Collège Doctoral du Languedoc-Roussillon

THÈSE

Pour obtenir le grade de Docteur

Délivré par l'**Université Montpellier II**

Préparée au sein de l'école doctorale **I2S***
Et de l'unité de recherche **UMR 5506**

Spécialité : **Informatique**

Présentée par **M. Fabien Givors**
fabien.givors@chezlefab.net

Vers une structure fine des calculabilités

Soutenue le 06/12/2013 devant le jury composé de :

M. Stéphane Bessy	MdC	Univ. Montpellier II	Examinateur
M. Arnaud Durand	Professeur	Univ. Paris VII	Examinateur
M. Bruno Durand	Professeur	Univ. Montpellier II	Directeur de thèse
M. Olivier Finkel	CR CNRS	Univ. Paris VII	Rapporteur
M. Serge Grigorieff	Prof. Émérite	Univ. Paris VII	Rapporteur
M. Lars Kristiansen	Professor	Univ. of Oslo	Examinateur
M. Grégory Lafitte	MdC	Univ. Montpellier II	Co-Directeur de thèse

Université Montpellier 2
Sciences et Techniques

[Les sous-calculabilités définissent une hiérarchie d'approximations de la calculabilité, comportant toujours plus de fonctions récursives et se rapprochant toujours plus du cas classique.]

* **I2S** : École doctorale Information Structures Systèmes

Sommaire

Remerciements

«*Nani gigantum humeris insidentes*[1]», c'est vraiment l'impression que cela donne de *terminer* sa thèse. Une longue escalade vers les hauteurs de sa spécialité, de géants en géants, avec ses proches pour tenir l'échelle.

Depuis les petites classes, mes professeurs n'ont jamais manqué de me transporter par les raisonnements mathématiques, dans toutes les classes scientifiques, et ont aiguisé ma curiosité au point qu'au lycée, je ne me voyais pas faire autre chose que de continuer à étudier, notamment les mathématiques et la physique. Merci à eux !

C'est en classe préparatoire, à La Martinière Monplaisir, section MPSI, que j'ai rencontré la personne qui m'a pour la première fois fait adorer les mathématiques. Merci à vous M. Feyler sans qui j'aurais probablement quitté cet univers là. C'est également dans cet environnement que j'ai eu la chance de découvrir l'informatique théorique, par la craie de M. Haberer et de ses cours qui m'ont immédiatement poussé à creuser le sujet. Merci enfin à M. Berne, mon professeur de mathématiques de deuxième année qui le premier m'a fait comprendre que l'ÉNS Lyon n'était pas une école inaccessible, c'est grâce à lui que j'ai tenté ma chance.

En entrant à l'ÉNS Lyon en 2006, Yves Robert alors directeur du LIP avait accueilli notre promotion de jeunes étudiants en informatique et il nous avait dit[2]. «Quand vous comparez ce que vous savez à l'heure actuelle, à ce qu'un chercheur doit savoir, cela doit vous donner une vague idée de l'infini.» Ironiquement, j'ai justement passé les années qui ont suivi à chercher une compréhension de moins en moins vague de celui-ci.

J'ai là-bas rencontré de nombreuses personnes grâce auxquelles j'ai pu m'émanciper et découvrir les branches de l'informatique fondamentale qui me parlaient le plus. Merci à Jaques Mazoyer pour ses cours de calculabilité si prenant, pour la vision du domaine qu'il communiquait et qui a formé mon regard sur le calcul. Merci également à Daniel Hirschkoff, pour nous avoir présenté le λ-calcul dans la joie et la bonne humeur.

Pendant cette année à Lyon, avec deux autres fous, nous avons contacté Enrico Formenti, qui voulait créer sur Nice un Master recherche antenne de l'ÉNS Lyon. Enrico nous a construit un cursus de rêve, piochant suivant nos demandes parmi les

1. Des nains sur des épaules de géants.
2. Ou il nous avait dit quelque chose d'approchant, plus ou moins... C'est en tout cas ce que j'en ai retenu.

experts locaux et moins locaux, dans des conditions d'études indécemment privilégiées. Merci également à Bruno Martin et à Sandrine Julia pour nous avoir accueillis comme leurs protégés dans cet environnement. Bien sûr, cette entreprise n'aurait pas été possible si Tarik Kaced et Marc Lasson n'avaient pas eux aussi décidé d'entreprendre ce parcours. Merci à eux pour avoir été de si chouettes collègues et colocs.

Pendant ce cursus, une personne, par son cours et sa passion scientifique, m'a particulièrement marqué. À tel point que, bien au courant de la difficulté et quasi-confidentialité du domaine, je l'ai suivi en stage puis en thèse, sans hésiter. Je me suis lancé sur ces bons conseils dans l'étude de papiers qui me dépassaient de quelques ordres de grandeur d'ordres de grandeur, et j'en suis sorti grâce aux nombreuses séance de travail qu'il m'a accordées, tantôt dans les Calanques, tantôt au CIRM, tantôt en ville, tantôt par téléphone à des heures improbables, parfois aussi au labo, mais également dans de nombreuses conférences passionnantes et épanouissantes, plus grand. Merci pour tout Greg !

Ces années de thèses ont été un peu mouvementées, entre les tensions politiques au LIF et le déménagement au LIRMM. Je n'étais pas tout le temps très à l'aise au milieu de ces changements, et j'en ai profité pour faire quelques âneries de mon côté pendant des périodes d'égarement. Heureusement pour moi, le grand chef de la famille ESCAPE est intervenu en ma faveur à de multiples reprises, assurant ainsi mes arrières et corrigeant mes boulettes. Merci Bruno !

Enfin, merci à Serge Grigorieff, pour avoir accepté d'être rapporteur de mon travail, après avoir été le directeur de thèse du directeur de thèse de mon directeur de thèse, ainsi que pour les quelques discussions épistémologiques et philosophiques que j'ai pu avoir avec lui. Merci à Olivier Finkel pour avoir été en tant que rapporteur un relecteur hors pair à faire rougir une maison d'édition et un programme de vérification de preuve tout à la fois. Merci à Lars Kristiansen, pour s'être déplacé de si loin pour venir participer comme examinateur à mon jury de thèse, et pour m'avoir ouvert des voies pour mes travaux futurs. Merci à Stéphane Bessy pour avoir eu le courage et la patience de se lancer dans la lecture de mon manuscrit pourtant très loin de son domaine de recherche, en tant qu'examinateur, et pour m'avoir aidé à de multiples reprises comme collègue de labo.

Sans toutes ces personnes, je n'aurais pas pu en arriver là. Il reste encore de nombreuses personnes à remercier, notamment les autres membres ou ex-membres de l'équipe ESCAPE, Nicolas Ollinger, Alexander Shen, Emmanuel Jeandel, Laurent Bienvenu, et ses ex-thésards et futurs ex-thésards, Gaëtan, Alexis, Martin, Alex, Pascal, Sabrina et Guilhem pour toutes les discussions que j'ai pu avoir avec eux, et les bons moments passés ensemble… Merci aussi à Marlène, Richard, Aurore, Caroline, Jérémie, Magali, Thibaud, Laure, Olivier, Chantal, Joël, Lætitia, Matthias, Alice, Bastien, Agathe et Ligaya, amis et camarades, pour le rôle qu'ils ont joué, parfois sans le savoir, en m'accompagnant durant ma thèse.

Mais je voudrais surtout remercier mes parents, Alain et Dany, mon frère Martin et le reste de ma famille qui m'ont soutenu depuis le début, toujours encouragé à faire ce que j'aimais, et à qui je dois, presque, tout. Merci à vous !

Préliminaires

Introduction

Calculabilités, incomplétude et ensembles constructibles

Calculabilité est habituellement un mot employé au singulier, il désigne un domaine de l'informatique théorique antérieur à la naissance des premiers ordinateurs, dont l'objectif est de répondre à ces questions qui nous interpellent encore aujourd'hui : que pouvons-nous calculer ? Qu'est-ce que calculer ?

Après de nombreuses tentatives pour formaliser un cadre général permettant de regrouper derrière une même théorie les fonctions que l'intuition disait être calculables, la communauté étudiant les questions de calculabilité a réalisé que la notion la plus élégante, correspondant réellement à ce que nous entendons être *le calcul*, ne s'identifiait pas à une classe de fonctions totales. Bien au contraire, nous savons depuis 1931, alors que Kurt Gödel publie ses résultats d'incomplétude de l'arithmétique, qu'une théorie du calcul englobant l'arithmétique ne peut pas être complète. En effet, une telle théorie ne peut pas prouver pour tout prédicat P si P ou $\neg P$ sont des théorèmes et donc en particulier, elle ne peut pas différencier ce qu'elle sait calculer de ce qu'elle ne sait pas. Les fonctions partielles ont alors été introduites, permettant des constructions qui aujourd'hui encore restent surprenantes comme le second théorème de la récursion de Stephen Cole Kleene [Kle36] : « si φ est une énumération acceptable des fonctions récursives et f est une fonction récursive totale, alors il existe un entier n tel que $\varphi_n \cong \varphi_{f(n)}$ ». Avec les fonctions partielles, la calculabilité était née et, dès sa naissance, elle était intrinsèquement liée à la prouvabilité des formules arithmétiques, et donc à la théorie de la preuve. Plus tard, vers 1950, afin de répondre aux interrogations de Emil Post sur la structure des degrés Turing et en particulier l'existence de degrés Turing *r.e.* non-récursifs mais non-complets, Richard Friedberg [Fri57] et Albert Abramovich Muchnik ont inventé la *méthode de priorité* dont les variantes sont utilisées aujourd'hui dans des constructions des plus ingénieuses pour prouver les résultats les plus pointus de la calculabilité moderne.

Définie sur les entiers, tout comme sa grande sœur l'arithmétique, la calculabilité utilise ces derniers pour coder des structures toujours plus complexes grâce entre autres aux *numérotations de Gödel*. Puis, avec l'essor des théories des ensembles telles **ZF** ou encore **KP**, la calculabilité a été généralisée aux ensembles via deux points de vue différents que nous rencontrerons par la suite : la théorie de la E-récursion de Dag Normann [Nor78] et Gerald Enoch Sacks [Sac90], et la théorie de la récursion sur les ordinaux admissibles, principalement étudiée par Jon Barwise [Bar75]. Ces deux points de vue divergent sur la notion d'effectivité du calcul. Pour nous rendre compte de cela, rappelons nous qu'en calculabilité, calculer une fonction récursive en un point revient à chercher le plus petit objet ayant une certaine propriété. Plus formellement, le théorème de la forme normale de Kleene nous apprend qu'il existe un prédicat *simple* [3] T et une fonction récursive primitive U tel que pour toute fonction calculable f, il existe un e tel que $\forall n, f(n) \cong U\left(\mu x, T\left(e, x, n\right)\right)$ [4]. Ce point de vue nous permet d'affirmer qu'une fonction récursive peut toujours être vue comme une fonction testant un certain prédicat sur tous les entiers, jusqu'à en trouver un le satisfaisant. En pratique, rechercher le plus petit entier n vérifiant une *propriété simple* P, revient à tester $P(0)$, puis, si la propriété est fausse, tester $P(1)$, *etc.* Dans ce contexte, il est facile, étant donné n, de construire $n+1$ et d'étudier ses propriétés. Dans le contexte des théories *classiques* [5] de la récursion sur les ensembles, la notion de *constructibilité* faisant référence est donnée par la hiérarchie des constructibles établie par Gödel. Elle exhibe un modèle intérieur de la théorie des ensembles **ZFC** [Jec03]. Le cadre de la récursion sur les admissibles suppose que l'on sait effectuer une recherche non bornée à travers tous les objets inclus dans ses modèles qui sont des niveaux particuliers de la hiérarchie des constructibles. Inversement, la E-récursion n'autorise la recherche de nouveaux objets que par des procédés utilisant des moyens et paramètres que nous avons déjà su produire. Par exemple, afin de pouvoir effectuer une recherche non bornée sur les entiers, il faut disposer de l'ensemble ω bornant cette recherche. Nous verrons comment ces deux points de vue transparaissent dans la calculabilité à condition d'en affaiblir les axiomes.

Ainsi, ces calculabilités d'ordre supérieur permettent de calculer d'avantage d'objets, d'identifier de nouveaux ensembles, que la calculabilité classique n'avait pas les moyens d'exprimer. Pour autant, bien que *dépassant* les problèmes et difficultés rencontrés dans le cas classique, elles ne permettent pas de les clore. Au contraire, les calculabilités d'ordre supérieur posent à nouveau l'ensemble des questions auxquelles il n'avait pas été fourni de réponse dans le cadre des fonctions récursives *à la* Church, Rosser et Turing, mais cette fois, dans un contexte déjà étudié sous d'autres points de vue, tel que ceux de la théorie de **KP**, des constructibles de Gödel ou comme nous le verrons, de l'analyse ordinale de certaines théories. Par exemple, les ordinaux admissibles sont aux théories de la récursion sur les ensembles ce que l'ordinal

3. Nous définirons par la suite une hiérarchie descriptive des formules : la hiérarchie arithmétique. Ici, nous entendons par *propriété simple* une formule Δ_0^0, c'est-à-dire sans quantificateur universel ou existentiel non borné.

4. Étant donné un prédicat Δ_0^0 P, le schéma μ, dit schéma de recherche non bornée, dénote par « $\mu x, P(x)$ » le plus petit entier x satisfaisant $P(x)$.

5. « *classique* » est ici à opposer à « *intuitionniste* ». D'autres notions de constructivisme ont été développées notamment en *Mathématiques Constructivives* par Andreï Andreïevitch Markov, voir [Kus06].

de Church-Kleene est à la calculabilité : ils représentent l'ensemble des ordinaux *récursifs* suivant le nouveau point de vue. De plus, les mêmes mécanismes sont en jeu, on retrouve par exemple un équivalent du théorème s_n^m, du second théorème de Kleene, du problème de l'arrêt, *etc.*

La généralisation de la calculabilité à un ordre supérieur nous permet cependant de prendre d'avantage de recul face à la notion de calcul. Elle nous montre toute une classe de structures satisfaisant aux axiomes de la calculabilité. Comme nous allons le voir, cette classe est structurée, ses éléments ne sont pas sans relation les uns avec les autres, de plus, tous ses éléments méritent de porter le nom de calculabilité. Bref, l'existence de tous ces modèles de calculabilité justifie le fait de parler des *calculabilités* au pluriel.

Ordinaux, sous-classes et énumérabilité

Initialement, nos intérêts particuliers dans les calculabilités concernaient les liens entre les calculabilités d'ordre supérieur et la calculabilité classique. Comme l'apparente similarité des structures et des problématiques était manifeste, et comme il semblait bien souvent plus simple de prouver ces propriétés dans les strates plus élevées du calcul où les outils de théorie des ensembles et de théorie de la preuve se révélaient très efficaces, il était naturel de s'interroger sur la projection de ces superstructures sur l'arithmétique. Cependant nous ne connaissions et ne comprenions pas assez le modèle d'arrivée : le modèle de la calculabilité. Nous savons quelles sont les fonctions définissables par récursion primitive, et nous doutons de celles définissables par un schéma de récursion sur un ordinal un peu plus élevé comme ω^2. Cependant, pour un ordinal récursif α, plus il est choisi proche de ω_1^{CK}, et moins l'ensemble des fonctions que l'on peut définir par un schéma d'α-récursion est évident. En fait, à partir d'un moment, il devient même très difficile d'exprimer ces grands ordinaux *proches* de ω_1^{CK}.

Nous avons donc décidé d'étudier plus en détail ces ordinaux. Avec chacune des sous-classes d'ordinaux récursifs, des sous-classes de fonctions calculables, induites par les schémas de récursion associés et composées uniquement de fonctions totales, font leur apparition. Nous savons qu'avec uniquement des fonctions totales, il est impossible de capturer toute la calculabilité, de prouver le second théorème de la récursion de Kleene ou de montrer l'existence d'une fonction universelle calculable. Pourtant, une grande partie des constructions utilisées en calculabilité sont récursives primitives, c'est-à-dire qu'elles sont réalisables via un opérateur de récursion bornée à la place du schéma μ de recherche non bornée ou de l'opérateur de point fixe général issu du théorème de Kleene. Dans sa thèse de doctorat [Laf02], Grégory Lafitte construit des *Systèmes acceptables de calcul généralisé* qui sont des calculabilités pour lesquelles il fait varier la puissance en fonction des ensembles récursivement énumérables qu'il lui prête. En poursuivant ce chemin, nous avons tenté d'identifier les objets effectivement descriptibles par les sous-classes de fonctions totales à notre disposition.

La première classe de fonctions récursives totales définie, assez complexe pour donner l'intuition derrière la notion de fonctions récursives, mais assez simple pour ne contenir que des fonctions totales est la classe des fonctions récursives primitives que nous avons mentionnée et que nous noterons par la suite p. Du point de vue des ordinaux récursifs, cette classe est obtenue dès lors que l'on autorise la récursion sur ω, c'est-à-dire dès lors que l'on autorise une fonction à faire des appels récursifs dont les arguments des appels décroissent suivant un ordre élémentaire isomorphe à ω. Close par composition, cette classe inclut en fait les schémas de récursion pour tous les ordinaux strictement inférieurs à ω^ω. Pour revenir à nos ensembles récursivement énumérables (*r.e.*), on pourrait vouloir s'intéresser à la collection des ensembles *r.e.* qui sont l'image d'une fonction récursive primitive. Or, il s'avère que cette classe est exactement [6] celle de tous les ensembles *r.e.* En effet, quel que soit un ensemble *r.e.* $\mathbf{W}_e$, il existe une fonction récursive primitive f qui l'énumère. Même l'image de la fonction d'Ackermann [Ack28] peut être énumérée par une fonction de p. La raison à cela est qu'il n'est nullement demandé aux énumérations d'être injectives, aussi, si φ_e est une fonction récursive totale [7] énumérant $\mathbf{W}_e$ et a un élément de $\mathbf{W}_e$, nous pouvons construire une fonction $f(\langle n, s\rangle)$ dans p qui simule $\varphi_e(n)$ pendant s étapes et renvoie $\varphi_e(n)$ si le calcul converge, ou a sinon. L'image de f est bien $\mathbf{W}_e$, mais f est *a priori* loin d'être injective. Inversement, en calculabilité classique, tout ensemble infini énumérable est énumérable injectivement.

Nous avons donc modifié la notion d'énumérabilité d'un ensemble en lui rajoutant des contraintes d'injectivité. Tous les ensembles *r.e.* ne sont plus « p-*énumérables* » pour cette nouvelle notion. En particulier, l'image de la fonction d'Ackermann ne l'est plus. Il est possible de discuter sur le choix de la restriction aux énumérations injectives. Ce choix n'est pas parfait, mais il assure déjà une bonne propriété sur les ensembles énumérables : il permet de produire un *nouvel* élément à la demande, ce qui est un critère vérifié dans le cas classique grâce au schéma de recherche non bornée. Pour rendre la classe des ensembles énumérables encore plus représentative de la puissance des fonctions qui les énumèrent, il faudrait enrichir la notion d'énumérabilité en considérant non plus qu'un ensemble d'entiers est énumérable, mais qu'une séquence d'entiers, de longueur ω, l'est. Nous n'adopterons pas ce point de vue dans cette thèse, ce qui nous permettra d'obtenir une notion d'énumérabilité intermédiaire entre celle induite par les fonctions (séquences ordonnées), et celle induite par la définition standard (ensembles avec répétitions).

Au final, pour chaque ordinal dénombrable α, nous obtenons en plus d'une classe de fonctions α-*récursives* [Rat99], une classe d'ensembles α-*énumérables.*

6. À codage près, l'ensemble vide peut lui aussi être énuméré.

7. Toute énumération peut-être rendue totale, mais une énumération totale ne peut être croissante que si l'ensemble est à la fois *r.e.* et de complémentaire *r.e.*.

Sous-calculabilités

De la même manière qu'en calculabilité classique, les fonctions partielles récursives sont l'ensemble des fonctions dont le graphe est *r.e.*, nous pouvons définir les fonctions partielles α-récursives comme étant celles dont le graphe est α-énumérable. Les classes de fonctions partielles et d'ensembles énumérables ainsi définies ne comptent pas tous les objets présents dans le cas classique. Du point de vue de la calculabilité, il en manque, il y a des *trous* qui révèlent les objets intrinsèquement difficiles à décrire.

Les structures de degrés et réductions basées sur des classes de fonctions totales nous permettent de disposer d'un second point de vue sur les ensembles d'entiers, et notamment sur les ensembles *r.e.* Alors que certains objets récursifs, tel l'ensemble des entiers positifs pairs, restent triviaux, nous voyons apparaître toute une hiérarchie de degrés à l'intérieur des degrés Turing. Au final, c'est toute une super-structure entre les différentes sous-calculabilités qui est mise à jour, dévoilant une famille d'ensembles hyperarithmétiques plus fine que celle connue dans le cas classique. Cette structure nous amène à considérer les sous-calculabilités comme un modèle plus général qui n'est pas restreint aux sous-classes des fonctions récursives classiques. De fait, il ne semble même pas nécessaire de modifier le modèle des sous-calculabilités pour y exprimer les calculabilités sur les admissibles. L'harmonisation avec la théorie de la E-récursion est, de son côté, plus délicate. Nous verrons quels en sont les éléments à adapter pour nous rapprocher du formalisme des sous-calculabilités et noterons les outils développés par celle-ci utilisables en l'état dans notre contexte.

Organisation de la thèse

Notre travail se découpe en quatre chapitres, une annexe et une conclusion sur les ouvertures et les poursuites de l'étude.

Classes de fonctions totales Dans ce premier chapitre, nous avons une approche axiomatique des classes de fonctions totales. Nous reprenons les travaux de Dexter Kozen [Koz80], les explicitant par quelques exemples bien connus, et transposons des résultats classiques sur ces structures.

Classes fondamentales En demandant aux classes de fonctions, en plus d'être énumérables, d'inclure les fonctions récursives primitives, nous obtenons des familles de fonctions sachant manipuler leurs indices plus aisément et fournissant déjà une notion d'énumérabilité et quatre notions de récursivités sur les ensembles, correspondant à des propriétés bien distinctes mais confondues dans le cas classique.

Sous-calculabilités En nous basant sur les classes fondamentales, nous définissons les fonctions $\mathtt{C}$-partielles associées. Nous étudions les objets présents et absents de cette classe. Enfin, nous explorons plus en profondeur la structure des degrés et les réductions entre ensembles associées à nos notions de récursivité. Nous montrons en particulier des résultats concernant différentes relativisations du problème de Post aux sous-calculabilités.

Premiers pas vers une structure fine généralisée Sous la forme de prospectives, nous étudions ensuite la portée des sous-calculabilités, en regardant dans un premier temps à quel point la calculabilité peut être capturée par ces classes. Dans un deuxième temps, nous cherchons à généraliser le domaine au delà de ω et les fonctions au delà des fonctions récursives usuelles grâce à la théorie de la récursion sur les admissible. Nous remarquons de nombreuses similarités entre les sous-calculabilités et ces théories mettant sur la voie d'une formalisation de celles-ci dans le cadre des sous-calculabilités.

Perspectives Enfin, nous présentons différentes pistes permettant d'atteindre les objectifs à plus long terme que nous nous étions fixés. L'une de ces pistes est l'absoluité : une notion permettant de transformer des exemples en théorèmes. Plus précisément, elle permet d'identifier les formules qui, si elles sont vérifiées dans un quelconque modèle transitif d'une théorie, sont vérifiées dans tous ses modèles. Enfin, nous présentons l'horizon que nous espérons atteindre à terme, en précisant les motivations et questionnements à l'origine de ce travail, et les raisons pour lesquelles nous pensons que celui-ci peut apporter une contribution intéressante au monde de la calculabilité.

Annexes Nous présentons en fin de document des rappels sur l'élaboration des séquences fondamentales utilisées dans le premier chapitre pour expliciter toute une hiérarchie de fonctions croissantes. Nous faisons également quelques rappels de récursion sur les ordinaux admissibles et de E-récursion explicitant les notions utilisées dans le chapitre des prospectives.

1 Classes de fonctions totales

Formalisation des classes de fonctions récursives totales

Sommaire

Il n'y a rien de plus commun qu'une fonction récursive. Chaque raisonnement, chaque preuve, chaque construction, chaque algorithme est une fonction récursive. Bien sûr, elles ne sont que rarement étudiées sous ce nom, mais au fond, dès lors qu'il s'agit d'un algorithme finiment descriptible travaillant sur un objet finiment descriptible et renvoyant une description finie, nous avons affaire à une fonction récursive.

Une conséquence du fait qu'elle soit finiment descriptible est qu'une fonction récursive peut manipuler cette description et surtout, elle peut l'interpréter et la transformer. En effet, une fonction récursive, grâce à la description d'une autre fonction récursive, peut la calculer, le summum étant atteint avec le second théorème de la récursion de Kleene quand une fonction se calcule à l'aide de sa propre description. Nous comprenons donc à quel point, dans l'étude des fonctions récursives, le calcul et la description du calcul sont liés et même intrinsèquement inséparables.

Ainsi une grande partie des problèmes de dénombrement ou de combinatoire dénombrable peut s'exprimer sous la forme de fonctions calculables. En particulier, toutes les formules existentielles du premier ordre sont équivalentes à l'arrêt d'une fonction récursive donnée en un point donné, comme nous allons le revoir lorsque nous rappellerons le théorème de la forme normale de Kleene.

À partir de là il est naturel de penser que, comme certaines formules paraissent plus compliquées que d'autres, comme par exemple la négation de la conjecture de Goldbach : « $\exists n \geqslant 2, \forall p \leqslant n, \neg(\text{Premier}(p) \wedge \text{Premier}(2n - p))$ », il doit en être de même des fonctions récursives. De plus, certaines fonctions doivent avoir des points communs ou des similitudes particulières entre elles en tant que fonctions effectuant des opérations de mêmes natures sur des descriptions de même type.

Pour tenter d'isoler ces fonctions de difficulté similaire, nous pouvons définir des ensembles de fonctions par un jeu d'axiomes qu'ils doivent ou que leurs membres doivent satisfaire. Ces axiomes comptent habituellement l'existence de fonctions de base, comme des fonctions constantes, mais nous rencontrons également des axiomes sur la structure de l'ensemble lui-même. Demander à un tel ensemble d'être clos par composition est en général un des premiers critères fixés afin d'avoir une classe capturant une notion intuitive.

Une des opérations de clôture que nous verrons être très porteuse de sens est la clôture via les opérateurs de récursion, qu'il s'agisse de l'opérateur de récursion non bornée (μ), de l'opérateur de récursion primitive, ou d'autres opérateurs que nous introduirons par la suite.

Cette approche axiomatique, présentant un nombre souvent fini voire dénombrable d'objets initiaux, et un nombre fini ou dénombrable d'opérateurs de construction, permet la plupart du temps d'établir des énumérations de ces classes dénombrables. Pour une classe donnée, cette énumération va nous permettre d'en identifier des propriétés élémentaires néanmoins fortement utilisées en calculabilité classique, qui ne dépendent pas des éléments de celle-ci. Ces propriétés sont en fait très générales et les sortir du contexte du calcul Turing sur les entiers permet d'identifier une structure très élémentaire valide dans tout système tentant de reproduire ou d'approximer la calculabilité.

Une particularité des énumérations de fonctions récursives est que leur complexité est directement liée à celles des éléments qu'elles énumèrent. Ainsi, dans le but d'identifier les propriétés simples d'une énumération sans que la complexité des fonctions manipulées ne vienne les cacher, nous nous intéresserons pour commencer à des classes de fonctions possédant elles-même une complexité très faible.

1.1 Notions de classes closes

La notion de classe close que nous allons utiliser dans cette première partie a été étudiée par Kozen [Koz72] et constitue une base très riche pour la suite, car elle permet de remarquer que tout un lot de propriétés génériques utiles en calculabilité ne découle en fait que d'un très faible fragment de celle-ci.

Une approche algébrique des classes de fonctions pour lesquelles il existe une énumération nous permettra d'identifier ou de reconnaître quelques unes de ces propriétés.

1.1.1 Définitions

Une des opérations les plus élémentaires que nous souhaitons autoriser entre les fonctions est la composition. Le critère principal pour rendre cette opération possible est la compatibilité des domaines de définition avec les ensembles d'arrivée. Une classe de fonctions sera donc un ensemble $\mathtt{C}$ de fonctions d'arités strictement positives et dont le domaine et l'ensemble d'arrivée devront être un même ensemble $\mathcal{D}$.

Rien ne nous oblige a priori à n'étudier seulement le cas $\mathcal{D} = \omega$. Comme nous le verrons dans le chapitre 4, il existe des structures de calculabilité sur des ensembles plus complexes comme les *ordinaux admissibles*, certains niveaux de la *hiérarchie des constructibles* de Gödel ou encore des structures E-closes (voir [Sac90].) Cependant, la généralisation à un domaine $\mathcal{D}$ moins contraint se traduit par une complexification des preuves, pouvant obscurcir l'intuition sous-jacente. Aussi, afin de ne pas mélanger les difficultés, nous nous cantonnerons dans les premiers chapitres au cas où $\mathcal{D}$ est l'ensemble des entiers naturels, nous n'explorerons qu'ensuite la généralisation ensembliste.

Lorsque nous souhaitons construire des fonctions, nous utilisons assez systématiquement différents outils que nous allons demander à nos classes de nous fournir. Pour commencer, nous demandons la présence des fonctions constantes pour chaque valeur du domaine $\mathcal{D}$. Ensuite, afin de nous libérer de toutes les contraintes liées à l'arité des fonctions, contraintes qui sont plutôt d'ordre syntaxique, nous demandons la présence de fonctions de projection et de paire. Nous demandons aussi la présence d'un opérateur conditionnel afin de pouvoir faire des choix dépendants d'un argument. Enfin, nous demandons à ce que la classe soit close par composition.

Définition 1.1.1 **Classes de fonctions closes.** *Une classe de fonctions* $\mathtt{C}$ *d'arités positives et de domaine* $\mathcal{D} = \omega$ *est* close *si elle satisfait les conditions suivantes :*

— $\mathtt{C}$ *contient les fonctions constantes :*

$$\forall d \in \mathcal{D}, \exists\, \mathfrak{c}_d \in \mathtt{C}, \forall x \in \mathcal{D}, \mathfrak{c}_d(x) = d,$$

— $\mathtt{C}$ *contient des fonctions de projections et de pairage, que nous fixerons :*

$$\begin{gathered} \exists \langle \cdot, \cdot \rangle : \mathcal{D}^2 \to \mathcal{D}, \pi_1^2, \pi_2^2 : \mathcal{D} \to \mathcal{D} \in \mathtt{C}, \\ \forall i \in \{1, 2\}, \forall d_1, d_2, \in \mathcal{D}, \pi_i^2(\langle d_1, d_2 \rangle) = d_i, \end{gathered}$$

— $\mathtt{C}$ *contient un opérateur conditionnel :*

$$\begin{gathered} \exists\, \mathbf{cond} : \mathcal{D}^4 \to \mathcal{D} \in \mathtt{C}, \forall d_1, d_2, a, b \in \mathcal{D}, \\ \mathbf{cond}(d_1, d_2, a, b) = \begin{cases} a & \textit{si } d_1 = d_2 \\ b & \textit{sinon} \end{cases}, \end{gathered}$$

— C *est stable par composition :*

$$\forall f, g \in \mathsf{C}, f \circ g \in \mathsf{C}.$$

Note 1.1.2 Arité des fonctions. *La présence de la fonction de pairage et des fonctions de projections nous permet de ramener la question de l'arité des fonctions à un simple problème syntaxique. Dans la suite, nous confondrons, pour toute fonction f et tout éléments x et y du domaine $f(x,y)$ avec $f(\langle x,y\rangle)$. En particulier, nous pouvons nous représenter C comme la clôture par composition d'une classe de fonctions unaires de ω dans ω auxquelles on ajoute la fonction de pairage binaire.*

Une classe close C est un ensemble de fonctions pouvant effectuer des opérations extrêmement basiques. Cependant, ces fonctions ne sont *a priori* pas capables de faire référence à elles-mêmes. En effet, nous n'avons pas encore requis l'existence d'une quelconque énumération de la classe.

Un des points clés donnant tout leur intérêt aux énumérations, et permettant entre autres choses les constructions d'indice non-triviales, est le fait que les objets manipulés par les fonctions et les éléments du domaine des fonctions de notre classe, sont en fait de la même nature que les descriptions des fonctions elles-mêmes.

En gardant ces considérations à l'esprit, nous pouvons demander aux fonctions de nos classes d'être capables de manipuler leurs descriptions via des fonctions de codage. En particulier, nous demandons de pouvoir générer des indices pour les fonctions constantes en fonction de la-dite constante, et de pouvoir générer des indices pour la composée de deux fonctions, en fonction des indices de celles-ci.

Inversement, si un élément du domaine est une description de fonction, il doit être interprétable par une fonction capable de la manipuler. La question de l'appartenance de cette fonction *universelle* à la classe sera traitée plus en détail par la suite. Rappelons-nous qu'une des raisons ayant conduit à considérer les fonctions partielles est le fait qu'il n'existe pas de fonction universelle totale calculable pour la classe de toutes les fonctions calculables totales.

Définition 1.1.3 Classes de fonctions énumérables. *Une classe de fonctions close C est énumérable s'il exsite une fonction $\mathsf{U}_{\mathsf{C}} : \omega^2 \to \omega$ telle que :*

— U_{C} *est une fonction universelle*[1] *:*

$$\mathsf{C} = \{x \mapsto \mathsf{U}_{\mathsf{C}}(e,x) : e \in \omega\}.$$

— C *contient une fonction renvoyant les indices des fonctions constantes :*

$$\exists\, \mathbf{const} \in \mathsf{C}, \forall c, d \in \mathcal{D}, \mathsf{U}_{\mathsf{C}}(\mathbf{const}(c), d) = \mathfrak{c}_c(d) = c$$

1. Cette définition propose une fonction universelle binaire pour des fonctions unaires de C. En utilisant les fonctions de pairage et de projections, il est bien entendu possible quelque soit $n > 0$ de créer une fonction $(n+1)$-aire universelle pour toutes les fonctions d'arité n. Pour les mêmes raisons, nous pourrons confondre $\mathsf{U}_{\mathsf{C}}(e,x)$ avec $\mathsf{U}_{\mathsf{C}}(\langle e,x\rangle)$ selon les besoins.

— C *contient une fonction prenant en argument deux indices de fonctions et renvoyant un indice de leur composée :*

$$\exists\,\mathbf{comp} \in \mathsf{C}, \forall f, g, d \in \mathcal{D}, \mathsf{U}_{\mathsf{C}}(\mathbf{comp}(f,g), d) = \mathsf{U}_{\mathsf{C}}(f, \mathsf{U}_{\mathsf{C}}(g,d))$$

— C *contient une fonction prenant en argument deux indices de fonctions et renvoyant un indice de leur paire :*

$$\exists\,\mathbf{pair} \in \mathsf{C}, \forall f, g, d \in \mathcal{D}, \mathsf{U}_{\mathsf{C}}(\mathbf{pair}(f,g), d) = \langle \mathsf{U}_{\mathsf{C}}(f,d), \mathsf{U}_{\mathsf{C}}(g,d)\rangle$$

L'énumération de la classe, qui est une surjection de ω sur C sera désignée par $(\Phi^{\mathsf{C}}_e)_{e\in\omega}$, avec $\forall e, x \in \omega, \Phi^{\mathsf{C}}_e(x) = \mathsf{U}_{\mathsf{C}}(e,x)$.

Si U_{C} est calculable, la classe énumérables C est dite récursive*.*

On remarque qu'étant donné une telle classe, il est toujours possible de supposer que son énumération dispose d'une fonction de bourrage élémentaire.

Note 1.1.4 Énumération et bourrage. *Étant donné une énumération U d'une classe énumérable C, il est toujours possible de se ramener à une énumération U' satisfaisant la propriété de bourrage. En effet, il suffit d'utiliser $\mathsf{U}' : \langle e, n\rangle, y \mapsto \mathsf{U}(e,y)$. La fonction $\mathfrak{p} : \langle e, n\rangle, c \mapsto \langle e, n+c\rangle$ est alors une fonction de bourrage récursive élémentaire pour U'.*

En calculabilité classique, nous savons que le problème de l'arrêt d'une fonction récursive donnée en paramètre sur une entrée donnée en paramètre est indécidable. Cependant, pour un entier positif n, il est possible de savoir si la fonction converge en moins de n étapes et ce de manière calculable. Cette propriété de simulation permet également la définition d'une notion de coût en étapes de calcul pour la simulation d'une fonction. Elle est rendue possible par le fait que nos fonctions sont Σ_1 en l'ensemble des entiers naturels [2]. Nous verrons plus tard que lorsque le domaine augmente, une définition similaire reste possible.

1.1.2 Théorème de la forme normale de Kleene

Revenons un instant sur le théorème de la forme normale de Kleene. Celui-ci nous indique que toute fonction calculable f peut s'écrire avec un prédicat récursif primitif et un schéma μ non borné :

2. C'est à dire que leur graphe est définissable via une formule comportant un seul quantificateur, qui est existentiel et porte sur les entiers naturels.

Théorème 1.1.5

Théorème de la forme normale de Kleene. *Il existe T un prédicat récursif primitif, et U une fonction récursive primitive tels que si ψ est une fonction récursive, alors il existe e tel que :*

$$\forall x, \psi(x) = U\left(\mu y.T\left(e,x,y\right)\right).$$

Si la recherche non bornée d'un élément y n'est pas possible dans nos classes énumérables récursives, la recherche dans un préfixe propre du domaine reste en général possible. En effet, la recherche bornée est une opération élémentaire et donc, en particulier, récursive primitive. C'est pourquoi nous introduisons la définition suivante, qui nous sera utile à de multiples reprises comme un pont entre la calculabilité classique et les classes fondamentales développées plus tard.

Définition 1.1.6 **Fonction de simulation pas-à-pas et fonctions de coût.** *Soit* C *une classe énumérable de fonction universelle* U_{C}*.* sim *est une fonction de* simulation bornée pas-à-pas *si elle est totale et vérifie :*

$$\forall e,x,\exists s, (\forall t < s, \mathrm{sim}(e,x,t) = 0 \land \mathrm{sim}(e,x,s) = \langle e,x,\mathsf{U}_{\mathsf{C}}(e,x)\rangle)$$

On note alors pour tout e coût_e *la fonction de coût pour e, vérifiant que pour tout x,* $\text{coût}_e(x)$ *est le plus petit s tel que* $\forall t < s, \mathrm{sim}(e,x,t) = 0$ *et* $\mathrm{sim}(e,x,s) = \langle e,x,\mathsf{U}_{\mathsf{C}}(e,x)\rangle$*.*

Dans la suite, on demandera à sim *et* coût *de vérifier les propriétés suivantes :*

— *la fonction* coût_e *doit être dans* C *et d'indice calculable dans* C *à partir de e via une fonction* use *telle que* $\forall e,x,\mathsf{U}_{\mathsf{C}}(\mathrm{use}\, e,x) = \text{coût}_e(x)$

— $\forall e,x,\mathsf{U}(\mathrm{use}(\mathrm{use}(e)),x) \leqslant \mathsf{U}(\mathrm{use}(e),x)$

— $\forall e,x,\mathsf{U}(\mathrm{use}(e),x) \geq \mathsf{U}(e,x)$

— *les fonctions de coûts sont cohérentes avec le système de codage des fonctions choisi (composition, produit, etc.)*

Note 1.1.7 Intuition sur les fonctions de simulation.

