \cong_\alpha m'$ (because a well-preorder is total[6]) and the previous lemma implies $\mathbf{S}(\mathrm{RAH}_{\alpha_{<m}}) = \mathbf{S}(\mathrm{RAH}_{\alpha_{<m'}})$. It is also valid for β.

   (b) Therefore, fixing $m \in \mathrm{M}_\alpha$ and $m' \in \mathrm{M}_\beta$, the induction hypothesis implies that $\mathbf{S}(\mathrm{RAH}_{\alpha_{<m}})$ and $\mathbf{S}(\mathrm{RAH}_{\beta_{<m'}})$ are equal.

   (c) Lemma 5.1.4.1 implies that $\mathbf{S}(\mathrm{RAH}_{\alpha_{<m}}) \subseteq \mathbf{Def^2}(\mathrm{RAH}_{\alpha_{<m}})$ and, as a consequence, $\mathbf{S}(\mathrm{RAH}_\alpha) = \mathbf{Def^2}(\mathrm{RAH}_{\alpha_{<m}})$.

   (d) This is also valid for β: $\mathbf{S}(\mathrm{RAH}_\beta) = \mathbf{Def^2}(\mathrm{RAH}_{\alpha_{<m'}})$.

   (e) Finally, the result follows from Proposition 5.1.4.1 saying that the operator $\mathbf{Def^2}(X)$ is well-defined.

3. Case: α is limit. It directly follows from the induction hypothesis as the two families are obtained as an union of the previously constructed families.

□

Corollary 5.2.2.1. *If $\mathcal{U}$ is a family of well-preorders, then* $\mathbf{S}(\mathrm{RAH}_{\sup(\mathcal{U})}) = \bigcup_{x \in \mathrm{Dom}_\mathcal{U}} \mathbf{S}(\mathrm{RAH}_{\mathcal{U}[x]})$.

$$\mathbf{PA2^-} + \mathrm{Fam}(\mathcal{U}) + (\forall x \in \mathrm{Dom}_\mathcal{U})\mathrm{WPO}(\mathcal{U}[x]) \vdash \forall X(X \in \mathbf{S}(\mathrm{RAH}_{\sup(\mathcal{U})}) \Leftrightarrow (\exists x \in \mathrm{Dom}_\mathcal{U})X \in \mathbf{S}(\mathrm{RAH}_{\mathcal{U}[x]}))$$

Proof. It is a consequence of the results of this subsection and of Theorem 4.2.2.1 characterizing the supremum of a family of well-preorders. □

5.2.3 Characterization of the elements of RAH

An element X of $\mathbf{S}(\mathrm{RAH}_\alpha)$ has an index of the form $\langle m, f, \rho_1, \rho_2\rangle$ where $m \in E_\alpha$ and $\langle f, \rho_1, \rho_2\rangle$ is a (second-order) definition of X in $\mathrm{RAH}_{\alpha_{<m}}$. The fact that all reals in RAH_α are indexed by an element of the field of α will be important to define a well-order on $\mathbf{S}(\mathrm{RAH}_\alpha)$.

Lemma 5.2.3.1. *All the elements of* RAH_α *have the shape* $\langle\langle l, t\rangle, x\rangle$ *for* $l \in E_\alpha$.

Proof. It is done by induction on α. □

Thus, by construction, RAH_α is "indexed" by the well-preorder α: individuals in RAH_α are indexed by the "level" at which they arrive in the hierarchy. This is the subject of the next lemma.

Lemma 5.2.3.2. *An individual* $\langle\langle l, t\rangle, x\rangle$ *is in* RAH_α *if and only if* $\langle t, x\rangle \in \mathbf{Def^2}(\mathrm{RAH}_{\alpha_{<l}})$.

$$\mathbf{PA2^-} + \mathrm{WPO}(\alpha) \vdash \ (\forall l \in E_\alpha)\forall t \forall x(\langle\langle l, t\rangle, x\rangle \in \mathrm{RAH}_\alpha \Leftrightarrow \langle t, x\rangle \in \mathbf{Def^2}(\mathrm{RAH}_{\alpha_{<l}}))$$

Proof. We show the implication ($\Rightarrow$). We reason by induction on α (i.e. we use Proposition 4.2.3.2).

1. Case: α is empty. RAH_α is empty and the result is immediate.

[6] see the comment after Definition 4.1.2.1

2. Case: α is successor. If $l \in \mathrm{M}_\alpha$ and $\langle\langle l, t\rangle, x\rangle \in \mathrm{RAH}_\alpha$. The definition of RAH implies that $\langle t, x\rangle \in \mathbf{Def^2}(\mathrm{RAH}_{\alpha_{<l}})$. If l is not maximal, it is in $\mathrm{RAH}_{\alpha_{<m}}$ for some m and the induction hypothesis gives the result.

3. Case: α is limit. It directly follows from the induction hypothesis.

□

We summarize the previous lemmas in the following theorem that gives a full characterization of the elements of RAH.

Theorem 5.2.3.1. *For a well-preorder α, all the elements of RAH_α have the shape $\langle\langle l, f, \rho_1, \rho_2\rangle, x\rangle$ and they satisfy $\langle f, \langle 0, x\rangle{::}\rho_1, \rho_2\rangle \in \mathrm{Sat}(\mathrm{RAH}_{\alpha_{<l}})$.*

1. $\mathbf{PA2}^- + \mathrm{WPO}(\alpha) \vdash (\forall y \in \mathrm{RAH}_\alpha)(\exists l \in E_\alpha)(\exists f \in \mathtt{Form})(\exists \rho_1 \in \mathtt{Val}^1)(\exists \rho_2 \in \mathtt{Val}^2)(\exists x \in \mathbb{B})$
$$y = \langle\langle l, f, \rho_1, \rho_2\rangle, x\rangle$$
2. $\mathbf{PA2}^- + \mathrm{WPO}(\alpha) \vdash (\forall l \in E_\alpha)(\forall f \in \mathtt{Form})(\forall \rho_1 \in \mathtt{Val}^1)(\forall \rho_2 \in \mathtt{Val}^2)(\forall x \in \mathbb{B})$
$$\langle\langle l, f, \rho_1, \rho_2\rangle, x\rangle \in \mathrm{RAH}_\alpha \Leftrightarrow \langle f, \langle 0, x\rangle{::}\rho_1, \rho_2\rangle \in \mathrm{Sat}(\mathrm{RAH}_{\alpha_{<l}})$$

Proof. Lemma 5.2.3.1 says that all elements of RAH_α are of the shape $\langle\langle l, t\rangle, x\rangle$ with $l \in E_\alpha$. For all individuals $l \in E_\alpha$, t and x:

$$\begin{aligned}\langle\langle l, t\rangle, x\rangle \in \mathrm{RAH}_\alpha \Leftrightarrow & \langle t, x\rangle \in \mathbf{Def^2}(\mathrm{RAH}_{\alpha_{<l}}) \\ \Leftrightarrow & (\exists f \in \mathtt{Form})(\exists \rho_1 \in \mathtt{Val}^1)(\exists \rho_2 \in \mathtt{Val}^2) \\ & (t = \langle f, \rho_1, \rho_2\rangle \wedge \langle f, \langle 0, x\rangle{::}\rho_1, \rho_2\rangle \in \mathrm{Sat}(\mathrm{RAH}_{\alpha_{<l}}))\end{aligned}$$

where the first equivalence is given by Lemma 5.2.3.2. □

This theorem shows that $\mathbf{S}(RAH_\alpha)$ was indeed obtained by iterating the operator $X \mapsto \mathbf{Def^2}(X)$ along the well-preorder α.

5.3 Well-order on $\mathbf{S}(\mathrm{RAH})$

The goal of this section is to define a well-order on $\mathbf{S}(\mathrm{RAH})$ This will be achieved in three steps:

1. For all well-preorders α, we construct a well-preorder $\mathcal{R}_\alpha$ on RAH_α (see Subsection 5.3.1).
2. Using Proposition 4.2.5.1, we deduce the existence of a well-order on $\mathbf{S}(\mathrm{RAH}_\alpha)$ (Subsection 5.3.2).
3. Finally, we construct a well-order on $\mathbf{S}(\mathrm{RAH})$ as the union of all the well-orders on $\mathbf{S}(\mathrm{RAH}_\alpha)$. (see Subsection 5.3.3).

5.3.1 Well-preorder on RAH_α

We consider the following functional relation:

$$\begin{aligned}&\mathcal{R}(X_1, X_2, Y) \triangleq \\ &\quad X_1 \notin \mathrm{PreOrd} \Rightarrow Y = \emptyset \quad \wedge \\ &\quad X_1 = \emptyset \Rightarrow Y = \{\langle 0, 0\rangle\} \quad \wedge \\ &\quad X_1 \in \mathrm{PreOrd} \wedge X_1 \in \mathrm{Succ} \Rightarrow \\ &\qquad Y = \bigcup_{m \in \mathrm{M}_{X_1}} (X_2[m] \uplus (\bigcup_{m_1, m_2 \in \mathrm{M}_{X_1}} (\{\langle m_1, m_2\rangle\}) \otimes \gamma^{\mathsf{F}} \otimes (\omega_{\mathbb{B}})^\omega \otimes (\omega \otimes X_2[m])^\omega)) \quad \wedge \\ &\quad X_1 \in \mathrm{PreOrd} \wedge X_1 \in \mathrm{Lim} \Rightarrow Y = \bigcup_{i \in E_{X_1}} X_2[i]\end{aligned}$$

Remark 5.3.1.1. As noted before for the functional RAH, if m and m' are maximal elements of α and if X_2 is constructed by induction along the formula $\mathcal{R}$, then $X_2[m]$ and $X_2[m']$ will in fact be equal. However, this formulation of $\mathcal{R}(X_1, X_2, Y)$ (using an union for the successor case) allows us to show directly that this formula denotes a total functional relation.

Remark 5.3.1.2. We strongly use the operations introduced in Proposition 4.1.2.2 and the well-order γ^{F} on the set of formulas (Definition 5.1.2.3).

In particular, the relation $\mathcal{R}(X_1, X_2, Y)$ is functional. Therefore, Theorem 4.2.3.1 implies that:

$$\mathbf{\Pi^1_1\text{-}CA^-_0} + \mathrm{WPO}(\alpha) \vdash \exists!\mathcal{U}\exists!R\ (\mathrm{Ind}_{(\alpha,\mathcal{R})}(\mathcal{U}) \wedge \mathcal{R}(\alpha,\mathcal{U},R))$$

We use the notations introduced for $(\alpha, \mathcal{R})$-inductive family:

1. $\mathcal{R}_\alpha$ will denote the unique set such that $\exists\mathcal{U}(\mathrm{Ind}_{(\alpha,\mathcal{R})}(\mathcal{U}) \wedge \mathcal{R}(\alpha,\mathcal{U},X))$.
2. $\mathcal{R}^\alpha$ will denote the unique set such that $\mathrm{Ind}_{(\alpha,\mathcal{R})}(\mathcal{R}^\alpha)$.

Our first goal is to show that $\mathcal{R}_\alpha$ is a well-preorder.

Lemma 5.3.1.1. *Each element $b \in E_{\mathcal{R}_\alpha}$ is either* 0 *or of the form $\langle l, f, \rho_1, \rho_2\rangle$ with $l \in E_\alpha$, $f \in \mathtt{Form}$, $\rho_1 \in \mathrm{Val}^1$, $\rho^2 \in \mathrm{Seq}(\mathbb{N} \times E_{\mathcal{R}_{\alpha_{<l}}})$.*

Proof. By induction on α. □

Proposition 5.3.1.1. *The relation $\mathcal{R}_\alpha$ is a well-preorder and for all $i \in E_\alpha$, $\mathcal{R}^\alpha[i]$ is equal to $(\mathcal{R}_\alpha)_{<\langle i,0,0,0\rangle}$.*

Proof. It is done by induction on α (i.e. using Proposition 4.2.3.2). We reason by case whether α is empty, successor or limit.

1. Case: α is empty. $\mathcal{R}_\alpha = \{\langle 0, 0\rangle\}$ and it is a well-preorder.
2. Case: α is successor. $\mathcal{R}_\alpha$ is obtained as an union of different relations indexed by the set of the maximal elements of α. However, all these relations are equal because $\mathcal{R}$ is functional and two maximal elements of α are isomorphic (if $i \cong_\alpha j$ then $\alpha_{<i} = \alpha_{<j}$ and $\mathrm{Ind}_{(\alpha,\mathcal{R})}(\mathcal{R}^\alpha)$ implies that $\mathcal{R}^\alpha[i] = \mathcal{R}^\alpha[j]$). Specifically, they are equal to the sum of two relations. These two relations are well-preorders (using the induction hypothesis and Proposition 4.2.1.1). Thus, by Proposition 4.2.1.1, $\mathcal{R}_\alpha$ is a well-preorder. Moreover, by the definition of $\uplus$ (or by induction hypothesis if $i \notin \mathrm{M}_\alpha$), $\mathcal{R}^\alpha[i] = (\mathcal{R}_\alpha)_{<\langle i,0,0,0\rangle}$.
3. Case: α is limit. It is enough to show that $\mathcal{R}^\alpha$ is an α-chain (by Proposition 4.2.1.1). It is the case because if $i <_\alpha j, \mathcal{R}^\alpha[i] = \mathcal{R}^\alpha[j]_{<\langle i,0,0,0\rangle}$ and if $i \cong_\alpha j, \mathcal{R}^\alpha[i] = \mathcal{R}^\alpha_{<\langle i,0,0,0\rangle} = \mathcal{R}^\alpha_{<\langle j,0,0,0\rangle} = \mathcal{R}^\alpha[j]$.

□

5.3.2 Well-order on $\mathbf{S}(RAH_\alpha)$

We show that $\mathcal{R}_\alpha$ induces a well-order on $\mathbf{S}(\mathrm{RAH}_\alpha)$. We use Proposition 4.2.5.1 and we thus have to show the two following properties.

1. The field of $\mathcal{R}_\alpha$ contains an index for every set in $\mathbf{S}(\mathrm{RAH}_\alpha)$. It is done in Lemma 5.3.2.1.
2. Two isomorphic elements $x \cong_{\mathcal{R}_\alpha} y$ denotes the same set in $\mathbf{S}(\mathrm{RAH}_\alpha)$. It is done by induction on α using Lemma 5.3.1.1 and Theorem 5.2.3.1.

Lemma 5.3.2.1. *The field of $\mathcal{R}_\alpha$ contains an index for every set in* $\mathbf{S}(\mathrm{RAH}_\alpha)$.

Proof. We prove this lemma by simultaneous induction on $\mathcal{R}$ and RAH. We reason by case whether α is empty, successor or limit.

1. Case: α is empty. $\mathbf{S}(\mathrm{RAH}_\alpha) = \{\emptyset\}$ and $\mathcal{R}_\alpha = \{\langle 0, 0\rangle\}$, in particular $\mathrm{RAH}_\alpha[0] = \emptyset$ and $0 \in E_{\mathcal{R}_\alpha}$.

2. Case: α is successor. Let $X = \mathrm{RAH}_\alpha[\langle l, f, \rho_1, \rho_2\rangle]$. We need to find $\rho_2' \in \mathrm{Seq}(\mathbb{N} \times E_{\mathcal{R}_{\alpha_{<l}}})$ such that for all $n \in \mathtt{FV}(f), \mathrm{RAH}_{\alpha_{<l}}[\mathtt{find}(\rho_2, n)] = \mathrm{RAH}_{\alpha_{<l}}[\mathtt{find}(\rho_2', n)]$. We prove the following property by induction on ω:

   $$(H_n) \triangleq (\exists \rho_2' \in \mathrm{Seq}(\mathbb{N} \times E_{\mathcal{R}_{\alpha_{<l}}}))(\forall m < n)\mathrm{RAH}_{\alpha_{<l}}[\mathtt{find}(\rho_2, n)] = \mathrm{RAH}_{\alpha_{<l}}[\mathtt{find}(\rho_2', n)].$$

   It is done as follow:

   (a) For the base case, take ρ_2' to be empty.

   (b) Assume the property true up to n and consider a valuation $\rho' \in \mathrm{Seq}(\mathbb{N} \times E_{\mathcal{R}_{\alpha_{<l}}})$ given by the induction hypothesis H_n. Then, by the (general) induction hypothesis, $\mathrm{RAH}_{\alpha_{<l}}[\mathtt{find}(\rho_2, n)]$ has an index t in $E_{\mathcal{R}_{\alpha_{<l}}}$ and we can consider the valuation $\rho_2' = \langle n, t\rangle ::\rho'$.

   Let n_0 be the smallest code of variables that does not appear in f and consider ρ_2' given by H_{n_0}. Therefore, $\langle l, f, \rho_1, \rho_2'\rangle \in E_{\mathcal{R}_\alpha}$ and

   $$\begin{aligned}\mathrm{RAH}_\alpha[\langle l, f, \rho_1, \rho_2\rangle] &= \mathbf{Def^2}(\mathrm{RAH}_{\alpha_{<l}})[\langle l, f, \rho_1, \rho_2\rangle] \\ &= \mathbf{Def^2}(\mathrm{RAH}_{\alpha_{<l}})[\langle l, f, \rho_1, \rho_2'\rangle]\end{aligned}$$

   using Lemma 5.2.3.2 and Lemma 5.1.4.2.

3. Case: α is limit. It follows from the induction hypothesis.

□

Corollary 5.3.2.1. *The well-preorder $\mathcal{R}_\alpha$ induces a well-order on* $\mathbf{S}(\mathrm{RAH}_\alpha)$.

Moreover, isomorphic well-preorders yield isomorphic well-orders.

Proposition 5.3.2.1. *If $\alpha \cong_{\mathrm{WPO}} \beta$, the well-preorders $\mathcal{R}_\alpha$ and $\mathcal{R}_\beta$ are isomorphic and the isomorphism respects the structure of the families* $\mathbf{S}(\mathrm{RAH}_\alpha)$ *and* $\mathbf{S}(\mathrm{RAH}_\beta)$*: an index for X in* $\mathbf{S}(\mathrm{RAH}_\alpha)$ *is sent to an index for X in* $\mathbf{S}(\mathrm{RAH}_\beta)$.

$$\mathbf{PA2^-} + \mathrm{WPO}(\alpha) + \mathrm{WPO}(\beta) + \alpha \cong_{\mathrm{WPO}} \beta \vdash$$
$$(\exists \mathcal{I})(\mathrm{Isom}(\mathcal{I}, R_\alpha, R_\beta) \wedge (\forall x \in \mathrm{Dom}(\mathcal{R}_\alpha))(\forall y \in \mathrm{Dom}(\mathcal{R}_\beta))(x \mathcal{I} y \Rightarrow \mathrm{RAH}_\alpha[x] = \mathrm{RAH}_\beta[y])$$

Proof. We use a simultaneous induction on $\mathcal{R}$ and RAH to show the existence of the desired isomorphism. □

Corollary 5.3.2.2. *In particular, if α and β are isomorphic, $X \leq_{\mathcal{R}_\alpha}^{\mathrm{RAH}_\alpha} Y$ if and only if $X \leq_{\mathcal{R}_\beta}^{\mathrm{RAH}_\beta} Y$.*

5.3.3 Well-order on $\mathbf{S}(\mathrm{RAH})$

We can now define a well-order on $\mathbf{S}(\mathrm{RAH})$.

Theorem 5.3.3.1. *Consider the formula $\leq_{\mathrm{RAH}}(X, Y)$ defined below:*

$$X \leq_{\mathrm{RAH}} Y :\triangleq (\exists \alpha \in \mathrm{WPO})(X \in \mathbf{S}(\mathrm{RAH}_\alpha) \wedge Y \in \mathbf{S}(\mathrm{RAH}_\alpha) \wedge X \leq_{\mathcal{R}_\alpha}^{\mathrm{RAH}_\alpha} Y)$$

Then, the formula $\leq_{\mathrm{RAH}}$ is a well-order on $\mathbf{S}(\mathrm{RAH})$. Moreover, for all well-preorders α, $\leq_{\mathcal{R}_\alpha}^{\mathrm{RAH}_\alpha}$ is an initial segment of $\leq_{\mathrm{RAH}}$.

Proof. By definition $\leq_{\mathcal{R}_\alpha}^{\mathrm{RAH}_\alpha}$ is an initial segment of $\leq_{\mathrm{RAH}}$. Because $\leq_{\mathrm{RAH}}$ is obtained as an increasing sequence on well-orders, it is also a well-order. □

5.4 $\mathbf{S}(\mathrm{RAH})$ is a model of PA2

5.4.1 Reflection principle

Thanks to the use of well-preorders and of the axiom scheme of collection, the Reflection principle in $\mathbf{PA2^-} + \mathbf{Coll}$ can be proven in the same way as in set theory [29]. This principle will be the key to show that $\mathbf{S}(\mathrm{RAH})$ satisfies the axiom scheme of comprehension.

Definition 5.4.1.1. Let $\phi(\vec{x}, \overrightarrow{X})$ be a formula of $\mathbf{PA2^-}$. We say that a well-preorder α suits ϕ when

$$\mathrm{Suit}_\phi(\alpha) \triangleq \forall \vec{x}(\forall \overrightarrow{X} \in \mathbf{S}(\mathrm{RAH}_\alpha))(\phi^{\mathbf{S}(\mathrm{RAH})}(\vec{x}, \overrightarrow{X}) \Leftrightarrow \phi^{\mathbf{S}(\mathrm{RAH}_\alpha)}(\vec{x}, \overrightarrow{X})).$$

We say that α strongly suits ϕ when α suits ϕ and all its strict subformulas, i.e. if $\phi_1, ..., \phi_n$ are the strict subformulas of ϕ:

$$\mathrm{SSuit}_\phi(\alpha) \triangleq \mathrm{Suit}_\phi(\alpha) \wedge \mathrm{Suit}_{\phi_1}(\alpha) \wedge ... \wedge \mathrm{Suit}_{\phi_n}(\alpha).$$

We use the next lemmas in the proof of the principle of Reflection.

Lemma 5.4.1.1. *If $\alpha \cong_{\mathrm{WPO}} \beta$ and α strongly suits ϕ then β also strongly suits ϕ.*

Proof. By external induction on ϕ. It is a consequence of the fact that the hierarchy $(\mathrm{RAH}_\alpha)_{\alpha \in \mathrm{WPO}}$ is well-defined. □

Lemma 5.4.1.2. *Let $\phi(\vec{x}, \overrightarrow{X})$ be a formula of $\mathbf{PA2^-}$. If $\mathcal{U}$ is a family of well-preorders that all suit ϕ, then the well-preorder $\sup(\mathcal{U})$ also suits ϕ.*

$$\mathbf{PA2^-} + \mathrm{Fam}(\mathcal{U}) + (\forall i \in \mathrm{Dom}_\mathcal{U})(\mathrm{WPO}(\mathcal{U}[i]) \wedge \mathrm{SSuit}_\phi(\mathcal{U}[i])) \vdash \mathrm{SSuit}_\phi(\sup(\mathcal{U})).$$

Proof. Let $\alpha \triangleq \sup(\mathcal{U})$. We reason by external induction on ϕ.

1. Case: ϕ atomic. Then, $\phi^{\mathbf{S}(\mathrm{RAH})} \triangleq \phi \triangleq \phi^{\mathbf{S}(\mathrm{RAH}_\alpha)}$ and the result is immediate.
2. Cases: $\phi \triangleq \psi_1 \Rightarrow \psi_2$ or $\phi \triangleq \forall x \psi$. It follows from the induction hypothesis.

3. Case: $\phi \triangleq \forall X \psi$. Because of the induction hypothesis, α strongly suits ψ. Let $\vec{t}$ be individuals and $\overrightarrow{E} \in \mathbf{S}(RAH_\alpha)$. On one hand, we observe that:

$$\begin{aligned}\phi^{\mathbf{S}(\mathrm{RAH})}\psi[\vec{x} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}] &\triangleq (\forall X \in \mathbf{S}(\mathrm{RAH}))\psi^{\mathbf{S}(\mathrm{RAH})}[\vec{x} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}] \\ &:\Rightarrow (\forall X \in \mathbf{S}(\mathrm{RAH}_\alpha))\psi^{\mathbf{S}(\mathrm{RAH})}[\vec{x} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}] \quad (1)\\ &:\Rightarrow (\forall X \in \mathbf{S}(\mathrm{RAH}_\alpha))\psi^{\mathbf{S}(\mathrm{RAH}_\alpha)}[\vec{x} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}] \quad (2)\\ &\triangleq \phi^{\mathbf{S}(\mathrm{RAH}_\alpha)}[\vec{x} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}]\end{aligned}$$

where (1) because $\mathbf{S}(\mathrm{RAH}_\alpha) \subseteq \mathbf{S}(\mathrm{RAH})$, (2) because α suits ψ.

On the other hand, Corollary 5.2.2.1 says that $\mathbf{S}(\mathrm{RAH}(\sup(\mathcal{U}))) = \bigcup_{x \in \mathrm{Dom}_\mathcal{U}} \mathbf{S}(\mathrm{RAH}_{\mathcal{U}[x]})$. Therefore, let $n \in \mathrm{Dom}_\mathcal{U}$ such that all sets $E \in \overrightarrow{E}$ are in $\mathbf{S}(\mathrm{RAH}_{\mathcal{U}[n]})$. Then, we observe that:

$$\begin{aligned}\phi^{\mathbf{S}(\mathrm{RAH}_\alpha)}[\vec{x} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}] &\triangleq (\forall X \in \mathbf{S}(\mathrm{RAH}_\alpha))\psi^{\mathbf{S}(\mathrm{RAH}_\alpha)}[\vec{x} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}] \\ &\Rightarrow (\forall X \in \mathbf{S}(\mathrm{RAH}_\alpha))\psi^{\mathbf{S}(\mathrm{RAH})}[\vec{x} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}] \quad (1)\\ &\Rightarrow (\forall X \in \mathbf{S}(\mathrm{RAH}_{\mathcal{U}[n]}))\psi^{\mathbf{S}(\mathrm{RAH})}[\vec{x} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}] \quad (2)\\ &\Rightarrow (\forall X \in \mathbf{S}(\mathrm{RAH}_{\mathcal{U}[n]}))\psi^{\mathbf{S}(\mathrm{RAH}_{\mathcal{U}[n]})}[\vec{x} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}] \quad (3)\\ &\triangleq \phi^{\mathbf{S}(\mathrm{RAH}_{\mathcal{U}[n]})}[\vec{x} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}] \\ &\Rightarrow \phi^{\mathbf{S}(\mathrm{RAH})}[\vec{x} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}] \quad (4)\end{aligned}$$

where (1) because α suits ψ, (2) because $\mathbf{S}(\mathrm{RAH}_{\mathcal{U}[n]}) \subseteq \mathbf{S}(\mathrm{RAH}_\alpha)$, (3) and (4) because $\mathcal{U}[n]$ strongly suits ϕ (and in particulars it suits ψ and ϕ).

□

We can now state and prove the Reflection principle. Note that the next theorem uses the axiom scheme of collection.

Theorem 5.4.1.1. *Let $\phi(\vec{x}, \overrightarrow{X})$ be a formula of* $\mathbf{PA2^-}$*. For every well-preorder α, there is a well-preorder λ greater than α that strongly suits ϕ. Formally, for every formula $\phi(\vec{x}, \overrightarrow{X})$, we design a compatible functional (up to $\cong_{\mathrm{WPO}}$) relation $\mathcal{F}_\phi(X, Y)$ such that:*

$$\mathbf{PA2^-} + \mathbf{Coll} \vdash \mathrm{Compat}_{\mathcal{F}_\phi} \wedge \mathrm{Func}^{\cong_{\mathrm{WPO}}}_{\mathcal{F}_\phi} \wedge (\forall \alpha \in \mathrm{WPO})(\forall \beta \in \mathrm{WPO})(\mathcal{F}_\phi(\alpha, \beta) \Rightarrow (\mathrm{SSuit}_\phi(\beta) \wedge \alpha \preccurlyeq_{\mathrm{WPO}} \beta))$$

Proof. We reason by external induction on ϕ.

1. Case: ϕ atomic. Take $\mathcal{F}_\phi(\alpha, \beta) \triangleq \beta \cong_{\mathrm{WPO}} \alpha$.

2. Case: $\phi \triangleq \psi_1 \Rightarrow \psi_2$. Given a well-preorder α, we consider the sequence $\mathcal{U}_\alpha$ defined by iteration along ω (see Theorem 4.2.4.1 and Remark 4.2.4.2, note that it uses the axiom scheme of collection) using the compatible functional relation

$$\mathcal{G}(n, \alpha, \beta) \triangleq (\exists m \in \mathbb{N})((n = 2m \wedge \mathcal{F}_{\psi_1}(\alpha, \beta)) \vee (n = 2m + 1 \wedge \mathcal{F}_{\psi_2}(\alpha, \beta))).$$

and starting at α. We check that

$$\begin{aligned}\sup(\mathcal{U}_\alpha) &= \sup(\mathcal{U}_\alpha|_{\{x | \mathrm{SSuit}_{\psi_1}(\mathcal{U}_\alpha[x])\}}) \\ &= \sup(\mathcal{U}_\alpha|_{\{x | \mathrm{SSuit}_{\psi_2}(\mathcal{U}_\alpha[x])\}})\end{aligned}$$

and Lemma 5.4.1.2 implies that $\sup(\mathcal{U}_\alpha)$ strongly suits ψ_1 and ψ_2. As a consequence, $\sup(\mathcal{U}_\alpha)$ strongly suits ϕ and we set $\mathcal{F}_\phi(\alpha,\beta) \triangleq \beta \cong_{\mathrm{WPO}} \sup(\mathcal{U}_\alpha)$.

3. Case: $\phi \triangleq \forall x\psi$. Take $\mathcal{F}_\phi(\alpha,\beta) \triangleq \mathcal{F}_\psi(\alpha,\beta)$.

4. Case: $\phi \triangleq \forall X\psi$. If $\psi(\vec{x},\overrightarrow{Y},X)$ is a formula of $\mathbf{PA2}^-$ and α a well-preorder, given individuals $\vec{t}$ and $\vec{s}$ that respectively match the length of $\vec{x}$ and $\overrightarrow{Y}$, we define the following formula saying that γ is a well-preorder "large" enough to contain counter-examples to the assertion $(\forall X \in \mathbf{S}(\mathrm{RAH}))\psi^{\mathbf{S}(\mathrm{RAH})}[\vec{x} := \vec{t}; \overrightarrow{Y} := \overrightarrow{\mathrm{RAH}_\alpha[s]}; X]$.

$$\mathrm{Wit}_\psi(\langle\vec{t},\vec{s}\rangle,\alpha,\gamma) \triangleq (\forall X \in \mathbf{S}(\mathrm{RAH}_\gamma))\psi^{\mathbf{S}(\mathrm{RAH})}[\vec{x} := \vec{t}; \overrightarrow{Y} := \overrightarrow{\mathrm{RAH}_\alpha[s]}; X]) \Rightarrow (\forall X \in \mathbf{S}(\mathrm{RAH}))\psi^{\mathbf{S}(\mathrm{RAH})}[\vec{x} := \vec{t}; \overrightarrow{Y} := \overrightarrow{\mathrm{RAH}_\alpha[s]}; X].$$

Note that by a classical reasoning $\forall t\exists\gamma\mathrm{Wit}_\psi(t,\alpha,\gamma)$. Therefore, the axiom of collection and the operation of supremum on families of well-preorders show the existence of a well-preorder γ such that $\mathrm{WitUniv}_\psi(\alpha,\gamma) \triangleq \forall t\, \mathrm{Wit}_\psi(t,\alpha,\gamma)$. Moreover, because this formula is compatible, there is a minimal well-preorder that satisfies it. Consequently, the following relation is compatible and functional:

$$\mathcal{G}(\alpha,\beta) \triangleq (\exists\gamma_0 \in \mathrm{WPO})(\mathrm{WitUniv}_\psi(\alpha,\gamma_0) \wedge (\forall\gamma \in \mathrm{WPO})(\mathrm{WitUniv}_\psi(\alpha,\gamma) \Rightarrow \gamma_0 \preccurlyeq_{\mathrm{WPO}} \gamma) \wedge \mathcal{F}_\psi(\alpha{+}\gamma_0,\beta)).$$

We can consider the family $\mathcal{U}_\alpha$ obtained by iteration of $\mathcal{G}$ along ω (starting at α) and define

$$\mathcal{F}_\phi(\alpha,\beta) \triangleq \beta \cong_{\mathrm{WPO}} \sup(\mathcal{U}_\alpha).$$

Now, let λ such that $\mathcal{F}_\phi(\alpha,\lambda)$. Therefore, λ is a well-preorder that strongly suits ψ (by induction hypotheses and Lemma 5.4.1.2). We show that λ suits ϕ. Let $\vec{t}$ be individuals and $\vec{E} \in \mathbf{S}(\mathrm{RAH}_\lambda)$. On one hand, we observe that:

$$\begin{aligned}
\phi^{\mathbf{S}(\mathrm{RAH})}[\vec{x} := t; \vec{X} := E] &\triangleq (\forall X \in \mathbf{S}(\mathrm{RAH}))\psi^{\mathbf{S}(\mathrm{RAH})}[\vec{x} := t; \overrightarrow{X} := E] \\
&\Rightarrow (\forall X \in \mathbf{S}(\mathrm{RAH}_\lambda))\psi^{\mathbf{S}(\mathrm{RAH})}[\vec{x} := t; \overrightarrow{X} := E] \qquad (1)\\
&\Rightarrow (\forall X \in \mathbf{S}(\mathrm{RAH}_\lambda))\psi^{\mathbf{S}(\mathrm{RAH}_\lambda)}[\vec{x} := t; \overrightarrow{X} := E] \qquad (2)\\
&\triangleq \phi^{\mathbf{S}(\mathrm{RAH}_\lambda)}[\vec{x} := t; \vec{X} := E]
\end{aligned}$$

where (1) because $\mathbf{S}(\mathrm{RAH}_\lambda) \subseteq \mathbf{S}(\mathrm{RAH})$, (2) because λ suits ψ.

On the other hand, let m such that $\overrightarrow{E} \subseteq \mathbf{S}(RAH_{\mathcal{U}_\alpha[m]})$ and n such that $\mathcal{G}(\mathcal{U}_\alpha[m],\mathcal{U}_\alpha[n])$. Then, we observe that:

$$\begin{aligned}
\phi^{\mathbf{S}(\mathrm{RAH}_\lambda)}[\vec{x} := t; \vec{X} := E] &\triangleq (\forall X \in \mathbf{S}(\mathrm{RAH}_\lambda))\psi^{\mathbf{S}(\mathrm{RAH}_\lambda)}[\vec{x} := t; \overrightarrow{X} := E] \\
&\Rightarrow (\forall X \in \mathbf{S}(\mathrm{RAH}_\lambda))\psi^{\mathbf{S}(\mathrm{RAH})}[\vec{x} := t; \overrightarrow{X} := E] \qquad (1)\\
&\Rightarrow (\forall X \in \mathbf{S}(\mathrm{RAH}_{\mathcal{U}_\alpha[n]}))\psi^{\mathbf{S}(\mathrm{RAH})}[\vec{x} := t; \overrightarrow{X} := E] \qquad (2)\\
&\Rightarrow (\forall X \in \mathbf{S}(\mathrm{RAH}))\psi^{\mathbf{S}(\mathrm{RAH})}[\vec{x} := t; \overrightarrow{X} := E] \qquad (3)\\
&\triangleq \phi^{\mathbf{S}(\mathrm{RAH})}[\vec{x} := t; \vec{X} := E]
\end{aligned}$$

where

- (1) because λ suits ψ,

- (2) because $\mathbf{S}(\mathrm{RAH}_{\mathcal{U}_\alpha[n]}) \subseteq \mathbf{S}(\mathrm{RAH}_\lambda)$,
- (3) because $\mathrm{WitUniv}_\psi(\mathcal{U}_\alpha[m],\mathcal{U}_\alpha[n])$ and $\overrightarrow{E} \subseteq \mathbf{S}(RAH_{\mathcal{U}_\alpha[m]})$.

□

5.4.2 S(RAH) is an inner model of PA2

Theorem 5.4.2.1. $\mathbf{S}(RAH)$ *is an inner model of* **PA2** *(in the sense of Definition 1.2.4.3) inside* $\mathbf{PA2^-} + \mathbf{Coll}$.

Proof. Because the relativization to the set $\mathbb{B}$ of the induction principle is provable in $\mathbf{PA2^-}$ and that it is a Π^1_1-formula, it is valid in $\mathbf{S}(\mathrm{RAH})$. It remains the case of the axiom scheme of Comprehension. Let $\phi(x, \vec{y}, \overrightarrow{X})$ a formula of **PA2**, we need to show

$$\mathbf{PA2^-} + \mathbf{Coll} \vdash (\forall \vec{x} \in \mathbb{B})(\forall \overrightarrow{X} \in \mathbf{S}(\mathrm{RAH}))(\exists X \in \mathbf{S}(\mathrm{RAH}))\forall x(x \in X \Leftrightarrow \phi(x, \vec{x}, \overrightarrow{X})^{\mathbf{S}(\mathrm{RAH})})$$

Let $\vec{t} \in \mathbb{B}$ be individuals and $\overrightarrow{E} \in \mathbf{S}(\mathrm{RAH})$. Because the hierarchy $\{\mathbf{S}(RAH_\alpha)\}_{\alpha\in\mathrm{WPO}}$ is increasing, there exists α such that $\overrightarrow{E} \in \mathbf{S}(\mathrm{RAH}_\alpha)$ By the Reflection principle, let $\beta \geq \alpha$ that strongly suits ϕ, then:

$$\forall x\, (\phi^{\mathbf{S}(\mathrm{RAH})}[x; \vec{y} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}] \Leftrightarrow \phi^{\mathbf{S}(\mathrm{RAH}_\beta)}[x; \vec{y} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}])$$

But then it is a definition of

$$\{x \mid \phi^{\mathbf{S}(\mathrm{RAH})}[x; \vec{y} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}]\}$$

with parameters in $\mathbf{S}(\mathrm{RAH}_\beta)$ and that relativizes to $\mathbf{S}(\mathrm{RAH}_\beta)$. Therefore, we can conclude that

$$\{x \mid \phi^{\mathbf{S}(\mathrm{RAH})}[x; \vec{y} := \vec{t}; \overrightarrow{X} := \overrightarrow{E}]\} \in \mathbf{Def^2}(\mathbf{S}(\mathrm{RAH}_\beta)) \subseteq \mathbf{S}(\mathrm{RAH}).$$

□

From now on, for all closed formula ϕ, we say that $\mathbf{S}(\mathrm{RAH})$ proves ϕ if

$$\mathbf{PA2^-} + \mathbf{Coll} \vdash \phi^{\mathbf{S}(\mathrm{RAH})}.$$

5.4.3 S(RAH) models the axiom scheme of the well-ordered universe

The axiom scheme of the well-ordered universe

We extend the language of $\mathbf{PA2^-}$ with a symbol $\leq_{\mathrm{U}}$ for a third-order binary relation. We consider the axiom saying that it is a well-order on the class of all sets, defined similarly as in Definition 4.2.5.1.