Plaçons-nous dans le cas d'une classe énumérable récursive. Considérons une fonction récursive d'indice e, une valeur d'entrée t et un nombre d'étape de calcul s. L'appel à $\mathrm{sim}\,\langle i,t,s\rangle$*, afin de tenter de calculer $\varphi_i(t)$ en s étapes de calcul, doit tenter de reconstruire l'arbre de calcul, tout en bornant chaque schéma μ de recherche par s. Ce faisant, la construction de l'arbre de calcul, quitte à ce qu'elle soit partielle, se fait bien de manière récursive primitive. De plus, sa taille est bornée de manière récursive primitive en s et donc sa manipulation reste facile.*

Pour un s assez grand, si la fonction d'indice e est bien définie en t, un arbre de calcul complet sera trouvé en utilisant cet algorithme.

Pour rendre sim *plus précise et faire en sorte qu'elle nous indique si le s en question est le plus petit permettant de dériver l'arbre de calcul, nous pouvons utiliser l'encodage suivant :*

$$\mathrm{sim} : \langle i, t, s\rangle \mapsto \begin{cases} \langle t, \varphi_i(t)\rangle + 1 & \textit{si le calcul termine pour } s \textit{ mais non pour } s-1, \\ 0 & \textit{sinon.} \end{cases}$$

Les classes énumérables récursives ne sont pas les seules classes énumérables dans lesquelles les définitions des fonctions de coût et de simulation sont possibles. En particulier, les classes de fonctions que nous verrons dans le chapitre 4 ne sont pas des classes récursives pour la notion de récursivité classique mais disposent tout de même de ces fonctions de simulation.

Définition 1.1.8 **Classes de fonctions énumérables avec fonctions de coût.** *Une classe de fonctions énumérable* $\mathtt{C}$ *est* avec fonctions de coût *s'il existe une fonction* use : $\omega \mapsto \omega$ *tel que pour toute fonction* $\Phi^{\mathtt{C}}_e$ *de* $\mathtt{C}$*,* $\Phi^{\mathtt{C}}_{\mathrm{use}(e)}$ *soit une fonction de coût (voir Définition 1.1.6) pour* $\Phi^{\mathtt{C}}_e$*.*

Cela implique notamment que chaque fonction dans $\mathtt{C}$ possède une fonction de coût elle-même dans $\mathtt{C}$.

1.1.3 Fonctions récursives totales et effectivité

Lorsqu'en calculabilité classique il est fait mention de l'ensemble des fonctions totales, souvent noté $\mathcal{TOT}$, celui-ci est introduit comme un ensemble de fonctions Π^0_2-complet [3] qui n'est donc pas calculable au sens Turing.

Cette complexité, dite descriptive, doit être considérée en regard du langage sur lequel elle est définie. Par exemple, dans la définition « $\{e : \forall x, \exists s, \varphi_{e,s}(x)\downarrow\}$ », les quantificateurs existentiel et universel ne sont bornés d'aucune manière et correspondent, une fois la formule rapportée à une version plus calculatoire, à des recherches et des vérifications sur tous les entiers.

Or, les entiers ne sont pas des objets si triviaux à obtenir. Il nous est certes possible de tous les énumérer sans grande difficulté, et même de les reconnaître par des formules très simples. Mais exhiber effectivement un de ces objets nécessite en réalité de puissants schémas récursifs afin de pouvoir exprimer les fonctions qui le calcule. De même, se demander combien d'étapes doit faire une machine de Turing donnée avant de s'arrêter a beau être un problème bien posé, le résoudre est en général indécidable. En pratique, on peut voir cela comme le fait qu'elle utilise des

3. Cela implique que sa définition nécessite exactement deux quantificateurs sur les entiers : un premier universel, puis un second existentiel. En particulier, on remarque qu'aucune fonction récursive (Σ^0_1) n'a une telle complexité.

schémas de récursion de plus en plus élevés, et que par conséquent il devient difficile de borner son nombre d'étapes en fonction des arguments qu'elle prend.

La classe des fonctions récursives totales entre bien dans notre définition des classes de fonctions totales, closes et énumérables. En revanche, elle n'est pas récursive puisqu'il n'existe pas de fonction universelle récursive pour les fonctions totales. Ne serait-ce le point soulevé plus haut sur la Π^0_2-complétude, un simple argument de diagonalisation suffit à s'en convaincre : soit ψ une fonction universelle pour la classe des fonctions totales et e son indice dans l'énumération qu'elle définit. Par composition, on peut construire une fonction ψ' qui, étant donné un entier a, renvoie $\psi(\langle a, a\rangle) + 1$. Cette fonction ψ' possède un indice e' et vérifie $\psi(\langle e', e'\rangle) = \psi'(e') = \psi(\langle e', e'\rangle) + 1$, ce qui est absurde.

Pour un plus large panel de résultats sur la calculabilité et les ensembles Δ^0_2 et Π^0_2, on pourra se référer à [Nie09] ou à [Soa87].

1.2 Fonctions à croissance rapide et hiérarchies

Afin de mettre plus facilement en regard les objets créés avec ceux de calculabilité classique déjà connus, nous observerons dans un premier temps des exemples vérifiant cette définition.

1.2.1 Exemples de classes de fonctions énumérables

La notion de complexité que nous allons utiliser implicitement par la suite et qui va nous guider dans notre étude est une complexité de *définissabilité* des fonctions en terme de schéma d'induction.

L'analyse ordinale d'une théorie repose notamment sur ce principe selon lequel un objet complexe est associé à un schéma d'induction fort, et ce dernier prend corps lorsque l'on considère les différents niveaux de la hiérarchie des ensembles constructibles de Gödel.

Pour mettre en avant ce découpage, nous allons identifier quelques classes bien connues de fonctions, closes, énumérables et récursives tout en focalisant notre attention sur les définitions de ces classes pour identifier quels opérateurs et quelles propriétés permettent au calcul d'émerger.

Fonctions élémentaires

Quand nous cherchons parmi les opérations sur les entiers, les premières à n'être pas triviales et à rendre compte de la complexité de ces derniers, ce que ne font pas les fonctions constante nulle et successeur, nous trouvons l'addition et la soustraction naturelle ($\dot{-} : x, y \mapsto \max\{0, x - y\}$). Il apparaît immédiatement que les fonctions n'utilisant que ces opérations se calculent facilement via un nombre d'étapes constant ou sous-linéaire suivant le codage des entiers utilisé.

Le nombre d'opérations et donc le déroulement du calcul commencent enfin à être dépendants des arguments donnés en entrée, avec les deux schémas de calcul que sont la somme bornée et le produit borné.

En ajoutant un schéma de composition des fonctions, nous obtenons une première caractérisation de notre classe :

Définition 1.2.1 **Fonctions élémentaires.** *La classe des fonctions élémentaires* ($\mathcal{E}$) *est la plus petite classe close formée des fonctions de base suivantes :*

1. $\mathbf{0} : x \mapsto 0$,
2. $\mathbf{s} : x \mapsto x + 1$,
3. $+ : x, y \mapsto x + y$,
4. $\dot{-} : x, y \mapsto \max\{0, x - y\}$,
5. $\pi^2_1, \pi^2_2, \langle \cdot, \cdot \rangle$;

et close par application des schémas ci-dessous :

6. *composition :* $f, g \in \mathcal{E}$ *implique* $f \circ g \in \mathcal{E}$,
7. *somme bornée :* $f : y, \vec{x} \mapsto f(y, \vec{x}) \in \mathcal{E}$ *implique* $g : m, \vec{x} \mapsto \sum_{i=0}^{i=m} f(i, \vec{x}) \in \mathcal{E}$,
8. *produit bornée :* $f : y, \vec{x} \mapsto f(y, \vec{x}) \in \mathcal{E}$ *implique* $g : m, \vec{x} \mapsto \prod_{i=0}^{i=m} f(i, \vec{x}) \in \mathcal{E}$.

Comme souvent, cette classe de fonctions va de paire avec une classe de prédicats, en l'occurrence, les prédicats dits élémentaires :

Définition 1.2.2 **Prédicats élémentaires.** *Un prédicat n-aire est élémentaire si sa fonction indicatrice*[4] *est élémentaire.*

Nous remarquons qu'il est facile de définir pour cette classe une fonction universelle qui soit récursive. Ainsi, les fonctions élémentaires forment bien une classe de fonctions close, énumérable et récursive.

Les schémas utilisés dans la définition ci-dessus sont strictement plus faibles que la récursion primitive dont nous parlerons plus tard. Intuitivement, ils ne permettent pas à une fonction de s'appeler récursivement et donc de travailler sur le résultat de son calcul.

En revanche, les fonctions élémentaires sont déjà capables de faire une recherche bornée :

4. Nous prendrons comme convention que la fonction indicatrice, notée χ, d'un ensemble A vaut 1 sur les éléments de A et 0 dans le reste de son domaine.

Propriété 1.2.3

La classe des fonctions élémentaires est stable par μ-récursion bornée.

Démonstration. Pour le vérifier, il suffit de montrer qu'il nous est possible de réécrire tout schéma μ borné en n'utilisant que les primitives des fonctions élémentaires.

Soit R un prédicat élémentaire, considérons la fonction suivante définie à $M \in \omega$ fixé, par schéma de μ-récursion borné :

$$f : x \mapsto \mu_{m \leqslant M}, R(x,m) = \begin{cases} \mu m, R(x,m) & \text{si } \exists m \leqslant M, R(x,m)\,; \\ 0 & \text{sinon} \end{cases}$$

Soit $\chi_{R'}$ la fonction indicatrice de $R' = \{(x,m) : R(x,m) \wedge \forall n < m, \neg R(x,n)\}$; $\chi_{R'}$ est élémentaire en χ_R :

$$\chi_{R'}(x,m) = \chi_R(x,m) \prod_{n \leqslant m-1} (1 - \chi_R(x,n));$$

Alors,

$$\mu m_{\leqslant M}, R(x,m) = \sum_{m \leqslant M} (m \times \chi_{R'}(x,m)).$$

□

Bien entendu, un résultat similaire existe sur les prédicats élémentaires.

Propriété 1.2.4

La classe des prédicats élémentaires est stable par connecteurs logiques et quantificateurs bornés.

Pour prouver ce résultat, il suffit de montrer que les opérations logiques (utilisation finie de connecteurs logiques et de quantification bornée) réalisables sur un prédicat sont transposables en opérations élémentaires sur une fonction indicatrice.

Ce résultat est une première indication du fait que la puissance de calcul d'une classe de fonctions récursives est liée à sa logique sous-jacente.

Fonctions récursives primitives

Les fonctions récursives primitives sont sans doute une des classes de fonctions récursives les plus étudiées. Elles introduisent le schéma de récursion borné qui, comme son nom l'indique, permet aux fonctions d'effectuer des appels récursifs un nombre borné de fois, tout en faisant décroître strictement un des arguments.

Elles sont habituellement définies comme suit :

Définition 1.2.5 **Fonctions récursives primitives.** *La classe des fonctions récursives primitives* p *est la plus petite classe close formée des fonctions de base suivantes :*

1. la fonction constante nulle est présente :

$$\mathbf{0} : x \mapsto 0\,;$$

2. la fonction successeur est présente :

$$\mathbf{s} : x \mapsto x + 1\,;$$

3. les projections sont présentes :

$$\pi_1^2, \pi_2^2.$$

et close par application des schémas ci-dessous :

4. le schéma de composition :

$$f, g \in \mathsf{p} \textit{ implique } f \circ g \in \mathsf{p}\,;$$

5. le schéma de récursion primitive :

$$\begin{aligned}&\textit{si } g, h \in \mathsf{p}, \textit{ alors } f : n, \vec{x} \mapsto \mathbf{rec}_{\mathsf{p}}(g, h, n, \vec{x}) \in \mathsf{p},\\ &\textit{avec } \mathbf{rec}_{\mathsf{p}}(g, h, n, \vec{x}) = \begin{cases} h(\vec{x}) & \textit{si } n = 0,\\ g(n, \vec{x}, \mathbf{rec}_{\mathsf{p}}(g, h, n-1, \vec{x})) & \textit{sinon.}\end{cases}\end{aligned}$$

Clairement, toutes les fonctions élémentaires sont récursives primitives. De plus, les schémas de clôture impliquant ceux utilisés pour la définition des fonctions élémentaires, nous comprenons pourquoi les prédicats récursifs primitifs, définis de manière analogue aux prédicats élémentaires, forment également une classe de prédicats stable par connecteurs logiques et quantificateurs bornés.

Il existe bien entendu une fonction récursive $\phi_p(\langle\cdot,\cdot\rangle)$ qui est universelle pour les fonctions primitives récursives. Elle peut être vue comme une fonction réduisant pas à pas un arbre de calcul déterminé par le code de la fonction primitive récursive passée en argument. Cependant, comme nous le verrons par la suite, cette fonction universelle n'est pas récursive primitive.

Fonctions α-récursives

Le fossé entre les fonctions récursives primitives et les fonctions récursives totales est très grand. La *première* des fonctions récursives qui ne soit pas primitive récursive est la fonction d'Ackermann-Péter, appelée par la suite fonction d'Ackermann.

Définition 1.2.6 **Fonction d'Ackermann.**

$$A : m, n \mapsto \begin{cases} n + 1 & \textit{si } m = 0,\\ A(m-1, 1) & \textit{si } m > 0 \textit{ et } n = 0\\ A(m-1, A(m, n-1)) & \textit{sinon.}\end{cases}$$

Cette fonction récursive ne peut pas être définie via le schéma de récursion primitive $\mathbf{rec}_{\mathsf{p}}$. En effet, ce schéma est un schéma de récursion sur ω, qui par composition finie ne peut atteindre le schéma de récursion sur ω^ω utilisé par la fonction d'Ackermann.

Afin de clarifier les choses, définissons le schéma d'α-récursion, permettant de construire une fonction récursive pour un ordinal récursif *presque*[5] quelconque α. Ce schéma est introduit par Michael Rathjen dans [Rat99].

Soit α un ordinal calculable pour lequel il existe une notation ordinale, c'est-à-dire un codage des ordinaux inférieurs à α par des entiers. Et notons $\lhd$ la relation d'ordre induite sur les entiers par cette notation ordinale.

Nous obtenons alors un schéma de récursion :

Définition 1.2.7 **Schéma d'α-récursion.** *Le schéma de récursion sur $\lhd$ d'ordinal type α s'exprime de la manière suivante :*

$$\mathbf{rec}_{\alpha,\lhd}(g,h,n,\vec{x}) = \begin{cases} g(n,\vec{x},\mathbf{rec}_{\alpha,\lhd}(g,h,\theta(n,\vec{x}),\vec{x})) & \textit{si } \overline{0} \lhd n \textit{ et } \theta(n,\vec{x}) \lhd n, \\ h(n,\vec{x}) & \textit{sinon,} \end{cases}$$

où $\overline{0}$ représente l'entier codant l'ordinal 0.

En posant $\alpha = \omega$, $\lhd = \in\restriction_\omega$ et $\theta = (n,\vec{x} \mapsto n-1)$, nous obtenons, avec g et h des fonctions primitives récursives, le schéma $\mathbf{rec}_{\mathsf{p}}$.

Pour $\alpha = \omega^\omega$, qui est isomorphe à l'ordre anti-lexicographique pour les listes finies d'entiers ($\lhd_{\text{antilex}}$), nous obtenons un codage très naturel de la fonction d'Ackermann :

$$A = n,m \mapsto \mathbf{rec}_{\omega^\omega,\lhd_{\text{antilex}}}(\mathbf{s}\circ\pi^2_2, \mathfrak{c}_1, \langle n,m\rangle),$$

c'est à dire $A = n,m, \mapsto f(\langle n,m\rangle)$ et

$$f : l \mapsto \begin{cases} f(\theta(l))+1 & \text{si } \theta(l) \neq \langle 0\rangle, \\ 0 & \text{sinon.} \end{cases}$$

avec

$$\theta : \begin{cases} \langle n\rangle & \mapsto \langle 0\rangle \\ \langle m_0,\cdots,m_k,0,n\rangle & \mapsto \langle m_0,\cdots,m_k,n+1\rangle \\ \langle m_0,\cdots,m_k,m+1,0\rangle & \mapsto \langle m_0,\cdots,m_k,m,1\rangle \\ \langle m_0,\cdots,m_k,m+1,n+1\rangle & \mapsto \langle m_0,\cdots,m_k,m+1,n\rangle \end{cases}$$

et $\mathbf{s}\circ\pi^2_2$ la fonction qui prend le deuxième argument (c'est à dire, dans le schéma d'α-récursion, le résultat de l'appel récursif, et l'incrémente. Ainsi, A compte simplement le nombre d'appels récursifs effectués suivant la fonction θ qui parcourt ω^ω de manière strictement décroissante.

5. Nous reviendrons sur ce point dans le chapitre traitant des sous-calculabilités.

Nous remarquons ici que θ est une fonction récursive primitive, de même que $\mathbf{s} \circ \pi_2^2$ et $\mathfrak{c}_1$. Le prédicat $\lhd_{\text{antilex}}$ est, lui, un prédicat récursif primitif.

Plus généralement, il est possible de définir la classe des fonctions α-récursives comme la classe des fonctions récursives primitives —ou de manière équivalente, élémentaires— close par application du schéma d'α-récursion à paramètres α-récursifs.

Définition 1.2.8 **Classes de fonctions α-récursives.** *À α donné, La classe des fonctions α-récursive $\mathtt{C}_\alpha$ est la plus petite classe close constituée des fonctions de base suivantes :*

1. *fonction constante nulle :* $\mathbf{0} : x \mapsto 0$ *;*
2. *fonction successeur :* $\mathbf{s} : x \mapsto x + 1$ *;*
3. *projections :* π_1^2, π_2^2 *;*

et close par application des schémas ci-dessous :

4. *composition :* $f, g \in \mathtt{C}_\alpha$ *implique* $f \circ g \in \mathtt{C}_\alpha$ *;*
5. *α-récursion :*

$$\mathbf{rec}_{\alpha,\lhd}(g, h, n, \vec{x}) = \begin{cases} g(n, \vec{x}, \mathbf{rec}_{\alpha,\lhd}(g, h, \theta(n, \vec{x}), \vec{x})) & \textit{si } \overline{0} \lhd n \textit{ et } \theta(n, \vec{x}) \lhd n, \\ h(n, \vec{x}) & \textit{sinon ;} \end{cases}$$

où $\chi_\lhd, g, h, \theta \in \mathtt{C}_\alpha$.

Ces classes de fonctions nous permettent d'établir une correspondance immédiate entre la complexité de *définissabilité* des fonctions et la complexité maximale de leurs preuves de terminaison. En effet, l'arbre de preuve de terminaison d'une fonction α-récursive est de profondeur au plus α (voir par exemple [Poh08] et [Rat99]).

1.2.2 Hiérarchies de croissance et fonctions α-récursives

Nous présentons en Annexe A un ensemble de constructions de séquences fondamentales, également décrites dans [Odi99], nous permettant de matérialiser tout un ensemble d'ordinaux calculables. Une fois que nous disposons de ces séquences fondamentales, il devient possible de créer des familles de fonctions récursives croissant toujours plus rapidement les unes que les autres.

Pour ce faire, il suffit d'itérer une opération récursivement à l'aide des séquences fondamentales. Suivant la manière dont les opérations sont itérées, il est possible d'obtenir différentes hiérarchies avec différents niveaux de croissance. Alors que la hiérarchie à croissance rapide avait déjà été définie en 1965 par Robbin dans sa thèse [Rob65], les hiérarchies à croissance modérée et faible ont été définies et étudiées plus tard par Wainer, voir [Wai72] et Constable, voir [Con70].

Voici les trois hiérarchies :

Définition 1.2.9 **(Wainer) Hiérarchie à Croissance Lente.** *La famille de fonctions $(f_\alpha)_{\alpha \in \omega_1}$ est définie de la manière suivante :*

$$\begin{array}{rcl} f_0(x) & = & 0 \\ f_{\alpha+1}(x) & = & f_\alpha(x)+1 \\ f_\alpha(x) & = & f_{\alpha_x}(x) \textit{ si } \alpha \textit{ est un ordinal limite.} \end{array}$$

La hiérarchie à croissance lente, introduite par Wainer en 1972 est une des dernières à avoir été définies. Nous l'utiliserons ici pour montrer un gros inconvénient de l'utilisation des séquences fondamentales, et pourquoi nous nous intéresserons principalement aux séquences fondamentales canoniques, qui sont élémentaires.

Comme nous l'avons vu plus haut, les séquences fondamentales sont arbitrairement choisies. Soit $(\alpha_n)_{n\in\omega}$ une séquence fondamentale pour ω, c'est-à-dire une suite strictement croissante d'entiers. Remarquons alors que pour tout entier n, $f_\omega(n) = \alpha_n$. Autrement dit, si nous ne faisons pas attention aux séquences fondamentales que nous choisissons, un choix malheureux nous permettrait de calculer Ackermann, voire le problème de l'arrêt, à des niveaux très bas de cette hiérarchie prétendument lente.

Définition 1.2.10 **(Wainer) Hiérarchie à croissance modérée.** *La famille de fonctions $(g_\alpha)_{\alpha\in\omega_1}$ est définie de la manière suivante :*

$$\begin{array}{rcl} g_0(x) & = & x+1 \\ g_{\alpha+1}(x) & = & g_\alpha(x+1) \\ g_\alpha(x) & = & g_{\alpha_x}(x) \textit{ si } \alpha \textit{ est un ordinal limite.} \end{array}$$

Définition 1.2.11 **(Robbin, Constable, Löb et Wainer, Schwichtemberg) Hiérarchie à Croissance Rapide.** *La famille de fonctions $(h_\alpha)_{\alpha\in\omega_1}$ est définie de la manière suivante :*

$$\begin{array}{rcl} h_0(x) & = & x+1 \\ h_{\alpha+1}(x) & = & h_\alpha^{(x)}(x) \\ h_\alpha(x) & = & h_{\alpha_x}(x) \textit{ si } \alpha \textit{ est un ordinal limite} \end{array}$$

Un préfixe de la hiérarchie à croissance rapide avait déjà été défini par Grzegorczyk mais uniquement pour les indices finis. Les classes d'équivalence de cette hiérarchie permettent de classer toutes les fonctions primitives. Aussi, cette hiérarchie à croissance rapide est parfois appelée « hiérarchie de Grzegorczyk étendue ».

Nous n'allons pas développer plus en détail ces hiérarchies, mais avant de refermer cette parenthèse, nous allons voir un exemple de fonction qui se trouve à différents niveaux de chacune d'entre elles. La fonction en question est la fonction diagonale des fonctions récursives primitives, identifiée ici par $h_\omega = n \mapsto h_n(n)$. En effet, quelle que soit la fonction récursive primitive f, il existe un m tel que $h_m > f$

en tous points, h_ω domine donc toutes les fonctions récursives primitives. Elle est d'ailleurs élémentairement équivalente à la fonction d'Ackermann. Autrement dit, la fonction d'Ackermann correspond au niveau ω de la hiérarchie à croissance rapide. Pour s'en rendre compte, il suffit d'exprimer la fonction d'Ackermann en fonction des puissances itérées de Donald Knuth, Nous obtenons alors $\mathrm{Ack}(m,n) = 2 \uparrow^{m-2} (n+3) - 3$, or $h_\omega(n) > 2 \uparrow^{n-1} n$. Et inversement, il est possible de montrer que h_ω est minimale pour la domination des fonctions primitives récursives, voir [Odi99] pour plus de détails.

Bien évidemment, plus la hiérarchie est à croissance lente et plus la fonction h_ω s'y trouvera à un rang élevé. Pour la hiérarchie à croissance modérée, h_ω correspond au rang ω^ω.

Théorème 1.2.12 **Hardy 1904, Wainer 1972**

Position de h_ω dans la hiérarchie à croissance modérée : $h_\omega \equiv_\mathcal{E} g_{\omega^\omega}$.

Dans le cas de la hiérarchie à croissance faible, il faut utiliser la hiérarchie de Veblen pour parvenir à décrire l'ordinal du rang de h_ω dans celle-ci. Un résultat de Jean-Yves Girard [GP81] nous permet de trouver la position de h_ω dans la hiérarchie à croissance faible :

Théorème 1.2.13 **Girard 1981**

Position de h_ω dans la hiérarchie à croissance faible : $h_\omega \equiv_\mathcal{E} f_{\phi_\omega(0)}$.

Les preuves de ces résultats un peu techniques peuvent être trouvées dans une présentation par Piergiorgio Odifreddi dans [Odi99].

1.3 Structure sur les classes énumérables

Une base commune ressort de tous les exemples que nous avons vu précédemment, tant du point de vue axiomatique où chaque classe semble être une sur-classe de la précédente, que du point de vue algébrique où l'énumération semble toujours posséder les propriétés familières que nous avons l'habitude de rencontrer lorsque nous manipulons les énumérations classiques.

Dans un premier temps, nous allons expliquer quels sont les résultats généraux que nous pouvons obtenir du fait que nos fonctions soient totales et que nos classes permettent la manipulation d'indices facilement. Ensuite, nous développerons plus particulièrement les conséquences de l'existence d'une énumération.

Nous nous plaçons donc dans la suite dans le cas d'une classe énumérable $\mathtt{C}$, de domaine $\mathcal{D} = \omega$ pour laquelle il existe une fonction de simulation sim et une fonction de calcul de la fonction de coût use.

1.3.1 Forme normale algébrique

À partir d'un ensemble de fonctions de base, il est très facile d'en construire de nouvelles (pas nécessairement différentes des précédentes) par composition, en effet, nos fonctions sont totales et de même domaine, la composition est ici une opération naturelle.

De plus, étant donné l'existence d'une fonction totale universelle pour les classes énumérables, toute une collection de fonctions peut être décrite en utilisant les fonctions de manipulation d'indices.

Par exemple, pour $e \in \mathcal{D}$, nous pouvons avoir envie de parler de

$$\text{« } \mathsf{U}_{\mathsf{C}}(\mathsf{U}_{\mathsf{C}}(e,e), \pi_1^2(\mathbf{s}(\mathbf{const}(\mathfrak{c}_{42}(x))))) \text{ ».}$$

Nous avons même envie de dire qu'il suffit de pouvoir l'écrire pour savoir le calculer. Kozen a formalisé cela par ce qu'il appelle les *termes* et les *termes itératifs*. Nous n'entrerons pas dans les détails ici et ne définirons pas le langage sous-jacent à ces termes, mais citerons simplement quelques propriétés facilitant notre raisonnement pour la suite. Pour plus de détails, le lecteur se reportera à [Koz80].

Les fonctions de manipulation d'indices

Lors de l'étude d'une classe possédant une fonction universelle, les fonctions sont souvent identifiées avec leurs descriptions sous forme d'indice. De la même manière que nous manipulons les fonctions en les composant, en les projetant et en leur appliquant des schémas, nous avons souvent l'habitude d'en faire de même avec leurs indices.

Par exemple, étant donné l'indice $e \in \mathcal{D}$ d'une fonction $f \in \mathsf{C}$, comme nous savons que $f \circ f \in \mathsf{C}$, nous pouvons vouloir exhiber un indice en témoignant. Les fonctions permettant de trouver ses indices sont celles que nous appellerons *fonctions de manipulation d'indices*. Elles forment une classe Ω_{C} relative à une classe de fonction et à son énumération considérée.

Définition 1.3.1 **Classes Ω_{C}.** *La classe Ω_{C} contient l'ensemble des fonctions de manipulation d'indices pour l'énumération de la classe C. Elle est la plus petite classe close qui contient les fonctions de base suivantes :*
- $\langle \cdot, \cdot \rangle : \mathcal{D}^2 \to \mathcal{D}, \pi_1^2, \pi_2^2 : \mathcal{D} \to \mathcal{D} \in \Omega_{\mathsf{C}}, \forall i \in \{1,2\}, \forall d_1, d_2, \in \mathcal{D}, \pi_i^2(\langle d_1, d_2 \rangle) = d_i$;
- $\mathbf{const} \in \mathsf{C}, \forall c, d \in \mathcal{D}, \mathsf{U}_{\mathsf{C}}(\mathbf{const}(c), d) = \mathfrak{c}_c(d) = c$;
- $\mathbf{comp} \in \mathsf{C}, \forall f, g, d \in \mathcal{D}, \mathsf{U}_{\mathsf{C}}(\mathbf{comp}(f,g), d) = \mathsf{U}_{\mathsf{C}}(f, \mathsf{U}_{\mathsf{C}}(g,d))$;
- $\mathbf{pair} \in \mathsf{C}, \forall f, g, d \in \mathcal{D}, \mathsf{U}_{\mathsf{C}}(\mathbf{pair}(f,g), d) = \langle \mathsf{U}_{\mathsf{C}}(f,d), \mathsf{U}_{\mathsf{C}}(g,d) \rangle$;

et est close par application du schéma de composition.

Le paragraphe suivant étudie le langage induit par cette classe et la correspondance entre l'existence d'une description d'indice et de la fonction associée.

Termes et termes itératifs

Dans cette section, nous allons étudier le langage utilisé informellement sur une classe de fonction dès lors que nous effectuons des constructions combinant différents objets issus de ces classes.

Les intérêts sont multiples. Le premier est d'isoler quels sont les composants nécessaires de ce langage, les briques élémentaires qui sont utiles pour construire des fonctions. Ensuite, si nous réussissons à prouver que toutes les constructions descriptibles via ce langage correspondent à des objets qui existent effectivement à l'intérieur de la classe étudiée, cela permettra de les justifier une fois pour toutes.

Définition 1.3.2 **Termes.** *Un* terme *est une expression composée uniquement de la fonction universelle, de constantes issues du domaine $\mathcal{D}$ et de variables libres évoluant dans $\mathcal{D}$.*

Les variables libres d'un terme t, sont indiquées entre crochets :

$$t\,[x_1, \cdots, x_n]\,.$$

Ce terme pourra être assimilé à la fonction f associée :

$$f : x_1, \cdots, x_n \mapsto t\,[x_1, \cdots, x_n]\,.$$

Les termes représentent simplement des valeurs à calculer, c'est-à-dire des expressions figées qui déterminent de manière unique le résultat du calcul. Pour encore plus de flexibilité, nous pouvons définir ce que Kozen a appelé les termes itératifs. Ces objets généralisent la notion de termes aux fonctions présentent dans $\Omega_{\mathtt{C}}$.

Définition 1.3.3 **Termes itératifs.** *La classe des termes itératifs est définie par induction. Soit $d_1, d_2, \ldots, d_n$ une famille de constantes de $\mathcal{D}$ ou de variables à valeurs dans $\mathcal{D}$.*

- *Soit p une variable fonctionnelle à valeur dans $\Omega_{\mathtt{C}}$, alors $p(d_1, \ldots, d_n)$ est un terme itératif.*
- *Soit t un terme itératif, alors $\Phi_t^{\mathtt{C}}(d_1, \ldots, d_n)$ est un terme itératif.*

Théorème 1.3.4

Les fonctions de transformation d'indices descriptibles via un terme itératif sur $\mathtt{C}$ sont membres de $\Omega_{\mathtt{C}}$.

1.3.2 Manipulations d'indices et arités de fonctions

Une des transformations d'indices les plus courantes reste la manipulation d'arités. Suivant la présentation de la classe de fonctions étudiée, ces manipulations sont axiomatiques, nécessitent une preuve plus complexe et bien souvent technique ou bien encore ces transformations peuvent être interdites.

Il reste que dans la plupart des classes définies de manière algébrique, comme dans notre cas par des propriétés de clôture via composition et projection, le théorème s_n^m s'applique.

Ici, il est possible d'en donner une preuve effective :

Théorème 1.3.5

s_n^m **pour les classes énumérables.** *On rappelle que* $\Phi_e^{\mathsf{C}}(n) = \mathsf{U}_{\mathsf{C}}(e,n)$. *Soit* $m, n \in \omega$, $e \in \mathcal{D}$ *et* $\Phi_{\cdot}^{\mathsf{C}}$ *une énumération de* C. *Alors, il existe une fonction* $s_n^m \in \mathsf{C}$ *vérifiant*

$$\Phi_e^{\mathsf{C}}(\langle x_1, \ldots, x_n, y_1, \ldots, y_m \rangle) = \Phi_{s_n^m(e,x_1,\ldots,x_n)}^{\mathsf{C}}(\langle y_1, \ldots, y_m \rangle).$$

Dans la suite on ommetra les chevrons « $\langle\rangle$ », l'arité pouvant être modifée via des fonctions de codage.

Démonstration. Il suffit pour s'en convaincre de remarquer que pour tout n et m, il nous est possible de construire la fonction suivante :

$$s_n^m = \mathbf{comp} \circ \left\langle \pi_1^{n+1}, \mathbf{pair} \circ \left\langle \mathbf{const} \circ \pi_2^{n+1}, \right.\right.$$
$$\left.\left. \mathbf{pair} \circ \left\langle \mathbf{const} \circ \pi_3^{n+1}, \ldots, \left\langle \mathbf{const} \circ \pi_{n+1}^{n+1}, \mathfrak{c}_{\mathfrak{c}_{\pi_1^1}} \right\rangle \right\rangle \right\rangle \right\rangle$$

Par exemple, dans le cas $m = n = 1$:

$$\begin{aligned}
\Phi_{s_1^1(e,x_1)}^{\mathsf{C}}(y_1) &= \Phi^{\mathsf{C}}_{\mathbf{comp}\circ\left\langle \pi_1^2, \mathbf{pair}\circ\left\langle \mathbf{const}\circ\pi_2^2, \mathfrak{c}_{\mathfrak{c}_{\pi_1^1}}\right\rangle\right\rangle(e,x_1)}(y_1) \\
&= \Phi^{\mathsf{C}}_{\mathbf{comp}\circ\left\langle e, \mathbf{pair}\circ\left\langle \mathfrak{c}_{x_1}, \mathfrak{c}_{\pi_1^1}\right\rangle\right\rangle}(y_1) \\
&= \Phi_e^{\mathsf{C}}(\Phi^{\mathsf{C}}_{\mathbf{pair}\circ\left\langle \mathfrak{c}_{x_1}, \mathfrak{c}_{\pi_1^1}\right\rangle}(y_1)) \\
&= \Phi_e^{\mathsf{C}}(\left\langle \Phi^{\mathsf{C}}_{\mathfrak{c}_{x_1}}(y_1), \Phi^{\mathsf{C}}_{\mathfrak{c}_{\pi_1^1}}(y_1) \right\rangle) \\
&= \Phi_e^{\mathsf{C}}(\langle x_1, y_1 \rangle).
\end{aligned}$$

□

Il convient de remarquer que les fonctions s_n^m ainsi définies appartiennent par construction à Ω_{C}. De plus, sans récursion primitive, il n'est pas possible de définir une fonction $s : (m, n, e, \langle x_1, \ldots, x_m\rangle) \mapsto s_n^m(e, x_1, \ldots, x_m)$ de manière uniforme. La notion de classe fondamentale que nous allons développer au chapitre 2 règlera ce problème, ces classes étant closes par récursion primitive.

1.3.3 Ensembles et fonctions remarquables

Bien que nous travaillions dans une classe de fonctions totales, nous pouvons étudier des ensembles clés en calculabilité classique.

L'ensemble diagonal

L'ensemble diagonal est habituellement l'exemple canonique de problème non calculable par une fonction récursive. La raison pour laquelle il n'est pas calculable est l'impossibilité de savoir a priori si un calcul arbitraire termine ou pas.

Il y a un pendant très similaire dans les classes de fonctions totales. En effet, comme nous l'avons démontré, ces classes ne possèdent pas de fonction universelle. Pourtant, le théorème de la forme normale de Kleene nous indique qu'il est possible de simuler *pas à pas* les fonctions récursives de manière primitive récursive. Ce qu'il manque pour faire une fonction universelle, c'est la connaissance de ce nombre de pas à effectuer pour terminer le calcul. Bien sûr, pour chaque fonction récursive totale, il est possible de construire à partir de son indice celui d'une autre fonction récursive totale qui calcule le nombre de pas effectués par la première et qui appartiendra vraisemblablement elle aussi à la même classe de fonctions. Cependant, cette transformation sur les indices a beau être uniforme, il manque toujours une fonction universelle pour l'exécuter. Or, une fonction universelle, autrement dit une fonction qui permette d'exécuter uniformément n'importe quelle fonction dont elle connaît l'indice, est nécessairement hors de la classe.

Aussi, ne pouvant pas simuler les fonctions entre elles, le fait de connaître la valeur d'une fonction en un point chez les fonctions totales est très similaire à connaître l'arrêt d'une fonction en un point en calculabilité classique.

Nous avons alors notre nouvel ensemble diagonal :

Définition 1.3.6 Ensemble diagonal pour une classe énumérable.

$$\mathsf{K}_{\mathsf{C}}^{\Phi} = \left\{e :\ \mathsf{U}_{\mathsf{C}}^{\Phi}(e, e) > 0\right\} = \left\{e :\ \Phi_e^{\mathsf{C}}(e) > 0\right\}$$

La fonction du castor affairé

La fonction du castor affairé introduite initialement par Tibor Radó pour les machines de Turing, voir [Rad62], et que nous allons généraliser ici, est très liée à la problématique de l'arrêt des fonctions totales. Cette fonction donne l'élément maximal (ou les éléments maximaux si $\mathcal{D}$ n'est pas un ordre total) que sont capables de produire les fonctions d'un préfixe de l'énumération.

Considérons la fonction *use* définie dans la Définition 1.1.6 telle que :

$$\forall x, e, \Phi_e^{\mathsf{C}}(x) = \pi_3^3 \circ \mathrm{sim}(e, x, \Phi_{\mathrm{use}(e)}^{\mathsf{C}}(x)).$$

Cette fonction existe pour toute classe de fonctions C capable de manipuler un schéma de codage à la numérotation de Gödel codant des opérations élémentaires dont la fonction de coût est présente dans C. C'est le cas par exemple des fonctions primitives récursives, de manière plus générale de toutes les classes α-récursives et même de toutes les classes énumérables récursives.

Définition 1.3.7 Fonctions du castor affairé pour une classe énumérable disposant des fonctions sim **et** use**.**

La version originale de cette fonction est la suivante :

$$\mathsf{BB}_{\mathsf{C}}^{\Phi'} : x \mapsto \max\{\Phi_i^{\mathsf{C}}(0) : i \leqslant x\} + x$$

Nous allons utiliser une variante forte $\mathsf{BB}_{\mathsf{C}}^{\Phi}$ *de la fonction du Castor Affairé, qui domine la fonction* $\mathsf{BB}_{\mathsf{C}}^{\Phi'}$ *mais peut à son tour être calculée à partir de celle-ci.*

$$\mathsf{BB}_{\mathsf{C}}^{\Phi} = x \mapsto max\left\{\Phi_i^{\mathsf{C}}(0), \Phi_{\mathrm{use}(i)}^{\mathsf{C}}(0) : i \leqslant x\right\} + x$$

On rappelle que dominer une fonction du Castor Affairé implique de savoir calculer celle-ci. Or justement, cette fonction domine $\mathsf{BB}_{\mathsf{C}}^{\Phi'}$, mais est dominée par

$$n \mapsto max\left(\mathsf{BB}_{\mathsf{C}}^{\Phi'}(n), \mathsf{BB}_{\mathsf{C}}^{\Phi'} \circ \mathrm{use}(n)\right).$$

Il est ensuite aisé de calculer uniformément en e et x, via une fonction $f \in \Omega_{\mathsf{C}}$, l'indice d'une fonction Φ_c^{C} telle que $\Phi_c^{\mathsf{C}}(0) = \Phi_{\mathrm{use}(e)}^{\mathsf{C}}(x)$.