Definition 5.4.3.1. The axiom scheme of the well-ordered universe states that the universe of sets is well-ordered by the third-order relation $\leq_{\mathrm{U}}$. Formally it contains the following formulas.

1. The relation $\leq_{\mathrm{U}}$ is an order, i.e.

   $$\mathrm{Ord}_{\leq_{\mathrm{U}}} \triangleq \forall X\forall Y\forall Z(X\leq_{\mathrm{U}}X\wedge(X\leq_{\mathrm{U}}Y \Rightarrow Y\leq_{\mathrm{U}}Z \Rightarrow X\leq_{\mathrm{U}}Z)\wedge(X\leq_{\mathrm{U}}Y \Rightarrow Y\leq_{\mathrm{U}}X \Rightarrow X = Y)).$$

2. For every non-empty class $\Theta(X)$, there is a $\leq_{\mathrm{U}}$-minimal set in it. Formally, for every formula with parameters $\phi(X)$:

   $$\mathrm{WO}_{(\leq_{\mathrm{U}},\phi)} \triangleq \forall X(\phi[X] \Rightarrow \exists M(\phi[X := M] \wedge \forall Y(\phi[X := Y] \Rightarrow M\leq_{\mathrm{U}}Y))).$$

Remark 5.4.3.1. This axiom is weaker than the principle of choice [29] stating that there exists a definable third-order relation that is a well-order on the universe of sets.

$\mathbf{S}$(RAH) satisfies the axiom scheme of the well-ordered universe

The definition of the relativization needs to be extended to the new third-order predicate $\leq_{\mathrm{U}}$. It will be interpreted as the order $\leq_{\mathrm{RAH}}$ defined in Theorem 5.3.3.1, which means that we extend the relativization $\phi \mapsto \phi^{\mathbf{S}(\mathrm{RAH})}$ with the clause

$$(X \leq_{\mathrm{U}} Y)^{\mathbf{S}(\mathrm{RAH})} \triangleq X \leq_{\mathrm{RAH}} Y.$$

Theorem 5.4.3.1. *When $\leq_{\mathrm{U}}$ is interpreted as $\leq_{\mathrm{RAH}}$, $\mathbf{S}$(RAH) satisfies the axiom scheme of the well-ordered universe.*

Proof. This is a reformulation of Theorem 5.3.3.1. □

Theorem 5.4.3.2. *The theory* **PA2** *extended with the axiom scheme of the well-ordered universe is relatively consistent to* $\mathbf{PA2}^{-} + \mathbf{Coll}$.

Remark 5.4.3.2. However, with the work we did, we were not able to show that $\leq_{\mathrm{RAH}}$ is internally a well-order on $\mathbf{S}$(RAH). In other words, we were not able to show $\mathrm{WO}^{\mathbf{S}(\mathrm{RAH})}_{(\leq_{\mathbf{S}(\mathrm{RAH})}, \phi)}$ in general. It is connected to our failure to show that $\mathbf{S}$(RAH) satisfies the axiom of constructibility.

5.5 Open problems, related works and future work

5.5.1 What about the axiom of constructibility?

The goal of this section is to explain why results that felt known in the community [15] are in fact still open problems. These results are connected with the problem of the consistency of the axiom of constructibility in second-order arithmetic (Open problems 2 page 112 and 3 page 113).

Definition 5.5.1.1. The axiom of constructibility in $\mathbf{PA2}^{-}$ is defined as the formula $\forall X(X \in \mathbf{S}(\mathrm{RAH}))$. Mimicking the notation of set theory, we will write $\mathrm{V} = \mathbf{S}(\mathrm{RAH})$ to denote the axiom of constructibility in $\mathbf{PA2}^{-}$.

The axiom of constructibility in set theory

The axiom of constructibility in set theory [21], written $\mathrm{V} = \mathrm{L}$, is defined as the formula $\forall x\, \mathrm{L}(x)$. It implies the well-order principle and the generalized continuum hypothesis. Gödel showed that the class L succeeded in interpreting this axiom and that it was the smallest inner model of set theory[21, 29, 23]. It is a consequence of a technical result about absoluteness of the definition of L and of the fact that all ordinals are in L. However, we were not able to adapt this proof to the realm of second-order arithmetic, namely we were not able to show that $\mathbf{S}$(RAH) contains "enough" well-preorders.

Yet an other technical result about RAH

The goal of this section is to show the following technical result: if $\alpha \in \mathbf{S}(\mathrm{RAH}_\beta)$, then $\mathrm{RAH}_\alpha \in \mathbf{S}(\mathrm{RAH}_{\beta+\alpha+\mathbf{1}})$ where $\mathbf{1} \triangleq \{\langle 0, 0\rangle\}$.

Remark 5.5.1.1. The well-preorder $\alpha+\mathbf{1}$ is used as a choice of a successor for the well-preorder α.

Lemma 5.5.1.1. *Let $\mathcal{U}$ be a family of sets satisfying the scheme of arithmetical comprehension and containing the set $\mathbb{B}$, then for all individuals p, $\mathbf{Def}^{\mathbf{2}}(\mathcal{U}[p]) \in \mathbf{S}(\mathbf{Def}^{\mathbf{2}}(\mathcal{U}))$.*

Proof. We need to show that the following set is in $\mathbf{S}(\mathbf{Def^2}(\mathcal{U}))$:

$$\begin{aligned}\mathbf{Def^2}(\mathcal{U}[p]) &\triangleq \{\langle\langle f, \rho_1, \rho_2\rangle, x\rangle \mid \langle f, \langle\langle 0, x\rangle, \rho_1\rangle, \rho_2\rangle \in \mathrm{Sat}(\mathcal{U}[p])\} \\ &\triangleq \{\langle\langle f, \rho_1, \rho_2\rangle, x\rangle \mid \langle f, \langle\langle 0, x\rangle, \rho_1\rangle, \rho_2\rangle \in G_{\mathcal{U}[p]}\}\end{aligned}$$

Because of the closure property of $\mathbf{S}(\mathbf{Def^2}(\mathcal{U}))$ (see Lemma 5.1.4.3), it is enough to show that $G_{\mathcal{U}[p]}$ is in $\mathbf{S}(\mathbf{Def^2}(\mathcal{U}))$.

The set $G_{\mathcal{U}[p]}$ is obtained by induction over the limit well-order $\gamma^{\mathtt{F}}$ using a functional relation denoted $\mathcal{H}$ with $\mathcal{U}[p]$ and $\gamma^{\mathtt{F}}$ as parameters (see Theorem 5.1.2.1). Therefore, the set $G_{\mathcal{U}[p]}$ has the following definition:

$$x \in G_{\mathcal{U}[p]} \Leftrightarrow (\exists f \in \gamma^{\mathtt{F}})\exists \mathcal{V}_f \exists R_f (\mathcal{H}(\gamma^{\mathtt{F}}_{<f}, \mathcal{V}_f, R_f, \mathcal{U}[p], \gamma^{\mathtt{F}}) \wedge x \in \mathcal{V}_f)$$

We show that this definition relativizes to $\mathbf{S}(\mathcal{U})$. First, we know that

$$(\forall f \in \mathtt{Form})\ \mathcal{H}(\gamma^{\mathtt{F}}_{<f}, G_{\mathcal{U}}|_{E_{\gamma^{\mathtt{F}}_{<f}}}, G_{\mathcal{U}}[f], \mathcal{U}[p], \gamma^{\mathtt{F}})$$

We then need to show:

1. The well-order $\gamma^{\mathtt{F}}$ is in $\mathcal{U}$. This is the case because of the closure property of $\mathcal{U}$ (it models the scheme of arithmetical comprehension and $\gamma^{\mathtt{F}}$ has an arithmetical definition with $\mathbb{B}$ as parameter).
2. For all codes f, the sets $G_{\mathcal{U}}[f]$ are in $\mathbf{S}(\mathcal{U})$. It is done by induction on $\gamma^{\mathtt{F}}$: for all codes f, there is an arithmetical definition of $G_{\mathcal{U}}[f]$ with parameters included in $\mathcal{U}[p]$ and $G_{\mathcal{U}}[f_i]$ for a finite number of subformulas f_i of f.
3. For all codes f, the set $G|_{E_{\gamma^{\mathtt{F}}_{<f}}}$ is in $\mathbf{S}(\mathcal{U})$. It is done by induction using the previous point.

□

Corollary 5.5.1.1. *If* $\alpha \in \mathbf{S}(\mathrm{RAH}_\beta)$ *then* $\mathrm{RAH}_\alpha \in \mathbf{S}(\mathrm{RAH}_{\beta+\alpha+\mathbf{1}})$.

Proof. This proof is done by induction on α. The hypothesis that $\alpha \in \mathbf{S}(\mathrm{RAH}_\beta)$ is used for the limit case where a definition of RAH_α that relativizes to $\mathbf{S}(\mathrm{RAH}_{\beta+\alpha})$ is produced, with α as parameter. □

Remark 5.5.1.2. This result can be generalized in the following manner. If $\alpha \in \mathbf{S}(\mathrm{RAH}_\beta)$ and $\mathrm{RAH}_{\alpha_{<i}} \in \mathbf{S}(\mathrm{RAH}_\beta)$ for all $i \in E_\alpha$, then $\mathrm{RAH}_\alpha \in \mathbf{S}(\mathrm{RAH}_{\beta+\mathbf{1}})$.

Russel paradox for RAH

We describe a variant of Russel paradox for the ramified analytic hierarchy. It will be used in the remainder of this section.

Lemma 5.5.1.2. *For all non-empty well-preorders* α, $\mathrm{RAH}_\alpha \notin \mathbf{S}(\mathrm{RAH}_\alpha)$.

Proof. Assume that $\mathrm{RAH}_\alpha \in \mathbf{S}(\mathrm{RAH}_\alpha)$. Because of the closure properties of $\mathbf{S}(\mathrm{RAH}_\alpha)$, the set $\{u \mid u \notin \mathrm{RAH}_\alpha[u]\}$ is in $\mathbf{S}(\mathrm{RAH}_\alpha)$. Let u_0 be an index for this set. Then $u_0 \in \mathrm{RAH}_\alpha[u_0]$ if and only if $u_0 \notin \mathrm{RAH}_\alpha[u_0]$. □

Remark 5.5.1.3. This proof can be adapted to well-preorders β isomorphic to α, showing that, in this case, $\mathrm{RAH}_\beta \notin \mathbf{S}(\mathrm{RAH}_\alpha)$.

The construction of RAH in ZF

The construction of RAH can be carried in **ZF** to obtain a model of **PA2**. We recall here the definition given by Apt and Marek [2].

Definition 5.5.1.2. The set R.A. (representing the ramified analytic hierarchy in ZF) is inductively defined by the clause

$$\begin{array}{rcl} \text{R.A.}_0 & \triangleq & \emptyset \\ \text{R.A.}_{\alpha+1} & \triangleq & \mathbf{Def^2}(\text{R.A.}_\alpha) \\ \text{R.A.}_\lambda & \triangleq & \bigcup_{\xi\in\lambda} \text{R.A.}_\xi \qquad \text{for } \lambda \text{ limit} \\ \text{R.A.} & \triangleq & \bigcup_{\alpha\in On} \text{R.A.}_\alpha \end{array}$$

where, in this context and for $X \subseteq \mathbb{N}$, $\mathbf{Def^2}(X)$ is the set of subsets of $\mathcal{P}(N)$ that can be defined using a second-order formula with second-order parameters in X and second-order quantifications relativized to X.

Note that the class R.A. is a set because it is included in $\mathcal{P}(\mathbb{N})$. Even more, Cohen observed that this sequence becomes stationary after a countable ordinal [14], called β_0.

Lemma 5.5.1.3. *There exists a countable ordinal β_0 which is a fixed point for* R.A.*:*

$$\text{R.A.}_{\beta_0+1} = \text{R.A.}_{\beta_0}.$$

Corollary 5.5.1.2. *The set* R.A. *is not closed by countable unions of well-preorders.*

Proof. If it was the case, one would be able to show that a well-preorder β of order-type β_0 is in R.A._{β_0} and therefore, by Lemma 5.5.1.1, that the set R.A._{β_0} is in $\text{R.A.}_{\beta_0+1} = \text{R.A.}_{\beta_0}$. It is in contradiction with the variant of Russel paradox proved in Lemma 5.5.1.2. □

An intent on proving that the class S(RAH) contains "enough" well-preorders

The goal of this section is to try to prove that "almost" all well-preorders are in $\mathbf{S}(\text{RAH})$. In other words, we want to prove the formula $(\forall\alpha \in \text{WPO})\alpha \in \mathbf{S}(\text{RAH})$. However, it is not reasonable because well-preorders can have representations that are not constructible. It seems better to focus on the formula $(\forall\alpha \in \text{WPO})(\exists\beta \in \text{WPO})(\alpha \cong_{\text{WPO}} \beta \wedge \beta \in \mathbf{S}(\text{RAH}))$.

In set theory, the stronger formula $\alpha \in \text{L}_{\alpha+1}$ is proved by induction on the class of ordinals. We follow this lead and we will try to to prove $(\forall\alpha \in \text{WPO})(\exists\beta \in \text{WPO})(\alpha \cong_{\text{WPO}} \beta \wedge \beta \in \mathbf{S}(\text{RAH}_{\alpha+1})$ by induction over the class of well-preorders.

In the limit case, we need to show that if all the proper initial segments of a well-preorder β are in $\mathbf{S}(\text{RAH}_\alpha)$ then this well-preorder β is also in $\mathbf{S}(\text{RAH}_{\alpha+1})$ But it is not the case, as shown in Corollary 5.5.1.2.

We will explain how this problem was tackled in the literature and conclude that it is not yet solved.

The notion of contributive well-preorders

As far as the author knows, there is only one work who studied the ramified analytic hierarchy in the syntax of **PA2**. It is the paper of Colson and Griogirief [15] where, in the appendix, they describe a formalization of RAH in second-order arithmetic. They overcome the issue presented in the last section by introducing the notion of contributive well-order.

Definition 5.5.1.3. A well-preorder α is contributive if $\mathbf{S}(\mathrm{RAH}_{\alpha+1}) \setminus \mathbf{S}(\mathrm{RAH}_\alpha) \neq \emptyset$.

Before analyzing more in depth their work, note that RAH is not closed by countable union of contributive well-preorders.

Lemma 5.5.1.4. *The set* R.A. *is not closed by countable union of contributive well-preorders.*

Proof. It is done as in Lemma 5.5.1.2 because all the initial segments of β_0 are contributive. □

Colson and Grigorieff show that any contributive well-order is isomorphic to a well-order in $\mathbf{S}$(RAH) (Proposition D.9.4). In fact, for α a contributive well-order, they show that the defined well-ordered on RAH_α (called $\mathcal{R}_\alpha$ here in Section 5.3.3 and λ_α in their work, see Theorem D.3.(a)) is in $\mathrm{RAH}_{\alpha+1}$ (Proposition D.9.3). I think that the proof of this fact is flawed. The problem is in the limit case:

1. Their proof brings a contradiction when applied in set theory to R.A. (the semantic version of RAH). In their proof, for a contributive limit well-order α (limit case of Proposition D.9.3), they do not use the fact that α is contributive, it is enough that all the initial segments of α are contributive. Therefore, by applying this proof to R.A., we would obtain that $\lambda_{\beta_0} \in \mathrm{R.A.}_{\beta_0+1} = \mathrm{R.A.}_{\beta_0}$. However, the order-type of λ_{β_0} is at least β_0. It is a contradiction, as shown in the proof of Corollary 5.5.1.2.

2. We describe here (precisely and technically) where the flaw is in their proof of the limit case (for a well-order α). They prove that the sets RAH_α and $\mathcal{R}_\alpha$ are in $\mathbf{S}(\mathrm{RAH}_{\alpha+1})$ by giving definitions of theses sets that relativize to $\mathbf{S}(\mathrm{RAH}_\alpha)$. However, theses definitions crucially use the well-order α as a parameter. But it is not yet proven that the well-order α appears in $\mathbf{S}(\mathrm{RAH}_\alpha)$.

Conclusion: an old open problem?

All in all, as far as the author's knowledge, the problem of the consistency of the axiom of constructibility in **PA2** remains unresolved[7].

Open problem 2. *Is the theory* $\mathbf{PA2} + \mathrm{V} = \mathbf{S}(\mathrm{RAH})$ *relatively consistent to the theory* **PA2***?*

I personally find that this is a very interesting and important question, connected to domains of logic that were deeply studied since more than 60 years (constructibility, second-order arithmetic, recursion theory...). Note that a very similar result appeared in the work of Vetulani [51] (described below). He worked in third-order analysis and used results of interpretability between set theories and higher-order analysis to show the relative consistency of the axiom of constructibility (formulated for the framework of higher-order logic). Therefore, while a solution of Open problem 2 is not yet explicitly written in the literature, it seems probable that one can be found by combining the work of Vetulani and the work of Simpson [47] (also described below). Concretely, Simpson showed the relative consistency of the axiom of choice with respect to **PA2** and Vetulani's work could be adapted to show the relative consistency of the axiom $\mathrm{V} = \mathbf{S}(\mathrm{RAH})$ with respect to $\mathbf{PA2}+\mathbf{AC}_\iota$; thus combining these two results would lead to a proof of the relative consistency of the axiom of constructibility with resepct to **PA2**.

However, in this thesis, we followed an other path and we tried to show that $\mathbf{S}$(RAH) contains "enough" well-orders, namely the contributive well-orders. Our work led to the following problem: can it be shown in $\mathbf{PA2}^- + \mathbf{Coll}$ that $\mathbf{S}$(RAH) contains all the contributive well-preorders?

[7] At least, it seems that it is not explicitly written in the literature.

On the necessity of the axiom scheme of collection and an other open problem

It seems that no proofs of the fact that RAH models the axiom scheme of comprehension was written internally in second-order arithmetic: the only fully written proof that saw the author in the literature is the one of Colson and Grigorieff. However, their proof uses the fact that the contributive well-orders have representatives in $\mathbf{S}$(RAH). Concretely, they use this fact to prove a principle of reflection (Theorem 5.4.1.1 in this thesis and Proposition D.10 in their work) which is crucial to show that $\mathbf{S}$(RAH) interprets the axiom scheme of comprehension. This leads to the formulation of an other potentially open problem.

Open problem 3. *Can it be shown in* **PA2** *that* $\mathbf{S}$(RAH) *models the axiom scheme of comprehension? Is the principle of reflection provable in* **PA2***?*

We noted that this flaw can be overcome by the use of an axiom of countable choice. However, a careful analysis of the reflection principle in set theory inspired us to work in second-order arithmetic without induction and to introduce a new logical scheme in this syntax: the scheme of collection.

Therefore, we answer partially to this problem in Theorem 5.4.2.1 stating that $\mathbf{PA2}^- + \mathbf{Coll}$ proves that $\mathbf{S}$(RAH) is an inner model of **PA2**.

5.5.2 Related works

The literature about RAH [26] is vast but is mainly concerned about the set-theoretic property of RAH, in connection for instance with recursivity [10] or with model theory of second-order arithmetic [2]. Consequently, I consider that all these works are not directly related to my research. I will mention the few sources that deal with a theme closer to the formalization of the ramified analytic hierarchy in second-order arithmetic.

The work of Colson and Griogorieff

This part of the thesis is very strongly inspired by the paper of Colson and Griogorieff [15] In fact, most of this chapter is a direct adaptation of their research: the structure of this chapter and the notations used are very close to the one of their work. The differences are the following:

1. We work here without induction. This is not a major difference as it is almost never used. However, small technical details appeared when working without induction. In order to define a well-order on $\mathbf{S}$(RAH), it is necessary to define a well-order on the set of first-order valuations. If they range over all the individuals, we will not be able to well-order them. Therefore, it is essential that the first-order valuations range over pure binary trees[8]. Consequently, the internalization of the notion of satisfiability is concerned with structures that have a standard first-order part. In other words, we internalize the satisfiability of formulas relativized to the set $\mathbb{B}$ of pure binary trees (Section 5.1).

2. We work with well-preorders instead of well-orders. It is also not a major difference: the results about well-orders extend mostly easily to well-preorders. However, it is notable that all the results about well-orders scale to well-preorders. Notably, the well-order on $\mathbf{S}$(RAH) is defined similarly when using well-orders or well-preorders to index this hierarchy! In fact, because the class of well-orders is only well-preordered, the well-order on $\mathbf{S}$(RAH) is defined as a quotient of a well-preorder in both oh these cases. In spite of all these similarities,

[8] Anyway, computing the value of a code of an individual enforces the code to denote a pure binary tree. Consequently, it is necessary for a first-order valuation to range over pure binary tree.

the class of well-preorders have one major advantage: the supremum of a family of well-preorders can be easily defined (Theorem 4.2.2.1). This ingredient, combined with the axiom scheme of collection, is the key to prove the principle of reflection.

3. The use of the axiom scheme of collection allows us to prove the reflection principle and to fix the flaw contained in their paper. It is specifically used in Theorem 4.2.4.1 that seems necessary to prove the principle of reflection (Theorem 5.4.1.1).

The work of Vetulani

Vetulani studied and formalized an adaptation of the ramified analytic hierarchy in third-order analysis, i.e. in third-order arithmetic with choice [51]. He constructed a variant of the ramified analytic hierarchy for third-order analysis and showed that it was a model satisfying (a variant of) the axiom of constructibility. In his work, he used results of interpretability between weak set theory and higher-order analysis. It seems possible that his work could be adapted to second-order analysis[9] and could (almost) give a solution to the Open Problem 2. Notably, I have never seen such an adaptation written. It remains as a future work for me to try and write it! Then, the next step would be to investigate if the axiom of choice is crucial in all his proofs. During this thesis, we chose to work directly in second-order arithmetic and not inside a set-theoretic conservative extension. Our main reason behind this choice is explained below.

I wanted to use the ramified analytic hierarchy to give a new meaning of impredicative quantification inside a type system (such as System F). The impredicative quantification, through relativization, would have been replaced by quantification along well-preorders and codes of formulas with parameters. This new way of understanding second-order quantification could have led to a new computational interpretation of the axiom of choice[10]. All in all, I thought that working in a set-theoretic interpretation of second-order arithmetic would blurred the computational content of RAH. Moreover, it may be impossible to adapt the proofs of Vetulani in a framework without axiom of choice ; thus this approach may not be suited for a computational interpretation of the axiom of choice.

The work of Simpson

Simpson formalized results about the constructible universe in a weak set theory that is a conservative extension of a subsystem of **PA2** [47]. In particular, he showed in this weak set theory that the class L satisfies the axiom of constructibility. Moreover, his work scales to stronger versions of arithmetic and, in particular, to full second-order arithmetic. Therefore, he proved that adding the principle of well-ordered universe to **PA2** does not lead to inconsistencies.

5.5.3 Future work: a translation of proof systems by relativization to the ramified analytic hierarchy

In this part, we sketch how to design a translation between two proof systems:

1. The source proof system $\boldsymbol{\lambda}\mathbf{PA2}_{\mathbf{AC}}$ is an extension of $\boldsymbol{\lambda}\mathbf{PA2}$ (presented in Figure 2.1 page 42) with extra instructions to interpret the axiom scheme of the well-ordered universe.

2. The target proof system $\boldsymbol{\lambda}\mathbf{PA2}^{-}_{\mathbf{Coll}}$ is an extension of $\boldsymbol{\lambda}\mathbf{PA2}^{-}$ (presented in Figure 2.2 page 45) obtained by adding a rule to use the axiom scheme of collection.

[9] Vetulani wrote at the end of his paper that his method adapts directly to second-order arithmetic with choice.
[10] However, it still remains as a future work.

This translation implements a relativization of the first-order quantifiers to the set $\mathbb{B}$ and a relativization of the second-order quantifiers to the class $\mathbf{S}(\mathrm{RAH})$. As in the negative translation described in Section 3.2.1, all the logical parts of a λ-term M of $\boldsymbol{\lambda}\mathbf{PA2_{AC}}$ will be given a computational meaning, thus describing (again) a translation from a Church-style calculus into a Curry-style calculus. We will see that, through these relativizations, the translation will interpret the principle of the well-ordered universe. Note that the principle of the well-ordered universe allows to extract a witness from a non-empty class of sets. Moreover, this witness extraction is extensional: the same witness is extracted from two extensionally equal classes. It is a very important difference with the non-extensional axiom of choice, which is realized in Krivine models [30]. However, this translation remains as a future work: we have yet to study the computational content of the "unformalized" proofs done in this chapter.

The source system: $\boldsymbol{\lambda}\mathbf{PA2_{AC}}$

The system $\boldsymbol{\lambda}\mathbf{PA2_{AC}}$ is obtained from $\boldsymbol{\lambda}\mathbf{PA2}^-$ by adding the necessary tools to interpret the axiom scheme of the well-ordered universe defined in Definition 5.4.3.1, specifically:

1. a new atomic formula $X \leq_{\mathrm{U}} Y$
2. proof terms $\mathrm{ord}_{\leq_{\mathrm{U}}}$ and $\mathrm{wo}_{\leq_{\mathrm{U}},\phi}$ for each formula $\phi(X)$ with one free second-order variable X
3. typing rules for the added proof terms

$$\frac{}{\Gamma \vdash \mathrm{ord}_{\leq_{\mathrm{U}}} : \mathrm{Ord}_{\leq_{\mathrm{U}}}} \qquad \frac{}{\Gamma \vdash \mathrm{wo}_{\leq_{\mathrm{U}},\phi} : \mathrm{WO}_{\leq_{\mathrm{U}},\phi}}$$

All in all, it is a proof system for $\mathbf{PA2}$ with the axiom of the well-ordered universe. We are interested in the computational behavior that can be given to the terms $\mathrm{wo}_{\leq_{\mathrm{U}},\phi}$. However, it will remain as a future work.

The target system: the system $\boldsymbol{\lambda}\mathbf{PA2}^-_{\mathbf{Coll}}$

The target system $\boldsymbol{\lambda}\mathbf{PA2}^-_{\mathbf{Coll}}$ is obtained from $\boldsymbol{\lambda}\mathbf{PA2}^-$ by adding the following typing rule

$$\frac{}{\vdash \lambda\xi.\xi : \forall Z \exists x \forall y \phi(x, Z[y]) \Rightarrow \exists x \forall Y \phi(x, Y)}$$

By the work we did in Section 2.3 and specifically in Theorem 2.4.0.1, we know that adding this rule to the system $\boldsymbol{\lambda}\mathbf{PA2}^-$ will not break the soundness of its interpretation by classical realizability (Theorem 2.3.3.1). Therefore, the soundness of this system is guaranteed and all the tools of classical realizability are available to study its computational properties.

The translation from $\boldsymbol{\lambda}\mathbf{PA2_{AC}}$ into $\boldsymbol{\lambda}\mathbf{PA2}^-_{\mathbf{Coll}}$

We follow the model of the negative translation done in Section 3.2.1 and we define 4 translations:

1. A translation $\phi \mapsto \phi^{\mathrm{RAH}}$ of formulas which is a relativization to the first-order quantifiers to the set $\mathbb{B}$ and to the second-order quantifiers to the class $\mathbf{S}(\mathrm{RAH})$.
2. A translation $t \mapsto t^*$ of individuals already done in Section 2.2.3, each individual t is translated to a proof term $t^* : t \in \mathbb{B}$.
3. A translation $E \mapsto E^*$ of sets, each set E is translated into a proof term $E : E \in \mathbf{S}(\mathrm{RAH})$. This part remains as future work.

4. A translation $M \mapsto M^*$ from the proof terms of $\boldsymbol{\lambda}\mathbf{PA2}_{\mathbf{AC}}$ into the proof terms of $\boldsymbol{\lambda}\mathbf{PA2}^-_{\mathbf{Coll}}$. This part, depending on the previous one, also remains as a future work.

The translation $\phi \mapsto \phi^{\mathrm{RAH}}$ from formulas of $\boldsymbol{\lambda}\mathbf{PA2}_{\mathbf{AC}}$ intro formulas of $\boldsymbol{\lambda}\mathbf{PA2}^-_{\mathbf{Coll}}$ is defined as follow

$$\begin{array}{rcl}
(t \in X)^{\mathrm{RAH}} & \triangleq & t \in X \\
\big(\mathrm{null}(t)\big)^{\mathrm{RAH}} & \triangleq & \mathrm{null}(t) \\
(\bot)^{\mathrm{RAH}} & \triangleq & \forall Z(0 \in Z) \\
(t = u)^{\mathrm{RAH}} & \triangleq & \forall Z(t \in Z \Rightarrow u \in Z) \\
(X \leq_{\mathrm{U}} Y)^{\mathrm{RAH}} & \triangleq & X \leq_{\mathrm{RAH}} Y \\
(\phi \Rightarrow \psi)^{\mathrm{RAH}} & \triangleq & \phi^{\mathrm{RAH}} \Rightarrow \psi^{\mathrm{RAH}} \\
(\forall x \phi)^{\mathrm{RAH}} & \triangleq & (\forall x \in \mathbb{B})\phi^{\mathrm{RAH}} \\
(\forall X \phi)^{\mathrm{RAH}} & \triangleq & \big(\forall X \in \mathbf{S}(\mathrm{RAH})\big)\phi^{\mathrm{RAH}}
\end{array}$$

Lemma 5.5.3.1. *If $\phi \simeq \phi'$ (in $\boldsymbol{\lambda}\mathbf{PA2}_{\mathbf{AC}}$), then $\phi^{\mathrm{RAH}} \simeq \phi'^{\mathrm{RAH}}$ (in $\boldsymbol{\lambda}\mathbf{PA2}^-_{\mathbf{Coll}}$).*

This translation extends to contexts and sets as follow:

$$\{x \mid \phi\}^{\mathrm{RAH}} \triangleq \{x \mid \phi^{\mathrm{RAH}}\} \quad \text{and} \quad (\xi_1 : \phi_1, ..., \xi_n : \phi_n)^{\mathrm{RAH}} \triangleq \xi_1 : \phi_1^{\mathrm{RAH}}, ..., \xi_n : \phi_n^{\mathrm{RAH}}$$

We associate a new proof variable ξ_X to each second-order variable X. From each finite list of second-order variables $\vec{X} \triangleq (X_1, ..., Xp)$, we associate a context $\Xi_{\vec{X}}$ defined by

$$\Xi_{\vec{X}} \triangleq \xi_{X_1} : X_1 \in \mathbf{S}(\mathrm{RAH}), ..., \xi_{X_p} : X_p \in \mathbf{S}(\mathrm{RAH}).$$

Future work 1. *For each set $E \triangleq \{x \mid \phi\}$ (in $\boldsymbol{\lambda}\mathbf{PA2}_{\mathbf{AC}}$), of free variables among $\vec{x} = x_1, ..., x_p$ and $\overrightarrow{X} = X_1, ..., X_q$, there exists a proof term E^* (in $\boldsymbol{\lambda}\mathbf{PA2}^-_{\mathbf{Coll}}$) with free variables among $\xi_{x_1}, ..., \xi_{x_p}, \xi_{X_1}, ..., \xi_{X_q}$ such that:*

$$\Xi_{\vec{x}}, \Xi_{\overrightarrow{X}} \vdash E^* : E^{\mathrm{RAH}} \in \mathbf{S}(\mathrm{RAH})^{\mathbb{B}}$$

The proof terms E^* should be constructed by induction on the formula ϕ underlying the set E, by carefully analyzing the proof of the reflection principle (Theorem 5.4.1.1).

Finally, a proof term M (in $\boldsymbol{\lambda}\mathbf{PA2}_{\mathbf{AC}}$) with free variables among $\vec{\xi}, \vec{x}, \overrightarrow{X}$ should be translated into a proof term M^* (in $\boldsymbol{\lambda}\mathbf{PA2}^-_{\mathbf{Coll}}$) with free variables among $\vec{\xi}, \xi_{\vec{x}}, \xi_{\overrightarrow{X}}$:

$$\begin{array}{rcl}
\xi^* & \triangleq & \xi \\
\mathrm{efq}(M, \phi)^* & \triangleq & M* \\
\mathrm{cc}(\phi, \psi)^* & \triangleq & \mathrm{cc} \\
\mathrm{ind}(E, M, N, t)^* & \triangleq & \mathrm{ind}^* M^* N^* t^* \\
\mathrm{refl}(t)^* & \triangleq & \lambda\xi.\xi \\
\mathrm{peel}(M, E, N)^* & \triangleq & M^* N^* \\
(\lambda\xi : \phi.M)^* & \triangleq & \lambda\xi.M^* \\
(MN)^* & \triangleq & M^* N^* \\
(\lambda x.M)^* & \triangleq & \lambda\xi_x.M^* \\
(Mt)^* & \triangleq & M^* t^* \\
(\Lambda X.M)^* & \triangleq & \lambda\xi_X.M^* \\
(ME)^* & \triangleq & M^* E^* \\
(\mathrm{ord}_{\leq_{\mathrm{U}}})^* & \triangleq & _ \\
(\mathrm{wo}_{\leq_{\mathrm{U}}, \phi})^* & \triangleq & _
\end{array}$$

However, the translations of the proof terms $\mathrm{ord}_{\leq_{\mathrm{U}}}$ and $\mathrm{wo}_{\leq_{\mathrm{U}},\phi}$ remain as future works.

Future work 2. *There exists a translation from the proof terms of* $\boldsymbol{\lambda}\mathbf{PA2}_{\mathbf{AC}}$ *into the proof terms of* $\boldsymbol{\lambda}\mathbf{PA2}^{-}_{\mathbf{Coll}}$ *such that* $\Gamma \vdash M : \phi$ *(in* $\boldsymbol{\lambda}\mathbf{PA2}_{\mathbf{AC}}$*) implies*

$$\Gamma^{\mathrm{RAH}}, \Xi_{\vec{x}}, \Xi_{\overrightarrow{X}} \vdash M^* : \phi^{\mathrm{RAH}} \qquad (\text{in } \boldsymbol{\lambda}\mathbf{PA2}^{-}_{\mathbf{Coll}})$$

where $\vec{x}$ *and* $\overrightarrow{X}$ *are the free first-order and second-order variables of* M*.*

This unfinished translation is very close to the negative translation described in Section 3.2.1. It interprets a Church-style λ-calculus inside a Curry-style λ-calculus, giving a computational meaning to the logical parts of proof terms of the system $\boldsymbol{\lambda}\mathbf{PA2}_{\mathbf{AC}}$. While the negative translation interprets sets as decidable predicates, this translation would interpret sets as elements of the ramified analytic hierarchy.

5.5.4 Motivation behind this translation

The target system has a well-known computational content, thanks to Krivine realizability. From a study of the image of the translation, we can imagine to extract reduction rules for the terms $\mathrm{ord}_{\leq_{\mathrm{U}}}$ and $\mathrm{wo}_{\leq_{\mathrm{U}},\phi}$. This could lead to the design of a classical type system incorporating the notion of constructibility and in which the principle of the well-ordered universe would be provable. The computational properties (strong normalization...) of such a system would be directly ensured by the translation. All in all, with this work, we could design a "calculus of constructibles" allowing to prove the full (and extensional) axiom of choice in a classical setting. However, this goal is very ambitious and is still quite far away right now.

Part II

Study of extension of equality in Higher Type Arithmetic

Introduction

Higher type arithmetic ($\mathbf{HA}^\omega$) is a first-order many-sorted theory. It is a conservative extension of Heyting arithmetic obtained by extending the syntax of terms to all of System T: the objects of interest here are the functionals of "higher types". While equality between natural numbers is specified by the axioms of Peano, how can equality between functionals be defined? From this question, different versions of $\mathbf{HA}^\omega$ arise, such as an extensional version ($\mathbf{E}$-$\mathbf{HA}^\omega$) and an intentional version ($\mathbf{I}$-$\mathbf{HA}^\omega$). The initial plan of this part was to study both of these versions and to show that they can be interpreted in a system with equality only on the base type $\mathbf{N}$. The idea behind these interpretations is shared with the translations presented before in this report. Namely, it is to restrict the range of quantification, leading us to describe yet an other translation by relativization.

1. For extensional equality, the range of the quantifiers should be restricted to "extensional" elements. It is done in Chapter 6 where we fully describe a translation by parametricity from $\mathbf{E}$-$\mathbf{HA}^\omega$ into $\mathbf{HA}^\omega$. This work was presented in an international workshop [11]. In the next chapter of this thesis, we include the paper coming from this work without modification. Note that this paper was already published [11].

2. For intentional equality, the range of the quantifiers should be restricted to "generalized recursive functions". My work on this aspect led to various presentations [12] but I was not able to include it in this thesis as it is still an on-going work. The main intuition of this work was to formalize the model **HRO** [49] as a syntactic translation from a type system for a variant of $\mathbf{I}$-$\mathbf{HA}^\omega$ into a type system for **HA**. Through this translation, a functional of $\mathbf{HA}^\omega$ (described in the syntax by a closed term of System T) would be interpreted as a natural number representing a code for its normal form. This translation would justify the addition of a family of instructions $\mathtt{quote}_\sigma$ (of type $\sigma \to \mathbf{N}$) to the syntax of System T, implementing quoting operators that return the source code of the normal form of its argument. The soundness of such a system and its operational semantic would be fully justified in the meta-theory. Notably, the system obtained from $\mathbf{HA}^\omega$ by adding these new instructions is not proof-theoretically stronger than $\mathbf{HA}^\omega$ (and, therefore, than **HA**). In particular, even with the quoting operators in the syntax, one cannot prove internally that System T is strongly normalizing. I want to emphasize that these results are not new: a model for a theory (called **HRO**) incorporating all the previous features was already described internally in **HA** [49]. Our idea was to reformulate this work as a syntactic translation; leading to the design of a type system for $\mathbf{I}$-$\mathbf{HA}^\omega$. In such a system, an interpretation for the full axiom of choice could be obtained by adapting and extending the computational analysis of the axiom of countable choice done through the Berardi-Bezem-Coquand (BBC) functional[11] [7]. It remains as a future work to write down this research. Finally, Pédrot adapted these results from first-order logic into dependent type theory; thus showing the compatibility of Church Thesis within dependent type theory [44]. As a last remark, we mention that by extending and combining the works of Pédrot [44], of Herbelin [22] and of Miquey [38], it could be possible to give a computational interpretation of the full axiom of choice in a dependent type theory where all types are decidable[12].

[11] It is already known that their interpretation can be extended to any decidable types.
[12] It is already known that the work of Herbelin and Miquey can be extended to any decidable types [39].