$\mathsf{BB}_{\mathsf{C}}^{\Phi}(c)$ donnera alors une borne supérieure du nombre d'étapes de simulation de e en x. Autrement dit, $e, x \mapsto \pi_3^3\left(\mu s \leqslant \mathsf{BB}_{\mathsf{C}}^{\Phi}(f(e,x)), \mathrm{sim}(e,x,s) > 0\right)$ est une fonction universelle pour C.

Ainsi, la fonction du castor affairé est la fonction naturelle permettant de calculer *l'arrêt* chez les fonctions totales.

1.3.4 Réductions entre fonctions et fonctions bornées

Les opérations de clôture choisies pour la définition de nos classes énumérables sous-tendent l'existence d'une relation de réduction entre les fonctions. Par exemple demander que si une fonction f est présente dans une classe C, alors, étant données les opérations de clôture de celle-ci, une autre fonction g donnée sera nécessairement présente.

Ces réductions nous permettent de discuter des différentes extensions des classes énumérables et de les comparer.

Réductions entre fonctions

La première réduction que nous allons définir est inspirée de la réduction *many-one* des problèmes de reconnaissance.

Définition 1.3.8 **Réduction *many-one*.** *f se $\mathtt{C}$-réduit* many-one *à h ($f \leqslant_m^{\mathtt{C}} h$) s'il existe $g \in \mathtt{C}$ telle que $f = h \circ g$.*

Notons que cette réduction ne demande pas à ce que f et g soient dans $\mathtt{C}$. Seule la fonction utilisée pour faire la réduction doit être dans $\mathtt{C}$. C'est donc une version $\mathtt{C}$-limitée de la réduction *many-one*.

Mais la composition à droite n'est qu'une des nombreuses opérations de clôture présentes dans les classes énumérables. Aussi, si nous cherchons à identifier les fonctions que nous pouvons construire via les opérations de clôture d'une classe $\mathtt{C}$ donnée, il nous faut définir une relation plus faible.

Définition 1.3.9 **Réduction de clôture.** *f se $\mathtt{C}$-réduit à g ($f \leqslant^{\mathtt{C}} g$) si f appartient à la plus petite classe close*[6] *incluant $\mathtt{C}$ et contenant g.*

Un cas particulier de cette relation qui mérite une étude approfondie est la réduction $\leqslant^{\Omega_{\mathtt{C}}}$. Elle nous permet de considérer toutes les extensions de la classe $\mathtt{C}$ qu'il nous est possible de réaliser à partir d'une fonction *oracle* f quelconque en utilisant toutes les fonctions de manipulation d'indices offertes par $\mathtt{C}$. Nous y reviendrons par la suite.

Enfin, une réduction qui nous intéressera tout particulièrement pour mettre en lumière les distinctions majeures entre la notion de récursivité dans le cas classique et les notions de récursivités que nous allons définir par la suite est la suivante. Cette fois inspirée de la réduction par table de vérité bornée, nous définissons la relation suivante :

Définition 1.3.10 **Réduction *btt* $\mathtt{C}$-limitées.** *f se $\mathtt{C}$-*btt*-réduit à h ($f \leqslant_{\mathrm{btt}}^{\mathtt{C}} h$) si pour un certain entier $k \in \omega$ il existe deux fonctions $e, g \in \mathtt{C}$ telles que :*

$$f = e \circ \left\langle \pi_1^1, \left\langle h \circ \pi_1^k, \ldots, h \circ \pi_k^k \right\rangle \circ g \right\rangle$$

En d'autres termes, il doit exister $k \in \omega$ et une suite de $k+1$ fonctions e', g_1, ..., g_k telles que pour tout x, $f(x) = e'(x, h(g_1(x)), \ldots, h(g_2(x)))$.

Bien entendu, ces trois réductions permettent de définir des classes d'équivalence de manière similaire qui seront notées si besoin $\equiv^{\mathtt{C}}$, $\equiv_m^{\mathtt{C}}$ et $\equiv_{\mathrm{btt}}^{\mathtt{C}}$.

Fonctions particulières

Ici, nous allons définir quelques fonctions classiques qui nous permettront d'identifier les spécificités de ces relations dans le contexte des classes closes et de remar-

6. Si $\mathtt{C}$ est énumérable, alors cette classe close le sera également.

quer comment se comportent les énumérations face aux réductions définies précédemment.

De manière générale, χ_E désigne la fonction indicatrice de l'ensemble E. C'est une fonction à valeurs dans $\{0,1\}$. Nous nous intéresserons particulièrement au cas de la fonction élémentaire du graphe d'une fonction universelle d'une classe énumérable fixée :

$$\chi_{\mathrm{graph}\,\mathsf{U}_{\mathsf{C}}} = x, y, z \mapsto \begin{cases} 1 & \text{si } \boldsymbol{\Phi}^{\mathsf{C}}_x(y) = z \\ 0 & \text{sinon.} \end{cases}$$

Une fonction qui dans le cas classique est équivalente à cette première est la fonction indicatrice de l'ensemble diagonal :

$$\chi_{\mathsf{K}^{\Phi}_{\mathsf{C}}} = x \mapsto \begin{cases} 1 & \text{si } \boldsymbol{\Phi}^{\mathsf{C}}_x(x) = 0 \\ 0 & \text{sinon.} \end{cases}$$

Une troisième fonction équivalente dans le cas classique est la fonction membre_x qui recherche uniformément x dans le domaine de définition de la fonction de la classe C d'indice donné en paramètre :

$$\mathrm{member}_y = x \mapsto \begin{cases} 1 & \text{si } \boldsymbol{\Phi}^{\mathsf{C}}_x(y) = 1 \\ 0 & \text{sinon.} \end{cases}$$

Remarquons tout d'abord que toutes ces fonctions, équivalentes dans le cas classique, le restent dans le cas des classes énumérables.

Proposition 1.3.11

Les fonctions bornées présentées sont équivalentes :

$$\chi_{\mathrm{graph}\,\mathsf{U}_{\mathsf{C}}} \equiv^{\Omega_{\mathsf{C}}}_m \chi_{\mathsf{K}^{\Phi}_{\mathsf{C}}} \equiv^{\Omega_{\mathsf{C}}}_m \mathrm{member}_y$$

Démonstration. Comme l'intuition peut le laisser penser, la fonction indicatrice du graphe de la fonction universelle permet de calculer les deux autres fonctions. Pour s'en convaincre, il suffit de considérer les deux égalités suivantes :

— $\chi_{\mathsf{K}^{\Phi}_{\mathsf{C}}} = x \mapsto \chi_{\mathrm{graph}\,\mathsf{U}^{\Phi}_{\mathsf{C}}}(x, x, 0)$

— $\mathrm{member}_y = x \mapsto \chi_{\mathrm{graph}\,\mathsf{U}^{\Phi}_{\mathsf{C}}}(x, y, 1)$

Au passage, remarquons que ces constructions n'utilisent en fait que des transformations d'indices simples et donc présentes dans Ω_{C}.

Nous venons de prouver $\chi_{\mathrm{graph}\,\mathsf{U}^{\Phi}_{\mathsf{C}}} \geqslant^{\Omega_{\mathsf{C}}}_m \chi_{\mathsf{K}^{\Phi}_{\mathsf{C}}}$ et $\chi_{\mathrm{graph}\,\mathsf{U}^{\Phi}_{\mathsf{C}}} \geqslant^{\Omega_{\mathsf{C}}}_m \mathrm{member}_y$. Il reste désormais à montrer les réductions dans l'autre sens.

Commençons par exprimer $\chi_{\mathrm{graph}\,\mathsf{U}^{\Phi}_{\mathsf{C}}}$ sous la forme $\chi_{\mathsf{K}^{\Phi}_{\mathsf{C}}} \circ g$ avec $g \in \Omega_{\mathsf{C}}$.

Pour ce faire, il nous faut trouver l'indice d'une fonction qui, appelée sur cet indice, calculera $\chi_{\mathrm{graph}\,\mathsf{U}^{\Phi}_{\mathsf{c}}}$. La construction suivante ne nécessite pas de point fixe et fait très bien l'affaire :

$$\Phi^{\mathsf{c}}_{g(x,y,z)}(w) = \begin{cases} 0 & \text{si } \Phi^{\mathsf{c}}_x(y) = z \\ 1 & \text{sinon.} \end{cases}$$

Nous vérifions que la fonction g peut s'écrire en utilisant uniquement les primitives de Ω_{c} de la manière suivante :

$$\begin{aligned} g = \mathbf{comp} \circ \big\langle \mathfrak{c}_{\mathbf{cond}}, \mathbf{pair} \circ \big\langle \mathbf{comp} \circ \big\langle \pi^3_1, \mathbf{const} \circ \pi^3_2 \big\rangle\,, \\ \mathbf{pair} \circ \big\langle \mathbf{const} \circ \pi^3_3,\ \mathbf{pair} \circ \big\langle \mathfrak{c}_{\mathbf{const}(0)}, \mathfrak{c}_{\mathbf{const}(1)} \big\rangle \big\rangle \big\rangle \big\rangle \end{aligned}$$

Le résultat est alors presque immédiat :

$$\begin{aligned} \chi_{\mathsf{K}^{\Phi}_{\mathsf{c}}}(g(x,y,z)) &= \begin{cases} 1 & \text{si } \Phi^{\mathsf{c}}_{g(x,y,z)}(g(x,y,z)) = 0 \\ 0 & \text{sinon.} \end{cases} \\ &= \begin{cases} 1 & \text{si } \Phi^{\mathsf{c}}_x(y) = z \\ 0 & \text{sinon.} \end{cases} \\ &= \chi_{\mathrm{graph}\,\mathsf{U}^{\Phi}_{\mathsf{c}}}(x,y,z) \end{aligned}$$

La construction permettant d'exprimer $\chi_{\mathrm{graph}\,\mathsf{U}^{\Phi}_{\mathsf{c}}}$ en fonction de membre_x étant très similaire, nous l'omettrons ici.

Ces deux réductions terminent la preuve de $\chi_{\mathrm{graph}\,\mathsf{U}^{\Phi}_{\mathsf{c}}} \equiv^{\Omega_{\mathsf{c}}}_m \chi_{\mathsf{K}^{\Phi}_{\mathsf{c}}} \equiv^{\Omega_{\mathsf{c}}}_m \mathrm{member}_y$.

□

Fonctions, fonctions bornées et réduction *btt*

Lorsque nous travaillons avec des classes de fonctions ne possédant pas de schéma de récursion non-borné, le fossé se creuse entre les fonctions bornées et les fonctions non-bornées. La notion de fonction bornée que nous allons utiliser est liée à la notion d'objet fini. Cependant, celle-ci change quand le domaine dépasse ω. Aussi, les résultats de ce paragraphe devront être prouvés à nouveau dans le cas où le domaine est plus grande que ω.

Définition 1.3.12 **Fonctions bornées.** *Une fonction est bornée si son image est de cardinalité finie.*

En particulier, dans une classe énumérable, la quantité d'information qu'il est possible de récupérer via un oracle borné est elle-même bornée, de telle sorte que dans ce cas la réduction de clôture de fonctions coïncide avec la relation de tables de vérité bornées :

Proposition 1.3.13

Si $f \leqslant^{\mathsf{C}} h$ *et* h *est bornée, alors* $f \leqslant^{\mathsf{C}}_{\mathrm{btt}} h$.

De cette propriété, nous pouvons déduire l'existence d'une fonction bornée et dans C qui réussit à éviter h en tous points :

Proposition 1.3.14

Si $f \leqslant^{\mathsf{C}} h$ *et* h *est bornée, alors il existe* $p \in \mathsf{C}$ *bornée telle que* $\forall d \in \mathcal{D}$, $p(d) \neq f(d)$.

Démonstration. Soit f et h tels que $f \leqslant^{\mathsf{C}} h$ et h soit bornée. Nous savons déjà d'après la proposition 1.3.13 que $f \leqslant^{\mathsf{C}}_{\mathrm{btt}} h$.

Soit g et e des témoins de cette réduction, tel que :

$$f = e \circ \left\langle \pi^1_1, \left\langle h \circ \pi^k_1, \ldots, h \circ \pi^k_k \right\rangle \circ g \right\rangle$$

.

Comme h est bornée, e ne peut prendre à d fixé qu'un nombre fini et déterminé d'arguments différents.

Posons M l'ensemble fini des valeurs prises par h et notons m sa cardinalité. Alors, pour tout $d \in \mathcal{D}$,

$$f(d) \in \{e(d, d_1, \ldots, d_k) : \forall i \in \{1, \ldots, k\}, d_i \in M\}$$

Rappelons-nous que e est une fonction membre de C. Il est donc possible pour p de calculer les m^k valeurs possibles de e sur l'entrée x. k et m étant fixés à l'avance, cette opération peut être faite en se restreignant à n'utiliser que des opérateurs conditionnels.

Comme p cherche une valeur qui n'est pas présente dans cet ensemble, il lui suffit de comparer les éléments avec un autre ensemble de $m^k + 1$ éléments distincts, comme $\left\{0, \ldots, m^k + 1\right\}$ et de choisir le plus petit absent. Le code de cette fonction est extrêmement peu agréable à lire car la combinatoire s'y retrouve *in extenso*. $\square$

Nous pouvons désormais montrer qu'aucune fonction bornée ne peut permettre de calculer la fonction universelle de la classe.

Proposition 1.3.15

Soit h *une fonction bornée, alors* $\mathsf{U}_{\mathsf{C}} \not\leqslant^{\mathsf{C}} h$.

Démonstration. Par contradiction, nous allons montrer que si ça n'était pas le cas, il serait possible de choisir h tel qu'il intersecte toutes les fonctions de C, entrant ainsi en conflit avec le résultat de la proposition 1.3.14.

Supposons que $\mathsf{U}_{\mathsf{C}} \leqslant^{\mathsf{C}} h$.

Considérons la fonction $x \mapsto \Phi^{\mathsf{C}}_x(x)$. Celle-ci vérifie elle aussi $x \mapsto \Phi^{\mathsf{C}}_x(x) \leqslant^{\mathsf{C}} h$ puisque $x \mapsto \Phi^{\mathsf{C}}_x(x) = \mathsf{U}_{\mathsf{C}} \circ \langle \pi^1_1, \pi^1_1 \rangle$.

Or, cette fonction intersecte au moins en un point toutes les fonctions de C, étant donné que $\forall d \in \mathcal{D}, \Phi^{\mathsf{C}}_d(d) = f(d)$. La contradiction recherchée étant démontrée, il était faux de dire que $\mathsf{U}_{\mathsf{C}} \leqslant^{\mathsf{C}} h$. D'où le résultat.

□

En particulier, nous remarquons que ni la fonction indicatrice du graphe de la fonction universelle, ni la fonction indicatrice de l'ensemble diagonal ne permettent de calculer la fonction universelle :

Corollaire 1.3.16

Les fonctions indicatrices du graphe de la fonction universelle et de l'ensemble diagonal ne permettent pas de construire la fonction universelle.

$$\begin{aligned} \mathsf{U}_{\mathsf{C}} &\not\leqslant^{\mathsf{C}} \chi_{\mathrm{graph}_{\mathsf{U}^{\Phi}_{\mathsf{C}}}} \\ \mathsf{U}_{\mathsf{C}} &\not\leqslant^{\mathsf{C}} \chi_{\mathsf{K}^{\Phi}_{\mathsf{C}}}. \end{aligned}$$

Extensions par fonctions bornées

Un autre corollaire assez pratique de la proposition précédente va nous aider à hiérarchiser les différentes classes énumérables. En effet, nous allons montrer que si deux extensions de classes par une même fonction bornée coïncident, alors aucune d'entre elles ne peut contenir de fonction universelle pour l'autre.

Tout d'abord, commençons par formaliser la notion d'extension, ou de relativisation :

Définition 1.3.17 **Relativisations.** C^f *désigne la relativisation de* C *à* f *:*

$$\mathsf{C}^f = \{g : g \leqslant^{\mathsf{C}} f\}.$$

Nous pouvons maintenant démontrer la proposition annoncée :

Proposition 1.3.18

Soit f une fonction bornée. Si $\mathsf{C}^f_1 = \mathsf{C}^f_2$, alors C^f_2 ne contient aucune fonction universelle pour C_1.

Démonstration. Soit C_1 une classe énumérable et $\mathsf{U}_{\mathsf{C}_1}$ une fonction universelle pour cette classe.

D'après la proposition 1.3.15, étant donné que f est bornée, $\mathsf{U}_{\mathsf{C}_1} \not\leq^{\mathsf{C}_1} f$.

Donc, $\mathsf{U}_{\mathsf{C}_1} \notin \mathsf{C}_1^f = \mathsf{C}_2^f$.

Et donc finalement, $\mathsf{U}_{\mathsf{C}_1} \notin \mathsf{C}_2$.

□

1.3.5 Énumérations acceptables

Les classes énumérables récursives permettent déjà la généralisation de la notion de *système d'indices acceptable* définie par Rogers dans [RJ58]. Celle-ci nous permettra au chapitre 2.2.2 de démontrer une version de l'isomorphisme de Rogers associées à nos classes de fonctions.

Définition 1.3.19 **C-système d'indices acceptable.** *Un* (C-)système d'indices acceptable *est une fonction $\Psi^{\mathsf{C}}_{\cdot}$ envoyant le domaine $\mathcal{D}$ sur la classe énumérable C tel qu'il existe des fonctions $f, g \in \mathsf{C}$ telles que $\forall e, \Phi^{\mathsf{C}}_e \cong \Psi^{\mathsf{C}}_{f(e)}$ et $\Psi^{\mathsf{C}}_e \cong \Phi^{\mathsf{C}}_{g(e)}$ (c'est à dire que $\Phi^{\mathsf{C}}_{\cdot}$ et $\Psi^{\mathsf{C}}_{\cdot}$ sont C-m-équivalentes) et munie d'une fonction $\mathfrak{p}_{\psi}$ de bourrage pour $\Psi^{\mathsf{C}}_{\cdot}$.*

Cette notion désigne donc une classe d'équivalence contenant l'énumération canonique.

Un C-système d'indices acceptable hérite de nombreuses propriétés de l'énumération canonique de la classe qui lui est associée. Le plus immédiat des héritages est celui concernant le théorème s^m_n.

Proposition 1.3.20

Héritage du théorème s^m_n. *Soit $m, n \in \omega$, $e \in \mathcal{D}$ et $\Phi^{\mathsf{C}}_{\cdot}$ une énumération de C. Alors, il existe une fonction $s'^m_n \in \mathsf{C}$ vérifiant*

$$\Psi^{\mathsf{C}}_e(x_1, \ldots, x_n, y_1, \ldots, y_m) = \Psi^{\mathsf{C}}_{s'^m_n(e, x_1, \ldots, x_n)}(y_1, \ldots, y_m).$$

Démonstration. La preuve découle directement de la stabilité de C par composition. Nous transformons a, indice dans $\Psi^{\mathsf{C}}_{\cdot}$, en un indice $f(a)$ pour l'énumération $\Phi^{\mathsf{C}}_{\cdot}$, nous appliquons le théorème s^m_n existant pour cette énumération, puis nous transformons le résultat en un indice pour $\Psi^{\mathsf{C}}_{\cdot}$ en utilisant la fonction g.

$$s'^m_n : a, x_1, \ldots, x_n \mapsto g(s^m_n(f(a), x_1, \ldots, x_n))$$

□

Montrons maintenant que la paramétrisation, c'est à dire le branchement conditionnel paramétré par une fonction est également héritée dans la nouvelle énumération. En effet, la paramétrisation reste présente dans tout système d'indices acceptable du moment que l'énumération canonique dispose d'une fonction de branchement conditionnel :

Proposition 1.3.21

Propriété de paramétrisation. *Soit* $\mathtt{C}$ *une classe énumérable et* $\Psi^{\mathtt{C}}_{\cdot}$ *un* $\mathtt{C}$-*système d'indices acceptable. Si la paramétrisation est vérifiée par l'énumération canonique, elle l'est par* $\Psi^{\mathtt{C}}_{\cdot}$ *: il existe un indice pour* $\Psi^{\mathtt{C}}_{\cdot}$, param_ψ *tel que pour tout* $i_0, i_1, j \in \mathcal{D}$,

$$\Psi^{\mathtt{C}}_{\Psi^{\mathtt{C}}_{\mathrm{param}_\psi}\langle i_0, i_1, \mathbf{j}\rangle(e)} \cong \begin{cases} \psi^{\mathtt{C}}_{i_0} & \textit{si } \Psi^{\mathtt{C}}_{\mathbf{j}}(e) = 0, \\ \psi^{\mathtt{C}}_{i_1} & \textit{sinon.} \end{cases}$$

Démonstration. Nous voulons en fait montrer qu'il existe une fonction $\mathbf{param}_\psi$ telle que :

$$\Psi^{\mathtt{C}}_{\mathbf{param}_\psi(\langle i_0, i_1, j\rangle)} : e \mapsto \begin{cases} i'_0 & \text{si } \Psi^{\mathtt{C}}_j(e) = 0, \\ i'_1 & \text{sinon.} \end{cases}$$

et telle que $\Psi^{\mathtt{C}}_{i'_0} \cong \Psi^{\mathtt{C}}_{i_0}$ et $\Psi^{\mathtt{C}}_{i'_1} \cong \Psi^{\mathtt{C}}_{i_1}$.

Il suffit pour ce faire de poser

$$\mathbf{param}_\psi = \mathbf{comp} \circ \left\langle \mathfrak{c}_{\mathbf{cond}}, \left\langle \mathbf{comp} \circ \left\langle \mathfrak{c} \circ \pi^3_3, \mathfrak{c}_{\pi^1_1} \right\rangle \right\rangle, \mathfrak{c}_0, \mathfrak{c} \circ \pi^3_1, \mathfrak{c} \circ \pi^3_2 \right\rangle$$

Nous vérifions qu'alors :

$$\Psi^{\mathtt{C}}_{\mathbf{param}_\psi(i_0, i_1, j)} \cong \Psi^{\mathtt{C}}_{\mathbf{comp} \circ \left\langle \mathfrak{c}_{\mathbf{cond}}, \left\langle \mathbf{comp} \circ \left\langle \mathfrak{c}_j, \mathfrak{c}_{\pi^1_1} \right\rangle \right\rangle, \mathfrak{c}_0, \mathfrak{c}_{i_0}, \mathfrak{c}_{i_1} \right\rangle}$$

Et donc, pour tout $e \in \mathcal{D}$,

$$\Psi^{\mathtt{C}}_{\Psi^{\mathtt{C}}_{\mathbf{param}_\psi(i_0, i_1, j)}(e)} \cong \Psi^{\mathtt{C}}_{\mathbf{cond} \circ \left\langle \Psi^{\mathtt{C}}_j \circ \pi^1_1(e), \mathfrak{c}_0, \mathfrak{c}_{i_0}, \mathfrak{c}_{i_1} \right\rangle} \cong \begin{cases} \Psi^{\mathtt{C}}_{i_0} & \text{si } \Psi^{\mathtt{C}}_j(e) = 0, \\ \Psi^{\mathtt{C}}_{i_1} & \text{sinon.} \end{cases}$$

□

1.3.6 Théorèmes classiques et classes sans universelle

Les classes énumérables ne possèdent jamais de fonction universelle. En effet, si elles en possédaient une, elles pourraient réaliser le point fixe en plus de la fonction s^m_n. Autrement dit, elles incluraient toutes les fonctions calculables.

Théorème 1.3.22

Absence de fonction universelle. *Soit* C *une classe énumérable contenant la fonction successeur et* U_C *une fonction universelle pour* C*, alors* $\mathsf{U}_\mathsf{C} \notin \mathsf{C}$.

Démonstration. Nous allons procéder très classiquement par diagonalisation. L'argument important ici est le fait que toutes les fonctions de l'énumération sont totales.

Supposons que U_C appartienne à C. Par composition, nous savons que la fonction $x \mapsto \mathbf{s}(\mathsf{U}_\mathsf{C}(x,x))$ est également dans C. Notons $d \in \mathcal{D}$ un de ses indices.

Nous remarquons alors l'égalité suivante : $\Phi^\mathsf{C}_d(d) = \mathbf{s}(\mathsf{U}_\mathsf{C}(d,d)) = \mathbf{s}(\Phi^\mathsf{C}_d(d))$, qui est effectivement une contradiction puisque Φ^C_d est totale et ne peut donc pas ne pas être définie en d.

Donc U_C n'appartient pas à C.

□

Le théorème du point fixe de Kleene est très lié à l'existence de fonctions partielles et d'une fonction universelle. En fait, il n'y a en général pas de second théorème de Kleene dans les classes de fonctions totales disposant d'une énumération.

Théorème 1.3.23

Absence du second théorème de la récursion de Kleene. *Soit* C *une classe énumérable contenant la fonction successeur.* $\exists f \in \mathsf{C}, \forall x \in \mathcal{D}, \exists y, \Phi^\mathsf{C}_x(y) \neq \Phi^\mathsf{C}_{f(x)}(y)$.

Démonstration. Soit s un indice de la fonction successeur $\mathbf{s}$.

Considérons la fonction suivante :

$$f : n \mapsto \mathbf{comp}(s,n).$$

Alors,

$$\forall d, \Phi^\mathsf{C}_{f(n)}(d) = \Phi^\mathsf{C}_{\mathbf{comp}(s,n)}(d) = \Phi^\mathsf{C}_s(\Phi^\mathsf{C}_n(d)).$$

Étant donné que nos fonctions sont totales,

$$\forall n, \forall d, \Phi^\mathsf{C}_n(d) \neq \Phi^\mathsf{C}_{f(n)}(d).$$

□

En revanche, plusieurs variantes plus faibles du théorème de la récursion sont vérifiées. Les premières sont décrites ci-dessous, les suivantes nécessites des notions qui seront développées dans le chapitre suivant.

Théorème 1.3.24

Théorème de la récursion de Kleene sans point fixe. *Soit* $x, y \in \mathcal{D}$. *Il existe* $f \in \mathtt{C}$ *tel que* $\Phi^{\mathtt{C}}_{f(x)}(y) = \Phi^{\mathtt{C}}_{x}(f(x), y)$.

Démonstration. En utilisant les fonctions de manipulation d'indice présentes par définition dans les classes énumérables, il est possible de construire une telle fonction f. Pour ce faire, on utilise la fonction g suivante :

$$g = \mathbf{comp} \circ \left\langle \pi^1_1, \mathbf{pair} \circ \left\langle \mathbf{comp} \circ \left\langle \mathfrak{c}_{s^1_1}, \mathbf{pair} \circ \langle \mathfrak{c}_{\pi_1}, \mathfrak{c}_{\pi_1} \rangle \right\rangle, \mathfrak{c}_{\pi_2} \right\rangle \right\rangle.$$

où pour une fonction h donnée, $\mathfrak{c}_h$ désigne la fonction constante valant l'indice de celle-ci. La fonction g est telle que :

$$\begin{aligned} \Phi^{\mathtt{C}}_{g(x)}(y, z) &= \Phi^{\mathtt{C}}_{x}(s^1_1(\pi_1(y, z), \pi_1(y, z)), \pi_2(y, z)) \\ &= \Phi^{\mathtt{C}}_{x}(s^1_1(y, y), z). \end{aligned}$$

On a donc, pour $f = s^1_1 \circ \langle g, g \rangle$:

$$\begin{aligned} \Phi^{\mathtt{C}}_{f(x)}(y) &= \Phi^{\mathtt{C}}_{s^1_1(g(x), g(x))}(y) \\ &= \Phi^{\mathtt{C}}_{g(x)}(g(x), y) \\ &= \Phi^{\mathtt{C}}_{x}(s^1_1(g(x), g(x)), y) \\ &= \Phi^{\mathtt{C}}_{x}(f(x), y). \end{aligned}$$

□

2 Classes fondamentales

Calculabilités sur les fonctions totales

Sommaire

Nous venons de voir qu'un grand nombre de propriétés classiques sont présentes dans des fragments très restreints de la calculabilité. Cependant, comme nous l'avons vu, afin de pouvoir simuler les fonctions récursives sur un nombre quelconque d'étapes, il est nécessaire de s'intéresser à des fragments un peu plus importants. Le théorème de la forme normale de Kleene nous indique, nous l'avons vu au paragraphe 1.1.2, qu'il existe un prédicat récursif primitif T et une fonction récursive primitive U tel que pour toute fonction calculable f il existe un e tel que $f(x) \cong U(\mu_s T(e, x, s))$.

Nous allons donc choisir comme fragment de base la classe des fonctions récursives primitives. Cette classe possède l'avantage d'avoir été très étudiée en calculabilité pour ses comportements simples et intuitifs bien que permettant d'écrire la majeure partie [1] des programmes rencontrés *naturellement* en programmation.

Un autre grand avantage que nous constaterons par la suite est que son pendant en logique mathématique, l'*Arithmétique Primitive Récursive*, est un outil clé pour étudier la puissance et la consistance des théories mathématiques. Par exemple, la preuve de consistance de **PA** par Gentzen ([Gen36]) a été réalisée dans ce système augmenté de l'induction transfinie sur ϵ_0. L'idée est que le schéma de récursion primitive permet de faire de la récursion sur tous les ordinaux présents dans le domaine $\mathcal{D}$. Ainsi, dans le cas classique, la récursion primitive permet de descendre sur l'ordre des entiers, mais dans un cas où $\mathcal{D}$ contiendrait un ordinal infini α, il serait alors possible de faire de l'α-récursion.

Dans ce chapitre et le suivant, nous allons nous concentrer sur l'étude des classes c énumérables récursives avec fonctions de coût. Nous demanderons à ces classes d'être définies sur un domaine $\mathcal{D} = \omega$. Cette condition impliquera l'existence de fonctions sim et use telles que définies respectivement dans la Définition 1.1.8 et la Définition 1.1.6. Ces fonctions nous permettrons de nous assurer de la présence des fonctions du Castor Affairé vues dans la Définition 1.3.7, qui possèdent des propriétés allant nous intéresser, notamment le fait qu'elles permettent de calculer la fonction universelle.

Ces conditions peuvent sembler très restrictives, mais elles nous ramènent en fait dans des modèles un peu plus *standards* de calculabilité. En particulier, toutes les classes α-récursives satisfont ces conditions.

2.1 Définitions

Nous avons déjà mis l'accent, lors de la définition des différentes notions de classes vues jusqu'à présent, sur les fonctions de manipulation d'indices relatives à une classe c. La question de la présence de ces fonctions à l'intérieur même de la classe ou non est d'une importance cruciale pour les preuves des théorèmes classiques, afin de savoir où se trouvent les fonctions que nous y construisons (par exemple, pour voir si la fonction s du théorème s_n^m est bien membre de c). C'est pour cette raison que nous avons considéré la classe Ω_{c} précédemment.

Dans la suite, nous souhaitons que toutes les transformations d'indices se fassent par les fonctions de la classe et demeurent à l'intérieur de la classe. En fait, nous demandons même à ce que ces fonctions de transformation soient dans p, ce qui est le cas pour la plupart des énumérations reposant sur un codage de Gödel.

C'est en particulier le cas pour les fonctions récursives primitives. Une autre

1. Certains diront *la totalité des programmes*, en considérant que l'univers (physique) ne permet que de stocker une quantité finie, bien que grande, d'information, alors cette assertion est évidente, toutes les fonctions devenant élémentaires en le plus grand nombre entier descriptible dans celui-ci...Sinon, il est toujours possible de se demander à quel point un programme nécessitant plus de ressources que la mémoire stockable dans l'univers est intuitif.

propriété de p que nous exigeons est celle du bourrage (*padding*), c'est à dire de l'existence calculable (dans p) d'une infinité d'indices pour une fonction d'indice donné relatif à son énumération canonique.

Pour rappeler toutes ces propriétés, nous introduisons une notion de *Schéma de codage récursif primitif* :

Définition 2.1.1 **Schéma de codage récursif primitif.** *Une classe énumérable avec fonctions de coût* c *possède un* schéma de codage récursif primitif *sur* $\mathcal{D} = \omega$ *permettant aux fonctions de manipuler leur indice si ses fonctions de manipulation d'indices sont récursives primitives :* $\Omega_{\mathsf{c}} \subset \mathsf{p}$*, et si les fonctions de simulation (voir Définition 1.1.6) et de calcul d'indice de fonction sont de coût récursif primitif. En particulier, elle satisfait les propriétés suivantes :*

— *il existe une fonction de* bourrage $\mathfrak{p} \in \mathsf{p}$ *telle que*

$$\forall x, c, (\mathfrak{p}(x,c) > x) \wedge (\forall d < c, \mathfrak{p}(x,c) > \mathfrak{p}(x,d)) \wedge (\forall c, d, \Phi^{\mathsf{c}}_x(d) = \Phi^{\mathsf{c}}_{\mathfrak{p}(x,c)}(d))$$

(on utilisera la notation $\mathfrak{p}(x)$ *pour désigner* $\mathfrak{p}(x,0)$*) ;*

— *il existe une fonction de* composition $\mathfrak{C} \in \mathsf{p}$ *telle que*

$$\forall c, d, e \in \mathcal{D}, \Phi^{\mathsf{c}}_e(\Phi^{\mathsf{c}}_d(c)) = \Phi^{\mathsf{c}}_{\mathfrak{C}(e,d)}(c)\ ;$$

— *il existe une fonction* sim $\in \mathsf{p}$ *permettant de simuler pas à pas un indice de fonction de* c *pendant un nombre d'étapes donné*

$$\forall x, e \in \mathcal{D}, \exists n \in \mathcal{D}, \Phi^{\mathsf{c}}_e(x) = \mathrm{sim}(e, x, n)\ ;$$

— *il existe une fonction* use $\in \mathsf{p}$ *permettant de transformer un indice de fonction de* c *en l'indice de sa fonction de* use *telle que*

$$\forall x, e, \Phi^{\mathsf{c}}_e(x) = \mathrm{sim}(e, x, \Phi^{\mathsf{c}}_{\mathrm{use}(e)}(x)).$$

En cas d'ambiguïté, ces indices seront nommés c*-indices pour les distinguer des indices de calculabilité classique.*

Nous disposons maintenant de tous les éléments pour définir une nouvelle notion de classes fondamentales qui, comme leur nom l'indique, serviront de *fondement* voire de *fondations* pour les sous-calculabilités à suivre :

Définition 2.1.2 **Classes Fondamentales.** *Une classe fondamentale est une classe énumérable* c *de fonctions totales :*

— *comprenant au moins la classe des fonctions primitives récursives* ($\mathsf{c} \supset \mathsf{p}$)*;*
— *comportant un schéma de codage récursif primitif incluant une fonction de bourrage, une fonction de simulation et une fonction de coût ;*
— *close par récursion primitive et par composition.*

L'énumération des fonctions totales de $\mathcal{D}$ vers $\mathcal{D}$ de $\mathtt{C}$ *est notée* $\Phi^{\mathtt{C}}_{\cdot}$ *et est indicée sur* $\mathcal{D}$.

Les éléments de $\mathtt{C}$ *sont appelés* fonctions $\mathtt{C}$-fondamentales.

Étant donné une classe fondamentale $\mathtt{C}$ *et une fonction* f *sur* $\mathcal{D}$*, on note* $\mathtt{C}[f]$ *la plus petite classe fondamentale contenant les fonctions de* $\mathtt{C}$ *et la fonction* f.

La récursion primitive nous permet de définir des fonctions de manipulation d'indice de manière bien plus uniforme. Dans les classes énumérables, la fonction s^m_n ne pouvait pas être définie uniformément en m et en n ça n'est plus le cas dans les classes fondamentales qui disposent elles d'une unique fonction s permettant de calculer tous les s^m_n pour m et n entiers.

2.2 Structure

Les classes fondamentales héritent bien entendu de la structure des classes énumérables. Cependant, la présence des fonctions récursives primitives nous permet de prouver l'existence d'une structure bien plus riche et bien plus proche de la calculabilité classique.

Le point le plus intéressant est sans doute la possibilité de prouver un théorème d'isomorphisme *à la* Rogers. Le théorème d'isomorphisme de Rogers en calculabilité classique est un jalon important qui nous indique que le choix de l'énumération ne conditionne pas les capacités de calcul ni la structure combinatoire de notre classe de fonctions récursives. Retrouver ce théorème dans les classes fondamentales leur donne une légitimité propre. En effet, c'est une assurance que les résultats obtenus ne sont pas le résultat du choix d'une énumération *ad-hoc*. L'énumération n'est pas juste un cas particulier, un point de vue que nous souhaitons avoir sur les classes.

Avant d'en arriver à prouver notre théorème d'isomorphisme *à la* Rogers, nous devons prouver le pendant dans les classes fondamentales du théorème de Myhill. Ce dernier établi l'équivalence entre la 1-équivalence et la m-équivalence. Nous allons donc dans le paragraphe suivant introduire ces notions puis faire la preuve de notre théorème *à la* Myhill. Pour lire les preuves des versions classiques de ces théorèmes, le lecteur pourra se référer à [Rog87].

Enfin, nous discuterons du second théorème de la récursion de Kleene, de l'impossibilité qu'il y a à l'établir dans les classes fondamentales, des raisons de ce fait, et de quelques alternatives possibles.

2.2.1 Théorème d'isomorphisme *à la* Myhill

Le théorème d'isomorphisme de Myhill [Myh55] est le pendant en calculabilité du théorème d'isomorphisme de théorie des ensembles connu sous le nom de *Théorème Cantor-Shröder-Bernstein.*

C'est une brique élémentaire importante puisqu'il va nous mener plus tard vers le théorème d'isomorphisme de Rogers sur les énumérations acceptables.

Le théorème de Myhill utilise la notion de 1-réduction entre ensembles, cas particulier de la réduction *many-one*. Nous les introduisons ici dans leurs versions relativisées aux classes fondamentales, pour le cas des classes récursives et des classes de fonctions disposant d'un schéma de recherche non-borné sur tous les éléments de leur domaine.

Définition 2.2.1 **C-réductions entre ensembles d'entiers.** *Soit A et B deux ensembles d'entiers.*

- *A se m-réduit à B, noté $A \preccurlyeq_m^{\mathsf{C}} B$ s'il existe $f \in \mathsf{C}$ tel que l'ensemble des antécédents par f des éléments de B soit A.*
- *A se 1-réduit à B, noté $A \preccurlyeq_1^{\mathsf{C}} B$ si A se m-réduit à B via une fonction f injective.*
- *A est (C-m-)équivalent à B, noté $A \equiv_m^{\mathsf{C}} B$ si $A \preccurlyeq_m^{\mathsf{C}} B$ et $B \preccurlyeq_m^{\mathsf{C}} A$.*
- *A est (C-1-)équivalent à B, noté $A \equiv_1^{\mathsf{C}} B$ si $A \preccurlyeq_1^{\mathsf{C}} B$ et $B \preccurlyeq_1^{\mathsf{C}} A$.*

La question qui se pose derrière le théorème de Myhill est de savoir s'il est suffisant d'avoir une fonction injective dans chaque direction pour pouvoir déduire l'existence d'une fonction bijective, c'est à dire une permutation entre les ensembles, calculable. Les deux ensembles sont alors dit isomorphes. Dans notre cas, nous demandons à ce que la permutation appartienne à notre classe fondamentale.

Définition 2.2.2 **C-isomorphismes.** *Soit A et B deux ensembles d'entiers.*

A et B sont (C-)isomorphes s'il existe une permutation $p \in \mathsf{C}$ telle que $p^{-1} \in \mathsf{C}$ et que l'image de A par p soit B. On note alors $A \equiv^{\mathsf{C}} B$.

Bien sûr, les classes fondamentales n'étant en général pas stables par inverse il nous faut demander à ce que la permutation et son inverse soient tous les deux dans C. Sans cette restriction, la relation d'équivalence ne serait pas symétrique et aurait une signification bien différente du point de vue des classes fondamentales.

La preuve classique du théorème de Myhill utilise le schéma de récursion non borné afin de construire la bijection entre A et B. On peut se ramener au cas d'une itération bornée par un élément du domaine. Le problème survient quand le domaine contient un élément infini. Dans les cas des récursions d'ordre supérieur, où ce genre d'itération est possible, cela ne présente pas de difficulté. En revanche, dans les cas où l'on ne dispose pas de cette opération et où le domaine possède des éléments infinis, nous ne savons pas encore répondre à cette question.

Note 2.2.3 Domaine de validité des résultats suivants. *La preuve suivante établit le résultat dans le cas des classes fondamentales, où $\mathcal{D} = \omega$. On entrevoit la possibilité de l'adapter aux cas où le domaine est plus vaste à condition de remplacer les fonctions récursives primitives et les encodages dans ω par des fonctions Δ_0 sur $\mathcal{D}$ et des encodages équivalents. Cela-dit, cette preuve ne fonctionne pas dans le* *cas général pour une classe énumérable quelconque. Le théorème d'isomorphisme de Rogers présenté dans la partie suivante a évidemment le même domaine de validité.*

Théorème 2.2.4

Myhill pour les classes fondamentales. *Soit* $\mathtt{C}$ *une classe fondamentale et* A *et* B *deux sous-ensembles de* ω.

$$A \equiv_1^{\mathtt{C}} B \iff A \equiv^{\mathtt{C}} B.$$

Démonstration. La preuve suivante adapte la preuve classique en construisant la bijection de manière récursive primitive.

L'implication de la droite vers la gauche est évidente. En effet, s'il existe $p \in \mathtt{C}$ telle que $p^{-1} \in \mathtt{C}$, $p``A = B$ et $p``B = A$ alors p et p^{-1} sont des témoins de $B \preccurlyeq_1^{\mathtt{C}} A$ et $A \preccurlyeq_1^{\mathtt{C}} B$. A et B sont donc bien équivalentes.

Pour l'implication de gauche à droite, nous adaptons la preuve habituelle du théorème de Myhill en calculabilité classique.

Soit f et g des témoins respectifs de $A \preccurlyeq_1^{\mathtt{C}} B$ et $B \preccurlyeq_1^{\mathtt{C}} A$.

f et g appartiennent à $\mathtt{C}$, et sont injectives respectivement sur A et B.

Nous allons construire une fonction graph_p qui énumère le graphe de p (et, par la même occasion, de p^{-1}).

Nous voulons que pour tout $s \in \mathcal{D}$, $\mathrm{graph}\, p(s+1)$ code un graphe fixant $p(s)$ et $p^{-1}(s)$.

Soit q_1 une fonction prenant comme arguments $\langle s, l\rangle$. l sera le codage d'une liste d'au moins $2s$ couples. Elle représente un sous-ensemble fini du graphe de p qui fixe les images et pré-images par p des éléments inférieurs à s.

Notons p_l l'approximation de p engendrée par l.

$q_1(\langle s, l\rangle)$ commence par vérifier si $p_l(s)$ est définie. Si c'est le cas, elle renvoie l.

Dans le cas contraire, f cherche le plus petit $n \leqslant 2s+1$ tel que $x = f \circ (p^{-1} \circ f)^{(n)}(s)$ n'appartienne pas à l'image de p_l.

Étant donné que $p(s)$ n'est pas encore défini, et sachant que f et p^{-1} sont injectives, nous savons qu'il n'y a pas de cycle dans $p^{-1} \circ f$. Le graphe encodé par l étant de cardinalité finie (bornée par $2s$), il n'y a pas de chemin de longueur $2s+1$ ou plus. Ainsi, nous savons que nous allons nous échapper de l'image de p_l pour un n inférieur à cette borne.

Une fois x trouvé, q_1 renvoie le nouveau graphe, l', qui étend p_l de sorte que $p_{l'}(s) = x$.

De la même manière, nous définissons $q_{-1}(s, l)$, qui prend en argument une approximation l du graphe de p et prolonge p_l^{-1} en s en utilisant cette fois l'autre injection, g.

Ces deux fonctions q_1 et q_{-1} définies, nous posons $q(s,l) = q_{-1}(s, q_1(s,l))$. q prolonge à la fois p_l et p_l^{-1} en $s+1$.

Il reste à définir graph_p qui, étant donné un entier s, renvoie l'approximation finie d'au plus $2s$ éléments du graphe de p, définissant p et p^{-1} sur tous les éléments inférieurs à s :

$$\mathrm{graph}_p(s) = \begin{cases} \langle\rangle & \text{si } s = 0, \\ q(s-1, \mathrm{graph}_p(s-1)) & \text{sinon.} \end{cases}$$

(où $\langle\rangle$ est un code pour la *liste vide*)

Ainsi, graph_p est bien définie et ce en utilisant simplement f, g et des opérations récursives primitives.

Il est ensuite aisé de construire p et p^{-1} à partir de graph_p, toujours de manière récursive primitive, puisqu'il s'agit simplement d'effectuer une recherche bornée dans une liste finie.

Ainsi, p et p^{-1} appartiennent bien à $\mathtt{C}$, ce qui termine la preuve du théorème. $\square$

2.2.2 Isomorphisme *à la* Rogers

Nous sommes maintenant fin prêts pour démontrer le théorème d'isomorphisme. Rappelons que ce résultat est essentiel si nous souhaitons plus tard prouver l'équivalence entre le modèle de calcul des classes fondamentales comme définies ici, et un autre modèle de calcul similaire, par exemple une restriction des machines de Turing ou encore du λ-calcul aux fonctions $\mathtt{C}$-fondamentales.

Théorème 2.2.5

Théorème d'isomorphisme de Rogers. *Soit $\mathtt{C}$ une classe fondamentale. Pour toute paire de $\mathtt{C}$-systèmes d'indices acceptables $\Phi^{\mathtt{C}}_{\cdot}$ et $\Psi^{\mathtt{C}}_{\cdot}$ il existe une permutation $p \in \mathtt{C}$ de $\mathcal{D}$, d'inverse $p^{-1} \in \mathtt{C}$ telle que pour tout $e \in \mathcal{D}$,*

$$\Phi^{\mathtt{C}}_e \cong \Psi^{\mathtt{C}}_{p(e)}.$$

Démonstration. Comme nous l'avons vu dans la définition, un système d'indices acceptable $\Psi^{\mathtt{C}}_{(\cdot)}$ vient avec une paire de fonctions f et g telle que

$$\forall e, \Phi^{\mathtt{C}}_e \;\cong\; \Psi^{\mathtt{C}}_{f(e)} \text{ et } \Psi^{\mathtt{C}}_e \;\cong\; \Phi^{\mathtt{C}}_{g(e)}.$$

Notre objectif est d'appliquer le théorème 2.2.4 aux deux fonctions f et g afin d'obtenir une permutation calculable p. Cependant, le théorème 2.2.4 requiert que f et g soit injectives.

Par définition des classes fondamentales, il nous est possible d'utiliser la fonction de bourrage sur l'énumération $\Psi^{\mathtt{C}}_{\cdot}$ des classes.

Nous allons donc transformer f de sorte qu'en utilisant la fonction de bourrage $\mathfrak{p}_\psi$ elle ne génère jamais un indice ayant déjà été produit pour une valeur d'entrée inférieure.

Pour se faire, créons une fonction f' qui, pour une valeur d'entrée $d \in \mathcal{D}$, calcule l'ensemble suivant :

$$\left\{\mathfrak{p}(f(d), n) : n \leqslant d\right\} \setminus \{f'(e) : e < d\}\,.$$

Une fois l'ensemble construit, il suffit à notre fonction f' d'en renvoyer le minimum. La fonction f' ainsi construite se comporte de la même manière que la fonction f mais est injective.

Il reste encore à rendre la fonction g injective. C'est déjà le cas pour les fonctions à support fini, par définition. Pour les autres, c'est cette fois-ci la fonction $\mathfrak{p}_\phi$ qui nous permet de faire du bourrage sur les indices de $\Phi^{\mathsf{c}}_{\cdot}$. Par la même construction que celle utilisée pour la fonction f', nous pouvons donc construire une fonction g' qui se comporte comme g tout en étant injective.

Les conditions sont alors réunies pour nous permettre d'appliquer le théorème 2.2.4 et ainsi obtenir la permutation p souhaitée.

□

2.2.3 Remarques sur l'inversibilité des fonctions

L'inverse d'une fonction n'est pas bien défini en règle générale quand celle-ci n'est pas injective. Une manière de palier à ce problème est de définir l'antécédent d'une valeur comme le plus petit des antécédents, ou encore n'importe quel antécédent. Dans le cas des fonctions récursives par exemple, inverser une fonction totale est facile, bien que le résultat soit une fonction a priori partielle. Par contre, inverser une fonction partielle peut être impossible si nous recherchons l'antécédent minimal.

Cela dit, comme nous l'avons remarqué, même dans le cas où nous nous contentons d'un antécédent quelconque pour l'inverse, si la fonction initiale n'est pas surjective, l'inverse est une fonction partielle, or, dans le contexte des fonction c-fondamentales, toutes les fonctions se doivent d'être totales. Une fois de plus, il est possible de s'en sortir en affectant une valeur par défaut aux points non-définis.

Cependant, là encore, toutes les fonctions ne sont pas inversibles.

Proposition 2.2.6

Instabilité par inverse. *En règle générale, les classes fondamentales ne sont pas stables par inverse.*

Démonstration. Considérons la variante forte de la fonction du Castor Affairé, nous remarquons que :

$$\mathsf{BB}_{\mathsf{C}}^{\Phi^{-1}} = n \mapsto min \left\{ i < n : \begin{array}{cc} & \exists s < n, \mathrm{sim}(i,0,s) + i = n \\ \vee & (\mathrm{sim}(i,0,n-i) > 0 \wedge \mathrm{sim}(i,0,n-i-1) = 0) \end{array} \right\}$$

Par construction, cette fonction est récursive primitive. Dominant $\mathsf{BB}_{\mathsf{C}}^{\Phi}$, elle ne peut pas être C-fondamentale, d'où le résultat. □

2.3 Énumérabilité

Dès lors que nous avons à notre disposition une classe de fonctions, nous pouvons étudier la classe des ensembles que celles-ci peuvent énumérer. Nous allons dans ce paragraphe définir une notion d'énumérabilité non-triviale basée sur les classes C-fondamentales.

2.3.1 Notions équivalentes d'énumérabilité classique

En calculabilité classique, il existe toute une collection de caractérisations d'un ensemble récursivement énumérable. Les caractérisations les plus connues étant les suivantes :

— $E \subset \omega$ est *r.e.* si et seulement s'il existe $e \in \omega$ tel que $E = \mathrm{range}(\varphi_e)$;

— $E \subset \omega$ est *r.e.* si et seulement s'il existe $e \in \omega$ tel que $E = \mathrm{domain}(\varphi_e)$.

Voici deux autres caractérisations qui, si elles sont très simples, et simples à prouver, sont moins communes.

D'abord, il est possible de demander à la fonction énumérante d'être injective sur son domaine tout en restant totale si l'ensemble est infini.

Théorème 2.3.1

Énumération injective des ensembles récursivement énumérables. *Soit E un ensemble récursivement énumérable. Il existe e tel que φ_e soit injective sur son domaine et que* $\mathrm{range}(\varphi_e) = E$. *De plus, si E est infini, φ_e peut être choisie totale.*

Démonstration. Nous voulons simplement que notre fonction trouve un nouvel élément qui n'ait pas encore été trouvé jusqu'à présent.

Nous savons que simuler pendant s étapes une fonction récursive d'indice e sur une entrée n et vérifier que, si elle converge, elle converge en m est une formule Δ_0^0 puisqu'elle est constituée d'opérations bornées.

Effectuer cette recherche pour tous les triplets $\langle s, n, m \rangle$ dont le code est inférieur à un entier donné est donc également une opération bornée.

Il nous est alors possible de considérer la fonction suivante définie dans sa forme normale de Kleene par un prédicat récursif primitif, qui cherche le premier élément énumérable par un triplet $\langle s, n, m\rangle$ supérieur à x :

$$f_e(x) = \pi_1^3 \left(\mu \langle n_1, n_2, n_3 \rangle \geq x, \quad \begin{array}{c} \varphi_{e,n_2}(n_3) = n_1 \\ \wedge \\ \forall \langle m_1, m_2, m_3 \rangle \geq x, \\ \langle m_1, m_2, m_3 \rangle < \langle n_1, n_2, n_3 \rangle \Rightarrow \varphi_{e,m_2}(m_3) \neq n_1 \end{array} \right).$$

La fonction $g : n \mapsto f_e\left((\mathbf{s} \circ f_e)^{(n)}\right)(0)$ énumère bien tous les éléments (puisqu'elle n'omet que les doublons) et est totale dans le cas où E est un ensemble infini (puisqu'alors il reste toujours un nouvel élément à énumérer.)

□

L'autre caractérisation annoncée concerne cette fois les fonctions récursives primitives. Celles-ci, vues comme bornées et limitées à côté des fonctions récursives, sont souvent laissées de côté lors de l'étude de la calculabilité classique. Cependant, il convient de noter qu'à l'exception notoire de l'ensemble vide, tous les ensembles récursivement énumérables sont énumérables par une fonction récursive primitive.

Théorème 2.3.2

Ensembles primitivement récursivement énumérables. *Soit E un ensemble récursivement énumérable non vide. Alors il existe f récursive primitive telle que $E = \mathrm{range}(f)$.*

Démonstration. Soit $e \in \omega$ tel que $\mathrm{range}(\varphi_e) = E$. Soit $a \in E$. On rappelle que la notation $\varphi_{e,s}(x)$ désigne le résultat du calcul de la fonction φ_e sur l'entrée x en s étapes de calcul.

De la même manière que dans la démonstration précédente, il va être question de simuler la fonction de référence φ_e sur une entrée fixée et en un temps borné.

Bien sûr, il n'est pas possible cette fois de rechercher une borne suffisamment grande pour que la fonction φ_e donne un résultat intéressant. Cependant, cette borne sera atteinte *in fine* et donc tous les éléments de l'image de φ_e seront énumérés par notre fonction primitive : $\forall n, \varphi_e(n) \downarrow \Rightarrow \exists s, \varphi_{e,s}(n) = \varphi_e(n)$.

Définissons f de la manière suivante :

$$f : x \mapsto \begin{cases} \varphi_{e,\pi_2^2(x)}(\pi_1^2(x)) & \text{si } \varphi_{e,\pi_2^2(x)}(\pi_1^2(x)) \downarrow, \\ \varphi_e(0) & \text{sinon.} \end{cases}$$

f est bien récursive primitive puisqu'elle effectue une simulation bornée de φ_e sur un argument fixé. De plus $\mathrm{range}(f) = \mathrm{range}(\varphi_e) = E$ puisque comme chaque élément x de E a un antécédent n par φ_e ($\forall x \in E, \exists n, \varphi_e(n) = x$), il existe un nombre d'étapes s tel que $\varphi_{e,s}(n) = x = f(\langle n, s\rangle$. Et réciproquement, si x appartient à l'image de f, c'est, par construction, qu'il appartient à l'image de φ_e.

□

Ces deux caractérisations nous apprennent, que considérer qu'un ensemble E est énumérable s'il est associé à une fonction capable d'énumérer *in fine* tous ses éléments, alors à peu de choses près, la notion est identique dans les fonctions récursives primitives et dans les fonctions récursives. En fait, elle peut être rendue strictement identique à codage près.

Note 2.3.3 Un codage pour énumérer également les ensembles vides avec les fonctions récursives primitives. *Soit f une fonction récursive primitive. Considérons l'ensemble E des éléments énumérés deux fois d'affilée par celle-ci : $E = \{x : \exists n, f(n) = f(n+1) = x\}$.*

Il est facile de modifier la preuve ci-dessus sur ce modèle, en considérant la fonction suivante :

$$\begin{array}{lll} f : 2x & \mapsto & \begin{cases} \varphi_{e,\pi_2^2(x)}(\pi_1^2(x)) & \textit{si } \varphi_{e,\pi_2^2(x)}(\pi_1^2(x)) \downarrow, \\ 0 & \textit{sinon.} \end{cases} \\ f : 2x+1 & \mapsto & \begin{cases} \varphi_{e,\pi_2^2(x)}(\pi_1^2(x)) & \textit{si } \varphi_{e,\pi_2^2(x)}(\pi_1^2(x)) \downarrow, \\ 1 & \textit{sinon.} \end{cases} \end{array}$$

2.3.2 Énumérabilité non triviale dans les classes fondamentales

Un autre point de vue possible, que nous avons adopté, est de considérer qu'un ensemble E est énumérable s'il existe une fonction capable de produire un nouvel élément à chaque appel, jusqu'à l'épuisement éventuel du stock. Autrement dit, nous demandons l'existence d'une fonction qui soit injective sur un segment initial de son domaine, et dont l'image de ce segment initial soit E.

Plus formellement :

Définition 2.3.4 Ensembles $\mathtt{C}$-récursivement énumérables. *Soit $\mathtt{C}$ une classe fondamentale et $E \subset \mathcal{D}$ un ensemble.*

*E est l'*énumération injective initiale *de f si f est injective sur $\mathcal{D}$ et E est l'image de f, ou s'il existe un $m \in \mathcal{D}$ tel que f soit injective sur $\{d \in \mathcal{D} : d < m\}$ mais tel que $f(m) = f(m+1)$ ou $f(m) \in \{f(d) : d \in \mathcal{D} \wedge d < m\}$ et $E = \{f(0), \ldots, f(m-1)\}$.*

$f_{\restriction\{d\in\mathcal{D}:d<m\}}$ est appelée la partie initiale injective de f.

E est $\mathtt{C}$-(récursivement)-énumérable *s'il existe $f \in \mathtt{C}$ tel que E soit l'énumération injective initiale de f.*

Pour tout $e \in \mathcal{D}$, $\mathbf{W}_e^{\mathtt{C}}$ désigne l'énumération injective initiale de $\mathbf{\Phi}_e^{\mathtt{C}}$.

Cette définition englobe bien les ensembles finis dont l'ensemble vide, et ce d'une manière qui ne trivialise pas ces ensembles au sens de Rice dans sa version relativisée aux classes fondamentales que nous verrons en fin de chapitre.

Note 2.3.5 *Une autre manière de justifier l'injectivité est la suivante : nous voulons être capable, avec les moyens de calcul de* c*, de produire un nouvel élément à la demande. Autrement dit, plutôt que de demander à ce qu'il existe* f *dans* c *qui soit injective, nous pourrions nous contenter de demander qu'il existe* f*,*g *deux fonctions de* c*, telle que pour tout* $n \in \mathcal{D}$ *fixé,* $\exists k \leqslant g(n)$ *tel que* $\forall m \leqslant n$*,* $f(n+k) \neq f(m)$*.*

Il est aisé de voir que cette caractérisation est équivalente à celle choisie dans la définition ci-dessus.

Bien qu'elle ressemble fortement à un mélange des deux caractérisations précédentes, cette nouvelle notion d'énumérabilité n'est pas équivalente à l'énumérabilité classique.

Un exemple tout simple nous en convaincra : le graphe de la fonction d'Ackermann n'est pas p-énumérable.

Proposition 2.3.6

Existence d'ensembles récursifs non p-énumérables. graph(Ack) *n'est pas* p*-énumérable.*

Démonstration. Supposons que le graphe de la fonction (unaire) d'Ackermann, c'est-à-dire l'ensemble des $\langle n, \mathrm{Ack}(n,n)\rangle$, soit p-énumérable. Par une propriété de la fonction de pairage : $\forall n > 0, \langle x_0, \dots, x_n\rangle > x_n$, il existerait donc une fonction $p \in \mathsf{p}$ injective qui serait supérieure à la fonction d'Ackermann en une infinité de points, et même une fonction $f \in \mathsf{p}$, $f : n \mapsto \sum_{i=0}^{i=n} p(i)$ qui dominerait la fonction d'Ackermann, ce qui est en contradiction avec le fait qu'elle soit récursive primitive.

Ainsi, le graphe de Ack ne peut pas être p-énumérable. □

Cette propriété reste vraie pour toute les classes c-fondamentales.

Théorème 2.3.7

Existence d'ensembles récursifs non c-énumérables. graph($\mathsf{BB}_{\mathsf{c}}^{\Phi}$) *n'est pas* c*-énumérable.*

Démonstration. On sait que la fonction du castor affairé relativisée à une classe fondamentale c n'est pas c-fondamentale et on sait également que dominer la fonction du castor affairé implique de savoir la calculer (voir Définition 1.3.7). De la même manière, énumérer son graphe revient à pouvoir la dominer et donc la calculer. Par contraposé, graph($\mathsf{BB}_{\mathsf{c}}^{\Phi}$) n'est pas c-énumérable (voir la preuve du Théorème 2.3.12 pour un développement plus détaillé). □

2.3.3 Compatibilité et hérédité

La notion d'énumérabilité classique et la notion de $\mathtt{C}$-énumérabilité ne coïncident donc pas. Il convient cependant de noter que, dans le cas qui nous intéresse, où $\mathcal{D} = \omega$, tout ensemble $\mathtt{C}$-énumérable est également récursivement énumérable. Puisque chaque fonction énumérante de $\mathtt{C}$ peut être transformée en une fonction équivalente en calculabilité classique.

Théorème 2.3.8

$\mathtt{C}$-énumérable implique récursivement énumérable. *Soit $\mathtt{C}$ une classe fondamentale et E un ensemble d'entiers naturels. Si E est $\mathtt{C}$-énumérable alors E est* r.e.

Démonstration. Soit $E \subset \omega$ un ensemble $\mathtt{C}$-énumérable et $f \in \mathtt{C}$ une fonction en témoignant.

Nous cherchons maintenant à construire une fonction récursive partielle dont l'image soit E. Il suffit pour cela de construire une fonction φ qui simulera f sur son segment initial injectif, et qui divergera en les autres points.

Comme f est totale, la simulation de celle-ci par φ ne présente aucune difficulté : il suffit que $\varphi(n)$ calcule les valeurs de $f(0)$ à $f(n+1)$ pour savoir si la fin éventuelle du segment initial injectif de f a été atteint ou pas.

Une telle fonction φ a bien pour image E et est bien récursive partielle donc E est *r.e.* comme annoncé.

□

Nous avons maintenant une notion de $\mathtt{C}$-énumérabilité qui est un raffinement, au sens de durcissement, de la notion habituelle. Il existe ainsi des ensembles *r.e.* qui sont $\mathtt{C}$-énumérables et d'autres qui ne le sont pas. Il serait agréable de trouver une caractérisation des ensembles *r.e.* également $\mathtt{C}$-énumérables. Cette caractérisation existe et prend en fait la forme d'un théorème d'hérédité.

Le théorème qui va suivre a été prouvé initialement par Koz'minyh dans [Koz72] en 1972 mais ne s'appliquait qu'aux ensembles énumérables par des fonctions primitives récursives injectives. Nous le généralisons ici à la notion de $\mathtt{C}$-énumérabilité.

Théorème 2.3.9

Héritage de la $\mathtt{C}$-énumérabilité. *Soit $\mathtt{C}$ une classe fondamentale et $e, i \in \omega$ deux indices tel que $\mathbb{W}_e^{\mathtt{C}}$ soit un ensemble $\mathtt{C}$-énumérable infini et $\mathbb{W}_i \supset \mathbb{W}_e^{\mathtt{C}}$ soit récursivement énumérable.*
Alors, $\mathbb{W}_i$ est $\mathtt{C}$-énumérable et son indice dans $\mathbb{W}_{\cdot}^{\mathtt{C}}$ est calculable de manière récursive primitive en fonction de e et i.

Démonstration. Nous utilisons ici la fonction sim définie dans la Définition 1.1.6 et telle que pour n'importe quels $i, t, s \in \omega$, $\mathrm{sim}(i, t, s)$ tente de calculer $\varphi_{i,s}(t)$ et renvoie soit 0 si le calcul diverge en s ou converge pour $s' < s$, soit $\langle t, \varphi_i(t)\rangle + 1$ si le calcul converge en exactement s étapes.

Cette fonction est belle et bien c-fondamentale puisque construite sur un prédicat Δ^0_0 elle est récursive primitive (voir Définition 1.1.6).

L'idée va être maintenant d'énumérer deux listes d'éléments de $\mathbf{W}_i$ une liste qui sera facile à générer par une fonction c-fondamentale injective, et qui nous permettra de ne jamais être à court d'élément nouveau à énumérer. Et une seconde liste qui contiendra elle tous les éléments de $\mathbf{W}_i$, mais qui potentiellement ne sera pas injective.

Nous apporterons ensuite un soin particulier lors de la construction de notre fonction de manière à ce qu'elle demeure injective, et qu'elle n'oublie aucun élément de $\mathbf{W}_i$.

Étant donné un entier n et l'indice j d'une fonction récursive, il est possible de calculer la liste, éventuellement vide, $l_j(n)$ des éléments du graphe de φ_j renvoyés par $\mathrm{sim}(j, t, s)$, pour tous les couples (t, s) d'entiers (positifs) vérifiant $t + s = n + 1$. Ce calcul est défini par une imbrication d'opérations bornées et est donc réalisable par une fonction p-fondamentale. Remarquons que tous les éléments du graphe de la fonction récursive d'indice j apparaissent une et une seule fois dans $l_j(n)$, à partir d'un n assez grand.

D'autre part, remarquons que puisque $\mathbf{W}_e$ est c-énumérable, alors e est tel que $\mathbf{\Phi}^{\mathsf{c}}_e$ est injective et totale et témoigne de ce fait.

En utilisant cette fonction, il est simple de construire pour n'importe quels n et i une seconde liste $\mathrm{enum}(e, i, n)$ constituée comme suit :

$$\begin{gathered} f_e(0), \\ \text{les éléments de } l_i(0), \\ f_e(1), \\ \text{les éléments de } l_i(1), \\ \dots \\ f_e(n), \\ \text{les éléments de } l_i(n). \end{gathered}$$

Comme nous effectuons encore et toujours des opérations bornées, notre construction demeure c-fondamentale. La longueur de la liste est même maîtrisée, puisqu'elle est bornée par $n + n^2$, tout en contenant au moins n éléments distincts.

Il ne reste plus maintenant qu'à utiliser cette liste pour énumérer à l'étape n un $n^{\text{ième}}$ élément distinct. Cette opération est une fois de plus bien bornée et donc c-fondamentale.

Finalement, l'ordre qu'induit la liste sur les éléments de $\mathbf{W}_i$ (en supprimant les doublons) étant bien d'ordinal type ω, nous savons qu'aucun élément ne sera laissé sur la touche lors de l'énumération. C'est donc bien tout $\mathbf{W}_i$ qui est énuméré injectivement, d'où le résultat.

□

Note 2.3.10 Cas des ensembles finis. *Le théorème précédent s'applique trivialement si $\mathbb{W}_i$ est fini. Cependant, dans le cas général, il échoue si $\mathbb{W}_e$ est fini, puisqu'alors il nous est impossible de nous assurer que nous disposons en permanence de suffisamment d'éléments nouveaux à énumérer.*

Ce résultat assure également que l'union de deux ensembles C-énumérable est effective, et qu'il est possible à partir des indices de deux ensembles C-énumérables de calculer de manière C-fondamentale l'indice de l'ensemble union.

Corollaire 2.3.11

Il existe une fonction récursive primitive H_C permettant de calculer l'indice de l'union de deux ensembles C-énumérables :

$$\mathbb{W}^{\mathsf{C}}_{\mathsf{H}_\mathsf{C}(\langle i,j\rangle)} = \mathbb{W}^{\mathsf{C}}_i \cup \mathbb{W}^{\mathsf{C}}_j.$$

Démonstration. A priori, la difficulté ici est de réussir à faire une énumération injective à partir de deux énumérations injectives. Un moyen de résoudre ce problème serait de voir que pour produire le $n^{ième}$ élément de l'ensemble union, il suffit d'énumérer les $n+1$ premiers éléments de chacun des ensembles initiaux afin d'en trouver un qui n'a pas déjà été énuméré.

En pratique, ce résultat est un corollaire du Théorème 2.3.9. En effet, $\mathbb{W}^{\mathsf{C}}_i \cup \mathbb{W}^{\mathsf{C}}_j$ est un ensemble récursivement énumérable contenant un sous-ensemble C-énumérable $\mathbb{W}^{\mathsf{C}}_i$. Comme à partir d'un C-indice a il est possible de calculer de manière récursive primitive un indice p tel que $\mathbb{W}^{\mathsf{C}}_a = \mathbb{W}_p$, le Théorème 2.3.9 s'applique.

□

2.3.4 Énumérations de graphes, d'images et de fonctions

Arrêtons-nous un instant sur les conséquences du fait que les différentes caractérisations des ensembles *r.e.* ne soient plus équivalentes dans le cas des classes fondamentales et des ensembles C-énumérables.

Cela va nous amener à faire des distinctions que nous n'avions pas à faire dans le cas classique. En particulier, considérons une fonction f sur les naturels. Dans le cas de la calculabilité classique, énumérer son graphe et énumérer son image sont des problèmes rigoureusement équivalents.

Cela n'est plus le cas lors de l'étude de la C-énumérabilité, comme l'indique le résultat suivant :

Théorème 2.3.12

Énumérer une image n'est ni plus facile ni plus difficile qu'énumérer un graphe. *Soit* C *une classe fondamentale. Il existe une fonction* f *dont le graphe est* C*-énumérable mais dont l'image ne l'est pas. Et inversement.*

Démonstration. Considérons la fonction suivante :

$$f : \begin{cases} 2x \mapsto & \mathsf{BB}_{\mathsf{C}}^{\Phi}(x) \\ 2x+1 \mapsto & 0 \end{cases}$$

Énumérer son image injectivement impliquerait de pouvoir dominer [2] la fonction du Castor Affairé. En effet, celle-ci étant croissante, pour tout n, la somme de $n+1$ éléments de l'image de f serait plus grande que la somme des n premières valeurs de $\mathsf{BB}_{\mathsf{C}}^{\Phi}$. Or, dominer $\mathsf{BB}_{\mathsf{C}}^{\Phi}$ implique de pouvoir la calculer à condition que la classe énumérable considérée possède les fonctions use et sim, ce qui est en contradiction avec le fait que $\mathsf{BB}_{\mathsf{C}}^{\Phi}$ n'est pas C-fondamentale. L'image de f n'est pas C-énumérable. En revanche, son graphe lui est C-énumérable.

Pour le voir, il suffit d'appliquer le Théorème 2.3.9. En effet, le graphe de f est récursivement énumérable, car $\mathsf{BB}_{\mathsf{C}}^{\Phi}$ est récursive. Il contient l'ensemble trivial suivant : $\{\langle 2x+1, 0\rangle : x \in \omega\}$, qui lui est C-énumérable de manière triviale. Donc, par application de la proposition, le graphe de f est C-énumérable, alors que l'image ne l'était pas.

Inversement, nous pouvons considérer la fonction suivante :

$$g : x \mapsto max\left(1, \sum_{0 \leqslant n \leqslant x} (\chi_{\mathsf{K}}(n))\right)$$

Où χ_{K} est la fonction indicatrice de l'ensemble diagonal classique.

L'image de g est $\omega \setminus \{0\}$, C-énumérable. En revanche, le graphe de g n'est même pas récursivement énumérable. □

Il est possible de trouver des comparaisons similaires pour ce qui est de la difficulté à énumérer le graphe de la fonction indicatrice d'un ensemble image d'une fonction. Nous verrons plus d'exemples à ce sujet lorsque nous étudierons les réductions entre ensembles dans les chapitres suivantes.

2.4 Récursivités d'un ensemble

En calculabilité classique, la notion de récursivité d'un ensemble possède un grand nombre de caractérisations différentes.

2. f domine g : $\exists N, \forall n, f(n) \leqslant g(n)$

Considérons par exemple la définition suivante d'un ensemble d'entiers récursif : $E \subset \omega$ est récursif s'il est *r.e.* et que son complémentaire $\overline{E}$ l'est également. Nous savons prouver qu'alors il va exister une énumération croissante des éléments de E. Nous en connaissons tous la preuve suivante :

Démonstration. Soit $\mathfrak{e}_E$, $\mathfrak{e}_{\overline{E}}$ deux fonctions récursives énumérant respectivement E et $\overline{E}$. Posons $\mathfrak{p}_E = \varphi_e$ pour e point fixe de Kleene de la fonction récursive totale f définie par :

$$\varphi_{f(e)} : x \mapsto \pi_1^3 \left(\mu \langle n_1, n_2, b \rangle \, , \ \wedge \begin{array}{c} \mathfrak{e}_E(n_2) = n_1 \\ \\ \forall m < n_2, \exists p < b, (m = \mathfrak{e}_{\overline{E}}(p) \vee m = \varphi_e(p)) \end{array} \right).$$

□

Nous pourrions facilement nous passer de l'application du point fixe ici, cependant, ce n'est en général pas le cas pour la recherche non bornée. De plus, l'astuce usuelle consistant à énumérer d'autres éléments en attendant de trouver celui recherché n'est pas envisageable puisque notre fonction doit être strictement croissante.

Une autre caractérisation d'un ensemble récursif est l'existence d'une fonction indicatrice (totale) pour celui-ci. Le même raisonnement que précédemment permet de voir que cette caractérisation est différente de la première. Nous verrons par la suite qu'elle est également différente de la seconde.

2.4.1 Approches des récursivités

La première caractérisation que nous allons considérer est l'existence d'une fonction indicatrice fondamentale. Nous appelons cette notion de récursivité la χ-récursivité en référence aux fonctions indicatrices souvent nommées χ.

Définition 2.4.1 **χ-récursivité.** *Un ensemble $E \subset \mathcal{D}$ est dit χ-$\mathtt{C}$-récursif si sa fonction indicatrice χ_E est fondamentale (*i.e. $\chi_E \in \mathtt{C}$.*)*

La fonction indicatrice, comme son nom l'évoque, nous indique si un élément est présent ou non dans un ensemble. En revanche, elle ne donne aucune information quant aux moyens d'énumérer l'ensemble ou son complémentaire. En effet, comme nous allons le voir, en l'absence de schéma μ la fonction indicatrice ne permet pas de calculer la fonction d'énumération, et réciproquement.

Une autre caractérisation classique de la récursivité d'un ensemble est le fait qu'il soit *r.e.* et co-*r.e.*. Relativisée aux classes fondamentales, nous appelons cette notion la récursivité faible :

Définition 2.4.2 **faible-$\mathtt{C}$-récursivité.** *Un ensemble $E \subset \mathcal{D}$ est dit faiblement-$\mathtt{C}$-récursif si à la fois E et son complémentaire dans $\mathcal{D}$ sont $\mathtt{C}$-énumérables.*

En utilisant la fonction indicatrice ou deux énumérations d'un ensemble et de son complémentaire, nous pouvons classiquement construire une énumération croissante de l'ensemble. C'est la troisième caractérisation que nous considérerons ici. Nous l'appelons énumérabilité forte bien que ce soit une notion de récursivité car elle repose sur l'existence d'une énumération particulière.

Définition 2.4.3 **forte-$\mathtt{C}$-énumérabilité.** *Un ensemble $E \subset \mathcal{D}$ est dit fortement-$\mathtt{C}$-énumérable s'il est $\mathtt{C}$-énumérable via une fonction croissante.*

Enfin, la récursivité forte, par analogie à la récursivité faible, correspond aux ensembles qui sont eux-mêmes ainsi que leur complémentaire fortement énumérables :

Définition 2.4.4 **forte-$\mathtt{C}$-récursivité.** *Un ensemble $E \subset \mathcal{D}$ est dit fortement-$\mathtt{C}$-récursif si lui et son complémentaire dans $\mathcal{D}$ sont $\mathtt{C}$-énumérables via des fonctions croissantes.*

2.4.2 Relations entre les différentes récursivités

En considérant les quatre notions de récursivité définies dans le paragraphe précédent, ainsi que la notion classique de récursivité d'un ensemble, nous avons à notre disposition cinq propriétés distinctes pour étudier la difficulté de description d'un ensemble donné.

La première chose à faire, est de vérifier que ces propriétés ne font pas doublons, et d'identifier desquelles découlent lesquelles. Nous allons établir une analyse complète des implications et non-implications entre ces cinq notions. Chacune des non-implications sera accompagnée d'un contre-exemple qui illustrera bien en quoi les différentes notions de récursivité ne sont pas équivalentes dans les circonstances données.

Implications entre les différentes notions

Tout d'abord, vérifions qu'au vu des définitions, les ensembles capturés par les notions de $\mathtt{C}$-récursivité ne touchent pas des ensembles qui ne sont pas récursifs au sens classique.

Théorème 2.4.5

$\mathtt{C}$-récursif implique classiquement récursif. *Soit $\mathtt{C}$ une classe fondamentale et $E \subset \omega$ un ensemble d'entiers. Si E est χ-$\mathtt{C}$-récursif, faiblement $\mathtt{C}$-récursif, fortement $\mathtt{C}$-énumérable ou fortement $\mathtt{C}$-récursif alors E est récursif au sens classique.*

Démonstration. Les définitions ayant été calquées sur diverses caractérisations de la récursivité dans le cas classique, il suffit de les considérer après avoir appliqué le Théorème 2.3.8.

E **χ-$\mathtt{C}$-récursif** Si la fonction indicatrice est $\mathtt{C}$-fondamentale, donc récursive, donc l'ensemble E est récursif.

E **faiblement $\mathtt{C}$-récursif** E est $\mathtt{C}$-énumérable et co-$\mathtt{C}$-énumérable, il est donc récursivement énumérable et co-récursivement énumérable, et donc récursif.

E **fortement $\mathtt{C}$-énumérable** E est énumérable de manière croissante par une fonction fondamentale, il l'est donc par une fonction récursive, donc il est récursif.

E **fortement $\mathtt{C}$-récursif** E est fortement $\mathtt{C}$-énumérable, donc récursif.

□

Le statut de la forte $\mathtt{C}$-énumérabilité est un peu ambiguë. Nous l'appelons énumérabilité mais la classons parmi les notions de récursivité. Le fait est qu'elle est plus forte qu'une $\mathtt{C}$-énumérabilité (faible) mais ne permet pas pour autant, comme nous allons le voir dans le prochain paragraphe, de déduire que le complémentaire est lui-même $\mathtt{C}$-énumérable. En revanche, il est possible de construire une fonction indicatrice à partir d'une énumération forte, comme le montre le résultat suivant :

Proposition 2.4.6

La forte $\mathtt{C}$-énumérabilité implique la χ-$\mathtt{C}$-récursivité.

Démonstration. Supposons E fortement $\mathtt{C}$-énumérable. Il existe alors $p \in \mathtt{C}$ énumérant E de manière croissante. Pour calculer $\chi_E(n)$, il suffit de calculer la liste $p(0), \ldots, p(n)$ et de regarder si n est dans celle-ci. Si c'est le cas, alors $\chi_E(n) = 1$, sinon, $\chi_E(n) = 0$. Cette opération étant récursive primitive (relativement à p), nous avons bien par les propriétés de clôture de $\mathtt{C}$ que si $p \in \mathtt{C}$, $\chi_E \in \mathtt{C}$. □

Non-implications et ensembles remarquables

Il reste à vérifier qu'aucune de ces notions ne fait doublon avec une autre et, bien entendu, qu'elles ne sont pas triviales, c'est à dire que pour toutes ces propriétés, il existe un ensemble qui l'ait, et un ensemble qui ne l'ait pas. Là où nous nous rendons compte que ces récursivités sont réellement porteuses de sens et identifient des notions différentes et intéressantes, c'est dans l'étude des ensembles diagonaux. Ces ensembles qui transcendent leurs classes de fonctions associées nous permettent d'identifier une grande partie de ces distinctions.