Chapter 6

An interpretation of E-HA$^\omega$ inside HA$^\omega$

In second-order logic, it can be shown as a meta-theorem that two extensionally equal predicates satisfy the same properties. It is not the case in higher-order logic: this is due to the potential existence of non extensional (higher-order) predicates. However, Gandy showed that axioms of extensionality could be consistently added by restraining the range of quantification to extensional elements [18]. A similar phenomenon occurs in higher type arithmetic (**HA**$^\omega$): one cannot prove in **HA**$^\omega$ that two extensionally equal functions satisfy the same formulas. It can be seen for instance by working in the model of Hereditary Recursive Operations **HRO** [49] where a functional can inspect the source code of its argument. But, again, axioms of extensionality can be added without loss of consistency: Zucker showed that every model of **N-HA**$^\omega$ (higher type arithmetic with equality at all levels of sort) can be turned into a model of **E-HA**$^\omega$ (higher type arithmetic with extensional equality at all levels of sort) [54].

In this work we tackle a similar problem. Starting from **HA**$^\omega$, we show that an extensional equality can be consistently added at all levels of sorts. Taking inspiration from syntactical models of type theory [9, 1], we chose to do it in a syntactical fashion: we design an interpretation of **E-HA**$^\omega$ in **HA**$^\omega$ that we express as a translation between two proof systems (without reduction rules). Concretely, we will compile a language with extensional equality at all levels of sorts to a language that merely has equality in the sort **N**. It will be done using techniques of parametricity, as one goal of this paper is to emphasize that parametricity can be used to extend equality.

After exposing a proof system λ**HA**$^\omega$ that captures higher type arithmetic (Section 6.1), we will study families (indexed by the sorts of System T) of (internal) partial equivalence relations that could be used to extend equality (Section 6.2). In particular, we will compare two potential candidates:

1. a family $=_\sigma^{\mathrm{ext}}$ generated from equality (over **N**) in an extensional fashion,
2. a family $=_\sigma^{\mathrm{pm}}$ generated from equality (over **N**) in a way reminiscent of binary parametricity [45].

While the former is reflexive, the latter is not. But being reflexive is not desirable in this context. Indeed, as explained above, one needs to restrict the range of quantifications before extending equality, specifically we will restrict quantifications on a sort σ to the domain of $=_\sigma^{\mathrm{pm}}$. Our first translation will be used to show that each closed term of System T is indeed in the domain of $=^{\mathrm{pm}}$: we translate judgments of System T into judgments of λ**HA**$^\omega$and we follow the typical

$$\frac{}{\Delta_1, x^\sigma, \Delta_2 \vdash_{\mathrm{T}} x^\sigma : \sigma}$$

$$\frac{\Delta, x^\sigma \vdash_{\mathrm{T}} t : \tau}{\Delta \vdash_{\mathrm{T}} \lambda x^\sigma.t : \sigma \to \tau} \qquad \frac{\Delta \vdash_{\mathrm{T}} t : \sigma \to \tau \quad \Delta \vdash_{\mathrm{T}} u : \sigma}{\Delta \vdash_{\mathrm{T}} tu : \tau}$$

$$\frac{}{\Delta \vdash_{\mathrm{T}} 0 : \mathbf{N}} \qquad \frac{\Delta \vdash_{\mathrm{T}} t : \mathbf{N}}{\Delta \vdash_{\mathrm{T}} \mathtt{s}\, t : \mathbf{N}}$$

$$\frac{\Delta \vdash_{\mathrm{T}} t : \sigma \quad \Delta \vdash_{\mathrm{T}} u : \sigma \to \mathbf{N} \to \sigma \quad \Delta \vdash_{\mathrm{T}} v : \mathbf{N}}{\Delta \vdash_{\mathrm{T}} \mathrm{Rec}^\sigma\, t\, u\, v : \sigma}$$

Figure 6.1: Derivation in System T

translation by parametricity, as it will allow us to show that typed terms satisfy the relation linked to their type [45, 52, 8]. Finally, by keeping the idea of a translation by parametricity, we will translate a proof system λ**E-HA**$^\omega$ (capturing **E-HA**$^\omega$) to λ**HA**$^\omega$ (Section 6.3). Before concluding, we will compare our result and our methodology to related works (Section 6.4).

6.1 A proof system for higher type arithmetic

6.1.1 System T

We use a version of Gödel's System T obtained by extending simply typed λ-calculus (à la Church) with a type constant $\mathbf{N}$ and native constructors to use it. Terms, sorts and signatures of System T are described as follows:

Sorts	σ, τ	$::=$	$\mathbf{N} \mid \sigma \to \tau$
Terms	t, u	$::=$	$x^\sigma \mid \lambda x^\sigma.t \mid tu$
			$\mid 0 \mid \mathtt{s}\, t \mid \mathrm{Rec}^\sigma\, t\, u\, v$
Signatures	Δ	$::=$	$\emptyset \mid \Delta, x^\sigma$

System T is presented in Church's style so terms come associated with a unique sort. Nevertheless, we use a type system (see Figure 6.1 page 121) to take into account in which signature (or environment) a term is considered. We may omit sort annotations on variables.

We consider the following rules on terms

$$\begin{array}{rcl} (\lambda x.t)\, u & \succ & t[x ::= u] \\ \mathrm{Rec}\, t\ u\, 0 & \succ & t \\ \mathrm{Rec}\, t\ u\, (\mathtt{s}\, v) & \succ & u\, (\mathrm{Rec}\, t\, u\, v)\, v \end{array}$$

from which we generate reduction and congruence

$$t \quad \rightsquigarrow \quad u \quad \text{and} \quad t \quad \cong \quad u$$

as respectively the least reflexive, transitive and closed by congruence relation containing $\succ$ and the least closed by congruence equivalence relation containing $\succ$. We define a substitution θ to

be a finite function from variables to terms. The action of a substitution θ on terms, denoted $t[\theta]$, corresponds to the simultaneous substitutions of free variables x in the domain of θ by $\theta(x)$.

Meta-theoretical results about System T can be found in the book of Girard, Lafont and Taylor [19], for instance:

1. terms of System T are strongly normalizable,
2. closed normal terms of type $\mathbf{N}$ are of the form $\mathtt{s}^n\,0$, closed normal terms of type $\sigma \to \tau$ are of the form $\lambda x^\sigma.t$.

Finally, we will use the two following facts.

Fact 1. *A generalized version of the weakening rule is admissible for this system:*

$$\textit{if} \quad \Delta \subseteq \Delta' \textit{ and } \Delta \vdash_T t : \sigma \quad \textit{ then } \quad \Delta' \vdash_T t : \sigma$$

where $\Delta \subseteq \Delta'$ is interpreted as the set theoretic inclusion (while seeing signatures as sets).

Fact 2. *If θ is a substitution then $t \cong u$ implies $t[\theta] \cong u[\theta]$.*

6.1.2 Higher type arithmetic

Higher type arithmetic ($\mathbf{HA}^\omega$) is a theory of many-sorted first-order logic. It is a conservative extension of **HA** obtained by extending the term language to the System T. Models of $\mathbf{HA}^\omega$ are described in the book of Troelstra [49], in particular the following will be used in the sequel:

1. the set theoretic model $\mathbf{M}$ defined by

$$\begin{array}{rcl} \mathbf{M}_{\mathbf{N}} & \triangleq & \mathbb{N} \\ \mathbf{M}_{\sigma\to\tau} & \triangleq & \mathbf{M}_\tau^{\mathbf{M}_\sigma} \end{array}$$

2. the model of Hereditary Recursive Operations **HRO** defined by

$$\begin{array}{rcl} \mathbf{HRO}_{\mathbf{N}} & \triangleq & \mathbb{N} \\ \mathbf{HRO}_{\sigma\to\tau} & \triangleq & \{e \in \mathbb{N} \mid \forall n \in \mathbf{HRO}_\sigma\ \{e\}(n)\downarrow \in \mathbf{HRO}_\tau\} \end{array}$$

   where $\{e\}(n)\downarrow \in E$ means that the computation of the function of index e terminates on the input n and that the result of this computation is in E.

We define a proof system $\lambda\mathbf{HA}^\omega$ that captures $\mathbf{HA}^\omega$. Formulas, proof terms and contexts of $\lambda\mathbf{HA}^\omega$ are generated by the following grammar:

$$\begin{array}{lrcl}
\text{Formulas} & \Phi, \Psi & ::= & t = u \mid \bot \mid \mathrm{null}(t) \\
& & & \mid \Phi \Rightarrow \Psi \mid \Phi \wedge \Psi \\
& & & \mid \forall x^\sigma \Phi \mid \exists x^\sigma \Phi \\
\text{Proof terms} & M, N & ::= & \xi \mid \mathrm{refl}\, t \mid \mathrm{peel}^{t,u}(M, \hat{x}.\Phi, N) \mid \mathrm{efq}(M, \Phi) \\
& & & \mid \lambda\xi.M \mid M\,N \\
& & & \mid (M, N) \mid M.1 \mid M.2 \\
& & & \mid \lambda x^\sigma.M \mid M\,t \\
& & & \mid [t, M] \mid \mathrm{let}\ [x, \xi] := M \ \mathrm{in}\ N \\
& & & \mid \mathrm{Ind}(\hat{x}.\Phi, M, N, t) \\
\text{Contexts} & \Gamma & ::= & \emptyset \mid \Gamma, \xi : \Phi
\end{array}$$

$$\frac{(\Delta;\Gamma)\ \mathbf{wf}}{\Delta;\Gamma\vdash\xi:\Phi}\,(\xi:\phi\in\Gamma)\qquad\frac{\Delta;\Gamma\vdash M:\bot}{\Delta;\Gamma\vdash \mathrm{efq}(M,\Phi):\Phi}\,(FV(\Phi)\subseteq\Delta)\qquad\frac{\Delta;\Gamma\vdash M:\Phi}{\Delta;\Gamma\vdash M:\Psi}\,(\Phi\simeq\Psi)$$

$$\frac{\Delta;\Gamma,\xi:\Phi\vdash M:\Psi}{\Delta;\Gamma\vdash\lambda\xi.M:\Phi\Rightarrow\Psi}\qquad\frac{\Delta;\Gamma\vdash M:\Phi\Rightarrow\Psi\quad\Delta;\Gamma\vdash N:\Phi}{\Delta;\Gamma\vdash M\,N:\Psi}$$

$$\frac{\Delta;\Gamma\vdash M_1:\Phi_1\quad\Delta;\Gamma\vdash M_2:\Phi_2}{\Delta;\Gamma\vdash(M_1,M_2):\Phi_1\wedge\Phi_2}\qquad\frac{\Delta;\Gamma\vdash M:\Phi_1\wedge\Phi_2}{\Delta;\Gamma\vdash M.i:\Phi_i}\,(i=1,2)$$

$$\frac{\Delta,x^\sigma;\Gamma\vdash M:\Phi}{\Delta;\Gamma\vdash\lambda x^\sigma.M:\forall x^\sigma\Phi}\,(x^\sigma\notin FV(\Gamma))\qquad\frac{\Delta;\Gamma\vdash M:\forall x^\sigma\Phi\quad\Delta\vdash_{\mathrm{T}}t:\sigma}{\Delta;\Gamma\vdash M\,t:\Phi[x^\sigma:=t]}$$

$$\frac{\Delta;\Gamma\vdash M:\Phi[x^\sigma:=t]\quad\Delta\vdash_{\mathrm{T}}t:\sigma}{\Delta;\Gamma\vdash[t,M]:\exists x^\sigma\Phi}\qquad\frac{\Delta;\Gamma\vdash M:\exists x^\sigma\Phi\quad\Delta,x^\sigma;\Gamma,\xi:\Phi\vdash N:\Psi}{\Delta;\Gamma\vdash\mathrm{let}\ [x,\xi]:=M\ \mathrm{in}\ N:\Psi}\,(x^\sigma\notin FV(\Gamma,\Psi))$$

$$\frac{(\Delta;\Gamma)\ \mathbf{wf}\quad\Delta\vdash_{\mathrm{T}}t:\mathbf{N}}{\Delta;\Gamma\vdash\mathrm{refl}\,t:t=t}\qquad\frac{\Delta;\Gamma\vdash M:t=u\quad\Delta;\Gamma\vdash N:\Phi[x^{\mathbf{N}}:=t]}{\Delta;\Gamma\vdash\mathrm{peel}^{t,u}(M,\hat{x}.\Phi,N):\Phi[x^{\mathbf{N}}:=u]}$$

$$\frac{\Delta;\Gamma\vdash M:\Phi[x^{\mathbf{N}}:=0]\quad\Delta;\Gamma\vdash N:\forall x^{\mathbf{N}}.(\Phi\Rightarrow\Phi[x^{\mathbf{N}}:=\mathtt{s}\,x^{\mathbf{N}}])\quad\Delta\vdash_{\mathrm{T}}t:\mathbf{N}}{\Delta;\Gamma\vdash\mathrm{Ind}(\hat{x}.\Phi,M,N,t):\Phi[x:=t]}$$

Figure 6.2: Proof derivations in $\lambda\mathbf{HA}^\omega$

This syntax contains three different λ-abstractions: two λ-abstractions at the level of proof terms ($\lambda\xi.M$ and $\lambda x^\sigma.M$) and the λ-abstraction of System T at the level of formulas ($\lambda x^\sigma.t$). The sort annotation on a variable may be omitted in the sequel if it can be inferred. In the proof terms $\mathrm{peel}^{t,u}(M,\hat{x}.\Phi,N)$ and $\mathrm{Ind}(\hat{x}.\Phi,M,N,t)$, the variable x is bound in Φ: the binder $\hat{x}$ is used to specify which variable will be substituted. The connectives $\top$ and $\vee$ are not included in $\lambda\mathbf{HA}^\omega$ but can be defined as $\top\triangleq\bot\Rightarrow\bot$ and $\Phi\vee\Psi\triangleq\exists x^{\mathbf{N}}\,(x=0\Rightarrow\Phi\wedge x\neq 0\Rightarrow\Psi)$ where the relation $x\neq y$ denotes $x=y\Rightarrow\bot$.

We consider sequents of the form $\Delta;\Gamma\vdash M:\Phi$ where

1. Δ is a signature of System T,
2. Γ is a context of $\lambda\mathbf{HA}^\omega$.

The typing rules of $\lambda\mathbf{HA}^\omega$ are presented in Figure 6.2 page 123. Note that equality is only defined on the sort $\mathbf{N}$. A pair of a signature and a context $(\Delta;\Gamma)$ is well formed when the free first-order variables of Γ are contained in Δ, i.e

$$(\Delta;\Gamma)\ \mathbf{wf}\quad\triangleq\quad FV(\Gamma)\quad\subseteq\quad\Delta.$$

This system is not equipped with reduction rules for proof terms: they are used as annotations for the derivation and they serve as a tool to formulate our work as a fully specified translation. The congruence relation $\Phi\simeq\Psi$ between formulas used in $\lambda\mathbf{HA}^\omega$ is generated from the reduction rules of System T and two extra rules:

$$\begin{array}{rcl}\mathrm{null}(0) & \succ & \top\\ \mathrm{null}(\mathtt{s}\,x) & \succ & \bot.\end{array}$$

Because of the rule of conversion, terms of System T are treated up to the equivalence $\cong$. For instance, one can prove $(\lambda x^{\mathbf{N}}.x)0 = 0$ in $\lambda\mathbf{HA}^\omega$. Moreover, all the axioms of $\mathbf{HA}^\omega$ [49] are derivable. In particular, the predicate $\mathrm{null}(t)$ is used to prove $\forall x^{\mathbf{N}}\ \mathtt{s}\, x \neq 0$.

Fact 3. *$\lambda\mathbf{HA}^\omega$ captures $\mathbf{HA}^\omega$ in the following manner: a closed formula Φ is derivable in $\lambda\mathbf{HA}^\omega$ if and only if the formula obtained from Φ by replacing all occurrences of subformulas $\mathrm{null}(t)$ by $t = 0$ is a logical consequence of $\mathbf{HA}^\omega$.*

The notion of substitutions extends from System T to $\lambda\mathbf{HA}^\omega$. Concretely, a (first-order) substitution θ is a finite function from first-order variables $(x, y...)$ to terms of System T while the operation of substitutions on proof terms M and formulas Φ is defined as before. The notation $\Gamma[\theta]$ is used to denote the application of the substitution θ to each terms and formulas in the context Γ. The system $\lambda\mathbf{HA}^\omega$ satisfies the following properties:

Fact 4. *If $\Delta; \Gamma \vdash M : \Phi$ then $FV(\Phi) \subseteq \Delta$.*

Fact 5. *A generalized version of the weakening rule is admissible for this system:*

$$\textit{if} \quad \Delta \subseteq \Delta',\ \Gamma \subseteq \Gamma' \ \textit{and}\ \Delta; \Gamma \vdash M : \Phi \quad \textit{then} \quad \Delta'; \Gamma' \vdash M : \Phi$$

where the set-theoretic inclusion is used to compare signatures and contexts.

Fact 6. *Let θ be a substitution of first-order variables, Δ a signature included in its domain and Δ' a signature containing all free variables of its image then*

$$\Delta; \Gamma \vdash M : \Phi \quad \textit{implies} \quad \Delta'; \Gamma[\theta] \vdash M[\theta] : \Phi[\theta].$$

Fact 7. *If $\Delta; \Gamma, \xi : \Psi \vdash M : \Phi$ and $\Delta; \Gamma \vdash N : \Psi$ then $\Delta; \Gamma \vdash M[\xi := N] : \Phi$.*

6.2 A preliminary study of possible extensions of equality

6.2.1 Two examples of Partial Equivalence Relation

Let σ be a sort of System T. A symbol of binary relation $\mathcal{R}$ on σ (added to the syntax of $\lambda\mathbf{HA}^\omega$) is a partial equivalence relation when it is symmetric and transitive. It is the case if the formulas

$$\begin{array}{rcl} \mathrm{Sym}_{\mathcal{R}} & \triangleq & \forall x^\sigma \forall y^\sigma\ x\mathcal{R}y \Rightarrow y\mathcal{R}x \\ \mathrm{Trans}_{\mathcal{R}} & \triangleq & \forall x^\sigma \forall y^\sigma \forall z^\sigma\ x\mathcal{R}y \Rightarrow y\mathcal{R}z \Rightarrow x\mathcal{R}z \end{array}$$

are provable in $\lambda\mathbf{HA}^\omega$.

A partial equivalence relation is an equivalence relation on its domain

$$x \in \mathrm{Dom}_{\mathcal{R}} \quad \triangleq \quad x\mathcal{R}x.$$

Moreover, using symmetry and transitivity, one can show that

$$x\mathcal{R}y \Rightarrow x \in \mathrm{Dom}_{\mathcal{R}} \wedge y \in \mathrm{Dom}_{\mathcal{R}}.$$

Therefore, a partial equivalence relation on σ is exactly an equivalence relation on a collection of individuals of sort σ (that is a formula with one free variable of sort σ).

Let $\{=_\sigma^{\mathrm{ext}}\}_\sigma$ and $\{=_\sigma^{\mathrm{pm}}\}_\sigma$ be two families of binary relations indexed by the sorts of System T and defined as follows:

$$\begin{array}{rcl} x^{\mathbf{N}} =_{\mathbf{N}}^{\mathrm{ext}} y^{\mathbf{N}} & \triangleq & x = y \\ f^{\sigma\to\tau} =_{\sigma\to\tau}^{\mathrm{ext}} g^{\sigma\to\tau} & \triangleq & \forall x\ f\,x =_\tau^{\mathrm{ext}} g\,x \end{array} \qquad \begin{array}{rcl} x^{\mathbf{N}} =_{\mathbf{N}}^{\mathrm{pm}} y^{\mathbf{N}} & \triangleq & x = y \\ f^{\sigma\to\tau} =_{\sigma\to\tau}^{\mathrm{pm}} g^{\sigma\to\tau} & \triangleq & \forall x \forall y\ x =_\sigma^{\mathrm{pm}} y \Rightarrow f\,x =_\tau^{\mathrm{pm}} g\,y. \end{array}$$

Note that

1. The relation $=^{\mathrm{ext}}$ is obtained from equality by extending it to higher sorts in an extensional fashion (two functions are in $=^{\mathrm{ext}}$ if they are extensionally equal).
2. The relation $=^{\mathrm{pm}}$ is obtained from equality by extending it to higher sorts in a parametric fashion (two functions are in $=^{\mathrm{pm}}$ if they send related entries to related outputs).