Ensemble diagonal des fonctions récursives classiques

Propriété 2.4.7

K *est* p*-énumérable mais n'est pas co-*r.e.

Démonstration. K a beau être un ensemble non-récursif, il contient des sous-ensembles triviaux. Prenons par exemple $e_0 = \mathbf{const}(0)$ un indice pour la fonction constante nulle. $\varphi_{e_0}(e_0)\downarrow$, donc $e_0 \in \mathsf{K}$. Considérons $E_0 = \left\{\mathfrak{p}(\mathbf{const}(0), n) : n \in \omega\right\}$, un ensemble d'indices de la fonction constante nulle généré par la fonction de bourrage, alors $E_0 \subset \mathsf{K}$ et de plus, E_0 est trivialement p-énumérable. Or, K est récursivement énumérable, donc par application du Théorème 2.3.9, nous vérifions que K est p-énumérable.

En revanche, nous savons que K n'est pas co-*r.e.* puisqu'il est *r.e.* mais non récursif. □

Ainsi, si un ensemble p-énumérable est nécessairement *r.e.*, il n'est pas pour autant nécessairement récursif. Ce contre-exemple est la preuve du corollaire suivant :

Corollaire 2.4.8

p*-énumérable n'implique pas co-*r.e.

Ensemble diagonal des fonctions fondamentales

Propriété 2.4.9

$\mathsf{K}_{\mathsf{c}}^{\Phi}$ *est récursif au sens classique,* p*-énumérable, co-*p*-énumérable et donc faiblement* p*-récursif. En revanche il n'est ni fortement* c*-énumérable, ni* χ*-*c*-calculable.*

Démonstration. Nous allons procéder étape par étape :

Récursivité Étant donné que c est une classe énumérable récursives de fonctions totales simulables, $\mathsf{K}_{\mathsf{c}}^{\Phi}$ est trivialement récursif.

Faible p-récursivité Pour montrer la faible p-récursivité, nous allons montrer la p-énumérabilité et la co-p-énumérabilité.
Les ensembles $\left\{\mathfrak{p}(\mathbf{const}(0), n) : n \in \omega\right\}$ et $\left\{\mathfrak{p}(\mathbf{const}(1), n) : n \in \omega\right\}$ sont des exemples d'ensembles c-énumérables inclus respectivement dans $\overline{\mathsf{K}_{\mathsf{c}}^{\Phi}}$ et $\mathsf{K}_{\mathsf{c}}^{\Phi}$
Une fois de plus, par application du Théorème 2.3.9 nous en déduisons leur c-énumérabilité respective. $\mathsf{K}_{\mathsf{c}}^{\Phi}$ est donc faiblement c-récursif.

Non χ-c-récursivité Par un simple procédé de diagonalisation, il est possible de montrer que la fonction indicatrice de $\mathsf{K}_{\mathsf{c}}^{\Phi}$ n'est pas dans c. En effet, si c'était le cas, nous pourrions construire une fonction f d'indice e telle que $f : n \mapsto 1 - \chi_{\mathsf{K}_{\mathsf{c}}^{\Phi}}(n)$. $f(e)$ vaudrait alors $1 - \chi_{\mathsf{K}_{\mathsf{c}}^{\Phi}}(e)$. C'est à dire qu'elle vaudrait 1 si et seulement si elle vaut 0, ce qui est absurde. Donc $\mathsf{K}_{\mathsf{c}}^{\Phi}$ n'est pas χ-c-récursif.

Non forte c-énumérabilité $\mathsf{K}_{\mathsf{c}}^{\Phi}$ n'est pas χ-c-récursif, il n'est donc pas fortement c-énumérable d'après la Proposition 2.4.6.

□

Ce contre exemple a pour corollaire le résultat suivant :

Corollaire 2.4.10

La faible c-récursivité n'implique pas la χ-c-récursivité.

Image de la fonction d'Ackermann La fonction d'Ackermann, aussi appelée fonction diagonale de la classe des fonctions récursives primitives, nous offre un autre ensemble diagonal pour p, mais dans une version bien plus *diluée* cette fois.

Propriété 2.4.11

L'image de la fonction d'Ackermann est χ-p-récursive, mais n'est pas p-énumérable. De plus, elle est co-fortement-p-énumérable.

Démonstration. Continuons l'étude de cette fonction, commencée dans le Théorème 2.3.7.

χ-p-récursive Nous cherchons à montrer qu'étant donné un entier e, il nous est possible de vérifier s'il appartient à l'image de la fonction d'Ackermann ou non. Pour ce faire, nous allons montrer qu'il est possible d'énumérer tout un ensemble borné d'arbres de calculs contenant, s'il existe, celui calculant e.

Le nombre d'étapes de calcul (ou encore la taille de l'arbre de calcul) de la fonction d'Ackermann sur une entrée n peut être borné de manière primitive récursive en la valeur de $\mathrm{Ack}(n)$. Il suffit d'écrire tous les arbres de calculs possibles de taille inférieure à cette borne, correspondant aux appels de la fonction avec des arguments inférieurs à cette borne, et de sélectionner ceux qui correspondent à un arbre de calcul correct et complet pour Ack. Sachant cela, vérifier si un entier e appartient à l'image de la fonction d'Ackermann est réalisable de manière récursive primitive. Il suffit de générer les arbres de calculs de taille bornée en fonction de e, et de regarder si e est le résultat d'un de ces calculs.

Non p-énumérabilité S'il nous était possible d'énumérer l'image de la fonction d'Ackermann de manière récursive primitive, nous pourrions alors majorer cette même fonction, ce qui est absurde (voir Théorème 2.3.7).

Co-forte-p-énumérabilité Nous savons que si un entier n appartient à l'image de Ack, alors il n'en est pas de même de $n+1$. Ainsi, la fonction p énumérant de manière croissante le complémentaire de l'image de Ack est définie comme suit :

$$p : \begin{cases} 0 & \mapsto \quad 0 \\ n+1 & \mapsto \quad \begin{cases} p(n)+1 & \text{si } \chi_{\text{range(Ack)}}(p(n)+1) = 0, \\ p(n)+2 & \text{sinon (car alors } \chi_{\text{range(Ack)}}(p(n)+2) = 0). \end{cases} \end{cases}$$

□

Image de la fonction du Castor Affairé Remarquons maintenant que la fonction du Castor Affairé joue le même rôle pour une classe fondamentale c que la fonction d'Ackermann pour la classe des fonctions récursives primitives p.

Propriété 2.4.12

L'image de la fonction $\mathsf{BB}^{\Phi}_{\mathsf{c}}$ est χ-c-récursive, mais n'est pas c-énumérable. De plus, elle est co-fortement-c-énumérable.

Démonstration. La démonstration se fait de manière similaire à la preuve de 2.2.6 en utilisant les fonctions de simulation. □

Cette étude nous fournit donc un contre-exemple prouvant les deux corollaires suivants :

Corollaire 2.4.13

La χ-récursivité n'implique aucune autre c-récursivité.

Corollaire 2.4.14

La forte c-énumérabilité n'implique pas la faible c-récursivité ni la co-forte-c-énumérabilité.

Démonstration. En effet, la faible c-récursivité, la forte c-récursivité et la forte c-énumérabilité impliquent toutes les trois la c-énumérabilité. □

	r.e.	co-*r.e.*	p-e	s-p-e	co-p-e	co-s-p-e	w-p-rec	s-p-rec	χ-p-rec
K	✓	×	✓	×	×	×	×	×	×
$\mathsf{K}_{\mathsf{p}}^{\Phi}$	✓	✓	✓	×	✓	×	✓	×	×
A	✓	✓	×	×	✓	✓	×	×	✓
B	✓	✓	×	×	✓	✓	×	×	✓
	r.e.	co-*r.e.*	c-e	s-c-e	co-c-e	co-s-c-e	w-c-rec	s-c-rec	χ-c-rec
K	✓	×	✓	×	×	×	×	×	×
$\mathsf{K}_{\mathsf{c}}^{\Phi}$	✓	✓	✓	×	✓	×	✓	×	×
C	✓	✓	×	×	✓	✓	×	×	✓

Table 2.1 – Propriétés des ensembles remarquables dans les classes fondamentales.

En résumé

Voici une petite liste des résultats que nous avons prouvés dans les deux paragraphes précédents et qui doivent conclure l'étude des implications et non-implications entre ces notions de récursivité. Le Tableau 2.1 résume les propriétés vues pour les ensembles suivants : $A = \mathrm{range}(\mathrm{Ack})$, $B = \mathrm{range}(\mathbf{BB}_{\mathsf{p}}^{\Phi})$, $C = \mathrm{range}(\mathbf{BB}_{\mathsf{c}}^{\Phi})$.

— Toutes les c-récursivités ainsi que la forte c-énumérabilité impliquent la récursivité classique (Théorème 2.4.5).
— La χ-c-n'implique aucune autre c-récursivité ni même l'énumérabilité (Corollaire 2.4.13).
— La faible c-récursivité n'implique aucune autre c-récursivité (Corollaire 2.4.10) mais elle implique la c-énumérabilité (par définition).
— La forte c-énumérabilité implique la faible c-énumérabilité (par définition) et la χ-c-récursivité (Corollaire 2.4.6) mais aucune autre c-récursivité.
— La forte c-récursivité implique toutes les autres c-récursivités (par définition en d'après les résultats précédents).

2.5 Réductions

Les notions de récursivité définies dans le point précédent mettent le doigt sur quatre notions différentes suivant les aspects de la complexité de description des ensembles qui nous intéressent. Bien sûr, ces notions de récursivités appellent chacune un ensemble de réductions qui lui sont associées.

En calculabilité classique, nous avons toute une zoologie de réductions entre ensembles. Une des plus enseignées est la réduction Turing, qui autorise à la fonction effectuant la réduction d'utiliser la fonction caractéristique de l'ensemble plus compliqué afin de calculer l'ensemble moins compliqué.

Cependant, d'autres réductions existent et nous avons déjà aperçu dans le chapitre précédent la réduction *many-one*, dite aussi m-réduction, et la réduction *btt* dite à table de vérité bornée. Nous reviendrons par la suite sur ces variantes possibles.

Pour la χ-récursion, nous l'avons définie comme le fait d'avoir une fonction indicatrice fondamentale. Être χ-récursif en un ensemble E signifiera donc être χ-récursif dans la plus petite classe fondamentale contenant $\mathtt{C}$ et χ_E :

Définition 2.5.1 **Réductions χ-$\mathtt{C}$-récursives.** *Soient A et B deux sous-ensembles de $\mathcal{D}$. A est dit χ-$\mathtt{C}$-récursif en B ($A \leqslant^{\chi}_{\mathtt{C}\text{-}T} B$) si $\chi_A \in \mathtt{C}[\chi_B]$.*

Le cas de la faible récursion est un petit peu plus délicat. Nous demandons qu'il existe une énumération d'un ensemble et de son complémentaire. Autrement dit, pour que la réduction se fasse d'un ensemble E vers un ensemble A, il faut que pour n'importe quel couple d'énumération de E et de son complémentaire $\overline{E}$ étendant $\mathtt{C}$, nous puissions trouver une énumération de A et une de son complémentaire.

Définition 2.5.2 **Réductions faiblement-$\mathtt{C}$-récursives.** *Soient A et B deux sous-ensembles de $\mathcal{D}$. A est dit faiblement-$\mathtt{C}$-récursif en B ($A \leqslant^{w}_{\mathtt{C}\text{-}T} B$) si pour tout $\mathfrak{e}_B, \mathfrak{e}_{\overline{B}}$ énumérant respectivement B et son complémentaire, il existe $\mathfrak{e}_A, \mathfrak{e}_{\overline{A}} \in \mathtt{C}[\mathfrak{e}_B, \mathfrak{e}_{\overline{B}}]$ énumérant respectivement A et son complémentaire.*

La forte énumération ne pose pas le même problème que précédemment puisque l'énumération croissante d'un ensemble est, par définition, unique.

Définition 2.5.3 ***A* fortement-$\mathtt{C}$-énumérable en *B*.** *Soient A et B deux sous-ensembles de $\mathcal{D}$. A est dit fortement-$\mathtt{C}$-énumérable en B ($A \leqslant^{s}_{\mathtt{C}\text{-}e} B$) si étant donné $\mathfrak{p}_B$ l'énumération croissante de B, il existe $\mathfrak{p}_A \in \mathtt{C}[\mathfrak{p}_B]$ énumérant A de manière croissante.*

De même, nous définissons la forte récursion comme nous l'avons fait pour l'énumération forte, mais en demandant à ce que les complémentaires se réduisent également l'un vers l'autre.

Définition 2.5.4 ***A* fortement-$\mathtt{C}$-récursif en *B*.** *Soient A et B deux sous-ensembles de $\mathcal{D}$. A est dit fortement-$\mathtt{C}$-récursif en B ($A \leqslant^{s}_{\mathtt{C}\text{-}T} B$) si étant donné $\mathfrak{p}_B$ et $\mathfrak{p}_{\overline{B}}$ énumérant respectivement B et son complémentaire de manière croissante, il existe $\mathfrak{p}_A, \mathfrak{p}_{\overline{A}} \in \mathtt{C}[\mathfrak{p}_B, \mathfrak{p}_{\overline{B}}]$ énumérant respectivement A et son complémentaire de manière croissante.*

Ces réductions seront étudiées plus en détail dans le chapitre suivant alors que nous aurons introduit des outils supplémentaires permettant ce travail. Nous étudierons alors les relations entre ces différentes notions, les structures de degrés engendrées par chacune d'entre elles et leurs liens avec la structure des degrés Turing classique. Nous aborderons également, mais sans entrer dans les détails, les autres variantes des réductions classiques que nous pouvons construire (*truth-table*, *weak truth-table*, *etc.*) et nous verrons comment elles se comparent avec les réductions déjà définies.

2.6 Hyper-structure

En calculabilité classique, nous avons une construction canonique pour élever la puissance de calcul de nos fonctions. Pour ce faire, nous rajoutons un oracle à nos fonctions, ce qui produit une nouvelle classe de fonctions, calculables étant donné cet oracle.

L'oracle le plus évident dans ce cadre est le problème de l'arrêt. Un pallier est ainsi franchi en ajoutant toute une classe de fonctions sachant déjà tout ce qu'il y a à savoir sur les fonctions récursives.

Nous savons qu'il nous est ensuite possible d'itérer ce saut un nombre quelconque de fois avant d'atteindre la limite de la hiérarchie arithmétique, sans que celle-ci ne s'effondre jamais.

Il existe un parallèle très clair dans le contexte des classes fondamentales. Certes, le problème de l'arrêt n'a pas de sens dans le cadre des fonctions totales. Mais comme nous l'avons déjà remarqué, le pendant de l'arrêt dans les classes ne disposant pas de schéma mu est le nombre d'étapes d'exécution avant arrêt.

De la même manière qu'en récursivité classique il n'est pas possible pour une fonction de savoir si un indice passé en paramètre correspond à une fonction qui termine sur l'entrée 0, il n'est pas possible dans les classes fondamentales pour une fonction de déterminer le nombre d'étapes nécessaires à une fonction dont l'indice est donné en paramètre avant que celle-ci ne termine.

Le problème du nombre maximal d'étapes est donc le problème de l'arrêt des fonctions fondamentales. La fonction qui le résout est la fonction du Castor affairé $\mathsf{BB}_{\mathsf{C}}^{\Phi}$ vue précédemment, voir définition 1.3.7.

C'est cette fonction qui va nous permettre de réaliser un *saut* entre classes fondamentales.

Définition 2.6.1 **Saut d'une Classes Fondamentales.** © *désigne la classe fondamentale définie par*

$$© = \mathsf{C}\left[\mathsf{BB}_{\mathsf{C}}^{\Phi}\right].$$

© est une classe fondamentale par définition de l'opérateur de relativisation.

La première chose à faire est de vérifier que cette hiérarchie ne s'effondre jamais elle non-plus.

Propriété 2.6.2

Le saut d'une classe fondamentale est une sur-classe stricte de celle-ci : $\mathsf{C}' \supsetneq \mathsf{C}$.

Démonstration. Comme nous l'avons vu, $\mathsf{BB}_{\mathsf{C}}^{\Phi} \notin \mathsf{C}$. Rappelons-nous d'ailleurs que la fonction $\mathsf{BB}_{\mathsf{C}}^{\Phi}$ nous permettait de calculer la fonction universelle U_{C} qui n'est elle-même pas dans C.

Nous avons donc au moins deux fonctions, $\mathsf{BB}_{\mathtt{C}}^{\Phi}$ et $\mathsf{U}_{\mathtt{C}}$ qui appartiennent à $© \setminus \mathtt{C}$.
$\square$

2.7 Théorèmes utilisant le saut

Le saut © d'une classe fondamentale $\mathtt{C}$ a la bonne propriété de posséder une fonction universelle (totale) pour $\mathtt{C}$. Aussi, elle nous permet de donner une forme altérée des théorèmes qu'il ne nous était pas possible de prouver précédemment par manque d'une fonction universelle.

Le premier théorème que nous retrouvons dans cette liste est le théorème de Kleene. Nous n'avons pas de vrai théorème *à la* Kleene dans les classes fondamentales. En revanche, nous avons une version un peu moins effective qui utilise ©.

De même, le classique théorème de Rice [Ric53] indiquant que pour un ensemble d'indices non-trivial X, soit l'ensemble diagonal, soit son complémentaire se 1-réduit à X, nécessite dans notre cas que cette 1-réduction se fasse dans © et est en général impossible dans $\mathtt{C}$.

2.7.1 Théorème de la récursion déséquilibré

Nous appelons ce théorème *déséquilibré* puisque, contrairement à la version classique, la fonction dont le point fixe est calculé et la fonction réalisée par le point fixe ne sont pas dans la même classe.

Théorème 2.7.1

Théorème déséquilibré de la récursion de Kleene. *Soit $\mathtt{C}$ une classe fondamentale et $f \in \mathtt{C}$. Alors, il existe $n \in \mathcal{D} = \omega$ tel que $\Phi_n^{©} \cong \Phi_{f(n)}^{\mathtt{C}}$.*

Démonstration. Soit $\mathtt{C}$ une sous-calculabilité et $f \in \mathtt{C}$ une fonction fondamentale. On définit une fonction q en utilisant les opérations sur les indices de ©, telle que pour tout e, $\Phi_{q(e)}^{©} = \Phi_{f \circ \Phi_e^{\mathtt{C}}(e)}^{\mathtt{C}}$. Cette fonction de transformation d'indice est primitive récursive car elle n'utilise que les opérations du schéma de codage. Soit i un de ses indices pour $\Phi_{\cdot}^{\mathtt{C}}$.

Notons n la valeur de $q(i) = \Phi_i^{\mathtt{C}}(i)$. On remarque alors que $\Phi_n^{©} = \Phi_{q(i)}^{©} = \Phi_{f(\Phi_i^{\mathtt{C}}(i))}^{\mathtt{C}}$ mais comme $\Phi_i^{\mathtt{C}}(i) = q(i)$, nous avons $\Phi_{f(\Phi_i^{\mathtt{C}}(i))}^{\mathtt{C}} = \Phi_{f(q(i))}^{\mathtt{C}} = \Phi_{f(n)}^{\mathtt{C}}$. $\square$

2.7.2 Théorème *à la* Rice

Un autre théorème majeur en calculabilité est le théorème de Rice. Ce dernier indique que les ensembles non-triviaux d'indices d'une énumération ne peuvent pas être calculables. Plus exactement, le théorème de Rice indique qu'un ensemble non-trivial d'indice permet de calculer par m-réduction soit l'arrêt, soit son complémentaire.

Le même phénomène se produit à l'intérieur des classes fondamentales, pour une des notions de réductions que nous avons définie : la χ-récursion. Ainsi, dans une classe fondamentale C, pour un ensemble d'indice donné P, l'ensemble diagonal de cette classe fondamentale se 1-réduit à l'ensemble d'indice ou à son complémentaire.

Définition 2.7.2 **Ensemble non-trivial.** *Un ensemble non-trivial $P \subset \mathcal{D} = \omega$ d'indices de C est un ensemble satisfaisant :*
- $P \neq \varnothing$ *et* $P \neq \mathcal{D}$
- $\forall x, y \in \mathcal{D}, \boldsymbol{\Phi}^{\mathsf{C}}_x = \boldsymbol{\Phi}^{\mathsf{C}}_y \Rightarrow (x \in P \leftrightarrow y \in P)$.

Théorème 2.7.3

Théorème *à la* Rice. *Soit C une classe fondamentale. Si P est un ensemble non-trivial d'indice de fonctions C-fondamentales, alors :*

$$\mathbf{K}^{\boldsymbol{\Phi}}_{\mathsf{C}} \preccurlyeq^{©}_{1} P$$
$$ou$$
$$\mathbf{K}^{\boldsymbol{\Phi}}_{\mathsf{C}} \preccurlyeq^{©}_{1} \overline{P}.$$

Démonstration. Afin d'établir ce théorème, nous allons construire la fonction f effectuant la 1-réduction. Cette fonction ne sera pas C-fondamentale puisqu'elle a besoin de connaître la fonction caractéristique de l'ensemble diagonal. En revanche, l'indice qu'elle calculera sera bien l'indice d'une fonction C-fondamentale.

Nous allons considérer deux indices témoins, le premier est un indice qui est par construction dans l'ensemble diagonal : $e_0 = \mathbf{const}^{©}(0)$, un indice tel que $\boldsymbol{\Phi}^{©}_{e_0} \cong \mathbf{0}$.

Il faut maintenant faire une disjonction de cas. Soit $e_0 \in \overline{P}$, soit $e_0 \in P$.

Si $e_0 \in \overline{P}$, alors nous montrons que $\mathbf{K}^{\boldsymbol{\Phi}}_{\mathsf{C}} \preccurlyeq^{©}_{1} P$ comme suit.

Étant donné que P n'est pas vide, nous pouvons trouver un élément $e_1 \in P$. C'est notre second indice témoin, celui-ci, contrairement à e_0 n'appartiendra pas à $\overline{P}$.

Comme P est un ensemble d'indices, $\boldsymbol{\Phi}^{\mathsf{C}}_{e_1} \neq \boldsymbol{\Phi}^{\mathsf{C}}_{e_0}$.

Construisons ensuite une fonction ©-fondamentale f comme suit :

$$f : x \mapsto \begin{cases} \mathfrak{p}(e_1, x) & \text{si } x \in \mathbf{K}^{\boldsymbol{\Phi}}_{\mathsf{C}}\,; \\ \mathfrak{p}(e_0, x) & \text{sinon.} \end{cases}$$

De cette manière, f est injective et vérifie bien :

$$\forall y, \boldsymbol{\Phi}^{\mathsf{C}}_{f(x)}(y) = \begin{cases} \boldsymbol{\Phi}^{\mathsf{C}}_{e_1}(y) & \text{si } x \in \mathbf{K}^{\boldsymbol{\Phi}}_{\mathsf{C}}, \\ 0 & \text{sinon.} \end{cases}$$

Ainsi, pour un élément appartenant à l'ensemble diagonal : $x \in \mathsf{K}_{\mathsf{c}}^{\Phi}$, nous avons l'équivalence entre fonctions $\Phi^{\mathsf{c}}_{f(x)} \cong \Phi^{\mathsf{c}}_{e_1}$. Et donc puisque e_1 est dans P, $f(x) \in P$.

Inversement, pour un élément n'appartenant pas à l'ensemble diagonal : $x \notin \mathsf{K}_{\mathsf{c}}^{\Phi}$, nous avons une autre équivalence entre fonctions : $\Phi^{\mathsf{c}}_{f(x)} \cong \Phi^{\mathsf{c}}_{e_0}$. Cette fois-ci, comme e_0 n'est pas dans P, $f(x) \notin P$

Si $e_0 \in P$, nous montrons alors que $\mathsf{K}_{\mathsf{c}}^{\Phi} \preccurlyeq_1^{\copyright} \overline{P}$ en remplaçant P par $\overline{P}$ dans la preuve précédente, et toujours en utilisant e_0 et e_1, cette fois témoins respectifs de P et $\overline{P}$.

$\square$

3 Sous-calculabilités

Calculabilités sur les fonctions partielles

Sommaire

3.1 Notions de sous-calculabilité

Dans ce chapitre, nous allons construire une classe de fonctions généralisant les fonctions totales des classes fondamentales aux fonctions partielles. Ce faisant, nous allons obtenir un ensemble de fonctions partielles aux propriétés de clôtures inhabituelles.

Tout au long de ce chapitre, nous allons tenter de démontrer que ce duo de classes de fonctions, fondamentales et partielles, se comporte réellement comme une approximation de la calculabilité usuelle. En fait, chaque niveau supérieur d'une sous-calculabilité sera une approximation plus fidèle de la calculabilité, comblant certains manques et rendant possibles certaines constructions qui jusqu'alors ne l'étaient pas.

Tout d'abord, nous allons expliquer comment nous avons choisi de définir les fonctions $\mathtt{c}$-partielles comme l'ensemble des fonctions descriptibles par les fonctions $\mathtt{c}$-fondamentales. En cela, si la classe des fonctions $\mathtt{c}$-partielle n'obéit pas aux règles de clôtures habituelles, elle hérite tout de même d'un nombre surprenant de propriétés de la calculabilité classique rarement présentes dans les sous-classes de celle-ci. Ainsi, nous retrouverons le théorème d'isomorphisme de Rogers, ainsi qu'une variante affaiblie du second théorème de la récursion de Kleene, qui cette fois-ci est assez forte pour pouvoir remplacer l'original dans la plupart des preuves.

3.1.1 Les classes fondamentales comme fondation

Intuitivement, pour une classe fondamentale $\mathtt{c}$, les fonctions $\mathtt{c}$-partielles sont l'ensemble des fonctions partielles qu'il est possible de calculer en utilisant les fonctions fondamentales de $\mathtt{c}$. Il reste à préciser ce que nous entendons ici par « en utilisant ».

Soit $\mathtt{c}$ une classe fondamentale de domaine $\mathcal{D} = \omega$. Une vision simplificatrice serait de considérer la classe des fonctions récursives primitives $\mathtt{c} = \mathtt{p}$. Cependant, tout le raisonnement développé par la suite fonctionne avec les autres classes.

Rappelons que la classe $\mathtt{c}$ vient avec une énumération de ses fonctions fondamentales :

$$(\boldsymbol{\Phi}^{\mathtt{c}}_{d}\cdot)_{d\in\mathcal{D}}\,.$$

Nous avons également défini dans le chapitre précédent une notion d'énumérabilité des ensembles relativement à une classe fondamentale. A est $\mathtt{c}$-(récursivement)-énumérable s'il est l'image du préfixe injectif d'une fonction fondamentale.

Cette considération nous permet d'obtenir une deuxième énumération, cette fois-ci d'ensembles $\mathtt{c}$-énumérables :

$$(\mathbf{W}^{\mathtt{c}}_{d})_{d\in\mathcal{D}}\,.$$

Sur ces ensembles récursivement énumérables sont définies quatre notions de récursivité qui ne sont pas équivalentes :

— la forte $\mathtt{c}$-énumérabilité : le fait d'être énumérable de manière croissante par une fonction $\mathtt{c}$-fondamentale ;

— la faible $\mathtt{c}$-récursivité : le fait d'être $\mathtt{c}$-énumérable et co-$\mathtt{c}$-énumérable ;

— la forte $\mathtt{c}$-récursivité : le fait d'être fortement $\mathtt{c}$-énumérable et co-fortement-$\mathtt{c}$-énumérable ;

— et finalement la χ-$\mathtt{c}$-énumérabilité : le fait d'avoir une fonction caractéristique $\mathtt{c}$-fondamentale.

Les relations entre ces différentes notions ont été étudiées dans le chapitre précédent.

Les classes fondamentales satisfont le théorème s_n^m, un théorème à la Myhill, un théorème d'isomorphisme à la Rogers, et viennent avec tout un lot de relativisations de notions de réductions entre sous-ensembles de $\mathcal{D}$, définissant plusieurs notions de degrés Turing et une structure riche sur ceux-ci.

Les sous-calculabilités héritent de l'énumération des fonctions fondamentales et donc de toutes ces propriétés qui en découlent. En plus de cela, elles rajoutent une nouvelle énumération de fonctions dites C-partielles définie en fonction des énumérations précédentes.

3.1.2 Fonctions C-partielles

En calculabilité classique, une des caractérisations des fonctions récursives partielles est d'avoir un graphe récursivement énumérable. En partant de ce principe, nous savons que l'ensemble des ensembles récursivement énumérables, compte entre autre tous les graphes de toutes les fonctions récursives partielles.

Dans le cas des sous-calculabilités, nous allons simplement suivre le chemin inverse, en partant des ensembles C-énumérables, et en s'en servant pour définir la classe des fonctions C-partielles.

Soit $d \in \mathcal{D}$. $\mathbf{W}_d$ est un ensemble d'éléments de $\mathcal{D}$ qui peut être vu, via la bijection $\langle, \cdot, \cdot\rangle : \mathcal{D} \times \mathcal{D} \mapsto \mathcal{D}$ comme un ensemble de paires $(\langle e_i, f_i\rangle)_{i\in\mathcal{D}}$. Lorsque nous énumérons un tel ensemble $\mathbf{W}_d$, nous pouvons veiller à ce que chacun des $e_i, i \in \mathcal{D}$ énumérés soient différents.

Pour ce faire, il suffit de considérer que l'énumération consiste uniquement en le préfixe sans doublon :

$$\mathcal{G}_d = \{\langle e_i, f_i\rangle \in \mathbf{W}_d : \forall j < k \leqslant i, e_j \neq e_k\}\,.$$

Cet ensemble $\mathcal{G}_d$ peut être calculé de manière récursive primitive depuis la fonction C-fondamentale $\mathbf{\Phi}_d^{\mathsf{C}}$ énumérant $\mathbf{W}_d$.

Bien sûr, pour chaque $d \in \mathcal{D}$, $\mathcal{G}_d$ identifie le graphe d'une fonction partielle. C'est cette notion-là de fonction descriptible par une fonction C-fondamentale que nous allons utiliser.

Nous venons donc de définir une nouvelle énumération de fonctions dites C-partielles, que nous noterons par similitude avec les fonctions récursives partielles usuelles :

$$(\varphi_d^{\mathsf{C}})_{d\in\mathcal{D}}$$

Définition 3.1.1 **Fonctions C-partielles.** *Pour tout $d \in \mathcal{D}$, φ_d^{C} désigne la fonction partielle définie par le graphe :*

$$\mathcal{G}_d = \{\langle e_i, f_i \rangle \in \mathbf{W}_d : \forall j < i, k \leqslant i, e_j \neq e_k\}.$$

Où les $(\langle e_i, f_i \rangle)_{i \in \mathcal{D}}$ *sont les éléments de* $\mathbf{W}_d$ *ordonnés par leur énumération canonique* Φ^{C}_d*, voir Définition 2.3.4.*

Note 3.1.2 *Il existe d'autres moyens de définir des fonctions partielles à partir des fonctions totales. L'un d'entre eux serait par exemple de considérer les graphes énumérables de manière croissante.*

Cependant, cette classe n'offre pas vraiment d'intérêt étant donné que nous pouvons étendre toutes ces fonctions pour en faire des fonctions totales. Et de fait, elles sont déjà présentes dans les fonctions C*-fondamentales.*

Pour s'en assurer, considérons f *une fonction* C*-fondamentale énumérant de manière croissante le graphe* G *d'une fonction partielle. La fonction* g *suivante est bien* C*-fondamentale, permet de déduire* f *facilement, et est totale :*

$$g : n \mapsto \begin{cases} 0 & \textit{si } \forall m \leqslant n, \pi^2_1(f(m)) \neq n, \\ \pi^2_2(f(m)) + 1 & \textit{s'il existe un } m \leqslant n \textit{ (alors unique) tel que } \pi^2_1(f(m)) = n. \end{cases}$$

Une troisième méthode envisageable serait, étant donné une fonction φ_e *récursive et une fonction* f C*-fondamentale, de considérer la fonction* ψ *partielle suivante*

$$\psi : n \mapsto \begin{cases} \uparrow & \textit{si } \varphi_{e,f(n)}(n) \uparrow \\ \varphi_e(n) & \textit{sinon.} \end{cases}$$

Où la notation $\varphi_{e,s}(n)$ *signifie* l'exécution de la fonction récursive d'indice e sur l'entrée n pendant s étapes.

Une fois de plus, en remplaçant $\uparrow$ *par* 0 *et* $\varphi_e(n)$ *par* $\varphi_e(n)+1$*, la fonction obtenue est* C*-fondamentale totale, permet de retrouver* ψ *et contient en plus les informations* d'arrêt *de cette fonction.*

Dans la suite du chapitre, nous allons tenter de décrire au mieux les objets présents et absents de cette énumération de fonctions, et la structure qu'il est possible de retrouver dans cette classe qui ne semble pourtant pas *bien* close.

3.1.3 Premières constatations

Les fonctions ainsi définies par C-énumérabilité de leur graphe forment une classe de fonctions récursives partielles assez inhabituelle. En effet, les schémas de clôture ne sont pas définis sur les fonctions elles-mêmes mais sur leurs fonctions énumérantes associées. Afin de mieux comprendre les clôtures induites sur cette nouvelle classe de fonctions partielles, nous allons chercher à y retrouver les propriétés classiques. Nous allons voir qu'en fait, ces propriétés sont parfois différentes de celles découvertes sur les classes fondamentales. Cette nouvelle classe est comme un second axe de lecture pour les fonctions fondamentales. Nous allons voir comment ces deux axes sont tout à fait complémentaires.

Le premier héritage des fonctions fondamentales que nous remarquons est l'inclusion de c dans la classe des fonctions c-partielles. En effet, toute fonction c-fondamentale est également c-partielle.

Proposition 3.1.3

Inclusion des fonctions fondamentales. *Soit c une classe fondamentale. Si f est une fonction de c. Alors f est également une fonction c-partielle.*

Démonstration. Pour montrer que chaque fonction c-fondamentale est également une fonction c-partielle, nous allons fixer une telle fonction et montrer comment la transformer en une fonction énumérant le graphe d'une c-partielle identique. Les opérations que nous allons utiliser pour effectuer la transformation sont capturées par les propriétés de clôture de la classe fondamentale, donc la nouvelle fonction créée sera bien c-fondamentale et son image le graphe d'une fonction c-partielle.

Soit $f : x \mapsto f(x) \in \mathsf{c}$. Alors, $g : x \mapsto \langle x, f(x)\rangle$ est injective par composition de $\langle\cdot,\cdot\rangle$ et de id : $x \mapsto x$ toutes deux injectives.

L'image par g injective de $\mathcal{D}$ est le graphe de f, ce qui par définition des fonctions c-partielles nous permet de conclure que f est également c-partielle. $\square$

Inversement, nous pouvons vérifier qu'il existe des fonctions totales c-partielles qui ne soient pas c-fondamentales. Il y en a même une quantité infinie dénombrable puisque la méthode que nous allons exhiber permet pour chaque fonction récursive totale $f \notin \mathsf{c}$ de construire une fonction $g \notin \mathsf{c}$ qui soit c-partielle.

Proposition 3.1.4

Soit c une classe fondamentale. Il existe des fonctions totales qui sont c-partielles et non c-fondamentale.

Démonstration. Soit f une fonction récursive non c-fondamentale. L'exemple canonique est $\mathbf{BB}^{\Phi}_{\mathsf{c}}$ (ou également Ack pour p).

f n'est pas dans c. Cependant, comme f est récursive, son graphe est récursivement énumérable (il est même récursif puisque f est totale).

Nous savons via le Théorème 2.3.9 que si un ensemble *r.e.* contient un ensemble c-énumérable, alors il est lui-même c-énumérable. Nous allons donc transformer le graphe de f : $\mathcal{G}_f$, en un graphe $\mathcal{G}_g$ qui sera cette fois c-énumérable.

Pour ce faire, il suffit de définir g comme suit :

$$g : \begin{cases} 2d & \mapsto \quad f(d) \\ 2d+1 & \mapsto \quad 0 \end{cases}$$

Cette fonction ne peut pas être dans $\mathtt{C}$, car sinon, $g \circ (n \mapsto 2n) = f$ serait également dans $\mathtt{C}$, ce qui est en contradiction avec notre hypothèse de départ.

Donc, g n'est pas dans $\mathtt{C}$. Il est cependant facile de montrer que g est $\mathtt{C}$-partielle.

En effet, l'ensemble $A = \{\langle 2d+1, 0\rangle : d \in \mathcal{D}\}$ est un sous-ensemble $\mathtt{C}$-énumérable de $\mathcal{G}_g$ qui est, comme nous l'avons vu, récursivement énumérable.

□

La proposition précédente semble nous entraîner vers l'idée que toutes les fonctions sont présentes dans notre sous-calculabilité. Cependant, il n'en est rien. Pour s'en convaincre, il suffit de se concentrer sur les fonctions strictement croissantes. Nous allons montrer dans la proposition suivante que chaque fonction totale strictement croissante $\mathtt{C}$-partielle est en fait dominée par une fonction $\mathtt{C}$-fondamentale. Comme nous savons que la croissance des fonctions fondamentales est fortement limitée, nous en déduisons qu'il en va de même pour les fonctions croissantes $\mathtt{C}$-partielles.

Pour ce faire, nous avons besoin de la proposition suivante, bien connue en récursivité classique :

Proposition 3.1.5

Extraction d'un sous-ensemble récursif. *Soit* $\mathtt{C}$ *une classe fondamentale. De tout ensemble* $\mathtt{C}$*-énumérable infini* A *il est possible d'extraire un ensemble fortement* $\mathtt{C}$*-récursif infini (et donc* χ*-*$\mathtt{C}$*-récursif et faiblement* $\mathtt{C}$*-récursif)* E.

Démonstration. Soit $f \in \mathtt{C}$ une fonction $\mathtt{C}$-fondamentale injective témoignant de la $\mathtt{C}$-énumérabilité de A.

Nous cherchons à calculer, via une fonction $\mathtt{C}$-fondamentale h et pour chaque élément $d \in \mathcal{D}$, un élément $h(d+1) > h(d)$. Défini de cette manière, le problème semble nécessiter l'utilisation du point fixe pour sa résolution.

En réalité, il n'en est rien. Considérons la fonction suivante :

$$g : d \mapsto \max\{f(i) : i \leqslant d+1\}$$

g calcule le max de $d+2$ éléments différents de A, parmi eux, il en est nécessairement un qui soit plus grand que $f(d)$. g satisfait donc $\forall x \in \mathcal{D}, g(f(x)) > f(x)$

Il ne reste alors plus qu'à composer cette fonction. En posant que $g^{(0)} = x \mapsto x$, nous définissons la fonction h de la manière suivante :

$h : d \mapsto g^{(d)}(f(0))$

$h(0)$ vaut $g^0(f(0)) = f(0)$, qui sera le premier élément de notre énumération. $h(0)$ est trivialement un élément de A.

Pour $d \in \mathcal{D}$, $h(d{+}1) = g^{d+1}(f(0)) > g^{(d)}(f(0))$. De plus, nous vérifions facilement que $h(d)$ est un élément de A.

h ayant été définie uniquement en utilisant f et des schémas de récursion de $\mathtt{C}$, elle est un bon candidat pour une énumération croissante d'éléments de A.

Cependant, si nous nous arrêtons là, l'image de h sera simplement fortement C-énumérable (et donc χ-C-énumérable), mais pas nécessairement co-fortement-C-énumérable.

En effet, même si nous disposons d'une énumération croissante d'un ensemble, nous n'avons pas nécessairement pour autant une énumération de son complémentaire, comme nous l'avons vu dans l'étude de l'image de la fonction d'Ackermann (Propriété 2.4.11), et le complémentaire de l'image d'Ackermann nous aide à en concevoir la raison. C'est un ensemble très dense, dont les *trous* ne se suivent jamais. Ainsi, essayer d'énumérer son complémentaire consiste à en trouver les trous, ce qui nécessite de rechercher un nombre non borné d'éléments, chose qui nous est interdite dans le contexte des classes fondamentales.

Pour pallier à ce problème, il suffit de nous assurer que deux éléments de notre ensemble ne se suivent jamais. Pour ce faire, considérons la fonction $h' = h \circ (d \mapsto 2d)$.

Notons enfin E l'image de h' qui satisfait bien tous les critères de récursivité que nous prétendions obtenir.

□

En utilisant le lemme précédent sur le graphe d'une fonction partielle, le résultat de domination suivant découle immédiatement :

Proposition 3.1.6

Domination par les fondamentales. *Soit* C *une classe* C*-fondamentale. Si f une fonction totale strictement croissante* C*-énumérable, alors il existe g une fonction totale* C*-fondamentale strictement croissante supérieure à f en tous points.*

Démonstration. En utilisant la construction de la Proposition 3.1.5, nous obtenons une fonction h produisant un sous-ensemble strictement croissant du graphe de f.

Cette fonction h est totale et strictement croissante par construction. De plus, comme $\langle x, y\rangle$ est plus grand que x et que y, h est plus grande que f.

$g = h$ est un témoin de l'existence d'une fonction dominant f dans C.

□

Note 3.1.7 *On remarquera que si f n'est pas strictement croissante,*

$$f' : n \mapsto n + \sum_{i \leqslant n} f(i)$$

l'est.

Ici, au travers de quelques résultats choisis, nous avons essayé de donner un aperçu de la population présente dans la classe des fonctions C-partielles. Dans le paragraphe suivant, nous allons tenter d'identifier la part de la structure de l'énumération qui peut être héritée des classes fondamentales, la part qui ne peut pas être héritée, et la part nouvelle dont, par exemple, le point fixe partiel à la Kleene.

3.2 Structure de calculabilité partielle

Nous venons de voir que nos sous-calculabilités possédaient toutes les fonctions fondamentales, des fonctions récursives arbitrairement complexes, nous avons entr'aperçu le fait qu'elles n'allaient pas être stables par composition (ce résultat peut se déduire de la Proposition 3.1.4) et pourtant nous allons voir que cette classe peut-être vue comme une approximation de la calculabilité classique. Cette approximation est même un terrain de jeu qui permet de manipuler, construire et vérifier l'existence de fonctions calculables prises dans les strates basses, en terme de complexité de leur définition, de la calculabilité.

Malgré tous leurs fonctions manquantes, les sous-calculabilités vont satisfaire un théorème d'isomorphisme à la Rogers et un théorème de la récursion à la Kleene, elles vont avoir un ensemble diagonal bien défini et Turing complet, elles vont nous permettre d'étendre nos réductions $\mathtt{C}$-fondamentales aux fonctions partielles, et enfin, ces nouvelles classes vont nous permettre de visualiser un peu mieux les liens entre la théorie de la preuve et calculabilité classique, ce qui était notre objectif initial.

3.2.1 Théorème *à la* Kleene partiel

Le second théorème de la récursion de Kleene est bien souvent l'apanage de la calculabilité classique. Il permet une telle souplesse dans la définition des fonctions, qu'il est difficile d'imaginer des classes de fonctions récursives le vérifiant mais ne possédant pas toutes les fonctions calculables. En fait, il est impossible de construire une classe de fonction récursive vérifiant le point fixe de Kleene, le théorème s_n^m et la présence d'une fonction universelle, sans définir une classe regroupant au moins toutes les fonctions récursives. Dans notre cas, c'est le théorème s_n^m qui fait défaut. Et c'est pourquoi ces classes de fonctions $\mathtt{C}$-partielles peuvent jouer leur rôle d'approximation, tout en nous faisant bénéficier du point fixe.

Ce théorème de point fixe n'est pas tout à fait le même que dans le cas classique. Il possède deux restrictions dont nous pouvons cependant nous accommoder sans perte de clarté dans les preuves l'utilisant.

Théorème 3.2.1

Théorème de récursion à la Kleene partiel. *Soit* $\mathtt{C}$ *une classe fondamentale,* h *une fonction* $\mathtt{C}$*-partielle, définie sur un ensemble infini* $\mathtt{C}$*-énumérable et co-*r.e. A *et* f *une fonction* $\mathtt{C}$*-fondamentale.*
Alors, il existe un $d \in \mathcal{D}$ *tel que* $(\varphi_d^{\mathtt{C}})_{\restriction \overline{A}} \cong \left(\varphi_{f(d)}^{\mathtt{C}}\right)_{\restriction \overline{A}}$ *et* $(\varphi_d^{\mathtt{C}})_{\restriction A} \cong h$.

Démonstration. Soit $f \in \mathtt{C}$ une fonction $\mathtt{C}$-fondamentale d'indice i_f, et a un indice pour la fonction h, $\mathtt{C}$-partielle et de domaine A $\mathtt{C}$-énumérable et co-*r.e.*.

Dans cette preuve, nous allons suivre un raisonnement similaire à celle du Théorème 2.3.9. Une fois de plus, la difficulté est que nous devons être en mesure de fournir un nouvel élément en un nombre d'étapes borné.

Dans le Théorème 2.3.9, pour prouver qu'un ensemble $\mathbf{W}_i$ *r.e.* contenait un sous-ensemble infini $\mathbf{W}_e^{\mathtt{C}}$ $\mathtt{C}$-*r.e.*, nous utilisions ce sous-ensemble $\mathbf{W}_e^{\mathtt{C}}$ comme source d'inspiration en cas de besoin et nous énumérions notre ensemble $\mathbf{W}_i$ le reste du temps.

Ici, pour que notre point fixe soit une fonction $\mathtt{C}$-partielle, il faut que son graphe soit $\mathtt{C}$-énumérable. Or, nous savons par avance que son graphe contiendra le graphe de h, qui est lui-même infini et $\mathtt{C}$-énumérable.

Il reste à prouver deux éléments importants. Premièrement nous devons montrer que la génération de l'indice de la fonction de point fixe est récursive primitive. Et deuxièmement, nous devons montrer que notre point fixe est effectivement récursivement énumérable.

Nous prouverons ces deux propriétés simultanément en exhibant une fonction possible réalisant le point fixe.

Considérons la fonction suivante :

$$u \mapsto \psi_x(u) = \begin{cases} \varphi_a^{\mathtt{C}}(u) & \text{si } u \in A \\ \varphi_{\varphi_x^{\mathtt{C}}(x)}^{\mathtt{C}}(u) & \text{sinon.} \end{cases}$$

Cette fonction est $\mathtt{C}$-partielle car elle est récursive et que c'est une extension d'une fonction $\mathtt{C}$-partielle de domaine infini (par application du Théorème 2.3.9).

Comme son énumération est définie en utilisant des primitives et des fonctions $\mathtt{C}$-fondamentales et via des schémas de composition d'indices, eux-même $\mathtt{C}$-fondamentaux, ψ_x est d'indice calculable par une fonction $\mathtt{C}$-fondamentale d_a, d'indice i_{d_a}, depuis le paramètre $x \in \mathcal{D}$.

Pour la suite, nous reprenons la méthode de preuve usuelle du théorème de la récursion. Posons $e_a = \mathbf{comp}(i_f, i_{d_a})$ un $\mathtt{C}$-indice pour $f \circ d_a$.

Remarquons alors que pour tout $u \in \mathcal{D}$:

$$u \mapsto \varphi_{d_a(e_a)}^{\mathtt{C}}(u) \cong \begin{cases} \varphi_a^{\mathtt{C}}(u) & \text{si } u \in A \\ \varphi_{\varphi_{e_a}^{\mathtt{C}}(e_a)}^{\mathtt{C}}(u) \cong \varphi_{f \circ d_a(e_a)}^{\mathtt{C}}(u) & \text{sinon.} \end{cases}$$

Il ne reste alors plus qu'à choisir $n = d_a(e_a)$, et nous avons que $\varphi_n^{\mathtt{C}}$ et $\varphi_{f(n)}^{\mathtt{C}}$ correspondent sur $\overline{A}$ et que $\varphi_n^{\mathtt{C}}$ et $\varphi_a^{\mathtt{C}} \cong h$ correspondent sur A. n est donc un “point fixe partiel” $\mathtt{C}$-partielle pour la fonction $\mathtt{C}$-fondamentale f.

$\square$

Ce théorème ne nous permet donc pas réellement de trouver un point fixe d'une fonction $\mathtt{C}$-fondamentale f, mais simplement un indice e de fonction qui se comportera, sur un ensemble presque arbitrairement choisi, de la même manière que son image par f. Dans certains cas, nous verrons que par une construction astucieuse, il est possible de faire coïncider les fonctions d'indices e et $f(e)$ sur tout leur domaine, ce qui nous fournira alors un véritable point fixe pour f.

3.2.2 Propriétés de l'énumération canonique

Avant de pouvoir généraliser la notion de système d'indices acceptable aux sous-calculabilités, il nous faut vérifier que l'énumération canonique $\varphi^{\mathtt{C}}_{\cdot}$ vérifie les propriétés nécessaires que sont la paramétrisation et le bourrage.

Proposition 3.2.2

Paramétrisation pour les fonctions $\mathtt{C}$-partielles. *Soit $\mathtt{C}$ une classe fondamentale. Il existe un indice pour $\varphi^{\mathtt{C}}_{\cdot}$,* param_0 *tel que pour tout $i_0, i_1, j \in \mathcal{D}$,*

$$\varphi^{\mathtt{C}}_{\Phi^{\mathtt{C}}_{\mathrm{param}_\psi}\langle i_0,i_1,\mathbf{j}\rangle}(e) \cong \begin{cases} \varphi^{\mathtt{C}}_{i_0} & si\ \Phi^{\mathtt{C}}_{\mathbf{j}}(e) = 0, \\ \varphi^{\mathtt{C}}_{i_1} & sinon. \end{cases}$$

Démonstration. Ce résultat est trivial étant donné que dans l'énumération canonique $\varphi^{\mathtt{C}}_{\cdot}$, i_0 et i_1 sont simplement des indices de fonctions $\mathtt{C}$-fondamentales. Autrement dit, nous appliquons le branchement conditionnel présent dans les classes $\mathtt{C}$-fondamentales aux fonctions $\mathtt{C}$-fondamentales énumérant les graphes des fonctions $\mathtt{C}$-partielles. □

Proposition 3.2.3

Lemme du bourrage. *Soit $\mathtt{C}$ une classe fondamentale. Il existe une fonction $\mathfrak{p}$ $\mathtt{C}$-fondamentale et donc $\mathtt{C}$-énumérable permettant pour tout $e \in \mathcal{D}$ de générer un indice $e' \in \mathcal{D}$ tel que $e' > e$ et $\varphi^{\mathtt{C}}_e \simeq \varphi^{\mathtt{C}}_{e'}$.*

Démonstration. Nous voulons montrer qu'étant donné h une fonction $\mathtt{C}$-énumérable, et B un ensemble fini d'indices pour cette fonction, nous pouvons de manière $\mathtt{C}$-fondamentale trouver un indice pour h qui n'est pas dans B.

En utilisant la paramétrisation démontrée dans la Proposition 3.2.2, nous définissons une fonction $\mathtt{C}$-fondamentale f telle que :

$$\varphi^{\mathtt{C}}_{f(e)} \cong \begin{cases} h & \text{si } e \notin B, \\ \mathbf{0} & \text{sinon.} \end{cases}$$

D'après la Proposition 3.1.5, il nous est possible d'extraire du graphe $\mathtt{C}$-énumérable de h un sous-ensemble fortement $\mathtt{C}$-récursif qui sera le graphe d'une fonction $\mathtt{C}$-partielle h_0. Notons A_0 le domaine fortement $\mathtt{C}$-récursif de cette fonction.

Les conditions sont alors réunies pour nous permettre d'appliquer le Théorème de la récursion partiel à la Kleene pour $\varphi^{\mathtt{C}}_{\cdot}$.

Nous obtenons donc un e satisfaisant le demi-point fixe suivant :

$$\varphi^{\mathtt{C}}_e|_{A_0} \cong h_0|_{A_0} \cong h|_{A_0} \text{ et } \varphi^{\mathtt{C}}_e|_{\overline{A_0}} \cong \varphi^{\mathtt{C}}_{f(e)}|_{\overline{A_0}}.$$

Deux cas sont alors possibles :

— Si $e \notin B$, nous sommes dans le premier cas de la définition de f, et donc

$$\varphi^{\mathtt{C}}_{e}|_{\overline{A_0}} \cong \varphi^{\mathtt{C}}_{f(e)}|_{\overline{A_0}} \cong h|_{\overline{A_0}}.$$

e est alors un indice pour h qui n'est pas présent dans B. De plus, il est possible de calculer de manière $\mathtt{C}$-fondamentale cet indice e à partir de B et d'un indice de h. Ce qui termine la preuve de ce cas.

— Si $e \in B$, alors nous sommes dans le second cas de la définition de f et $\varphi^{\mathtt{C}}_{e}|_{\overline{A_0}}$ est la fonction constante nulle $\mathbf{0}|_{\overline{A_0}}$.
Mais comme e est un élément de B, cela signifie que c'est un indice de h, et par conséquent $h|_{\overline{A_0}} \cong \mathbf{0}|_{\overline{A_0}}$.
Afin de parvenir à trouver un nouvel indice pour h, il va nous falloir définir une nouvelle fonction $\mathtt{C}$-partielle g par paramétrisation :

$$\varphi^{\mathtt{C}}_{g(a)} \cong \begin{cases} \mathbf{1} & \text{si } a \in B, \\ \mathbf{0} & \text{sinon.} \end{cases}$$

Nous devons maintenant appliquer à nouveau le théorème du demi-point fixe, ce qui nous permet d'obtenir un a tel que :

$$\varphi^{\mathtt{C}}_{a}|_{\overline{A_0}} \cong \varphi^{\mathtt{C}}_{g(a)}|_{\overline{A_0}}.$$

La partie est maintenant gagnée étant donné que a ne peut pas être un élément de B, sans quoi $h|_{\overline{A_0}}$ serait la fonction constante égale à 1 : $\mathbf{1}|_{\overline{A_0}}$, alors que l'application précédente du point fixe nous a indiqué que c'était la fonction constante égale à 0 : $\mathbf{0}|_{\overline{A_0}}$.

La seule possibilité est donc le deuxième cas, c'est à dire $a \notin B$ et

$$\varphi^{\mathtt{C}}_{a}|_{\overline{A_0}} \cong \mathbf{0}|_{\overline{A_0}}.$$

Par conséquent, a est un indice pour h qui n'est pas dans B, et qui est calculable de manière $\mathtt{C}$-fondamentale à partir de B. Ce qui conclut la preuve. □

3.2.3 Isomorphisme *à la* Rogers pour les sous-calculabilités

Afin de confirmer la légitimité de l'énumération des fonctions $\mathtt{C}$-partielles, nous allons poursuivre notre étude des propriétés classiques et, en particulier, nous allons nous intéresser une fois de plus au théorème d'isomorphisme *à la* Rogers. Ce théorème, bien plus qu'une simple propriété algébrique confortable d'un point de vue technique et élégante d'un point de vue abstrait, nous offre en fait l'assurance que l'énumération canonique n'est pas une énumération *ad-hoc* cachant ses bonnes propriétés dans la complexité de sa définition et de sa construction mais qu'au contraire elle est interchangeable avec n'importe quelle énumération de la classe en isomorphisme $\mathtt{C}$-calculable à partir de celle-ci.

Dans le cas des classes $\mathtt{C}$-fondamentales, nous avions défini la notion de système d'indices acceptable en nous intéressant à l'énumération de nos fonctions présentes dans $\mathtt{C}$. Dans le cas des sous-calculabilités, nous avons maintenant deux énumérations à prendre en compte. Cependant, contrairement à ce qu'ont pu laisser entendre les définitions et preuves faites jusqu'à présent, ces deux énumérations ne sont pas nécessairement appareillées, bien que la composition des isomorphismes nous assure qu'il est toujours possible de revenir à des paires corrélées, c'est à dire dont l'indice pour $\varphi^{\mathtt{C}}_{\cdot}$ correspond à la fonction énumératrice dans $\boldsymbol{\Phi}^{\mathtt{C}}_{\cdot}$.

Définition 3.2.4 **Généralisation de la notion de système acceptable.** *Un système d'indices acceptable pour sous-calculabilité est une paire* $\left(\boldsymbol{\Psi}^{\mathtt{C}}_{\cdot}, \psi^{\mathtt{C}}_{\cdot}\right)$ *d'énumérations, respectivement pour les fonctions $\mathtt{C}$-fondamentales et pour les fonctions $\mathtt{C}$-partielles, telle que $\boldsymbol{\Psi}^{\mathtt{C}}_{\cdot}$ soit acceptable pour la classe fondamentale au sens de la Définition 1.3.19 et qu'il existe deux fonctions $\mathtt{C}$-fondamentales f et g vérifiant :*

$$\forall e \in \mathcal{D}, \quad \psi^{\mathtt{C}}_{e} \;\cong\; \varphi^{\mathtt{C}}_{f(e)} \quad \textit{et} \quad \varphi^{\mathtt{C}}_{e} \;\cong\; \psi^{\mathtt{C}}_{g(e)}.$$

Ce faisant, nous définissons à nouveau une classe d'équivalence contenant la paire formée par les énumérations canoniques $\boldsymbol{\Phi}^{\mathtt{C}}_{\cdot}$ et $\varphi^{\mathtt{C}}_{\cdot}$.

Nous avons déjà prouvé que la paramétrisation était effective dans les classes $\mathtt{C}$-fondamentales, qu'elle se transmettait par héritage à tous les systèmes d'indices acceptables pour les $\mathtt{C}$-fondamentales. Nous avons également montré qu'elle était effective pour les classes $\mathtt{C}$-partielles, il reste à montrer qu'elle se transmet également par héritage chez les classes $\mathtt{C}$-partielles. Nous devrons également expliquer pourquoi il en va de même pour la propriété de bourrage. Ces outils en main, nous pourrons vérifier la validité du Théorème de Rogers pour les sous-calculabilités.

Proposition 3.2.5

Héritage de la paramétrisation et du bourrage. *Soit $\mathtt{C}$ une classe fondamentale et $\left(\boldsymbol{\Psi}^{\mathtt{C}}_{\cdot}, \psi^{\mathtt{C}}_{\cdot}\right)$ une énumération acceptable des fonctions $\mathtt{C}$-fondamentales et $\mathtt{C}$-partielles de la $\mathtt{C}$-calculabilité engendrée. Alors $\psi^{\mathtt{C}}_{\cdot}$ possède les propriétés de paramétrisation et de bourrage.*

Démonstration. Ayant déjà fait la plus grande partie de cette preuve en amont, dans ce chapitre et dans le précédent, nous allons discuter ces résultats afin de mettre en avant les arguments les démontrant.

Paramétrisation La fonction de paramétrisation est une fonction $\mathtt{C}$-fondamentale. Il est donc possible de lui appliquer les fonctions de changements d'indice f et g sans sortir de la classe des $\mathtt{C}$-fondamentales, grâce à la propriété de clôture par composition de celle-ci. Ainsi, à partir du moment où le changement d'indice à faire est $\mathtt{C}$-fondamental, ce qui est le cas dans les systèmes d'indices acceptables, la propriété de paramétrisation est directement issue de la structure des fonctions $\mathtt{C}$-fondamentales.

Bourrage La preuve du bourrage pour les sous-calculabilités faite dans la Proposition 3.2.3 est liée à une énumération, $\varphi^{\mathtt{C}}_{\cdot}$ dans le cas du lemme en question, mais non pas propre à celle-ci. Du moment que les changements d'indice sont possibles, ce qui est le cas dans le contexte des systèmes d'indices acceptables, la preuve fonctionne avec n'importe quelle énumération (qui sera récursive puisque les changements d'indices seront calculables). Il suffit alors pour prouver l'héritage du bourrage de refaire la preuve à l'identique, en remplaçant $\varphi^{\mathtt{C}}_{\cdot}$ par $\psi^{\mathtt{C}}_{\cdot}$.

□

Théorème 3.2.6

Rogers pour les sous-calculabilités. *Soit* $\mathtt{C}$ *une classe fondamentale, considérons sa sous-calculabilité engendrée. Pour tout système d'indices acceptable de sous-calculabilité* $(\boldsymbol{\Psi}^{\mathtt{C}}_{\cdot}, \psi^{\mathtt{C}}_{\cdot})$*, il existe permutation* h *de* $\mathcal{D}$ $\mathtt{C}$*-fondamentale telle que :*

$$\forall e \in \mathcal{D}, \quad \varphi^{\mathtt{C}}_{e} \;\cong\; \psi^{\mathtt{C}}_{h(e)}.$$

De plus, le Théorème 2.2.5 s'applique pour les énumérations acceptables de la classe fondamentale.

Démonstration. La preuve est identique à celle du théorème 2.2.5, en remplaçant les énumérations $\boldsymbol{\Phi}^{\mathtt{C}}_{\cdot}$ et $\boldsymbol{\Psi}^{\mathtt{C}}_{\cdot}$ par $\varphi^{\mathtt{C}}_{\cdot}$ et $\psi^{\mathtt{C}}_{\cdot}$. □

Ainsi, la notion de système d'indices acceptable reste pertinente dans le contexte des sous-calculabilités.

3.2.4 Cas de la projection et de la composition

Comme nous l'avons remarqué, une propriété de clôture habituellement naturelle va manquer à la classe des fonctions $\mathtt{C}$-partielles. Par essence, la classe des fonctions $\mathtt{C}$-partielles n'est pas stable par projection. C'est à dire qu'une fonction définie sur un ensemble ne le sera pas nécessairement sur un de ses sous-ensembles, de la même manière qu'un sous-ensemble E d'un ensemble récursif R n'est pas nécessairement récursif lui-même.

Les conséquences en seront que ni la composition, ni l'application de la fonction de manipulation d'arité s^m_n ne seront possibles en règle générale.

Théorème 3.2.7

Absence de théorème s-m-n pour les fonctions $\mathtt{C}$-partielles. $f : x, y \mapsto \langle x, \mathsf{BB}^{\Phi}_{\mathtt{C}}(y) \rangle$ *est* $\mathtt{C}$*-partielle mais* $\mathsf{BB}^{\Phi}_{\mathtt{C}}$ *ne l'est pas.*

Démonstration. Supposons que $\mathsf{BB}^{\Phi}_{\mathtt{C}}$ soit une fonction fonction $\mathtt{C}$-partielle. On sait qu'elle est totale et strictement croissante. Donc, par la Proposition 3.1.6, on sait qu'il existerait une fonction $\mathtt{C}$-fondamentale la dominant. Or, une fonction dominant le Castor Affairé permet de calculer celui-ci. Ce qui impliquerait que $\mathsf{BB}^{\Phi}_{\mathtt{C}}$ est $\mathtt{C}$-fondamentale, ce que nous avons prouvé être faux. Donc $\mathsf{BB}^{\Phi}_{\mathtt{C}}$ n'est pas $\mathtt{C}$-partielle.

La première partie du résultat s'obtient par le Théorème 2.3.9 en remarquant que le graphe est *r.e.* et contient un sous-ensemble trivial $\mathtt{C}$-énumérable. □

Cependant, bien que nous ne puissions pas directement utiliser le théorème s^m_n pour les fonctions $\mathtt{C}$-partielles, certains corollaires de la version $\mathtt{C}$-fondamentale peuvent toujours nous être utiles. En particulier, comme à application du théorème d'isomorphisme à la Rogers pour les classes fondamentales près, les fonctions partielles sont énumérées par les fonctions fondamentales, il est possible d'utiliser s^m_n pour construire nos fonctions énumérantes.

Nous pouvons nous assurer facilement de la non-stabilité de la classe des fonctions $\mathtt{C}$-partielles par composition en considérant le résultat suivant :

Théorème 3.2.8

Instabilité par composition. *Soit* $\mathtt{C}$ *une classe fondamentale. Il existe* f *et* g $\mathtt{C}$*-partielles telles que l'image de* g *soit incluse dans le domaine de* f*, mais que* $f \circ g$ *ne soit pas* $\mathtt{C}$*-partielle.*

Démonstration. Cette fois, nous allons nous appuyer sur le Théorème 2.3.9 pour créer une fonction $\mathtt{C}$-énumérable qui ne sera pas bien définie sur un sous-ensemble de son domaine de définition.

Soit f telle que $f : 2x \mapsto \mathsf{BB}^{\Phi}_{\mathtt{C}}(x)$ et $f : 2x + 1 \mapsto 0$ et g tel que $g : x \mapsto 2x$.

f et g sont toutes deux $\mathtt{C}$-partielles. En effet, f est récursive et contient un sous-ensemble $\mathtt{C}$-énumérable (pour les x impairs), et g est $\mathtt{C}$-fondamentale et donc $\mathtt{C}$-partielle.

Or, $f \circ g = \mathsf{BB}^{\Phi}_{\mathtt{C}}$, qui n'est pas $\mathtt{C}$-partielle.

□

Les deux résultats précédents, nous permettent de mieux comprendre les résultats suivants : la présence dans les sous-calculabilités de fonctions universelles pour les $\mathtt{C}$-partielles, et même de fonctions universelles Turing.

Proposition 3.2.9

Il existe une fonction $\mathsf{U}_{T,\mathsf{p}}$ p*-partielle qui soit universelle pour la classe de toutes les fonctions récursives partielles.*

Démonstration. Par application du Théorème 2.3.9, légitimée par l'inclusion de sous-ensembles p-énumérables comme le graphe de la fonction constante nulle, dans le graphe *r.e.* de la fonction universelle récursive, nous obtenons directement que ce graphe est p-énumérable.

□

Proposition 3.2.10

Soit c *une classe fondamentale. Il existe une fonction* U_c p*-partielle qui soit universelle pour la classe des fonctions* c*-partielles.*

Démonstration. La démonstration pour le cas de la fonction universelle pour c est similaire, en remplaçant $\mathbf{const}_T$ par **const** et en utilisant le graphe de la fonction c-universelle à la place de la fonction universelle classique.

Il nous faut cependant vérifier que le graphe de la fonction c-universelle est bien *r.e.*. C'est effectivement le cas puisque notre sous-calculabilité est fondée sur une classe fondamentale c qui est une classe énumérable récursive.

□

3.3 Retour sur les réductions

Pour chacune des quatre notions récursivités sur les ensembles dont nous disposons, nous avons défini dans le chapitre précédent au parapgraphe 2.5 une notion de réduction *à la* Turing qui lui est associée. Nous avons également en vue du théorème de Myhill défini la c-1-réduction et la c-*many-one*-réduction.

3.3.1 Variantes des c-réductions

Rappelons d'abord les principales réductions vues précédemment. Étant donné une classe fondamentale c et deux ensembles A et B,

— A est c-énumérable en B si toute énumération de B permet de c-calculer une énumération de A ($\mathfrak{e}_A \in \mathsf{c}[\mathfrak{e}_B]$) ;

— A est fortement c-énumérable en B si l'énumération croissante de A est c-calculable en l'énumération croissante de B ($\mathfrak{p}_A \in \mathsf{c}[\mathfrak{p}_B]$) ;

— lorsque l'on considère à la fois l'énumération de l'ensemble et celle de son complémentaire, on parle de (forte) $\mathtt{C}$-récursivité en B ;
— A est χ-$\mathtt{C}$-récursive en B si la fonction caractéristique de A est $\mathtt{C}$-calculable en celle de B ($\chi_A \in \mathtt{C}[\chi_B]$).

Un autre type de réductions bornant les ressources utilisables lors de la réduction sont les réductions à *tables de vérité* (*truth-table*).

Définition 3.3.1 **Réductions par tables de vérité.** *Une fonction f est* weak truth-table *réductible à un ensemble Y, noté $f \leqslant_{\mathrm{wtt}} Y$, s'il existe une fonction récursive φ_e et une fonction totale calculable r tel que $f = \varphi_e^Y$ et pour tout n, le calcul de $\varphi_e^Y(n)$ ne questionne l'oracle Y que sur des valeurs inférieures à $r(n)$.*

Une fonction f est truth-table *réductible à un ensemble Y, noté $f \leqslant_{\mathrm{tt}} Y$, s'il existe une fonction récursive φ_e tel que $f = \varphi_e^Y$ et tel que pour tout oracle Z, φ_e^Z soit totale.*

On généralise ces deux réductions aux ensembles en les confondant avec leur fonction caractéristique.

Nous pourrions penser que ces réductions, étant donné qu'elles contraignent l'utilisation de l'oracle de la réduction par une fonction calculable, voire par un entier fini, sont semblables. En fait, comme nous allons le voir, chaque $\mathtt{C}$-réduction est simplement un raffinement de la réduction classique associée, mais reste en général incomparable avec les autres réductions.

Proposition 3.3.2

Compatibilité des réductions par tables de vérité. *Soit $\mathtt{C}$ une classe fondamentale. Si deux ensembles se réduisent via $\mathtt{C}$-*wtt *ou $\mathtt{C}$-*tt*, alors ils se réduisent respectivement via* wtt *ou via* tt*.*

Démonstration. Les fonctions $\mathtt{C}$-fondamentales étant récursives, l'implication est immédiate.

□

La notion de réduction $\preccurlyeq^{\chi}_{\mathtt{C}\text{-}T}$ s'avère être l'une des plus fortes de nos réductions, elle est même plus forte que la réduction *truth-table* :

Proposition 3.3.3

Soit $\mathtt{C}$ une classe fondamentale. $\preccurlyeq^{\chi}_{\mathtt{C}\text{-}T}$ est strictement plus forte que la réduction à table de vérité $\preccurlyeq_{tt}$ et est donc également plus forte que la version affaiblie $\preccurlyeq_{wtt}$ et comme nous l'avons vu, que la version Turing $\preccurlyeq_T$.

En revanche, la faible $\mathtt{C}$-réduction est incomparable avec la réduction à table de vérité et la faible réduction à table de vérité.

Proposition 3.3.4

Soit C *une classe fondamentale.* $\preccurlyeq^{w}_{\mathsf{C}\text{-}T}$ *est incomparable avec* $\preccurlyeq_{tt}$ *et* $\preccurlyeq_{wtt}$.

Démonstration. Soit A un ensemble *effectively hypersimple*, c'est à dire un ensemble Turing-complet mais non *truth-table*-complet. Alors, $B = \{3a : a \in A\} \cup \{3n+1 : n \in \mathbb{N}\}$ est faiblement C-Turing complet mais n'est pas *truth-table* complet.

□

3.3.2 Ensembles caractéristiques et complétude

Nous voulons maintenant étudier la *puissance de calcul* de nos classes de fonctions C-partielles. Nous allons montrer qu'il est possible d'étudier ces classes de la même manière que sont étudiées les fonctions récursives partielles classiques. L'un des outils permettant notamment de repérer la difficulté de prédiction des fonctions dans la structure de degrés est l'ensemble diagonal. Nous allons donc définir un nouvel ensemble diagonal, lié à une classe de fonctions C-partielles. Cette fois-ci, nous pourrons coller de plus près à la définition classique étant donné que nos fonctions sont partielles.

Définition 3.3.5 **Ensemble diagonal pour les fonctions C-partielles.**

$$\mathsf{K}_{\mathsf{C}} = \{e : \ \mathsf{U}_{\mathsf{C}}(e,e) \downarrow\} = \{e : \ \varphi^{\mathsf{C}}_{e}(e) \downarrow\}$$

Nous avons noté lors de l'étude des classes fondamentales qu'un ensemble diagonal était lié à une fonction de Castor Affairé. Voici donc la version sous-calculabilité de cette fonction :

Définition 3.3.6 **Fonction du castor affairé pour une sous-calculabilité.**

$$\mathsf{BB}_{\mathsf{C}} : x \mapsto \begin{cases} \max\{\varphi^{\mathsf{C}}_{i}(0) : i \leqslant x\} & \textit{s'il est défini ;} \\ 0 & \textit{sinon.} \end{cases}$$

Nous allons vérifier que là encore, l'image de la fonction du castor affairé et l'ensemble diagonal relatifs à une sous-calculabilité fondée sur C sont tous deux Turing-complets via une C-réduction *many-one*.

Proposition 3.3.7

L'ensemble C*-récursivement énumérable* K_{C} *est Turing-complet via* C*-réduction* many-one. *La fonction* BB_{C} *est complète via* C*-réduction* many-one *sur les fonctions.*

Démonstration. Nous allons montrer deux des équivalences :

$\mathsf{K}_{\mathsf{C}} \equiv_m^{\mathsf{C}} \mathsf{K}$ Nous voulons montrer que K_{C} est m-équivalent à K. Comme nous savons que K_{C} est récursivement énumérable, il en découle immédiatement que $\mathsf{K}_{\mathsf{C}} \preccurlyeq_m \mathsf{K}$.

Il reste à montrer la réduction inverse, $\mathsf{K} \preccurlyeq_m \mathsf{K}_{\mathsf{C}}$, c'est à dire qu'il est possible de réduire l'arrêt des fonctions récursives partielles à l'arrêt des fonctions C-partielles.

Pour ce faire, nous allons montrer qu'il existe une fonction C-fondamentale f telle que $x \in \mathsf{K} \Leftrightarrow f(x) \in \mathsf{K}_{\mathsf{C}}$, c'est à dire $\varphi_x(x) \downarrow \Leftrightarrow \varphi^{\mathsf{C}}_{f(x)}(f(x)) \downarrow$.

Pour ce faire, nous partons d'un indice $x \in \mathsf{K}$ et nous allons tenter de le transformer en indice de fonction C-partielle.

Soit a l'indice d'une fonction C-partielle jamais nulle. Nous définissons alors une fonction C-fondamentale f_x telle que pour tout $e \in \mathcal{D}$, $f_x(e)$ soit un indice pour une fonction C-partielle vérifiant :

$$\varphi^{\mathsf{C}}_{f_x(e)} : y \mapsto \begin{cases} \varphi_x(x) & \text{si } y = a \text{ ou } y = e \\ 0 & \text{sinon.} \end{cases}$$

Nous voulons trouver un point fixe de cette fonction, c'est à dire un indice n tel que $\varphi^{\mathsf{C}}_n = \varphi^{\mathsf{C}}_{f_x(n)}$ et nous voulons le faire de manière calculable.

Notre théorème de récursion partiel *à la* Kleene (Théorème 3.2.1) va nous le permettre.

Soit z un indice pour une fonction C-partielle constante égale à zéro, et $A = \left\{\mathfrak{p}(z, 2n) : z \geqslant 1\right\}$ un ensemble infini calculable d'indices pour cette fonction. Remarquons en particulier que $a \notin A$ et que A est fortement C-récursif [1].

Soit g une fonction C-partielle telle que :

$$g : y \mapsto \begin{cases} 0 & \text{si } y \in A, \\ \uparrow & \text{sinon.} \end{cases}$$

Cette fonction est nécessairement C-partielle puisque A est fortement C-calculable.

g définie sur A satisfait donc les conditions nécessaires pour appliquer le théorème du point fixe partiel à f_x sur A. D'après ce dernier, il existe donc nécessairement un n tel que :

$$\varphi^{\mathsf{C}}_n : y \mapsto \begin{cases} 0 & \text{si } y \in A, \\ \varphi^{\mathsf{C}}_{f_x(n)}(y) & \text{sinon.} \end{cases}$$

Analysons la situation : si $y \notin A$, nous sommes dans le cas du point fixe partiel qui nous donne l'égalité $\varphi^{\mathsf{C}}_n(y) = \varphi^{\mathsf{C}}_{f(n)}(y)$.

1. Il est C-énumérable de manière croissante ($\mathfrak{p}(z, \cdot)$ étant croissante et C-fondamentale) et ne contient jamais deux éléments consécutifs, ce qui permet une énumération aisée du complémentaire.

Regardons maintenant ce qu'il se passe si y appartient à A. Deux cas se présentent, soit $y = n$, soit $y \neq n$. Si $y \neq n$, par définition de f_x et sachant que comme $a \notin A$, $y \neq a$, nous savons que $\varphi^{\mathsf{c}}_{f_x(n)}(y) = 0$. Nous avons donc à nouveau l'égalité $\varphi^{\mathsf{c}}_n(y) = \varphi^{\mathsf{c}}_{f_x(n)}(y)$.

Il ne reste plus que le dernier cas à étudier, c'est à dire quand $n = y$. D'après la définition de f_x, nous nous trouvons dans le cas où $\varphi^{\mathsf{c}}_{f_x(n)}(y) = \varphi_x(x) = \varphi^{\mathsf{c}}_{f_x(n)}(a)$.

Comme $y \in A$ et $n = y$, $n \in A$, c'est à dire que n est un indice pour la fonction constante nulle, et donc en particulier $\varphi^{\mathsf{c}}_n(n) = \varphi^{\mathsf{c}}_n(a)$. Or, comme $a \notin A$, $\varphi^{\mathsf{c}}_n(a) = \varphi^{\mathsf{c}}_{f_x(n)}(a)$ par définition de n. Ainsi, $\varphi^{\mathsf{c}}_{f_x(n)}(y) = \varphi^{\mathsf{c}}_{f_x(n)}(a) = \varphi^{\mathsf{c}}_n(a) = \varphi^{\mathsf{c}}_n(n)$ et nous avons donc là aussi réalisé le point fixe.

Nous avons donc construit via f_x et g un indice n tel que $\varphi^{\mathsf{c}}_{f_x(n)} \cong \varphi^{\mathsf{c}}_n$, soit un véritable point fixe pour f_x. L'indice n a été construit via le théorème de récursion partielle de Kleene, il est donc c-fondamental en x. Si l'on note g la fonction calculant n à partir de x, on a bien que $n = f(x) \in \mathsf{K}_{\mathsf{c}}$ si et seulement si $x \in \mathsf{K}$. Ce qui prouve que K se c-m-réduit à K_{c} via la fonction c-fondamentale g.

Nous avons bien prouvé que $\mathsf{K} \preccurlyeq^{\mathsf{c}}_m \mathsf{K}_{\mathsf{c}}$, et donc que $\mathsf{K} \equiv^{\mathsf{c}}_m \mathsf{K}_{\mathsf{c}}$. Il en découle bien sûr, comme $\equiv^{\mathsf{c}}_m$ est plus forte que $\equiv_m$ que $\mathsf{K} \equiv_m \mathsf{K}_{\mathsf{c}}$.

$\chi_{\mathsf{K}_{\mathsf{c}}} \preccurlyeq^{\mathsf{c}}_m \mathsf{BB}_{\mathsf{c}}$ Comme d'habitude, la fonction du castor affairé permet de calculer le nombre maximal d'étapes de calcul nécessaires avant la terminaison d'une fonction, elle permet donc de calculer l'arrêt facilement, via les fonctions de simulation et de coût.

□

Tout comme les fonctions relatives aux fonctions classiques ou aux fonctions fondamentales, cette fonction-là n'est pas calculable par la classe qu'elle étudie, elle n'est donc pas c-partielle.

Théorème 3.3.8

Soit c une classe fondamentale. BB_{c} n'est pas une fonction c-partielle. De plus, $\overline{\mathrm{graph}(\mathsf{BB}_{\mathsf{c}})}$ n'est pas c-énumérable.