With an (external) induction on the sorts of System T, it can be shown that for all sorts σ:

1. $=^{\mathrm{ext}}_{\sigma}$ is an equivalence relation.
2. $=^{\mathrm{pm}}_{\sigma}$ is a partial equivalence relation.

We exhibit proof terms that are used on forthcoming translations:

$$\begin{array}{rcl} \vdash & \mathrm{sym}^{\mathrm{pm}}_{\sigma} & : \quad \mathrm{Sym}_{=^{\mathrm{pm}}_{\sigma}} \\ \vdash & \mathrm{trans}^{\mathrm{pm}}_{\sigma} & : \quad \mathrm{Trans}_{=^{\mathrm{pm}}_{\sigma}} \\ \vdash & \mathrm{refl}^{\mathrm{pm}}_{\sigma} & : \quad \forall x^{\sigma} \forall y^{\sigma}\, x =^{\mathrm{pm}}_{\sigma} y \Rightarrow (x =^{\mathrm{pm}}_{\sigma} x \wedge y =^{\mathrm{pm}}_{\sigma} y) \end{array}$$

They are defined by induction on the sort of System T:

$$\begin{array}{rcl} \mathrm{sym}^{\mathrm{pm}}_{\mathbf{N}} & \triangleq & \lambda x \lambda y.\lambda \xi.\mathrm{peel}(\xi, \hat{z}.(z = x), \mathrm{refl}\, x) \\ \mathrm{sym}^{\mathrm{pm}}_{\sigma\to\tau} & \triangleq & \lambda f \lambda g.\lambda \xi.\lambda x \lambda y.\lambda \eta.\mathrm{sym}^{\mathrm{pm}}_{\tau}(f\, y)(g\, x)(\xi\, y\, x\, (\mathrm{sym}^{\mathrm{pm}}_{\sigma}\, x\, y\, \eta)) \\ \mathrm{trans}^{\mathrm{pm}}_{\mathbf{N}} & \triangleq & \lambda x \lambda y \lambda z.\lambda \xi \lambda \eta.\mathrm{peel}(\eta, \hat{w}.x = w, \xi) \\ \mathrm{trans}^{\mathrm{pm}}_{\sigma\to\tau} & \triangleq & \lambda f \lambda g \lambda h.\lambda \xi \lambda \eta.\lambda x \lambda y.\lambda \chi.\mathrm{trans}_{\tau}(f\, x)(g\, y)(h\, y)(\xi\, x\, y\, \chi)(\eta\, y\, y(\mathrm{trans}_{\sigma}\, y\, x\, y(\mathrm{sym}^{\mathrm{pm}}_{\sigma} x\, y\, \chi)\chi)) \\ \mathrm{refl}^{\mathrm{pm}}_{\sigma} & \triangleq & \lambda x^{\sigma} \lambda y^{\sigma} \lambda \xi.(\mathrm{trans}^{\mathrm{pm}}_{\sigma}\, x\, y\, x\, \xi(\mathrm{sym}^{\mathrm{pm}}\, x\, y\, \xi), \mathrm{trans}^{\mathrm{pm}}_{\sigma}\, y\, x\, y\, (\mathrm{sym}^{\mathrm{pm}}\, x\, y\, \xi)\xi). \end{array}$$

One cannot prove inside $\lambda\mathbf{HA}^{\omega}$ that $=^{\mathrm{pm}}_{\sigma}$ is reflexive for all sorts σ as it can be seen by working in **HRO**. Let

$$\mathtt{quote} \in \mathbf{HRO}_{(\mathbf{N}\to\mathbf{N})\to\mathbf{N}}$$

be an index for the identity function[1] and

$$p, q \in \mathbf{HRO}_{\mathbf{N}\to\mathbf{N}}$$

two distinct indexes for the same total unary function. Note that

$$\begin{array}{rcl} \mathbf{HRO} & \vDash & p =^{\mathrm{pm}}_{\mathbf{N}\to\mathbf{N}} q \\ \mathbf{HRO} & \vDash & \{\mathtt{quote}\}(p) \neq^{\mathrm{pm}}_{\mathbf{N}} \{\mathtt{quote}\}(q). \end{array}$$

Consequently

$$\mathbf{HRO} \quad \vDash \quad \mathtt{quote} \neq^{\mathrm{pm}}_{(\mathbf{N}\to\mathbf{N})\to\mathbf{N}} \mathtt{quote}$$

and $=^{\mathrm{pm}}_{(\mathbf{N}\to\mathbf{N})\to\mathbf{N}}$ is not reflexive in **HRO**.

Because $=^{\mathrm{ext}}_{(\mathbf{N}\to\mathbf{N})\to\mathbf{N}}$ is reflexive, the previous result shows that in **HRO**, $=^{\mathrm{ext}}_{(\mathbf{N}\to\mathbf{N})\to\mathbf{N}}$ is not included in $=^{\mathrm{pm}}_{(\mathbf{N}\to\mathbf{N})\to\mathbf{N}}$. Therefore, one cannot prove in $\lambda\mathbf{HA}^{\omega}$

$$\forall x \forall y\; x =^{\mathrm{ext}}_{(\mathbf{N}\to\mathbf{N})\to\mathbf{N}} y \Rightarrow x =^{\mathrm{pm}}_{(\mathbf{N}\to\mathbf{N})\to\mathbf{N}} y.$$

[1] `quote` is a functional that takes a function as argument and returns its source code. Here the index of the identity function is used as a polymorphic object. For all sorts σ, it can denote in $\mathbf{HRO}_{\sigma\to\mathbf{N}}$ a functional that returns the source code of its argument ; thus giving a possible interpretation for $\mathtt{quote}_{\sigma}$.

Going one step higher in the hierarchy of sorts, one can show that

$$\forall x \forall y\ x =^{\mathrm{pm}}_{((\mathbf{N}\to\mathbf{N})\to\mathbf{N})\to\mathbf{N}} y \Rightarrow x =^{\mathrm{ext}}_{((\mathbf{N}\to\mathbf{N})\to\mathbf{N})\to\mathbf{N}} y$$

is not provable in $\lambda\mathbf{HA}^\omega$. Indeed, consider a variant $\mathbf{HRO^o}$ of $\mathbf{HRO}$ where natural numbers denote recursive functions that can access an oracle deciding if its entry is an index of the identity function[2] (i.e for instance it returns 1 if it accepts and 0 otherwise). Let

$$n \in \mathbf{HRO^o}_{((\mathbf{N}\to\mathbf{N})\to\mathbf{N})\to\mathbf{N}}$$

be an index for this oracle and

$$m \in \mathbf{HRO^o}_{((\mathbf{N}\to\mathbf{N})\to\mathbf{N})\to\mathbf{N}}$$

be an index of the constant function $x \mapsto 0$. It turns out that

$$\begin{array}{rcl} \mathbf{HRO^o} & \vDash & n =^{\mathrm{pm}}_{(\mathbf{N}\to\mathbf{N})\to\mathbf{N})\to\mathbf{N}} m \\ \mathbf{HRO^o} & \vDash & n \neq^{\mathrm{ext}}_{(\mathbf{N}\to\mathbf{N})\to\mathbf{N})\to\mathbf{N}} m \end{array}$$

because indexes of the identity function are not in the domain of $=^{\mathrm{pm}}_{(\mathbf{N}\to\mathbf{N})\to\mathbf{N}}$.

Finally, in the set theoretic model $\mathbf{M}$ of $\mathbf{HA}^\omega$ (and in fact in any extensional model), one can show

$$\mathbf{M} \quad \vDash \quad \forall x \forall y\ x =^{\mathrm{ext}}_\sigma y \Leftrightarrow x =^{\mathrm{pm}}_\sigma y$$

for all σ (by induction on the sorts of System T).

Therefore, in $\lambda\mathbf{HA}^\omega$, one cannot prove that the relations $=^{\mathrm{pm}}$ and $=^{\mathrm{ext}}$ are different. We wrap up all these results in the following theorem.

Theorem 6.2.1.1. *In $\lambda\mathbf{HA}^\omega$ one cannot prove that*

1. $=^{\mathrm{pm}}_{(\mathbf{N}\to\mathbf{N})\to\mathbf{N}}$ *is reflexive*
2. $=^{\mathrm{ext}}_{(\mathbf{N}\to\mathbf{N})\to\mathbf{N}} \subseteq =^{\mathrm{pm}}_{(\mathbf{N}\to\mathbf{N})\to\mathbf{N}}$
3. $=^{\mathrm{pm}}_{((\mathbf{N}\to\mathbf{N})\to\mathbf{N})\to\mathbf{N}} \subseteq =^{\mathrm{ext}}_{((\mathbf{N}\to\mathbf{N})\to\mathbf{N})\to\mathbf{N}}$
4. $=^{\mathrm{pm}}_\sigma \subsetneq =^{\mathrm{ext}}_\sigma$ *and* $=^{\mathrm{ext}}_\sigma \subsetneq =^{\mathrm{pm}}_\sigma$ *(for any sort σ).*

where the symbols $\subseteq, \subsetneq$ and the property of being reflexive are defined inside $\lambda\mathbf{HA}^\omega$ in the usual way.

6.2.2 A first translation: from System T into $\lambda\mathbf{HA}^\omega$

Although one cannot prove inside $\lambda\mathbf{HA}^\omega$ that $=^{\mathrm{pm}}$ is reflexive, it can be shown that all closed terms of System T are in its domain. With this goal in mind, we design a translation from System T into $\lambda\mathbf{HA}^\omega$:

$$(\Delta \vdash_{\mathrm{T}} t : \sigma)^{\mathrm{pm}} \quad \rightsquigarrow \quad \Delta^1, \Delta^2; \Delta^{\mathrm{pm}} \vdash t^{\mathrm{pm}} : t^1 =^{\mathrm{pm}}_\sigma t^2.$$

1. Declarations of variables in signatures are duplicated. Fixing $i = 1, 2$:

$$\begin{array}{rcl} \emptyset^i & \triangleq & \emptyset \\ (\Delta, x^\sigma)^i & \triangleq & \Delta^i, (x^i)^\sigma \end{array}$$

[2] Here again, the index of the identity function is seen as an interpretation of `quote`.

where x^i are fresh distinct variables.

2. Terms of System T are duplicated. Fixing $i := 1, 2$:

$$t^i \quad \triangleq \quad t[\theta_t^i]$$

will denote the term obtained by substituting all free variables x of t by x^i (i.e θ_t^i is the substitution defined on the free variables of t and that associates to a variable x the variable x^i).

3. Signatures of System T are translated into contexts of $\lambda\mathbf{HA}^\omega$:

$$\begin{array}{rcl} \emptyset^{\mathrm{pm}} & \triangleq & \emptyset \\ (\Delta, x^\sigma)^{\mathrm{pm}} & \triangleq & \Delta^{\mathrm{pm}}, x^{\mathrm{pm}} : x^1 =_\sigma^{\mathrm{pm}} x^2. \end{array}$$

4. Terms of System T are translated into proof terms of $\lambda\mathbf{HA}^\omega$:

$$\begin{array}{rcl} (x)^{\mathrm{pm}} & \triangleq & x^{\mathrm{pm}} \\ (\lambda x.t)^{\mathrm{pm}} & \triangleq & \lambda x^1 \lambda x^2 . \lambda x^{\mathrm{pm}} . t^{\mathrm{pm}} \\ (t\,u)^{\mathrm{pm}} & \triangleq & t^{\mathrm{pm}}\, u^1\, u^2\, u^{\mathrm{pm}} \\ 0^{\mathrm{pm}} & \triangleq & \mathrm{refl}\, 0 \\ (\mathtt{s}\, t)^{\mathrm{pm}} & \triangleq & \mathrm{peel}\big(t^{\mathrm{pm}}, \hat{x}.(\mathtt{s}\, t^1 = \mathtt{s}\, x), \mathrm{refl}\,(\mathtt{s}\, t^1)\big) \\ (\mathrm{Rec}\, t\, u\, v)^{\mathrm{pm}} & \triangleq & \mathrm{Ind}\big(\hat{x}.(\forall y^{\mathbf{N}}\ x = y \Rightarrow \mathrm{Rec}^\sigma\, t^1\, u^1\, x =_\sigma^{\mathrm{pm}} \mathrm{Rec}^\sigma\, t^2\, u^2\, y), \\ & & \quad \lambda y.\lambda\xi.\mathrm{peel}(\xi, \hat{z}.(t^1 =_\sigma^{\mathrm{pm}} \mathrm{Rec}^\sigma\, t^2\, u^2\, z), t^{\mathrm{pm}}), \\ & & \quad \lambda x.\lambda\eta.\lambda y.\lambda\xi.\mathrm{peel}(\xi, \hat{z}.(u^1(\mathrm{Rec}\, t^1\, u^1\, x)x =_\sigma^{\mathrm{pm}} (\mathrm{Rec}\, t^2\, u^2\, z)), \\ & & \qquad\qquad u^{\mathrm{pm}}(\mathrm{Rec}\, t^1\, u^1\, x)(\mathrm{Rec}\, t^2\, u^2\, x)(\eta\, x\, (\mathrm{refl}\, x))\, x\, x\, (\mathrm{refl}\, x)), \\ & & \quad v^1)v^2\, v^{\mathrm{pm}} \end{array}$$

The translation works as follows:

1. An abstraction in System T is interpreted as 3 abstractions in $\lambda\mathbf{HA}^\omega$: 2 abstractions of first-order variables and one of proof variable. Indeed, $(\lambda x^\sigma.t)^{\mathrm{pm}}$ should be of type $\forall x^1 \forall x^2\, x^1 =_\sigma^{\mathrm{pm}} x^2 \Rightarrow t^1 =^{\mathrm{pm}} t^2$ (assuming $\lambda x^\sigma.t : \sigma \to \tau$).

2. Symmetrically, an application in System T is interpreted as 3 applications.

3. Because the relation $=_{\mathbf{N}}^{\mathrm{pm}}$ is merely the equality, 0 and $\mathtt{s}\, t$ are interpreted as equality proofs.

4. Finally, the recursor is interpreted using an induction. During the induction, the synchronization between the two copies of the term v (of sort $\mathbf{N}$) is lost. Therefore, we need an extra generalization in the hypothesis to retrieve that these terms are equal.

Theorem 6.2.2.1. *If $\Delta \vdash_T t : \sigma$ then $\Delta^1, \Delta^2; \Delta^{\mathrm{pm}} \vdash t^{\mathrm{pm}} : t^1 =_\sigma^{\mathrm{pm}} t^2$. In particular, $\vdash t^{\mathrm{pm}} : t =_\sigma^{\mathrm{pm}} t$ for all closed terms of sort σ.*

Proof. By induction on the derivations of System T. □

We deduce from the previous theorem that each closed terms of System T are in the domain of $=^{\mathrm{pm}}$.

The following terms are used lated:

$$\Delta^1, \Delta^2; \Delta^{\mathrm{pm}} \quad \vdash \quad \mathrm{Elim}^i_{\hat{z}.t} : \forall z^1 \forall z^2\ z^1 =_\sigma^{\mathrm{pm}} z^2 \Rightarrow t^i[z^i = z^1] =_\tau^{\mathrm{pm}} t^i[z^i = z^2].$$

for $i = 1, 2$. Note that these proof terms are indexed by a term t and a variable z.

These terms are constructed using the previous translation as follows:

$$\begin{array}{rcl}
\mathrm{Elim}^1_{\tilde{z}.t} & \triangleq & \lambda z^1 \lambda z^2 . \lambda z^{\mathrm{pm}} . \mathrm{trans}^{\mathrm{pm}}\, t^1\, t^2 t^1 [z^1 := z^2] t^{\mathrm{pm}} \\
& & \qquad (\mathrm{sym}^{\mathrm{pm}} t^1 [z^1 := z^2] t^2 t^{\mathrm{pm}} [z^1 := z^2][z^{\mathrm{pm}} := (\mathrm{refl}^{\mathrm{pm}} z^1 z^2 z^{\mathrm{pm}}).2]) \\
\mathrm{Elim}^2_{\tilde{z}.t} & \triangleq & \lambda z^1 \lambda z^2 . \lambda z^{\mathrm{pm}} . \mathrm{trans}^{\mathrm{pm}}\, t^2 [z^2 := z^1]\, t^1 t^2 \\
& & \qquad (\mathrm{sym}^{\mathrm{pm}} t^1 t^2 [z^2 := z^1] t^{\mathrm{pm}} [z^2 := z^1][z^{\mathrm{pm}} := (\mathrm{refl}^{\mathrm{pm}}\, z^1 z^2 z^{\mathrm{pm}}).1]) t^{\mathrm{pm}})
\end{array}$$

They are well typed as soon as $FV(t) \subseteq \Delta, z$.

6.3 Interpreting extensional equality: from λE-HA$^\omega$ into λHA$^\omega$

6.3.1 A preliminary step: a translation from λHA$^\omega$ into λHA$^\omega$

Although all closed terms of System T are in the domain of $=^{\mathrm{pm}}$, one cannot prove

$$\forall x^\sigma \; x =^{\mathrm{pm}}_\sigma x$$

in λ**HA**$^\omega$ (for $\sigma = (\mathbf{N} \to \mathbf{N}) \to \mathbf{N}$ for instance). Nevertheless, building on the intuition of restricting quantifications and on the work of the previous section, we can design a translation

$$(\Delta; \Gamma \vdash M : \Phi)^{\mathrm{pm}} \quad \rightsquigarrow \quad \Delta^1, \Delta^2; \Delta^{\mathrm{pm}}, \Gamma^{\mathrm{pm}} \vdash M^{\mathrm{pm}} : \Phi^{\mathrm{pm}}.$$

from λ**HA**$^\omega$ into itself that will serve as a basis to interpret extensional equality.

1. This translation extends the one defined formerly. In particular

   $$\Delta^i, \Delta^{\mathrm{pm}}, t^i \text{ and } t^{\mathrm{pm}}$$

   are already defined.