Démonstration. Supposons que BB_{c} soit c-partielle. Comme BB_{c} est une fonction totale, cela signifie que c'est une fonction récursive (au sens classique) totale.

Or, il est impossible que BB_{c} soit récursive totale puisque $\mathsf{BB}_{\mathsf{c}} \succcurlyeq_m \chi_{\mathsf{K}_{\mathsf{c}}}$ et $\mathsf{K}_{\mathsf{c}} \equiv_T \mathsf{K}$ (Proposition 3.3.7.) Donc BB_{c} n'est pas c-partielle.

La fonction classique du Castor Affairé n'est approximable que par en dessous, et BB_{c} hérite de cette propriété puisqu'il existe une fonction p-fondamentale f telle que $\forall e, \mathsf{BB}(e) \leqslant \mathsf{BB}_{\mathsf{c}}(f(e))$.

En effet, considérons l'ensemble $G_e = \{\langle i, 0\rangle : i > 0\} \cup \{\langle 0, \varphi_e(0)\rangle\}$. Cet ensemble est C-énumérable, d'indice C-fondamental calculable à partie de e (par application du Théorème 2.3.9), par une fonction C-fondamentale que nous noterons g. Posons alors

$$f : e \mapsto \begin{cases} g(e) & \text{si } e = 0, \\ \mathfrak{p}(g(e), f(e-1)) & \text{sinon.} \end{cases}$$

f est bien C-fondamentale, car définie par un schéma de récursion primitive à partir de fonctions C-fondamentale, et $\varphi^{\mathsf{C}}_{f(e)}$ désigne bien une fonction telle que $\varphi^{\mathsf{C}}_{f(e)}(0) = \varphi^{\mathsf{C}}_e(0)$ et $\forall d > 0, \varphi^{\mathsf{C}}_{f(e)}(d) = 0$.

Alors, s'il était possible d'approximer BB_{C} par en dessus, il en serait de même pour BB, ce qui est absurde puisque si on pouvait approximer BB par en dessus, on pourrait le calculer. Par conséquent, BB_{C} n'est pas approximable par en dessus et donc $\overline{\mathrm{graph}(\mathsf{BB}_{\mathsf{C}})}$ ne peut pas être C-énumérable.

□

3.3.3 Degrés intermédiaires

Nous l'avons déjà vu en observant les éléments de la classe des fonctions C-fondamentales, les objets que peuvent décrire ces fonctions ne sont pas nécessairement triviaux, ni même récursifs. En fait, un grand nombre de résultats sur la structure des degrés Turing et les propriétés de certains peut être démontré à l'intérieur même des sous-calculabilités. Nous nous sommes pour l'instant particulièrement intéressés aux problèmes Δ^0_2 qui trouvent leurs pendants assez naturellement dans notre cadre.

Dans la suite de ce paragraphe, nous allons étudier les preuves de plusieurs solutions à différentes variantes du problème de Post. Il s'agit de la question de trouver un degré Turing intermédiaire récursivement énumérable, non récursif ($> \mathbf{0}$) et incomplet ($< \mathsf{K}$). La question initialement posée par Emil L. Post dans [Pos03], a finalement trouvé une réponse indépendamment par Richard M. Freidberg dans son papier [Fri57] et par Albert Abramovich Muchnik comme le reporte [VS03]. Ces deux preuves introduisent du même coup la méthode de priorité. Depuis, d'autres solutions ont été apportées, voir par exemple [Joc84].

Il y a différents points de vue possibles pour l'étude de ce problème. Nous allons commencer par étudier l'existence d'un ensemble compris entre $\mathbf{0}$ et K suivant les C-réductions que nous avons définies.

Théorème 3.3.9

Solutions au C-problème de Post. *Soit* C *une classe fondamentale. Il existe un ensemble* r.e. X *tel que* $\varnothing \prec_{\mathsf{C}\text{-}T} X \prec_{\mathsf{C}\text{-}T} \mathsf{K}$ *pour* $\prec_{\mathsf{C}\text{-}T} = \prec^{\chi}_{\mathsf{C}\text{-}T}$ *et* $\prec_{\mathsf{C}\text{-}T} = \prec^{s}_{\mathsf{C}\text{-}T}$.

Démonstration. Un ensemble de degré intermédiaire est intuitivement un ensemble dont la complexité est intermédiaire entre la trivialité récursive et la complétude de l'arrêt. Nous avons défini deux candidats naturels : les ensembles diagonaux de C et de la classe des C-partielles.

Considérons donc K_{C} et $\mathsf{K}^{\Phi}_{\mathsf{C}}$. Ces deux ensembles sont *r.e.* et même p-énumérables. Aucun d'entre eux n'est ni fortement C-énumérable ni χ-C-récursif. Ils vérifient donc

$$\begin{array}{lll} \mathsf{K}_{\mathsf{C}} & \succ_{\mathsf{C}\text{-T}} & \varnothing \\ \mathsf{K}^{\Phi}_{\mathsf{C}} & \succ_{\mathsf{C}\text{-T}} & \varnothing \end{array}$$

pour $\prec_{\mathsf{C}\text{-}T} = \prec^{\chi}_{\mathsf{C}\text{-}T}$ et $\prec_{\mathsf{C}\text{-}T} = \prec^{\mathrm{s}}_{\mathsf{C}\text{-}T}$.

Nous voulons maintenant montrer que $\mathsf{K}_{\mathsf{C}} \succ_{\mathsf{C}\text{-T}} \mathsf{K}^{\Phi}_{\mathsf{C}}$ pour ces deux réductions.

Pour démontrer que $\mathsf{K}^{\Phi}_{\mathsf{C}} \prec^{\chi}_{\mathsf{C}\text{-}T} \mathsf{K}_{\mathsf{C}}$, montrons que $\chi_{\mathsf{K}^{\Phi}_{\mathsf{C}}} \in \mathsf{C}[\chi_{\mathsf{K}_{\mathsf{C}}}]$, mais que l'inverse n'est pas vrai. Soit $e \in \omega$, pour savoir si $e \in \mathsf{K}^{\Phi}_{\mathsf{C}}$, on veut savoir si $\Phi^{\mathsf{C}}_{e}(e) > 0$. Pour ce faire, construisons une fonction Φ^{C}_{p} qui sur l'entrée n commence par calculer $\Phi^{\mathsf{C}}_{e}(e)$, et qui si le calcul renvoie 0, retourne 0, sinon retourne $\langle n, 0\rangle$. De par la définition des fonctions C-partielles, φ^{C}_{p} est soit nul part définie si $\Phi^{\mathsf{C}}_{e}(e) = 0$, soit partout définie. Autrement dit, $\chi_{\mathsf{K}^{\Phi}_{\mathsf{C}}}(e) = \chi_{\mathsf{K}_{\mathsf{C}}}(p_e)$. Nous avons donc notre réduction. La réduction dans l'autre sens est impossible puisque $\mathsf{K}^{\Phi}_{\mathsf{C}}$ est récursif alors que K_{C} ne l'est pas.

Pour démontrer que $\mathsf{K}^{\Phi}_{\mathsf{C}} \prec^{s}_{\mathsf{C}\text{-}T} \mathsf{K}_{\mathsf{C}}$, le raisonnement est similaire. La difficulté est de pouvoir connaître à chaque fois le prochain élément à énumérer pour $\mathsf{K}^{\Phi}_{\mathsf{C}}$ et son complémentaire. Il suffit en fait de tous les essayer. Il n'y a qu'un nombre fini d'essais à faire, dont nous pouvons établir la borne grâce à $n \mapsto \mathfrak{p}(\mathbf{const}(1), n)$ et $n \mapsto \mathfrak{p}(\mathbf{const}(0, n))$. Donc l'inégalité est vérifiée et également stricte dans ce cas.

Ce qui termine la preuve du résultat. $\square$

Un autre point de vue possible dans l'étude du problème de Post est de considérer le véritable problème de Post de la calculabilité classique, d'en prendre une solution et de s'intéresser à ce que les sous-calculabilités peuvent dire sur le degré ainsi trouvé.

Théorème 3.3.10

Solutions C-énumérables au problème de Post. *Soit C une classe fondamentale. Il existe un ensemble X C-énumérable vérifiant $\varnothing <_T X <_T \varnothing'$.*

Démonstration. Pour montrer qu'il existe un ensemble de degré intermédiaire qui soit C-énumérable, nous pouvons soit en construire un de toutes pièces, soit l'obtenir par transformation d'un ensemble de degré intermédiaire existant.

Pour une preuve directe, il est possible d'utiliser la construction usuelle de priorité avec nombre fini de corrections (*finite injury*) de Firedberg-Muchnik dans la construction de Kučera ([Kuč86]). Cette méthode nous permet de construire un ensemble promptement simple[2] se réduisant Turing à un ensemble *low* non *r.e.*

2. Dans la quête pour un degré Turing *r.e.* intermédiaire, de nombreuses notions de *simplicité* ont été définies. Il y a eu les ensembles *simples*, les ensembles *hyper-simples*, *etc.*. Les ensembles *promptement simple* (*promptly simple*, définis par Maass en 1982) sont des ensembles récursivement énumérables intersectant tous les ensembles récursivement énumérables selon une certaine relation de domination.

Cependant, ce résultat peut aussi être artificiellement déduit de l'existence d'un ensemble *r.e.* de degré intermédiaire X. En effet, soit l'ensemble $I = \{2x : x \in X\} \cup \{2x+1 : x \in \mathbb{N}\}$, nous vérifions sans difficulté que I est de degré intermédiaire et que par le Théorème 2.3.9, I est c-énumérable.

$\square$

Il reste encore de nombreux aspects à étudier dans la structure des degrés des sous-calculabilités. Par exemple, la question de l'existence des paires de degrés *r.e.* minimales, résolue par Alistair H. Lachlan dans [Lac66] trouve très sûrement un écho ici. Il serait intéressant pour la suite de tenter de construire une telle paire de degrés c-énumérables de manière directe.

3.3.4 Liens entre les degrés Turing et les degrés c-Turing

Les degrés c-Turing étant définis dans les sous-calculabilités de manière similaire aux degrés Turing classiques, il n'y a rien d'étonnant à ce qu'une certaine partie de la structure soit commune. Nous allons voir que c'est particulièrement vrai pour l'ensemble des degrés récursivement énumérables, pour lesquels il est possible d'exhiber une injection uniforme entre les ensembles témoins récursivement énumérables de ces degrés et leurs versions c-Turing. Nous verrons également que la question précédente de l'existence de degrés intermédiaires *r.e.* se trouve éclaircie par cette injection, qui laisse présager que toute une famille de résultats classiques vont pouvoir être reportés à notre contexte de manière similaire. Enfin, nous étudierons le cas de certains ensembles standards comme les ensembles diagonaux ont des propriétés qui méritent d'être étudiées à la fois dans les sous-calculabilités et dans la calculabilité classique afin de concrétiser les liens entre ces différentes notions.

Propriété 3.3.11

*Soit c une classe fondamentale. K_{c} est p-énumérable mais non co-*r.e.

Démonstration. La preuve est très proche de celle concernant K :

p-énumérabilité Tout comme K, K_{c} est p-énumérable par application du Théorème 2.3.9 car il contient un ensemble trivial généré en utilisant la fonction de bourrage et est lui-même récursivement énumérable.

non co-*r.e.* K_{c} est p-énumérable et donc *r.e.*, or, d'après la Proposition 3.3.7, il est Turing-complet. Donc, K_{c} ne peut pas être co-*r.e.*

$\square$

Proposition 3.3.12

Soit C *une classe fondamentale. Tout* 1*-degré récursivement énumérable possède un représentant* r.e. *non-*C*-énumérable. Tout* m*-degré récursivement énumérable possède un représentant* C*-énumérable.*

Démonstration. Soit φ une énumération injective d'un ensemble *r.e.* E infini, qui existe forcément. Le problème est que φ est peut-être C-fondamentale. Nous allons donc chercher à transformer φ et donc E afin d'obtenir un ensemble qui ne sera cette fois pas C-énumérable.

Considérons la fonction $\psi = \mathsf{BB}^{\Phi}_{\mathsf{C}} \circ \varphi$ d'image F. F est un sous-ensemble infini d'un ensemble *r.e.* non-C-énumérable, par contraposée du Théorème 2.3.9, F ne peut pas être C-énumérable.

Or, φ et $\mathsf{BB}^{\Phi}_{\mathsf{C}}$ sont toutes deux des fonctions totales calculables injectives, il en est donc de même de ψ.

Nous pouvons alors inverser ψ de manière calculable, afin d'obtenir une fonction partielle injective ψ^{-1} vérifiant $\psi^{-1}``F = E$, et donc, $E \equiv_1 F$.

F est donc un ensemble non-C-énumérable de même 1-degré que E.

On s'intéresse maintenant aux m-degrés. Soit F un ensemble récursivement énumérable énuméré par une fonction calculable ψ. Nous pouvons supposer sans perte de généralité que ψ est une fonction calculable totale injective. Remarquons ensuite que l'ensemble $E = \{2 * n + 1\} \cup \{2\phi(n) : n \in \omega\}$ est nécessairement C-énumérable.

Il reste à démontrer qu'il est de même m-degré que F. Clairement, un élément $d \in \mathcal{D}$ appartient à E s'il est impair ou si $d/2$ appartient à F. Nous pouvons facilement construire une m-réduction de E en F : Soit $f_0 \in F$, on pose f la fonction telle que :

$$f : d \mapsto \begin{cases} f_x & \text{si } d \text{ est impair ;} \\ d/2 & \text{sinon.} \end{cases}$$

Notons que cette réduction n'est pas injective et qu'il n'est pas tout le temps possible de la rendre injective.

Pour réduire F en E, la fonction $g : d \mapsto 2d$ convient.

E est donc un ensemble C-énumérable de même m-degré que F. □

Cette propriété étant établie, il nous est facile de tirer le corollaire suivant indiquant que toutes les solutions au problème de Post classique ne sont pas nécessairement des solutions au C-problème de Post :

Théorème 3.3.13

Solutions *r.e.* et non C-énumérables au problème de Post. *Soit* C *une classe fondamentale. Il existe un ensemble* r.e. *et non* C*-énumérable* X *vérifiant* $\varnothing <_T X <_T \varnothing'$.

Démonstration. Il s'agit d'une application simple des résultats précédents. Soit φ une énumération injective pour un ensemble E *r.e.* de degré intermédiaire.

Par la Proposition 3.3.12, nous pouvons construire un ensemble *r.e.* non $\mathtt{C}$-énumérable F de même 1-degré que E. Cet ensemble F satisfait alors $\varnothing <_T W <_T \varnothing'$ et est bien non-$\mathtt{C}$-énumérable.

□

Les ensembles *low* étant particulièrement bas dans la hiérarchie des degrés Turing, il est naturel de se demander s'ils sont tous $\mathtt{C}$-énumérables. Le corollaire suivant nous indique que ça n'est pas le cas.

Corollaire 3.3.14

Soit $\mathtt{C}$ *une classe fondamentale. Il existe un ensemble* W r.e., low *qui ne soit pas* $\mathtt{C}$*-énumérable.*

Démonstration. Pour justifier ce résultat, il suffit de reprendre la preuve précédente avec un ensemble *r.e. low.* □

Note 3.3.15 *Pour définir la notion de* $\mathtt{C}$-low*, il faut définir le* $\mathtt{C}$*-Turing-jump* à la *sous-calculabilité.*

X *est alors* $\mathtt{C}$-low *si* $X'^{\mathtt{C}} \preccurlyeq_{\mathtt{C}\text{-}T} \varnothing'$.

Par un argument de priorité, il est possible de s'assurer que l'ensemble low *promptement simple* X *défini en 3.3.14 soit* $\mathtt{C}$-low *mais non* $\mathtt{C}_-$-low*, avec* $\mathtt{C}_-$ *une classe fondamentale strictement contenue dans* $\mathtt{C}$*. La preuve se généralise également aux ensembles* superlow*, voir* [*Nie09, Ex. 1.6.7, p. 386*].

3.3.5 Une structure fine récursive

Nous avons noté dans les paragraphes 3.1 et 3.2 que toutes les fonctions récursives n'étaient pas présentes parmi les fonctions $\mathtt{C}$-partielles et les fonctions $\mathtt{C}$-fondamentales. Dans notre étude des $\mathtt{C}$-réductions et de leurs degrés $\mathtt{C}$-Turing associés, ces *trous* prennent la forme de réductions *manquantes.* Ainsi, certains ensembles qui se réduisent l'un à l'autre au sens Turing classique ne se $\mathtt{C}$-réduisent pas nécessairement l'un à l'autre.

Les résultats précédents nous démontrent que chaque degré Turing peut être découpé en une multitude de $\mathtt{C}$-degrés, suivant les différentes notions de récursivité que nous avons définies. Nous voyons que le Théorème 2.3.9 trivialise la preuve de l'existence de tous les degrés Turing dans notre relativisation de celle-ci. Cependant, les ensembles obtenus ne sont pas les mêmes. Ainsi, si nous nous plaçons à l'échelle d'un degré Turing et tentons d'en classifier les membres dans les différents degrés $\mathtt{C}$-Turing

Ce travail se rapproche notamment de l'étude des *degrés honnêtes* menée par Lars Kristiansen, voir [KSPW11], [Kri96], [Kri98] et [Kri99], dans laquelle il identifie une classe de fonctions *honnêtes élémentairement énumérables*, dont l'étude de la structure fait apparaître des objets similaires aux nôtres : degrés incomparables, opérateur de *jump*, etc. Si les outils utilisés dans ces travaux sont différents, l'arithmétique élémentaire posant d'autres problématiques que l'arithmétique primitive récursive, cette structure semble être un raffinement supplémentaire, révélant une fois de plus une hiérarchie de *modèles de calculabilités* que nous étudierons d'avantage par la suite.

3.4 Productivité et créativité

Les ensembles productifs sont des ensembles qui ne sont pas bien approximables par des ensembles *r.e.* Pour un ensemble productif X, il existe une fonction *productive* f tel que pour tout sous-ensemble *r.e.* $\mathbf{W}_e$ de X, $f(e)$ soit un élément de X n'appartenant pas à $\mathbf{W}_e$. Dans le cas classique, cette fonction f est une fonction récursive quelconque[3]. Nous nous intéressons dans ce chapitre à la notion de productivité obtenue lorsque l'on rajoute la contrainte que les fonctions productives doivent être C-fondamentales.

Définition 3.4.1 **Ensembles productifs et créatifs.** *Un ensemble X est dit* C-productif *s'il existe une fonction injective C-fondamentale ψ, appelée la* fonction productive *pour X telle que, pour tout $x \in \mathcal{D}$, si $\mathbf{W}^{\mathsf{C}}_x \subseteq X$, alors la fonction productive ψ permet de générer un élément $\psi(x)$ appartenant à l'ensemble différence $X \setminus \mathbf{W}^{\mathsf{C}}_x$.*

Un ensemble C-énumérable X est dit C-créatif *si son complémentaire $\overline{X}$ est C-productif.*

Tout d'abord, remarquons que cette notion n'est pas trivialement équivalente à celle de productivité classique :

Proposition 3.4.2

Il existe des ensembles qui sont productifs mais qui ne sont pas C-productifs. *Soit C une classe fondamentale. L'ensemble $\mathbf{BB}^{\Phi}_{\mathsf{C}}$"$\overline{\mathbf{K}}$ est productif, mais il n'est pas C-productif.*

Démonstration. Nous voulons montrer que l'ensemble en question ne peut pas être C-productif. Pour ce faire, nous allons considérer une de ses fonctions de productions et remarquer qu'elle ne peut pas être C-fondamentale.

3. On peut toutefois toujours supposer f injective en la remplaçant par g tel que $g : e \mapsto f(e')$ avec e' un indice de l'ensemble $\mathbf{W}_e \cup \{g(i) : i < e\}$.

On considère l'ensemble $X = \mathsf{BB}_{\mathsf{C}}^{\Phi}\text{``}\overline{\mathsf{K}}$. Comme $\overline{\mathsf{K}}$ est productif[4] et que $\mathsf{BB}_{\mathsf{C}}^{\Phi}$ est une injection récursive, $X$ est un ensemble productif. En effet, étant donné un ensemble productif $A$ de fonction de production $p$ et une fonction récursive injective $h$, $h\text{``}A$ est productif car pour tout ensemble énumérable $\mathsf{W}_e \subset h\text{``}A$, si on note $e'$ l'indice tel que $\mathsf{W}_{e'} = h^{-1}\text{``}\mathsf{W}_e$, alors $p(e') \in A \setminus \mathsf{W}_{e'}$ et donc $h \circ p(e') \in h\text{``}A \setminus \mathsf{W}_e$.

Supposons que X soit C-productif et notons f une fonction de production C-fondamentale pour X.

Soit x tel que $\mathsf{W}_x^{\mathsf{C}}$ soit un sous-ensemble C-énumérable infini de X. Par application du corollaire 2.3.11, il existe une fonction C-fondamentale h telle que $\mathsf{W}_{h(x)}^{\mathsf{C}} = \mathsf{W}_x^{\mathsf{C}} \cup \{f(x)\}$.

Alors, $g : n \mapsto f \circ h^{(n)}(x)$ est une fonction C-fondamentale qui énumère injectivement un sous-ensemble de l'image de $\mathsf{BB}_{\mathsf{C}}^{\Phi}$ et donc domine la fonction du castor affairé, ce qui est absurde.

Et donc finalement, $\mathsf{BB}_{\mathsf{C}}^{\Phi}\text{``}\overline{\mathsf{K}}$ n'est pas C-productif. □

Nous remarquons également que comme dans la calculabilité classique, l'ensemble diagonal est créatif :

Proposition 3.4.3

Soit C une classe fondamentale. K_{C} est créatif, K est C-créatif.

Démonstration. L'ensemble K_{C} étant m-complet, il est créatif (voir par exemple [Odi88], Théorème III.6.6).

L'ensemble $\overline{\mathsf{K}}$ est productif via la fonction identité. Cette fonction étant dans p, $\overline{\mathsf{K}}$ est C-productif, et K est C-créatif. □

La limitation de la fonction productive se comporte bien avec les notions de C-énumérabilité, en particulier, on obtient la propriété suivante :

Propriété 3.4.4

Soit C une classe fondamentale. Si X est un ensemble C-productif, alors il n'est pas C-énumérable et donc pas faiblement C-récursif.

Démonstration. Cette preuve ne présente aucune difficulté, nous la présentons comme une application directe de la définition 3.4.1.

Supposons que X soit C-productif, de fonction productive f, et C-énumérable. Alors, il existe un y tel que $X = \mathsf{W}_y^{\mathsf{C}}$. Comme $\mathsf{W}_y^{\mathsf{C}} \subset X$, alors $f(y)$ devrait nous donner un élément de $\mathsf{W}_y^{\mathsf{C}} \setminus X = \varnothing$, ce qui est absurde.

4. $\overline{\mathsf{K}}$ est productif avec pour fonction de production π_1^1. En effet, quelque soit $e \in \omega$ tel que $\mathsf{W}_e \subset \overline{\mathsf{K}}$, $e \notin \mathsf{W}_e$, sans quoi $e \in \mathsf{K}$ et donc $e \notin \overline{\mathsf{K}}$, ce qui contredirait l'inclusion. Donc $e \notin \mathsf{W}_e$, ce qui implique que $e \in \overline{\mathsf{K}}$. On a donc bien $\pi_1^1(e) = e \in \overline{\mathsf{K}} \setminus \mathsf{W}_e$.

Donc un ensemble $\mathtt{C}$-énumérable ne peut pas être productif, et réciproquement.

Un ensemble $\mathtt{C}$-productif ne pouvant pas être $\mathtt{C}$-énumérable, il ne peut donc pas être $\mathtt{C}$-récursif. □

Note 3.4.5 *Remarquons en particulier que l'ensemble $\mathbf{K}_{\mathtt{C}}^{\Phi}$, qui est faiblement $\mathtt{C}$-calculable, est nécessairement non $\mathtt{C}$-productif et non $\mathtt{C}$-créatif.*

Propriété 3.4.6

Soit $\mathtt{C}$ une classe fondamentale. Si X est un ensemble $\mathtt{C}$-productif, alors X contient un sous-ensemble infini $\mathtt{C}$-énumérable.

Démonstration. Soit X un ensemble $\mathtt{C}$-productif de fonction de production f et d un indice de $\mathcal{D}$ pour l'ensemble vide : $\mathbf{W}_d^{\mathtt{C}} = \varnothing$.

Par application du corollaire 2.3.11, il existe une fonction $\mathtt{C}$-fondamentale h telle que $\mathbf{W}_{h(x)}^{\mathtt{C}} = \mathbf{W}_x^{\mathtt{C}} \cup \{f(x)\}$.

Il nous suffit alors de considérer l'ensemble $W = \{f(d), f \circ h(d), f \circ h^2(d), \ldots\} = \{f \circ h^n(d) : n \in \mathcal{D}\}$. Cet ensemble est bien un sous-ensemble $\mathtt{C}$-énumérable. □

En calculabilité classique, il existe une caractérisation des ensembles productifs par 1-réduction de $\overline{\mathbf{K}}$. Le lemme suivant nous indique que l'on peut espérer un résultat similaire dans le contexte des sous-calculabilités :

Proposition 3.4.7

Soit $\mathtt{C}$ une classe fondamentale. Si X est un ensemble $\mathtt{C}$-productif et se $\mathtt{C}$-1-réduit à un ensemble A ($X \preccurlyeq_1^{\mathtt{C}} A$) via une bijection $\mathtt{C}$-fondamentale d'inverse f^{-1} également $\mathtt{C}$-fondamentale, alors A est $\mathtt{C}$-productif.

Démonstration. Soit X un ensemble $\mathtt{C}$-productif de fonction de production ψ et un ensemble A tel que $X \preccurlyeq_1^{\mathtt{C}} A$ via une fonction f injective, $\mathtt{C}$-fondamentale f d'inverse f^{-1} injective et $\mathtt{C}$-fondamentale.

Nous cherchons à construire une fonction $\mathtt{C}$-productive pour A. Pour ce faire, nous allons utiliser la bijection f afin de composer une nouvelle fonction g qui sera productive pour A.

Posons e_f et $e_{f^{-1}}$ des indices pour respectivement la fonction f et la fonction f^{-1} et soit e et A_0 tel que $A_0 = \mathbf{W}_e^{\mathtt{C}}$ soit un sous-ensemble $\mathtt{C}$-énumérable de A. Quitte à se ramener à l'énumération canonique, nous pouvons supposer que e est l'indice de la fonction fondamentale énumérant $\mathbf{W}_e^{\mathtt{C}}$. Dans ce cas, $\mathbf{comp}(e_{f^{-1}}, e)$ est l'indice d'un sous-ensemble X_0 récursivement énumérable de X. Considérons maintenant l'ensemble $\mathtt{C}$-énumérable X_1 d'indice $\mathbf{comp}(e_\psi, \mathbf{comp}(e_{f^{-1}}, e))$. X_1 contient au moins un élément $a \notin X_0$. L'image $f(a)$ ne peut pas être dans A_0, sans quoi a serait dans X_0. En revanche, $f(a)$ est présente dans $A_1 = f``X_1 = \mathbf{W}_{\mathbf{comp}(e_f,\mathbf{comp}(e_\psi,\mathbf{comp}(e_{f-1},e)))}$. De plus, comme f est une injection de X dans A, A_1 est bien un sur-ensemble de A_0, contenu dans A. Finalement, ces manipulations d'indices de fonctions C-fondamentale injectives sont elles-mêmes C-fondamentales, la fonction résultante $g = \Phi^{\mathsf{C}}_{\mathbf{comp}(e_f,\mathbf{comp}(e_\psi,\mathbf{comp}(e_{f-1},e)))}$ est elle-même injective, C-fondamentale, et calculée de manière C-fondamentale en f et ψ.

g est donc bien une fonction C-productive pour A. □

Nous avons vu que K était C-créatif, et que cette notion différait de la créativité classique. Nous n'avons pas poussé l'étude plus loin, le nombre de pistes à étudier étant très important. L'idéal serait de parvenir à une caractérisation de la créativité via par la m-complétude comme en récursivité classique, où être productif pour un ensemble X implique $\overline{\mathsf{K}} \preccurlyeq_1 X$ (voir [Odi88], Théorème III.6.6).

3.5 De la récursion primitive vers la calculabilité

Nous allons maintenant illustrer ces modèles et considérer quelques-uns d'entre eux. La première sous-calculabilité que nous allons étudier est la sous-calculabilité primitive récursive. Nous en avons déjà couvert un large pan dans le premier chapitre alors que nous traitions des classes énumérables qui comptent la classe des fonctions primitives récursives.

Nous allons ensuite montrer combien il est aisé d'identifier toute une famille de sous-calculabilités en se basant sur des systèmes de notations ordinales appelées *EORS* (pour *Système Élémentaire de Représentation Ordinale*) définis et étudiés par Michael Rathjen, Harvey Friedman et Michael Sheard (voir [Rat99] et [FS95]) alors qu'ils développaient des outils permettant de faciliter l'analyse ordinale des théories mathématiques. Ces *EORS* font une utilisation intensive des séquences fondamentales sur lesquelles ils reposent. Nous n'aborderons pas ces EORS dans ce chapitre mais les utiliserons implicitement pour la construction des schémas d'α-récursion.

Nous verrons plus tard, dans le chapitre suivant qu'il est possible de se passer complètement de ces notations pour peu que les ordinaux que nous cherchons à décrire soient présents dans notre domaine de départ.

Un premier moyen élémentaire mais déjà évoqué de réaliser l'existence des nombreux niveaux de sous-calculabilités nous est donné par les diverses hiérarchies de croissance rapide explicitées au chapitre précédent.

L'idée est simple : si une fonction domine toutes les fonctions présentes dans notre classe, elle ne peut pas en faire partie, or la distance entre les fonctions primitives récursives et les fonctions totales récursives est *grande*...

3.5.1 Sous-calculabilité primitive

Nous présentons ici, sous forme d'un tableau, les différents résultats obtenus sur cette sous-calculabilité. Pour des questions de présentation, les noms des propriétés ont été abrégés. ***r.e.***, **s**, **w**, **e** et **rec** abrègent respectivement *récursivement énumérable* (au sens classique), *fortement*, *faiblement*, *énumérable* et *récursif*.

Dans le tableau suivant, nous utiliserons les notations suivantes : $A = \text{range}(\text{Ack})$, $B = \text{range}(\mathsf{BB}_{\mathsf{p}}^{\Phi})$, $B' = \text{range}(\mathsf{BB}_{\mathsf{p}})$.

	r.e.	co-*r.e.*	p-e	s-p-e	co-p-e	co-s-p-e	w-p-rec	s-p-rec	χ-p-rec
K	✓	×	✓	×	×	×	×	×	×
K_{p}	✓	×	✓	×	×	×	×	×	×
$\mathsf{K}_{\mathsf{p}}^{\Phi}$	✓	✓	✓	×	✓	×	✓	×	×
A	✓	✓	×	×	✓	✓	×	×	✓
B	✓	✓	×	×	✓	✓	×	×	✓
B'	×	×	×	×	✓	✓	×	×	✓

Table 3.1 – Propriétés des ensembles remarquables de p

3.5.2 Sous-calculabilité ω^ω-récursive

Nous allons maintenant nous intéresser à la sous-calculabilité engendrée par la classe des fonctions ω^ω-récursives.

Pour réaliser un schéma d'ω^ω-récursion sur des entiers, il est nécessaire d'avoir un encodage des ordinaux déchiffrable par nos fonctions. Heureusement, il est simple de représenter les ordinaux inférieurs à ϵ_0 sous la forme de listes de listes d'entiers. Dans le cas des ordinaux inférieurs à ω^ω, comme dans l'exemple qui va suivre, de simples listes d'entiers suffisent. Ces schémas de codage nous permettent en fait de construire les schémas récursifs associés aux ordinaux ω^ω et ϵ_0.

A priori, avec ces schémas de récursion, nous devrions être capables de construire des fonctions qui n'étaient pas primitives récursives. C'est effectivement le cas. En particulier, la première fonction impossible à calculer dans p était la fonction d'Ackermann.

La voici dans sa définition équivalente sous forme de fonction diagonale de la classe des récursives primitives, définie en utilisant le schéma de récursion sur ω^ω.

Définition 3.5.1 **Fonction diagonale primitive récursive.** $g_{\omega^\omega} : x \mapsto f(\langle \underline{\omega}^{\overline{x}}, x \rangle)$, *où*

$$f : \langle x, m \rangle \mapsto \begin{cases} f(\theta(\langle x, m \rangle)) & \text{si } \overline{0} \triangleleft x \triangleleft \underline{\omega}^{\omega^{\overline{1}}}, \\ m & \text{sinon,} \end{cases}$$

où θ est définit comme suit :

$$\theta : \langle x, m\rangle \mapsto \begin{cases} \langle y, m+1\rangle & \textit{si } x,\ y \textit{ sont des notations ordinales pour} \\ & \textit{respectivement } \beta+1 \textit{ et } \beta, \\ \langle y_m, m\rangle & \textit{si } x,\ y_m \textit{ sont des notations ordinales pour} \\ & \textit{respectivement un ordinal limite } \alpha \textit{ et } \alpha_m, \\ \langle \overline{0}, 0\rangle & \textit{sinon.} \end{cases}$$

Nous en déduisons donc que la fonction d'Ackermann est ω^ω-récursive et donc ω^ω-fondamentale. Pour nous calquer sur la hiérarchie de croissance modérée qui coïncide avec nos schémas de récursion, nous pouvons dire qu'elle nécessite en fait un schéma de récursion sur ω^ω, qui n'est pas définissable dans la classe des fonctions primitives récursives puisqu'il consiste d'une certaine manière en une imbrication non-bornée de schémas de récursion primitive.

Théorème 3.5.2

La fonction d'Ackermann est ω^ω-fondamentale.

Par le même argument et une construction similaire, nous pouvons nous rendre compte que la fonction de Goodstein, *i.e.* la fonction calculant la longueur des séquences éponymes[5] est ϵ_0-récursive et donc ϵ_0-fondamentale.

Posons $A_0 = \mathrm{range}(g_{\omega^\omega})$ et $A_1 = \mathrm{range}(g_{\epsilon_0})$. Le tableau 3.2 contient un résumé des propriétés de nos ensembles dans la calculabilité des fonctions ω^ω-récursive. Nous remarquerons que ces propriétés sont similaires au cas récursif primitif.

	re	co-*r.e.*	ω^ω-e	s-ω^ω-e	co-ω^ω-e	co-s-ω^ω-e	w-ω^ω-rec	s-ω^ω-rec	χ-ω^ω-rec
K	✓	×	✓	×	×	×	×	×	×
$\mathsf{K}_{\omega^\omega}$	✓	×	✓	×	×	×	×	×	×
$\mathsf{K}^{\Phi}_{\omega^\omega}$	✓	✓	✓	×	✓	×	✓	×	×
A_0	✓	✓	✓	✓	✓	✓	✓	✓	✓
A_1	✓	✓	×	×	✓	✓	×	×	✓

Table 3.2 – Propriétés des images de g_{ω^ω} et g_{ϵ_0} dans la sous-calculabilité sur ω^ω

5. Les séquences de Goodstein sont définies comme suit : le premier terme u_0 est un entier $a > 0$ de notre choix. Pour obtenir le terme u_{n+1} à partir du terme u_n, il faut écrire u_n en base héréditaire $n+2$ (exemple : pour $n = 0$ et $a = 7$: $9 = 2^{2^{2^0}+2^0} + 2^0$), de remplacer toutes les occurrences de la base par $n+3$, puis de retrancher 1 au résultat obtenu. Ces séquences convergent toujours vers 0 en un nombre fini d'étapes, c'est ce dernier que calcule la fonction de Goodstein, en fonction de a. En utilisant les résultats de Gentzen [Gen36], Goodstein a prouvé en 1944 que l'axiomatique de Peano n'est pas capable de prouver la convergence de sa fonction.

3.5.3 Sous-calculabilité ζ-récursive

Nous avons vu qu'il était possible de construire des séquences fondamentales jusqu'à l'ordinal de Bachmann-Howard : ζ, soit, dans les notations que nous avons définies dans le paragraphes traitant des séquences fondamentales, ϕ_{ω_1+1}, et même beaucoup plus loin. En particulier, celles-ci vont nous permettre de construire un schéma de codage pour l'ordinal ζ. En fait, ce que nous faisons ici pour ζ est possible pour une multitude d'ordinaux récursifs compris entre ϵ_0 et ω_1^{CK}.

Une fois de plus, remarquons que les propriétés de la fonction *diagonale* de la classe fondamentale, g_ζ, sont les mêmes que les propriétés du Castor Affairé relatif à celle-ci (BB_ζ^Φ.)

Les deux peuvent être choisies pour réaliser le *saut* de la classe fondamentale.

Cette fois, posons $A = \mathrm{range}(g_\zeta)$:

	$r.e.$	co-$r.e.$	ϵ_0-e	s-ϵ_0-e	co-ϵ_0-e	co-s-ϵ_0-e	w-ϵ_0-rec	s-ϵ_0-rec	χ-ϵ_0-rec
A	✓	✓	×	×	✓	✓	×	×	✓

	$r.e.$	co-$r.e.$	ζ-e	s-Γ_0-e	co-Γ_0-e	co-s-Γ_0-e	w-Γ_0-rec	s-Γ_0-rec	χ-Γ_0-rec
A	✓	✓	✓	✓	✓	✓	✓	✓	✓

TABLE 3.3 – Propriétés de range(g_ζ) dans les sous-calculabilités sur ϵ_0 et sur Γ_0.

4 Premiers pas vers une structure fine généralisée

Portée des sous-calculabilités récursives et généralisation aux ensembles

Sommaire

Nous avons jusqu'à présent étudié les classes fondamentales de fonctions totales et leur extension aux classes de fonctions partielles, les sous-calculabilités, en donnant quelques exemples de celles-ci et en prouvant qu'il existait toute une hiérarchie de telles sous-calculabilités récursives sur les entiers.

Cet exposé amène naturellement une question : y a-t-il une limite dans la part de calculabilité qui peut être intégrée à une sous-calculabilité ? Nous n'avons pas encore de réponse à celle-ci. Cependant, nous savons que les ordinaux et particulièrement les outils que l'on a pour manipuler les ordinaux récursifs [1] jouent un rôle important. Nous aborderons donc cette question dans le premier chapitre.

Une autre question a été laissée en suspend pendant cette étude : celle de la généralisation du domaine. La calculabilité a été généralisée depuis longtemps aux objets mathématiques plus complexes que les entiers. En particulier, la théorie de la récursion sur les admissibles [2] nous permet d'identifier une structure de calculabilité sur n'importe quel ensemble admissible. Nous y avons vu une structure semblable à celles des sous-calculabilité et sans pouvoir encore dire que ces calculabilités possèdent toutes les bonnes propriétés des classes fondamentales et des sous-calculabilités, nous allons montrer qu'elles peuvent déjà s'écrire dans leur formalisme. Nous évoquerons enfin une deuxième généralisation de la calculabilité aux ensembles, assez distincte de la récursion sur les admissibles, mais dont l'intuition est assez proche de celle des sous-calculabilités.