2. Formulas of λ**HA**$^\omega$ are translated into formulas of λ**HA**$^\omega$:

   $$\begin{array}{rcl}
   (t = u)^{\mathrm{pm}} & \triangleq & t^1 =^{\mathrm{pm}}_{\mathbf{N}} u^2 \\
   \bot^{\mathrm{pm}} & \triangleq & \bot \\
   (\Phi \Rightarrow \Psi)^{\mathrm{pm}} & \triangleq & \Phi^{\mathrm{pm}} \Rightarrow \Psi^{\mathrm{pm}} \\
   (\Phi \wedge \Psi)^{\mathrm{pm}} & \triangleq & \Phi^{\mathrm{pm}} \wedge \Psi^{\mathrm{pm}} \\
   (\forall x^\sigma \Phi)^{\mathrm{pm}} & \triangleq & \forall x^1 \forall x^2 \; x^1 =^{\mathrm{pm}}_\sigma x^2 \Rightarrow \Phi^{\mathrm{pm}} \\
   (\exists x^\sigma \Phi)^{\mathrm{pm}} & \triangleq & \exists x^1 \exists x^2 \; x^1 =^{\mathrm{pm}}_\sigma x^2 \wedge \Phi^{\mathrm{pm}}.
   \end{array}$$

3. Contexts of λ**HA**$^\omega$ are translated into contexts of λ**HA**$^\omega$:

   $$\begin{array}{rcl}
   \emptyset^{\mathrm{pm}} & \triangleq & \emptyset \\
   (\Gamma, \xi : \Phi)^{\mathrm{pm}} & \triangleq & \Gamma^{\mathrm{pm}}, \xi : \Phi^{\mathrm{pm}}.
   \end{array}$$

4. Proof terms of λ**HA**$^\omega$ are translated into proof terms of λ**HA**$^\omega$:

$$
\begin{array}{rcl}
(\xi)^{\mathrm{pm}} & \triangleq & \xi \\
(\lambda\xi.M)^{\mathrm{pm}} & \triangleq & \lambda\xi.M^{\mathrm{pm}} \\
(M\,N)^{\mathrm{pm}} & \triangleq & M^{\mathrm{pm}}N^{\mathrm{pm}} \\
(M,N)^{\mathrm{pm}} & \triangleq & (M^{\mathrm{pm}},N^{\mathrm{pm}}) \\
(M.i)^{\mathrm{pm}} & \triangleq & M^{\mathrm{pm}}.i \\
(\lambda x.M)^{\mathrm{pm}} & \triangleq & \lambda x^1\lambda x^2\lambda x^{\mathrm{pm}}.M^{\mathrm{pm}} \\
(M\,t)^{\mathrm{pm}} & \triangleq & M^{\mathrm{pm}}t^1t^2t^{\mathrm{pm}} \\
([t,M])^{\mathrm{pm}} & \triangleq & [t^1,[t^2,(t^{\mathrm{pm}},M^{\mathrm{pm}})]] \\
(\mathrm{let}\,[x,\xi]:=M\,\mathrm{in}\,N)^{\mathrm{pm}} & \triangleq & \mathrm{let}\,[x^1,\eta]:=M^{\mathrm{pm}}\,\mathrm{in}\,\mathrm{let}\,[x^2,\chi]:=\eta\,\mathrm{in}\,N^{\mathrm{pm}}[x^{\mathrm{pm}}=\chi.1][\xi^{\mathrm{pm}}:=\chi.2] \\
(\mathrm{efq}(M,\Phi))^{\mathrm{pm}} & \triangleq & \mathrm{efq}(M^{\mathrm{pm}},\Phi^{\mathrm{pm}}) \\
(\mathrm{refl}\,t)^{\mathrm{pm}} & \triangleq & t^{\mathrm{pm}} \\
\big(\mathrm{peel}^{t,u}(M,\hat{x}.\Phi,N)\big)^{\mathrm{pm}} & \triangleq & \mathrm{peel}\big(M_2^{\mathrm{pm}},\hat{x^2}.\Phi^{\mathrm{pm}}[x^1:=u^1],\mathrm{peel}(M_1^{\mathrm{pm}},\hat{x^1}.\Phi^{\mathrm{pm}}[x^2:=t^2],N^{\mathrm{pm}})\big) \\
\big(\mathrm{Ind}(\hat{x}.\Phi,M,N,t)\big)^{\mathrm{pm}} & \triangleq & \mathrm{Ind}(\hat{x}.\forall y\;x=y\Rightarrow\Phi^{\mathrm{pm}}[x^1:=x][x^2:=y], \\
 & & \quad \lambda y\lambda\xi.\mathrm{peel}(\xi,\hat{z}.\Phi^{\mathrm{pm}}[x^1:=0][x2:=z],M^{\mathrm{pm}}), \\
 & & \quad \lambda x\lambda\eta\lambda y\xi.\mathrm{peel}(\xi,\hat{z}.\Phi^{\mathrm{pm}}[x^1:=\mathtt{s}\,x][x^2:=z],N^{\mathrm{pm}}x\,x\,(\mathrm{refl}\,x)(\eta\,x\,(\mathrm{refl}\,x)), \\
 & & \quad t^1)\,t^2t^{\mathrm{pm}}
\end{array}
$$

where in the translation of $\mathrm{peel}(M,\hat{x}.\Phi,N)$, M_i^{pm} denotes a proof of $t^i=u^i$:

$$
\begin{array}{rcll}
M_1^{\mathrm{pm}} & \triangleq & \mathrm{trans}_{\mathbf{N}}^{\mathrm{pm}}t^1\,u^2\,u^1\,M^{\mathrm{pm}}(\mathrm{sym}_{\mathbf{N}}^{\mathrm{pm}}u^1\,u^2u^{\mathrm{pm}}) & : \quad t^1=u^1 \\
M_2^{\mathrm{pm}} & \triangleq & \mathrm{trans}_{\mathbf{N}}^{\mathrm{pm}}t^2\,t^1\,u^2(\mathrm{sym}_{\mathbf{N}}^{\mathrm{pm}}t^1\,t^2t^{\mathrm{pm}})M^{\mathrm{pm}} & : \quad t^2=u^2.
\end{array}
$$

Here, the translation of peel is ad hoc: it is merely done by using peel on two distinct equalities. However, it will not be the case in the last translation where we will need an external recursion on the formulas of the source system to interpret it.

The translation of induction follows the same principle as the translation of the recursor done in Section 6.2.2: because the synchronization between the copies of t is lost, we need to generalize the induction hypothesis.

Theorem 6.3.1.1. *If* $\Delta;\Gamma\vdash M:\Phi$ *then* $\Delta^1,\Delta^2;\Delta^{\mathrm{pm}},\Gamma^{\mathrm{pm}}\vdash M^{\mathrm{pm}}:\Phi^{\mathrm{pm}}$.

The proof of this theorem is by induction on the derivation of $\lambda\mathbf{HA}^\omega$ and it uses three lemmas:

Lemma 6.3.1.1. *If* $t\cong u$ *then* $t^i\cong u^i$ *for* $i=1,2$ *and* t,u *terms of System T.*

Lemma 6.3.1.2. *If* $t(x^\sigma)$ *and* u *are terms with* u *of sort* σ*,then* $\big(t[x:=u]\big)^i\triangleq t^i[x^i:=u^i]$ *for* $i=1,2$.

Lemma 6.3.1.3. *If* $\Phi(x^\sigma)$ *is a formula and* t *is a term of sort* σ*, then* $\big(\Phi[x:=t]\big)^{\mathrm{pm}}\triangleq\Phi^{\mathrm{pm}}[x^1:=t^1][x^2:=t^2]$.

6.3.2 Extending equality through parametricity: A translation from λE-HA$^\omega$ into λHA$^\omega$

Our next goal is to extend the previous translation to give an interpretation of extensional equality inside $\lambda\mathbf{HA}^\omega$.

Consider an extension $\lambda\mathbf{E}$-$\mathbf{HA}^\omega$ of $\lambda\mathbf{HA}^\omega$ obtained by extending equality in an extensional way to all higher sorts, i.e by adding

$$\frac{(\Delta;\Gamma)\ \mathbf{wf} \quad \Delta \vdash_{\mathrm{T}} t : \sigma}{\Delta;\Gamma \vdash_{\mathrm{e}} \mathrm{refl}_\sigma\, t : t =_\sigma t} \qquad \frac{\Delta;\Gamma \vdash_{\mathrm{e}} M : t =_\sigma u \quad \Delta;\Gamma \vdash_{\mathrm{e}} N : \Phi[x^\sigma := t]}{\Delta;\Gamma \vdash_{\mathrm{e}} \mathrm{peel}^{t,u}_\sigma(M, \hat{x}.\Phi, N) : \Phi[x^\sigma := u]}$$

$$\frac{\Delta;\Gamma \vdash_{\mathrm{e}} M : \forall x^\sigma\, f\, x =_\tau g\, x}{\Delta;\Gamma \vdash_{\mathrm{e}} \mathrm{ext}_{\sigma,\tau}(M) : f =_{\sigma\to\tau} g} \qquad \frac{\Delta;\Gamma \vdash_{\mathrm{e}} M : f =_{\sigma\to\tau} g \quad \Delta;\Gamma \vdash_{\mathrm{e}} N : t =_\sigma u}{\Delta;\Gamma \vdash_{\mathrm{e}} \mathrm{app}_{\sigma,\tau}(M, t, u, N) : f\, t =_\tau g\, u}$$

Figure 6.3: Additional typing rules for extensional equality

1. atomic formulas $t =_\sigma u$ for all sorts σ,
2. proof terms $(\mathrm{refl}_\sigma\, t)$, $\mathrm{peel}^{t,u}_\sigma(M, \hat{x}.\Phi, N)$, $\mathrm{ext}_{\sigma,\tau}(M)$ and $\mathrm{app}_{\sigma,\tau}(M, t, u, N)$ for all sort σ and τ,
3. typing rules for the added proof terms presented in Figure 6.3 page 130.

The symbol $\vdash_{\mathrm{e}}$ will be used to denote sequents (and provability) in λ**E-HA**$^\omega$. The translation $(\_)^{\mathrm{pm}}$ can be extended to a translation from λ**E-HA**$^\omega$ into λ**HA**$^\omega$. Indeed, one interprets

$$(t =_\sigma u)^{\mathrm{pm}} \quad \text{as} \quad t^1 =^{\mathrm{pm}}_\sigma u^2.$$

It is then possible to extend the translation with

$$(\mathrm{refl}_\sigma\, t)^{\mathrm{pm}} \quad \triangleq \quad t^{\mathrm{pm}}$$

and to preserve adequacy. The case of $\left(\mathrm{peel}^{t,u}_\sigma(M, \hat{x}.\Phi, N)\right)^{\mathrm{pm}}$ is more involved and it is treated below. We first construct a family of terms $\mathrm{Elim}_{\hat{x^\sigma}.\Phi}$ satisfying that if

$$FV(\Phi) \quad \subseteq \quad \Delta, x^{\mathrm{pm}}$$

then

$$\Delta^1, \Delta^2; \Delta^{\mathrm{pm}} \quad \vdash \quad \mathrm{Elim}_{\hat{x^\sigma}.\Phi} : \forall x^1 \forall x^2 \forall y^1 \forall y^2\ x^1 =^{\mathrm{pm}}_\sigma y^1 \Rightarrow x^2 =^{\mathrm{pm}}_\sigma y^2 \Rightarrow \Phi^{\mathrm{pm}} \Rightarrow \Phi^{\mathrm{pm}}[x^1 := y^1][x^2 := y^2].$$

It is done by induction on the syntax of formulas:

$$\begin{array}{rcl}
\mathrm{Elim}_{\hat{x}.t=_\sigma u} & \triangleq & \lambda x^1 \lambda x^2 \lambda y^1 \lambda y^2 \lambda \xi^1 \lambda \xi^2 \lambda \xi.\mathrm{trans}^{\mathrm{pm}}\, t^1[x^1 := y^1]\, t^1\, u^2[x^2 := y^2] \\
& & \qquad (\mathrm{Elim}^1_{\hat{x}.t}\ y^1\, x^1\, (\mathrm{sym}^{\mathrm{pm}}\, x^1\, y^1\, \xi^1)) \\
& & \qquad (\mathrm{trans}^{\mathrm{pm}}\, t^1\, u^2\, u^2[x^2 := y^2]\xi(\mathrm{Elim}^2_{\hat{x}.u}\ x^2\, y^2\, \xi^2)) \\
\mathrm{Elim}_{\hat{x}.\bot} & \triangleq & \lambda x^1 \lambda x^2 \lambda y^1 \lambda y^2 \lambda \xi^1 \lambda \xi^2 \lambda \xi.\xi \\
\mathrm{Elim}_{\hat{x}.(\Phi\Rightarrow\Psi)} & \triangleq & \lambda x^1 \lambda x^2 \lambda y^1 \lambda y^2 \lambda \xi^1 \lambda \xi^2 \lambda \xi.\lambda \eta.\mathrm{Elim}^+_{\hat{x}.\Psi}(\xi\ (\mathrm{Elim}^-\eta)) \\
\mathrm{Elim}_{\hat{x}.(\Phi\wedge\Psi)} & \triangleq & \lambda x^1 \lambda x^2 \lambda y^1 \lambda y^2 \lambda \xi^1 \lambda \xi^2 \lambda \xi.(\mathrm{Elim}^+_{\hat{x}.\Phi}\xi.1, \mathrm{Elim}^+_{\hat{x}.\Psi}\xi.2) \\
\mathrm{Elim}_{\hat{x}.(\forall z\Phi)} & \triangleq & \lambda x^1 \lambda x^2 \lambda y^1 \lambda y^2 \lambda \xi^1 \lambda \xi^2 \lambda \xi.\lambda z^1 \lambda z^2 \lambda z^{\mathrm{pm}}.\mathrm{Elim}_{\hat{x}.\Phi}(\xi\, z^1 z^2 z^{\mathrm{pm}}) \\
\mathrm{Elim}_{\hat{x}.(\exists z\Phi)} & \triangleq & \lambda x^1 \lambda x^2 \lambda y^1 \lambda y^2 \lambda \xi^1 \lambda \xi^2 \lambda \xi.\mathrm{let}\, [z^1, \eta] := \xi\, \mathrm{in}\, \mathrm{let}\, [z^2, \chi] := \eta\, \mathrm{in}\, [z^1, [z^2, (\eta.1, \mathrm{Elim}^+_{\hat{x}.\Phi}\eta.2)]]
\end{array}$$

where

$$\begin{array}{rcl}
\mathrm{Elim}^+_{\hat{x}.\Phi} & \triangleq & \mathrm{Elim}_{\hat{x}.\Phi}\, x^1\, x^2\, y^1\, y^2\, \xi^1\, \xi^2 \\
\mathrm{Elim}^- & \triangleq & \mathrm{Elim}_{\hat{y}.\Phi[x:=y]}\, y^1\, y^2\, x^1\, x^2\, (\mathrm{sym}^{\mathrm{pm}}\, x^1\, y^1\, \xi^1)(\mathrm{sym}^{\mathrm{pm}}\, x^2\, y^2\, \xi^2).
\end{array}$$

The proof that $\mathrm{Elim}_{\hat{x}.\Phi}$ satisfies the given property is by induction on the syntax, where the hypothesis is used to treat the case of equality.

We can now define

$$\begin{array}{rcl}\big(\mathrm{peel}^{t,u}_{\sigma}(M,\hat{x}.\Phi,N)\big)^{\mathrm{pm}} & \triangleq & \mathrm{Elim}_{\hat{x}.\Phi}t^1t^2u^1u^2(\mathrm{trans}^{\mathrm{pm}}t^1u^2u^1M^{\mathrm{pm}}(\mathrm{sym}^{\mathrm{pm}}u^1u^2u^{\mathrm{pm}}))\\ & & (\mathrm{trans}^{\mathrm{pm}}t^2t^1u^2(\mathrm{sym}^{\mathrm{pm}}t^1t^2t^{\mathrm{pm}})M^{\mathrm{pm}})N^{\mathrm{pm}}.\end{array}$$

Finally, we set

$$\begin{array}{rcl}\big(\mathrm{ext}_{\sigma\to\tau}(M)\big)^{\mathrm{pm}} & \triangleq & \lambda x^1,x^2\lambda x^{\mathrm{pm}}.M^{\mathrm{pm}}\,x^1\,x^2\,x^{\mathrm{pm}}\\ \big(\mathrm{app}_{\sigma\to\tau}(M,t,u,N)\big)^{\mathrm{pm}} & \triangleq & M^{\mathrm{pm}}\,t^1\,u^2\,N^{\mathrm{pm}}.\end{array}$$

Theorem 6.3.2.1. *If* $\Delta;\Gamma\vdash_{\mathrm{e}} M:\Phi$ *then* $\Delta^1,\Delta^2;\Delta^{\mathrm{pm}},\Gamma^{\mathrm{pm}}\vdash M^{\mathrm{pm}}:\Phi^{\mathrm{pm}}$.

Proof. The case of $\big(\mathrm{peel}_{\sigma}(M,\hat{x}.\Phi,N)\big)^{\mathrm{pm}}$ uses the property of $\mathrm{Elim}_{\hat{x}.\Phi}$ and a generalization of Fact 4 saying that if $\Delta;\Gamma\vdash_{\mathrm{e}} M:\Phi$ then $FV(\Phi)\subseteq\Delta$. □

Corollary 6.3.2.1. *If* $\lambda\mathbf{HA}^{\omega}$ *is consistent, then so is* $\lambda\mathbf{E}$-$\mathbf{HA}^{\omega}$.

Proof. If $\lambda\mathbf{E}$-$\mathbf{HA}^{\omega}$ is inconsistent, there exists a proof term M such that $\vdash_e M:\bot$. By the previous translation, one gets a derivation $\vdash M^{\mathrm{pm}}:\bot$ and concludes that $\lambda\mathbf{HA}^{\omega}$ is inconsistent. □

6.3.3 Characterizing the image of the translation

In the previous section, we showed that if a closed formula Φ is provable in $\lambda\mathbf{E}$-$\mathbf{HA}^{\omega}$ then Φ^{pm} is provable in $\lambda\mathbf{HA}^{\omega}$. The goal of this section is to prove the converse: if Φ^{pm} is provable in $\lambda\mathbf{HA}^{\omega}$ then Φ is provable in $\lambda\mathbf{E}$-$\mathbf{HA}^{\omega}$. It shows that the system $\lambda\mathbf{E}$-$\mathbf{HA}^{\omega}$ fully characterizes the image of the translation we designed.

We first show that the relation $=^{\mathrm{pm}}$ collapses to the equality relation in $\lambda\mathbf{E}$-$\mathbf{HA}^{\omega}$. For every sort σ, we construct a proof term

$$\vdash_e \mathrm{Collaps}_{\sigma}:\forall x^{\sigma}\forall y^{\sigma}\ x=_{\sigma}y\Leftrightarrow x=^{\mathrm{pm}}_{\sigma}y$$

by external induction on the sorts of System T:

$$\begin{array}{rcl}\mathrm{Collaps}_{\mathbf{N}} & \triangleq & \lambda x\lambda y\big(\lambda\xi.\xi,\lambda\xi.\xi\big)\\ \mathrm{Collaps}_{\sigma\to\tau} & \triangleq & \lambda f\lambda g(\lambda\xi\lambda x\lambda y\lambda\eta.\mathrm{Collaps}_{\tau}.1\,(f\,x)\,(g\,y)\,\mathrm{app}_{\sigma,\tau}(\xi,x,y,\mathrm{Collaps}_{\sigma}.2\,x\,y\,\eta),\\ & & \lambda\xi.\mathrm{ext}_{\sigma,\tau}(\lambda z.\mathrm{Collaps}_{\tau}.2\,(f\,z)\,(g\,z)(\xi\,z\,z\,(\mathrm{Collaps}_{\sigma}.1\,z\,z,(\mathrm{refl}\,z)))))\big)\end{array}$$

Proposition 6.3.3.1. *For every sort* σ*, the binary relations* $=^{\mathrm{pm}}_{\sigma}$ *and* $=^{\mathrm{ext}}_{\sigma}$ *collapse to* $=_{\sigma}$ *in* $\lambda\mathbf{E}$-$\mathbf{HA}^{\omega}$.

Proof. We proved that $=^{\mathrm{pm}}_{\sigma}$ collapses to $=_{\sigma}$ and it can be proved in a similar fashion that $=^{\mathrm{ext}}_{\sigma}$ also collapses to $=_{\sigma}$ in $\lambda\mathbf{E}$-$\mathbf{HA}^{\omega}$. □

Using this fact, we can now show that the image of the last translation is fully characterized by the type system $\lambda\mathbf{E}$-$\mathbf{HA}^{\omega}$.

We exhibit a family of proof terms Equiv^i_{Φ} for $i=1,2$ satisfying for any formula Φ and any signatures Δ containing the free variables of Φ

$$\begin{array}{rcl}\Delta^1,\Delta^2;\Delta^{\mathrm{pm}} & \vdash_e & \mathrm{Equiv}^1_{\Phi}:\Phi^1\Rightarrow\Phi^{\mathrm{pm}}\\ \Delta^1,\Delta^2;\Delta^{\mathrm{pm}} & \vdash_e & \mathrm{Equiv}^2_{\Phi}:\Phi^{\mathrm{pm}}\Rightarrow\Phi^1\end{array}$$

as follows:

$$\begin{array}{rcl}
\mathrm{Equiv}^1_{t=_\sigma u} & \triangleq & \lambda\xi.\mathrm{trans}_\sigma\, t^1\, u^1\, u^2(\mathrm{Collaps}_\sigma.1\, t^1\, u^1\, \xi)\, u^{\mathrm{pm}} \\
\mathrm{Equiv}^2_{t=_\sigma u} & \triangleq & \lambda\xi.\mathrm{Collaps}_\sigma.2\, t^1\, u^1\, (\mathrm{trans}_\sigma\, t^1\, u^2\, u^1\, \xi\, (\mathrm{sym}^{\mathrm{pm}}\, u^1\, u^2\, u^{\mathrm{pm}})) \\
\\
\mathrm{Equiv}^1_{\Phi\Rightarrow\Psi} & \triangleq & \lambda\xi\lambda\eta.\mathrm{Equiv}^1_\Psi(\xi\, (\mathrm{Equiv}^2_\Phi\, \eta)) \\
\mathrm{Equiv}^2_{\Phi\Rightarrow\Psi} & \triangleq & \lambda\xi\lambda\eta.\mathrm{Equiv}^2_\Psi(\xi\, (\mathrm{Equiv}^1_\Phi\, \eta)) \\
\\
\mathrm{Equiv}^1_{\Phi\wedge\Psi} & \triangleq & \lambda\xi.(\mathrm{Equiv}^1_\Phi\xi.1, \mathrm{Equiv}^1_\Psi\xi.2) \\
\mathrm{Equiv}^2_{\Phi\wedge\Psi} & \triangleq & \lambda\xi.(\mathrm{Equiv}^2_\Phi\xi.1, \mathrm{Equiv}^2_\Psi\xi.2) \\
\\
\mathrm{Equiv}^1_{\forall x^\sigma\,\Phi} & \triangleq & \lambda\xi\lambda x^1\lambda x^2.\lambda x^{\mathrm{pm}}.\mathrm{Equiv}^1_\Phi\, (\xi\, x^1) \\
\mathrm{Equiv}^2_{\forall x^\sigma\,\Phi} & \triangleq & \lambda\xi\lambda x^1.\mathrm{Equiv}^2_\Phi[x_2 := x_1][x^{\mathrm{pm}} := (\mathrm{Collaps}_\sigma.1\, x^1\, x^1\, (\mathrm{refl}_\sigma\, x^1))] \\
 & & \qquad (\xi\, x^1\, x^1\, (\mathrm{Collaps}_\sigma.1\, x^1\, x^1\, (\mathrm{refl}_\sigma\, x^1))) \\
\\
\mathrm{Equiv}^1_{\exists x^\sigma\,\Phi} & \triangleq & \lambda\xi.\mathrm{let}\,[x,\eta] := \xi\,\mathrm{in}\,[x, [x, (\mathrm{Collaps}_\sigma.1\, x\, x\, (\mathrm{refl}_\sigma\, x), \mathrm{Equiv}^1_\Phi\xi)]] \\
\mathrm{Equiv}^2_{\exists x^\sigma\,\Phi} & \triangleq & \lambda\xi.\mathrm{let}\,[x^1,\eta] := \xi\,\mathrm{in}\,\mathrm{let}\,[x^2,\chi] := \eta\,\mathrm{in}\,[x^1, \mathrm{Equiv}^2_\Phi\chi.2]
\end{array}$$

We can then conclude that a closed formula Φ is provable in λ**E-HA**$^\omega$ if and only if its translation is provable in λ**E-HA**$^\omega$.

Theorem 6.3.3.1. *For a closed formula Φ of λ**E-HA**$^\omega$:*

$$\vdash_e (\mathrm{Equiv}^1_\Phi, \mathrm{Equiv}^2_\Phi) : \Phi \Leftrightarrow \Phi^{\mathrm{pm}}.$$

*In particular, if Φ^{pm} is provable in λ**HA**$^\omega$ then Φ is provable in λ**E-HA**$^\omega$.*

6.3.4 Adding reduction rules: a conjecture

The proof systems used here lack of computational rules, such as

$$\begin{array}{rcl}
(\lambda x.M)\, t & \succ_\beta & M[x := t] \\
(\lambda\xi.M)\, N & \succ_\beta & M[\xi := N] \\
\mathrm{let}\,[x,\xi] := [t, M]\,\mathrm{in}\, N & \succ_\beta & N[x := t][\xi := M] \\
(M_1, M_2).i & \succ_\beta & M.i \\
\mathrm{Ind}(\hat{x}.\Phi, M, N, 0) & \succ_\iota & M \\
\mathrm{Ind}(\hat{x}.\Phi, M, N, S\, t) & \succ_\iota & N\, t\, \mathrm{Ind}(\hat{x}.\Phi, M, N, t) \\
\mathrm{peel}(\mathrm{refl}\, t, \hat{x}.\Phi, N) & \succ_\iota & N
\end{array}$$

and in λ**E-HA**$^\omega$

$$\mathrm{app}(\mathrm{ext}(M), t, t, \mathrm{refl}\, t) \quad \succ \quad M\, t.$$

One could try to figure out if the translation $(\_)^{\mathrm{pm}}$ respects reductions, i.e to obtain a result of the shape

$$M \rightsquigarrow N \quad \text{implies} \quad M^{\mathrm{pm}} \simeq_{\beta,\iota,\eta} N^{\mathrm{pm}}.$$

While it will be true for β-reductions, it seems that it will not be the case in general for the reduction of $\mathrm{Ind}(\hat{x}.\Phi, M, N, \mathtt{s}\, t)$ if t contains free first-order variables. Indeed, the term $\mathtt{s}\, t$ will be translated as a proof term asserting an equality and if it does not compute into $\mathrm{refl}\,(\mathtt{s}\, t)$, it will block the reduction of the subterm peel(...) inside $(\mathrm{Ind}(\hat{x}.\Phi, M, N, S\, t))^{\mathrm{pm}}$. In the case of a closed term t, we conjecture it will compute as desired. Concretely, we think that the proof will

use the meta properties of System T described at the end of Section 6.1.1: one will use that every closed term of sort $\mathbf{N}$ is β-equivalent to a term of the shape $\mathtt{s}^n\,0$ and that the translation of such a term computes into $\mathrm{refl}\,(\mathtt{s}^n\,0)$, assuming that the translation from System T to $\lambda\mathbf{HA}^\omega$respects reductions.

Conjecture 1. *If M is a proof term of λ**E-HA**$^\omega$ without free first-order variables and if M does not contain* peel *then*

$$M \rightsquigarrow N \quad \textit{implies} \quad M^{\mathrm{pm}} \simeq_{\beta,\iota} N^{\mathrm{pm}}.$$

However, the case of the reduction rule of peel seems more difficult to treat as the translation of peel relies on an external induction over the syntax but also because it uses proofs of symmetry and transitivity of $=^{\mathrm{pm}}$ that do not seem to compute as needed.

6.4 Related works

The idea to build a syntactic model satisfying extensionality axioms is already present in the work of Gandy [18]. Concretely, in Higher Order Logic, Gandy defined a syntactic model by restricting the elements of discourse to parametric ones and proved that, in this model, extensionally equal elements satisfy the same properties. Here, we adapt this construction to the theory $\mathbf{HA}^\omega$ and, using ideas of the Curry-Howard correspondence, we formulate it as a translation of proof systems. Our translation slightly differs from the one of Gandy because we use techniques from parametricity. This choice comes from the idea that parametric translation can serve to extend equality. Nevertheless, because extensionality and parametricity relations collapse to equality in an extensional model, it is somehow a matter of design.

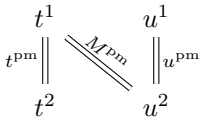
Zucker already proved a result of relative consistency between **E-HA**$^\omega$ and **N-HA**$^\omega$ [54, 49]. He did it in a semantical fashion by transforming models of **N-HA**$^\omega$ into models of **E-HA**$^\omega$. The method he used is similar to the method of Gandy but, in this context, it suffices to restrict the domain to parametric elements and to check that the relation of extensionality is an equivalence relation that is congruent (thus suited to interpret equality). In our work, rather from **N-HA**$^\omega$, we start with **HA**$^\omega$: we reconstruct the equality from scratch and show that it respects Leibniz principle. Despite this slight difference, our work can be seen as the syntactical counterpart of the result of Zucker. One advantage is that a syntactical translation comes with an explicit translation of proofs, that we formulate here as a translation of proof terms.

The ideas behind the proof system $\lambda\mathbf{HA}^\omega$ are folklore[3]. For instance, representing the axiom scheme of induction as an inference rule can be seen in many other proof systems as for instance in Martin-Löf Type Theory [33]. The idea to use the predicate $\mathrm{null}(t)$ to derive the forth axiom of Peano already appears in the work of Miquel [35]. The terminology peel that we use to denote the eliminator of equality is similar to the one used in some presentations of type theory [42], however our own motivation to use it is to emphasize that Leibniz principle is recovered by doing an external induction on the formulas or, more graphically, by peeling out the syntax.

6.5 Conclusion

We designed a translation from λ**E-HA**$^\omega$ into $\lambda\mathbf{HA}^\omega$ using techniques reminiscent of parametricity and we proved a result of relative consistency: if $\lambda\mathbf{HA}^\omega$ is consistent, then so is λ**E-HA**$^\omega$. The following diagram shows an intuition of the translation:

[3]This paragraph explains similar ideas as the one I wrote in Subsection 2.2.4.

λ**E-HA**$^\omega$		λ**HA**$^\omega$
t	$\rightsquigarrow$	
$M : t = u$	$\rightsquigarrow$	

A first-order term t is interpreted as a proof of (parametric) equality between two copies of itself and an equality proof $M : t = u$ is translated into a proof of (parametric) equality between a copy of t and a copy of u. While the choice to translate M as an equality between t^1 and u^2 is ad hoc (in the sense that it is not imposed by the translation), it is notable that equality proofs are translated into (parametric) equality proofs without need of higher order structures (that do not exist in this framework).

Bibliography

[1] T. Altenkirch, S. Boulier, A. Kaposi, and N. Tabareau. Setoid type theory—a syntactic translation. In Mathematics of Program Construction, pages 155–196. Springer International Publishing, 2019.

[2] K. R. Apt and W. Marek. Second order arithmetic and related topics. Annals of Mathematical Logic, 6(3):177–229, 1974.

[3] H. P. Barendregt. The lambda calculus - its syntax and semantics, volume 103 of Studies in logic and the foundations of mathematics. North-Holland, 1985.

[4] M. J. Beeson. Foundations of Constructive Mathematics. A Series of Modern Surveys in Mathematics. Springer, Berlin, 1985.

[5] E. Beffara, F. Castro, M. Guillermo, and É. Miquey. Concurrent Realizability on Conjunctive Structures. In Marco Gaboardi and Femke van Raamsdonk, editors, 8th International Conference on Formal Structures for Computation and Deduction (FSCD 2023), volume 260 of Leibniz International Proceedings in Informatics (LIPIcs), pages 28:1–28:21. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2023.

[6] J. L. Bell. The Axiom of Choice. In Edward N. Zalta, editor, The Stanford Encyclopedia of Philosophy. Metaphysics Research Lab, Stanford University, Winter 2021 edition, 2021.

[7] S. Berardi, M. Bezem, and T. Coquand. On the computational content of the axiom of choice. Journal of Symbolic Logic, 63(2):600–622, 1998.

[8] J.-P. Bernardy, P. Jansson, and R. Paterson. Parametricity and dependent types. ACM SIGPLAN Notices, 45:345–356, 09 2010.

[9] S. Boulier, P.-M. Pédrot, and N. Tabareau. The next 700 syntactical models of type theory. In Proceedings of the 6th ACM SIGPLAN Conference on Certified Programs and Proofs, CPP 2017, page 182–194, New York, NY, USA, 2017. Association for Computing Machinery.

[10] R. Boyd, G. Hensel, and H. Putnam. A recursion-theoretic characterization of the ramified analytical hierarchy. Transactions of the American Mathematical Society, 141, 07 1969.

[11] F. Castro. An interpretation of **E-HA**$^\omega$ inside **HA**$^\omega$. Electronic Proceedings in Theoretical Computer Science, 396:52–66, November 2023.

[12] F. Castro. Revisiting the model of Hereditary Recursive Operations, a presentation. `https://www.irif.fr/_media/users/castro/ihaw.pdf`, 2024.

[13] F. Castro, A. Miquel, and K. Worytkiewicz. Implicative assemblies. `https://arxiv.org/abs/2304.10429`, 2023.

[14] P. J. Cohen. A minimal model for set theory. Bulletin of the American Mathematical Society, 69:537–540, 1963.

[15] L. Colson and S. Grigorieff. Syntactical truth predicates for second order arithmetic. Journal of Symbolic Logic, 66, 03 2001.

[16] L. Fontanella and G. Geoffroy. Preserving cardinals and weak forms of Zorn's lemma in realizability models. Mathematical Structures in Computer Science, 30(9):976–996, October 2020.

[17] H. Friedman. The consistency of classical set theory relative to a set theory with intuitionistic logic. The Journal of Symbolic Logic, 38(2):315–319, 1973.

[18] R. O. Gandy. On the axiom of extensionality–part i. The Journal of Symbolic Logic, 21(1):36–48, 1956.

[19] J.-Y. Girard, Y. Lafont, and P. Taylor. Proofs and Types, volume 7 of Cambridge Tracts in Theoretical Computer Science. Cambridge University Press, 1989.

[20] T. G. Griffin. A formulae-as-type notion of control. POPL '90, page 47–58, New York, NY, USA, 1989. Association for Computing Machinery.

[21] K. Gödel. The consistency of the axiom of choice and the generalized continuum hypothesis. Proceedings of the National Academy of Sciences of the United States of America, 24:556–7, 01 1938.

[22] H. Herbelin. A Constructive Proof of Dependent Choice, Compatible with Classical Logic. In LICS 2012 - 27th Annual ACM/IEEE Symposium on Logic in Computer Science, Proceedings of the 27th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS 2012, 25-28 June 2012, Dubrovnik, Croatia, pages 365–374, Dubrovnik, Croatia, June 2012. IEEE Computer Society.

[23] T. Jech. Set Theory: The Third Millennium Edition. Springer, 2003.

[24] P. T. Johnstone. Sketches of an elephant: a Topos theory compendium. Oxford logic guides. Oxford Univ. Press, New York, NY, 2002.

[25] S. C. Kleene. On the interpretation of intuitionistic number theory. Journal of Symbolic Logic, 10:109 – 124, 1945.

[26] S. C. Kleene. Quantification of number-theoretic functions. Compositio Mathematica, 14:23–40, 1959-1960.

[27] J.-L. Krivine. Lambda-calculus, Types and Models. Ellis Horwood, Upper Saddle River, NJ, USA, 1993.

[28] J.-L. Krivine. A general storage theorem for integers in call-by-name λ-calculus. Theoretical Computer Science, 129(1):79–94, 1994.

[29] J.-L. Krivine. Théorie des ensembles. 1998.

[30] J.-L. Krivine. Realizability in classical logic. 2009.

[31] J.-L. Krivine. Realizability algebras II: new models of ZF + DC. Logical Methods in Computer Science, Volume 8, Issue 1, February 2012.

[32] J.-L. Krivine. A program for the full axiom of choice. Logical Methods in Computer Science, Volume 17, Issue 3, September 2021.

[33] P. Martin-Löf. Intuitionistic Type Theory. Bibliopolis, 1984.

[34] C. McCarty. Realizability and recursive set theory. Annals of Pure and Applied Logic, 32:153–183, 1986.

[35] A. Miquel. Relating classical realizability and negative translation for existential witness extraction. In Typed Lambda Calculi and Applications, pages 188–202. Springer Berlin Heidelberg, 2009.

[36] A. Miquel. Forcing as a program transformation. pages 197–206, 06 2011.

[37] A. Miquel. Implicative algebras: a new foundation for realizability and forcing. Mathematical Structures in Computer Science, 30(5):458–510, May 2020.

[38] É. Miquey. A sequent calculus with dependent types for classical arithmetic. In LICS 2018 - 33th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS '18 Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science, pages 720–729, Oxford, United Kingdom, July 2018. ACM.

[39] É. Miquey. A constructive proof of dependent choice in classical arithmetic via memoization, 2019.

[40] Y. N. Moschovakis. Determinacy and prewellorderings of the continuum. In Y. Bar-Hillel, editor, Mathematical Logic and Foundations of Set Theory, volume 59 of Studies in Logic and the Foundations of Mathematics, pages 24–62. Elsevier, 1970.

[41] Y. N. Moschovakis. Descriptive Set Theory. North-Holland Mathematics Studies. North-Holland Publishing Company, 1980.

[42] B. Nordström, K. Petersson, and J. M. Smith. Programming in Martin-Löf's Type Theory: An Introduction. Clarendon Press, USA, 1990.

[43] P. Odifreddi. Classical Recursion Theory: The Theory of Functions and Sets of Natural Numbers. Number vol. 1 in Classical Recursion Theory. North-Holland, 1989.

[44] P.-M. Pédrot. Upon this quote I will build my Church Thesis. In Proceedings of the 39th Annual ACM/IEEE Symposium on Logic in Computer Science, LICS '24, New York, NY, USA, 2024. Association for Computing Machinery.

[45] J. C. Reynolds. Types, abstraction and parametric polymorphism. In IFIP Congress, 1983.

[46] X. Shi. Projective prewellorderings vs projective wellfounded relations. The Journal of Symbolic Logic, 74(2):579–596, 2009.

[47] S. G. Simpson. Subsystems of second order arithmetic. In Perspectives in mathematical logic, 1999.

[48] T. Streicher. Introduction to constructive logic and mathematics, lecture notes. `https://www2.mathematik.tu-darmstadt.de/~streicher/`.

[49] A. S. Troelstra. Metamathematical Investigation of Intuitionistic Arithmetic and Analysis. New York: Springer, 1973.

[50] J. van Oosten. Realizability: An Introduction to its Categorical Side. ISSN. Elsevier Science, 2008.

[51] Z. Vetulani. Ramified analysis and the minimal β-models of higher order arithmetics. Fundamenta Mathematicae, 121, 01 1984.

[52] P. Wadler. Theorems for free! In Proceedings of the Fourth International Conference on Functional Programming Languages and Computer Architecture, FPCA '89, page 347–359, New York, NY, USA, 1989. Association for Computing Machinery.

[53] E. Zermelo. Beweis, daß jede menge wohlgeordnet werden kann. Mathematische Annalen, 59:514–516, 1904.

[54] J. I. Zucker. Proof theoretic studies of systems of iterated inductive definitions and subsystems of analysis. PhD thesis, Standford University, 1971.