1. Voir Définition 4.1.5.

2. Un ensemble admissible est un ensemble modélisant la théorie des ensembles de Kripke-Platek : **KP**.

4.1 Portée des sous-calculabilités récursives

Dans chacun des exemples de classe α-récursive que nous avons donnés jusqu'à présent, notre construction dépendait d'une méthode de codage des ordinaux dans les entiers. Certes, par définition, tous les ordinaux dénombrables, et donc *a fortiori* les ordinaux récursifs, sont en bijection avec les entiers naturels. Cependant il n'est pas évident de savoir quand cette bijection peut-être élémentaire, récursive primitive, *etc.* ni même d'obtenir un moyen de l'exprimer sous forme d'une fonction appartenant à une telle classe.

Ces problèmes sont bien connus dans le domaine de l'analyse ordinale des théories, où un des objectifs a longtemps été de tenter d'exprimer des représentations ordinales pour des ordinaux toujours plus grands. Le paragraphe ci-dessous traite d'un des outils les plus utilisés dans cette optique : les EORS.

Le paragraphe suivant met alors en regard les connaissances que l'on a sur les EORS avec les limites que l'on peut espérer pour les sous-calculabilités.

4.1.1 Au delà de la récursion primitive

Dans le chapitre 2 présentant les classes fondamentales, nous avons traité le cas des classes de fonctions α-récursives, sans trop nous attarder sur la définition de l'α-récursivité, ni sur son domaine de validité ou en tout cas, sans expliciter pour quelle famille d'ordinaux il était intéressant de considérer les classes α-récursives associées.

Les ordinaux ne sont pas tous aussi facilement manipulables les uns que les autres. En particulier avant de s'autoriser à faire de la récursion sur un ordinal α, il est raisonnable de vouloir se placer dans un contexte dans lequel l'ordinal est en quelque sorte *effectivement* accessible et *effectivement* manipulable. C'est là qu'interviennent les *EORS* en tant que structures dont l'existence assure une utilisation effective de l'ordinal associé.

En réalité, un EORS va s'assurer qu'un certain nombre de propriétés liées à l'ordinal sont vraies pour une théorie élémentaire : l'*Arithmétique Récursive Élémentaire* (*ERA*) :

Définition 4.1.1 **Arithmétique Récursive Élémentaire.** *L'Arithmétique Récursive Élémentaire,* **ERA***, est un système faible de théorie des nombres, avec pour langage* $0, 1, +, \times, E$ *(exponentiation),* $<$ *et pour axiomes :*

1. *les axiomes habituels de la récursion pour* $+$, $\times$, E *et* $<$ *;*
2. *l'induction sur les formules* Δ_0 *à variables libres.*

Cette axiomatique repose sur la base de l'induction des formules Δ_0, notée habituellement $I\Delta_0$, à laquelle ont été rajoutées les opérations usuelles des fonctions élémentaires. Et, de fait, les fonctions que cette théorie peut prouver totales sont exactement les fonctions élémentaires.

Propriété 4.1.2

Les fonctions prouvablement récursives selon **ERA** *sont exactement les fonctions élémentaires introduites dans le paragraphe 1.2.1.*

Le point qui nous intéresse est le fait que cette théorie, bien que minimale, nous permet de construire les *EORS*, clés de voûte de l'α-récursion.

Un EORS est une petite structure définissant des opérations sur une relation bien fondée $\triangleleft$ d'ordinal type λ et vérifiant de bonnes propriétés dans ERA :

Définition 4.1.3 **Systèmes Élémentaires de Représentation Ordinale.** *Un système élémentaire de représentation ordinale (EORS) pour un ordinal limite λ est une structure $\langle A, \triangleleft, n \mapsto \lambda_n, +, \times, x \mapsto \omega^x \rangle$ telle que :*

1. *A est un sous-ensemble élémentaire de ω ;*
2. *$\triangleleft$ est un bon ordre sur A ;*
3. *$|\triangleleft| = \lambda$;*
4. **ERA** *prouve que pour tout n, $\triangleleft_{\restriction \lambda_n}$ est un segment initial propre de $\triangleleft$ et que $\bigcup_{n \in \omega} \triangleleft_{\restriction \lambda_n} = \triangleleft$;*
5. **ERA** *prouve que $\triangleleft$ est un ordre total sur A et que pour tout x et y comparables par $\triangleleft$, x et y appartiennent à A ;*
6. *$+$ et $\times$ sont des opérations binaires, $x \mapsto \omega^x$ est unaire. Elles sont élémentaires et représentent respectivement l'addition, la multiplication et l'exponentiation en base ω ;*

   $n \mapsto \lambda_n$ est une fonction élémentaire ;
7. *$\langle A, \triangleleft, +, \times, \omega^x \rangle$ satisfait les propriétés algébriques d'associativité et distributivité des trois opérations ordinales considérées ainsi que les clauses usuelles de définition de celles-ci, et ce, prouvablement dans* **ERA** *;*
8. *si $\overline{n}$ dénote le $n^{ième}$ élément de l'ordre sur A, alors $n \mapsto \overline{n}$ est élémentaire ;*
9. *Soit $\alpha = \omega^{\beta_1} + \cdots + \omega^{\beta_k}$, $\beta_1 \geq \cdots \geq \beta_k$ un ordinal en forme normale de Cantor, alors la correspondance $\alpha \leftrightarrow \langle \beta_1, \ldots, \beta_k \rangle$ est élémentaire.*

Un *EORS* nous donne donc accès à une notation ordinale sur les ordinaux inférieurs à un certain α, ordinal-type d'une relation d'ordre $\triangleleft$ sur A. Nous répétons ci-dessous la définition 1.2.7 du schéma d'α-récursion. Il prend tout son sens maintenant que nous savons que celle-ci se base sur un *EORS*. En effet, c'est ce dernier qui fournit la fonction de notation $n \mapsto \overline{n}$ (et donc la notation $\overline{0}$) et la fonction de comparaison pour la relation $\lhd$.

Définition 4.1.4 **Schéma d'α-récursion.** *Le schéma de récursion sur $\lhd$ d'ordinal type α s'exprime de la manière suivante :*

$$\mathbf{rec}_{\alpha,\lhd}(g,h,n,\vec{x}) = \begin{cases} g(n,\vec{x},\mathbf{rec}_{\alpha,\lhd}(g,h,\theta(n,\vec{x}),\vec{x})) & \text{si } \overline{0} \lhd n \lhd \alpha \text{ et } \theta(n,\vec{x}) \lhd n, \\ h(n,\vec{x}) & \text{sinon,} \end{cases}$$

où $\overline{0}$ représente l'entier codant l'ordinal 0.

En utilisant un EORS sur α, la théorie de la preuve nous permet directement de construire une énumération des fonctions α-récursives.

Il est naturel de se demander jusqu'où il est possible d'aller avec les *EORS* et donc avec ce schéma d'α-récursion. Pour l'instant, toutes les théories arithmétiques qui ont été analysées avec succès l'ont été par le biais d'un *EORS*, ce qui laisse présager que la construction d'un *EORS* sur un ordinal *naturel* devrait la plupart du temps être possible. Cela n'est pas pour autant prouvé, et les notations ordinales sont des outils souvent très complexes à mettre à jour. C'est notamment pour s'en affranchir que nous verrons dans le paragraphe 4.2 comment faire de la récursion sur des ordinaux admissibles quelconques.

4.1.2 Jusqu'à la calculabilité

Nous voyons que les sous-calculabilités définissent une collection d'approximations de la calculabilité, comportant toujours plus de fonctions récursives, et se rapprochant toujours plus du cas classique. Un des moyens que nous avons utilisé pour construire celles-ci est l'α-récursion. Nous allons maintenant tenter de savoir si toutes les fonctions récursives sont capturées par l'α-récursion. Dans cette optique, nous étudierons notamment les ordinaux récursifs et le $\mathcal{O}$ de Kleene [Kle38], représentant tous les bons ordres récursifs, relativisés aux sous-calculabilités.

Fonctions récursives et sous-calculabilités

Une question qui se pose naturellement est celle des limites des sous-calculabilités en générale et de celles définies par α-récursion en tant que familles de classes de fonctions basées sur des notations ordinales issues d'analyses ordinales en particulier. Nous voudrions savoir si pour toute fonction récursive ψ il existe un α ordinal dénombrable récursif tel que ψ soit α-récursive.

Rappelons d'abord qu'un ordinal est récursif si la calculabilité classique permet de calculer récursivement une relation d'ordre isomorphe.

Définition 4.1.5 **Ordinaux récursifs.** *Un ordinal $\alpha \in \omega_1$ est récursif s'il existe*

R une relation binaire et f une fonction récursive telle que R soit isomorphe à $\in\restriction_\alpha$, et f soit la fonction caractéristique de R.

Il est possible d'identifier l'ensemble des ordinaux calculables en tant que segment initial des ordinaux dénombrables.

Propriété 4.1.6

Les ordinaux récursifs forment un préfixe des ordinaux. *Le plus petit ordinal non-récursif est ω_1^{CK}, et aucun ordinal supérieur à lui ne l'est.*

Démonstration. En effet, supposons que γ soit un ordinal supérieur à ω_1^{CK} et que S soit une relation binaire récursive isomorphe à $\in\restriction_\gamma$, alors, du fait que $\omega_1^{\mathrm{CK}} \in \gamma$, il existerait $e \in S$ tel que $R = \{\langle x, y\rangle : S(x,y) \wedge S(y,e)\}$ soit isomorphe à $\in_{\omega_1^{\mathrm{CK}}}$. Ce qui est absurde, donc tous les ordinaux récursifs sont membres de ω_1^{CK}, donc ω_1^{CK} contient exactement tous les ordinaux récursifs.

Nous savons également que $\omega_1^{\mathrm{CK}} \in \omega_1$ puisque tout ordinal récursif est nécessairement dénombrable.

□

En particulier, les EORS dont nous avons parlé sont tous associés à un ordinal récursif α.

Théorème 4.1.7 **Rathjen**

Soit T_A la théorie $\mathbf{PA} + \cup_{\alpha\in A} TI(A_{\overline{\alpha}}, \lhd_{\overline{\alpha}})$ une théorie récursivement axiomatisable contenant l'arithmétique.
Alors, les fonctions récursives pour T_A, c'est à dire les fonctions prouvablement totales dans T_A, sont exactement les fonctions α-récursives, pour $\alpha \in A$.

De manière générale, si nous disposons de l'analyse ordinale d'une théorie T et d'un EORS associé, celle-ci permet d'énumérer les fonctions prouvablement récursives dans cette théorie.

À cela, Michael Rathjen rajoute que, pour l'instant, les théories qui ont pu être analysées nous ont fourni un *EORS*. Nous pouvons donc imaginer que dans toute théorie raisonnable prouvant la totalité de certaines fonctions, il sera possible de trouver un EORS associé permettant d'établir la liste de ses fonctions prouvablement totales.

Remarquons ici que nous parlons de théories basées sur **PA** et non sur **ERA**. En fait, $\mathbf{ERA} + \cup_{\alpha\in A} TI(A_{\overline{\alpha}}, \lhd_{\overline{\alpha}})$ pour $\lhd$ d'ordinal type ϵ_0 est un fragment du premier ordre de **PA**.

Note 4.1.8 (Rathjen) *Toutes les analyses ordinales des théories intuitionnistes ou classiques connues à ce jour viennent de paire avec un* EORS *$\langle A, \lhd, \dots\rangle$ tel que T se réduise du point de vue de la théorie de la preuve à $\mathbf{PA} + \cup_{\alpha\in A} TI(A_{\overline{\alpha}}, \lhd_{\overline{\alpha}})$*

Les travaux de Rathjen nous permettent de déduire que pour toutes les fonctions prouvablement totales dans une théorie logiquement équivalente, pour un EORS $\langle A, \lhd \rangle$ à $\mathbf{PA} + \cup_{\alpha \in A} TI(A_{\overline{\alpha}}, \lhd_{\overline{\alpha}})$, vont pouvoir être définissables par α-récursion et donc être présentent à partir d'un certain *niveau* dans les sous-calculabilités.

Ce résultat n'est pas suffisant pour prouver que toutes les fonctions calculables sont α-récursives pour un certain α. Cependant nous en faisons la conjecture.

Conjecture 4.1.9

Toute fonction récursive est α-récursive pour un certain ordinal α récursif ($\alpha \in \omega_1^{CK}$.)

Lorsque nous cherchons une raison pour laquelle cette conjecture pourrait être fausse, nous pensons immédiatement au procédé de diagonalisation, ce qui nécessiterait d'avoir une fonction totale qui utiliserait suivant l'argument donné, des schémas de récursion arbitrairement grand. Le fait est que pour faire cela, la fonction devrait, soit être capable de discerner les relations bien fondées des autres, ce qui est un problème Π_1^1, soit se tromper parfois, et donc être partielle.

Une autre piste pour tenter de prouver que la conjecture est fausse serait d'identifier un ordinal récursif à partir duquel nous ne puissions plus construire d'EORS. Au vu de la remarque de Rathjen, cette piste semble peu probable, et quand bien même elle serait avérée, il nous serait toujours possible d'envisager l'utilisation d'*ORS* basés sur une théorie plus puissante que *ERA*.

Note 4.1.10 *N'oublions pas cependant qu'il est impossible de trouver un ordinal α tel que la sous-calculabilité engendrée par les fonctions fondamentales α-récursives soit égale à la calculabilité classique. En effet, les classes fondamentales récursives requièrent qu'il existe une fonction universelle récursive permettant de les simuler toutes. Or il n'y a pas de telle fonction.*

Aussi, s'il est possible d'espérer montrer que pour toute fonction récursive totale f il existe un ordinal récursif α tel que f soit α-fondamentale, il n'existe en revanche aucun ordinal récursif α tel que pour toute fonction récursive totale f, f soit α-récursive.

Ordinaux récursifs et sous-calculabilités

En calculabilité classique, il existe un système standard de représentation de l'ensemble des relations d'ordre récursive, $\mathcal{O}$: le *grand «o»* de Kleene.

$\mathcal{O}$ possède diverses propriétés, notamment de Π_1^1-complétude, ce qui fait de lui un oracle polyvalent et facilement utilisable. Un premier point intéressant est le fait que $\mathcal{O}$ soit d'ordinal type ω_1^{CK}, il permet donc de représenter chacun des ordinaux récursifs. Sa structure reflète ainsi un aspect de la complexité de la classe des fonctions récursives, puisqu'il associe à chaque ordinal récursif l'une d'entre elles.

Deuxièmement, $\mathcal{O}$ est classiquement utilisé pour construire la classe des ensembles hyperarithmétiques, et nous voudrions étudier la relativisation de ceux-ci aux sous-calculabilités.

Pour rappel, voici la définition de $\mathcal{O}$ dans le cas classique :

Définition 4.1.11 **Notation $\mathcal{O}$ de Kleene.** *L'ensemble $\mathcal{O}$ de Kleene est défini par induction avec son ordre $<_\mathcal{O}$ de la manière suivante :*

— $0 \in \mathcal{O}$;
— *si* $n \in \mathcal{O}$, *alors* $2^n \in \mathcal{O}$ *et* $n <_\mathcal{O} 2^n$;
— *si* φ_e *est totale et* $\forall n, \varphi_e(n) <_\mathcal{O} \varphi_e(n+1)$ *alors* $3.5^e \in \mathcal{O}$ *et* $\forall n, \varphi_e(n) <_\mathcal{O} 3.5^e$;

Grâce à ce $\mathcal{O}$, nous pouvons notamment définir la famille des *H-ensembles*, encore appelés *ensembles hyperarithmétiques*, dont la collection forme l'ensemble **HYP**.

Définition 4.1.12 **Ensembles hyperarithmétiques.** *Les ensembles hyperarithmétiques (H-ensembles) sont définis par induction transfinie sur $\mathcal{O}$ de la manière suivante :*

— $H_1 = \varnothing$
— $H_{2^n} = (H_n)'$ *(avec $'$ le saut Turing)*
— $H_{3.5^e} = \{\langle m, n\rangle : m \in H_{\varphi_e(n)}\}$

$\mathbf{HYP} = \{H_e : e \in \mathcal{O}\}$ *désigne l'ensemble des ensembles hyperarithmétiques.*

Les ensembles hyperarithmétiques nous permettent entre autre de parler des itérations ordinales du *saut*. C'est à dire de considérer pour tout $\alpha \in \omega_1^{\mathrm{CK}}$, le degré $\mathbf{0}^{(\alpha)}$, correspondant dans la hiérarchie hyperarithmétique aux formules Σ^0_α.

Dans le cas des sous-calculabilités, nous définissons $\mathcal{O}_\mathsf{C}$ et $\mathbf{HYP}_\mathsf{C}$ de manière similaire :

Définition 4.1.13 **Notation $\mathcal{O}_\mathsf{C}$ relativisée.** *Soit C une classe fondamentale. L'ensemble $\mathcal{O}_\mathsf{C}$ de Kleene relativisé à la sous-calculabilité basée sur C est défini par induction avec son ordre $<_{\mathcal{O}_\mathsf{C}}$ de la manière suivante :*

— $0 \in \mathcal{O}_\mathsf{C}$;
— *si* $n \in \mathcal{O}_\mathsf{C}$, *alors* $2^n \in \mathcal{O}_\mathsf{C}$ *et* $n <_{\mathcal{O}_\mathsf{C}} 2^n$;
— *si* Φ^C_e *est totale et* $\forall n, \Phi^\mathsf{C}_e(n) <_{\mathcal{O}_\mathsf{C}} \Phi^\mathsf{C}_e(n+1)$ *alors* $3.5^e \in \mathcal{O}_\mathsf{C}$ *et* $\forall n, \Phi^\mathsf{C}_e(n) <_{\mathcal{O}_\mathsf{C}} 3.5^e$;

Avec cette définition, il n'est pas clair que tous les ordinaux récursifs soient représentés par un élément de $\mathcal{O}_\mathsf{C}$.

Il est possible d'additionner les éléments de $\mathcal{O}_\mathsf{C}$ de manière à respecter les additions sur les ordinaux :

Propriété 4.1.14

Soit $\mathtt{C}$ *une classe fondamentale. Il existe un opérateur binaire* $\mathtt{C}$*-fondamental* $+_{\mathcal{O}_{\mathtt{C}}}$ *tel que*

$$\forall x <_{\mathcal{O}_{\mathtt{C}}} y \in \mathcal{O}, \begin{cases} & o\left(\{\langle u,v\rangle : u <_{\mathcal{O}_{\mathtt{C}}} v \wedge u <_{\mathcal{O}_{\mathtt{C}}} x \wedge v <_{\mathcal{O}_{\mathtt{C}}} x\}\right) \\ + & o\left(\{\langle u,v\rangle : u <_{\mathcal{O}_{\mathtt{C}}} v \wedge u <_{\mathcal{O}_{\mathtt{C}}} y \wedge v <_{\mathcal{O}_{\mathtt{C}}} y\}\right) \\ = & o\left(\{\langle u,v\rangle : u <_{\mathcal{O}_{\mathtt{C}}} v \wedge u <_{\mathcal{O}_{\mathtt{C}}} (x +_{\mathcal{O}_{\mathtt{C}}} y) \wedge v <_{\mathcal{O}_{\mathtt{C}}} (x +_{\mathcal{O}_{\mathtt{C}}} y)\}\right). \end{cases}$$

Démonstration. Nous voulons construire une fonction $\Phi^{\mathtt{C}}_c$ telle qu'elle représente $+_{\mathcal{O}_{\mathtt{C}}}$. Pour ce faire, nous allons l'écrire un peu comme s'il s'agissait des ordinaux eux-mêmes. L'idée est de transformer le deuxième indice de sorte qu'en considérant l'arbre engendré par son énumération, les feuilles ne portent plus l'étiquette 0 mais l'étiquette du premier indice. Ainsi, nous *branchons* le premier ordinal à chaque feuille du second, ce qui correspond bien à l'addition d'ordinaux.

Soit g une fonction $\mathtt{C}$-fondamentale telle que

$$\Phi^{\mathtt{C}}_{g(e,m,n)} : k \mapsto \Phi^{\mathtt{C}}_e(m, \Phi^{\mathtt{C}}_n(k))$$

qu'il est possible de définir grâce au théorème s^m_n, et $\Phi^{\mathtt{C}}_c$ une autre fonction $\mathtt{C}$-fondamentale telle que

$$\Phi^{\mathtt{C}}_c : \langle m,n\rangle \mapsto \begin{cases} m & \text{si } n = 1, \\ 2^{\Phi^{\mathtt{C}}_c(m,k)} & \text{si } n = 2^k, \\ 3.5^{g(c,m,k))} & \text{si } n = 3.5^k, \\ 7 & \text{sinon.} \end{cases}$$

Une fois de plus, remarquons que le point fixe utilisé peut être remplacé par un schéma de récursion primitive, ce que nous ne ferons pas pour préserver la lisibilité de la fonction.

Posons $\cdot +_{\mathtt{O}_{\mathtt{C}}} \cdot = \Phi^{\mathtt{C}}_c(\langle\cdot,\cdot\rangle)$, nous remarquons qu'alors l'addition se comporte belle et bien comme souhaité.

□

Nous allons maintenant tenter de trouver une caractérisation de ces ordinaux représentés dans $\mathcal{O}_{\mathtt{C}}$. Une des données intéressantes du $\mathcal{O}$ de Kleene sur les calculabilités, est que chaque élément e permet de reconstruire de manière récursive un ordre sur les naturels qui est isomorphe à l'ensemble des prédécesseurs de e dans $\mathcal{O}$ pour $<_{\mathcal{O}}$.

Cette opération n'est pas donnée d'avance dans le cas des sous-calculabilités du fait de notre contrainte d'injectivité. Aussi, nous adaptons la définition d'un ordinal récursif comme suit :

Définition 4.1.15 **Ordinaux héréditairement faiblement-$\mathtt{C}$-récursifs.** *Soit*

$\mathtt{C}$ *une classe fondamentale et* α *un ordinal dénombrable.* α *est héréditairement faiblement-*$\mathtt{C}$*-récursif si tous les points suivants sont satisfaits :*

— *il existe une relation* R *d'ordinal-type* α *;*
— *il existe* e *tel que* $R = \mathbf{W}^{\mathtt{C}}_e$ *;*
— *il existe une fonction* f $\mathtt{C}$*-fondamentale telle que*
  — $\forall a \in \mathrm{range}(R)$, $\mathbf{W}^{\mathtt{C}}_{f(e,a)} = \{\langle x, y\rangle : \{\langle x, y\rangle, \langle x, a\rangle, \langle y, a\rangle\} \subset R\}$,
  — $\mathbf{W}^{\mathtt{C}}_{f(e,x)}$ *est faiblement-*$\mathtt{C}$*-récursif.*

Il se trouve que cette définition est une bonne caractérisation des ordinaux représentables par $\mathcal{O}_{\mathtt{C}}$.

Propriété 4.1.16

Soit $\mathtt{C}$ *une classe fondamentale et* β *un ordinal tel que la classe des fonctions* β*-récursives est inclue dans* $\mathtt{C}$*.*
Si β *est héréditairement faiblement-*$\mathtt{C}$*-récursif alors il est représenté par un élément de* $\mathcal{O}$*, et réciproquement.*

Démonstration. Étant donné $e \in \mathcal{O}_{\mathtt{C}}$, nous voulons construire une relation R faiblement-$\mathtt{C}$-récursive isomorphe à l'ordinal représenté par e.

On note f la fonction associée à β (par la Définition 4.1.15). Soit $r = r_e$ et $s = s_e$ deux fonctions $\mathtt{C}$-fondamentales définies comme suit en fonction de e :

$$\Phi^{\mathtt{C}}_{r_e(a)} : \begin{cases} 0 & \mapsto \ 1 \\ n+1 & \mapsto \ \Phi^{\mathtt{C}}_{f(e,a)}(n) \end{cases}$$
$$\Phi^{\mathtt{C}}_{s_e(a)} : \begin{cases} 0 & \mapsto \ 1 \\ n+1 & \mapsto \ \Phi^{\mathtt{C}}_{s_e(a)}(n) +_{\mathcal{O}_{\mathtt{C}}} 2^{\Phi^{\mathtt{C}}_{r_e(a)}(n+1)} \end{cases}$$

Ces deux fonctions sont $\mathtt{C}$-fondamentales puisque l'utilisation du point-fixe dans la deuxième est en fait un schéma de récursion primitive.

Finalement, nous définissons $p = p_e$ telle que

$$\Phi^{\mathtt{C}}_{p_e(i)} : n \mapsto \begin{cases} 0 & \text{si } \Phi^{\mathtt{C}}_e(0) = \Phi^{\mathtt{C}}_e(1), \\ \Phi^{\mathtt{C}}_i(f(e,n)) & \text{sinon.} \end{cases}$$

Ici, il convient de remarquer un élément important. L'indice i donné à la fonction p va servir à réaliser un point-fixe. Ce point-fixe ne pourra être réalisé par les fonctions $\mathtt{C}$-fondamentales seulement s'il peut être remplacé par un schéma de récursion. Or, l'argument utilisé ici est un élément de $\mathcal{O}_{\mathtt{C}}$ qui décroît suivant $<_{\mathcal{O}_{\mathtt{C}}}$. Nous en déduisons que si l'ordre représenté par e, $<_{\mathcal{O}_{\mathtt{C}}}\restriction_e$, est d'ordinal-type β il suffit que la classe des fonctions $\mathtt{C}$-fondamentales étudiée contienne les fonctions β-récursives.

L'utilisation du schéma de récursion permet alors de calculer l'indice c d'une fonction telle que

$$\Phi^{\mathtt{C}}_{2.5^{p_e(c)}} \cong \Phi^{\mathtt{C}}_c.$$

Inversement, si e est un élément de $\mathcal{O}_{\mathsf{c}}$, son indice permet par construction de $\mathcal{O}_{\mathsf{c}}$ d'énumérer les éléments inférieurs à e de manière c-fondamentale. Afin de préserver l'injectivité de l'énumération, il est possible de commencer par se ramener à un ordinal limite e' en appliquant tant que possible $\log_2$ à e. Grâce à cela, nous pourrons nous assurer d'avoir en permanence un élément nouveau $m +_{\mathcal{O}_{\mathsf{c}}} 1$ à énumérer, où m est le plus grand élément inférieur à e' énuméré jusqu'à présent.

□

Avec cette caractérisation et les idées et difficultés rencontrées dans la preuve, notamment dans le remplacement du point fixe par un schéma de récursion, nous pouvons nous lancer dans quelques conjectures quant aux ordinaux types des différents $\mathcal{O}_{\mathsf{c}}$ définis.

Conjecture 4.1.17

Nous conjecturons les résultats suivant pour les ordinaux-types de quelques c-calculabilités :

- *l'ordinal-type de $\mathcal{O}_{\mathsf{p}}$ est ω^ω,*
- *l'ordinal-type de $\mathcal{O}_{\mathsf{c}_{\omega^\omega}}$ est ϵ_0.*

Maintenant que $\mathcal{O}_{\mathsf{c}}$ est défini, nous pouvons définir des ensembles hyperarithmétiques de façon similaire au cas classique.

Définition 4.1.18 **Ensembles hyperarithmétiques relativisés.** *Les H_{c}-ensembles sont définis par induction transfinie sur $\mathcal{O}_{\mathsf{c}}$ de la manière suivante :*

- $H_{\mathsf{c},1} = \varnothing$
- $H_{\mathsf{c},2^n} = (H_n)'$ *(avec $'$ le saut des sous-calculabilités)*
- $H_{\mathsf{c},3.5^e} = \left\{\langle m, n\rangle : m \in H_{\Phi_e^{\mathsf{c}}(n)}\right\}$

$\mathbf{HYP}_{\mathsf{c}} = \{H_{\mathsf{c},e} : e \in \mathcal{O}\}$ *désigne cette fois l'ensemble des ensembles hyperarithmétiques relativement à c, que nous appellerons* ensembles c-hyperarithmétiques.

Si notre conjecture 4.1.17 est avérée, alors $\mathbf{HYP}_{\mathsf{p}}$ nous fournit une hiérarchie complète de p-degrés entre la p-calculabilité et jusqu'à son ω^ω-saut. Et de même entre la $\mathsf{c}_{\omega^\omega}$-calculabilité et son α-saut, où α est le *supremum* des γ tel que les fonctions γ-récursives sont inclues dans $\mathsf{c}_{\omega^\omega}$.

Conjecture 4.1.19

Nous conjecturons qu'il en va de même pour toute une famille de sous-calculabilités dont la limite serait la calculabilité classique, sans que celle-ci ne soit jamais atteinte.

Note 4.1.20 *En effet, la limite ne peut jamais être atteinte puisque la hiérarchie n'a aucune chance d'être récursive. Si c'était le cas, nous aurions une notation ordinale jusqu'à ω_1^{CK} qui serait récursive, ce qui est impossible.*

Notons d'ailleurs que le $\mathcal{O}$ de Kleene de la calculabilité classique, d'ordinal type ω_1^{CK}, est un ensemble Π_1^1-complet qui transcende tous les degrés hyperarithmétiques, et est donc très loin d'être récursif. Pour autant, tous ses préfixes sont récursivement énumérables.

4.2 Récursion sur les admissibles et sous-calcualbilités

La calculabilité classique, dont nous avons revu quelques résultats au fil des chapitres précédents, s'effectue sur les entiers, dans le cas particulier où $\mathcal{D} = \omega$. C'est une situation qui permet d'assimiler une grande partie de la calculabilité à la logique du premier ordre, et inversement. Cependant, il est également possible de se représenter la calculabilité classique d'un point de vue ensembliste. Le *codage de von Neumann*[3] permet de représenter chaque entier par l'ensemble des représentations des entiers positifs qui le précèdent et, à partir de là, les opérations de base de successeur, prédécesseur, puis d'addition et de soustraction, *etc.* sont facilement transposables. Ces axes de lecture pour le calcul mènent tous les deux à la même notion de calculabilité, il s'agit, en quelque sorte, simplement d'implémentations différentes.

L'intérêt survient lorsque ces notions sont généralisées aux ensembles dénombrables dans un premier temps, puis quelconque ensuite. Cette généralisation soulève tout un lot de questions sur l'interprétation des opérateurs classiques dans le cadre généralisé. Par exemple le cas du schéma μ et du schéma de récursion primitive, deux opérateurs clés de la calculabilité classique. Dans le cas $\mathcal{D} = \omega$, le schéma μ est un schéma de recherche sur tous les éléments de l'univers, c'est à dire de $\mathcal{D}$, qui consiste également en l'ensemble des éléments descriptibles finiment de l'univers, puisque tous les entiers sont de cofinalité nulle. Ce n'est plus le cas si $\mathcal{D} = \omega_1$. Il est possible soit de généraliser le schéma μ d'une manière inductive, en permettant l'itération sur tous les éléments énumérables, soit au contraire de permettre une recherche non bornée sur tous les éléments de l'univers qui existent. Ces descriptions simplificatrices donnent en fait l'intuition permettant de discerner la théorie de la E-récursion de G. E. Sacks, et la théorie de la Σ-récursion, ou théorie de la récursion sur les admissibles, qui sont les deux extensions majeures de la calculabilité aux ensembles. Revenons maintenant au cas de l'opérateur de récursion primitive. Ce dernier permet dans le cas classique d'effectuer un nombre fini d'opérations en effectuant une descente bornée d'un ordinal fini, de plus, il implique le schéma μ borné, effectuant une recherche bornée d'éléments de l'ordinal fini considéré. Nous

3. Si on note par $\overline{n}$ la représentation d'un entier n, $\overline{0} = \varnothing$ et $\overline{n} = \{\overline{i} : i < n\}$.

disposons également de deux généralisations possibles de celui-ci. On peut soit permettre de descendre n'importe quel ordinal (ce qui n'est possible qu'en un nombre, certes non-borné, mais fini d'étapes), soit permettre la recherche Δ_0 d'éléments dans l'ordinal donné. Différents points de vue sont adoptés dans la récursion que les admissibles (que nous allons traiter ici) et la E-récursion, que nous exposons en Annexe C.

La théorie des ensembles admissibles a été développée par Jon Barwise et est largement traitée dans son ouvrage [Bar75]. Elle s'intéresse aux ensembles satisfaisant les axiomes de la théorie de Kripke-Platek (**KP**), qui est elle-même une *petite* théorie des ensembles, dans le sens où des axiomes de la théorie des ensembles de Zermelo-Franenkel (**ZF**) sont restreints à la classe des formules Δ_0, c'est à dire aux formules sans quantificateurs non bornés.

De ce fait, une certaine analogie avec la calculabilité émane naturellement. En effet, en calculabilité classique, les fonctions Δ_0^0 sont toutes totales calculables, et les fonctions partielles sont définissables comme l'ensemble des fonctions définissables par une formule Σ_1^0.

De même, **KP** satisfaisant l'axiome du Σ-remplacement, il est possible de définir les fonctions partielles (via leurs graphes) de la même manière qu'en calculabilité classique. Ainsi, toute une théorie de la récursion a été développée sur les ensembles admissibles, dont nous rappellerons certains des résultats fondamentaux que nous mettrons en parallèle avec les résultats et propriétés observés précédemment dans les classes des sous-calculabilités.

Cette thèse n'est pas vouée à donner une présentation complète ou détaillée de la récursion sur les ensembles. Le lecteur trouvera en Annexe B et C une présentation concise respectivement de la théorie des ensembles de Kripke-Platek, base de la récursion sur les admissibles, et de la E-recursion, Nous l'encourageons cependant à se référer à [Sac90] et [Bar75] pour une étude bien plus riche du domaine.

Ces objets disposent en fait de nombreuses propriétés proches de celles de la calculabilité. Sur un ensemble admissible $\mathbb{A}$, nous allons identifier une structure riche d'une énumération $\mathbb{A}$-récursive, d'un théorème à la s_n^m, d'un théorème de la récursion, d'un théorème de paramétrisation. Seulement le formalisme habituel des calculabilités sur les admissibles est assez déroutant par rapport aux calculabilités usuelles. C'est pourquoi, nous avons par la suite rappelé les définitions et les résultats sur les calculabilités admissibles en utilisant le formalisme introduit précédemment pour les sous-calculabilités.

4.2.1 Récursion sur les ordinaux admissibles

Nous avons succinctement présenté, dans le paragraphe précédent, la structure sous-jacente aux modèles de la théorie de **KP**. Ce faisant, nous avons commencé à identifier un certain nombre d'ensembles dont les définitions étaient proches des objets que nous avons l'habitude de manipuler en calculabilité classique. Dans ce paragraphe, nous allons donner un peu plus de sens à ces ensembles en fournissant une interprétation calculabiliste de ceux-ci.

Nos objets de base étant cette fois-ci non pas des fonctions mais des ensembles, nous commençons par définir les notions de récursivité sur des relations, les fonctions n'étant alors qu'un cas particulier de celles-ci.

Définition 4.2.1 **Ensembles E-récursivement énumérables.** *Soit $\mathbb{A}$ un ensemble admissible et R une relation sur $\mathbb{A}$.*

— *R est $\mathbb{A}$-récursivement énumérable ($\mathbb{A}$-r.e.) si R est Σ_1 sur $\mathbb{A}$;*
— *R est $\mathbb{A}$-récursive si R est Δ_1 sur $\mathbb{A}$;*
— *R est $\mathbb{A}$-finie si $R \in \mathbb{A}$;*

La notion de $\mathbb{A}$-énumérabilité d'une relation est équivalente à la Σ_1-définissabilité en $\mathbb{A}$. Autrement dit, une relation R est $\mathbb{A}$-énumérable s'il est possible de trouver une formule ϕ de complexité Σ_1, telle que $R = \{x \in \mathbb{A} : \phi(x, y_1, \ldots, y_n)\}$ pour $y_1, \ldots, y_n$ des éléments de $\mathbb{A}$.

La définition de fonction est ensuite calquée sur celle de la calculabilité et correspond également à celle définie dans les sous-calculabilités :

Définition 4.2.2 **Fonctions $\mathbb{A}$-récursives.** *Une fonction f dont le domaine et l'image sont des sous-ensembles de $\mathbb{A}$ est $\mathbb{A}$-récursive si son graphe est* $\mathbb{A}$-r.e.

Notons en particulier que si une fonction f est totale, il est alors possible d'énumérer le complémentaire de son graphe, c'est à dire de l'exprimer de manière Σ_1. Une des justifications possibles est très similaire au cas de la calculabilité classique, dans lequel le raisonnement peut-être le suivant : pour savoir si x est un élément du complémentaire du graphe de f, regardons ce que vaut $f(\pi_1^2(x))$ et s'il est égal à $\pi_2^2(x)$ alors non, sinon, oui. Plus formellement, si le graphe de f est l'ensemble des x satisfaisant une formule $\phi(x)$ de classe Σ_1^0, le complémentaire du graphe de f est l'ensemble des x vérifiant $\exists y, (\phi(y) \wedge \pi_1^2(y) = \pi_1^2(x) \wedge \pi_2^2(y) \neq \pi_2^2(x))$, qui est bien une formule Σ_1^0. Remarquons que la transposition dans le langage de la théorie **KP** fonctionne sans problème, les opérations de pairage et projection étant Δ_0, la formule résultante est bien Σ_1. Cette propriété s'exprime par le résultat suivant :

Propriété 4.2.3

Soit $\mathbb{A}$ un ensemble admissible. Une fonction totale $\mathbb{A}$-récursive a un graphe $\mathbb{A}$-récursif.

Nous obtenons ainsi une propriété similaire à la calculabilité classique.

4.2.2 Énumération

Avant de pouvoir étudier plus en détail les propriétés structurelles de points fixes ou encore de manipulation d'indices, il nous faut définir la notion d'indice et donc d'énumération pour les calculabilités sur les admissibles.

Dans le cas classique, les nombres entiers conviennent tout à fait pour définir une énumération, ils permettent d'encoder via une numérotation à la Gödel les codes des programmes. Ici, il serait possible de faire de même en encodant les syntaxes des formules Σ_1. Cependant, ces formules sont à paramètres dans $\mathbb{A}$, un ensemble arbitrairement grand (tous les cardinaux infinis étant admissibles) et donc dont les éléments ne sont pas nécessairement tous encodables par des entiers. Heureusement pour nous, pour les modèles admissibles de la forme $\mathbb{A} = \mathbb{L}_\alpha$, il existe une bijection naturelle et calculable (Σ_1) vers α. Nous pouvons donc commencer par énumérer tous les objets de $\mathbb{A}$, de la même manière que nous énumérions nos classes fondamentales dans les sous-calculabilités.

Nous allons considérer comme fonctions fondamentales les fonctions définissables à partir des $\mathbb{A}$-finis, que nous confondrons avec les ensembles $\mathbb{A}$-finis.

Définition 4.2.4 **Fonctions $\mathbb{A}$-fondamentales.** *Les fonctions $\mathbb{A}$-fondamentales sont les fonctions dont le graphe est un élément de $\mathbb{A}$. Par commodité, nous considérerons qu'un ensemble qui n'est pas un graphe représente quand même une fonction : celle dont le graphe est inclus dans $\mathbb{A}$ et est le plus grand possible suivant la relation d'ordre entre ensembles constructibles. Cette relation d'ordre étant Δ_0 cela n'induira pas d'artefact en terme de puissance de calcul des fonctions.*

Il est possible de générer *mécaniquement* les éléments de $\mathbb{L}_\alpha$ à partir des opérations de Gödel et de l'ordinal α. Cette génération nous définit une bijection entre l'ordinal admissible et son modèle des constructibles associé :

Propriété 4.2.5

Si α est un ordinal admissible, il existe une bijection Σ_1-définissable entre $\mathbb{L}_\alpha$ et α.

De cela, nous déduisons comme premier résultat l'existence d'une énumération pour les fonctions $\mathbb{A}$-fondamentales :

Théorème 4.2.6

Soit $\mathbb{A}$ un ensemble admissible. Il existe une énumération Σ_1 des fonctions $\mathbb{A}$-fondamentales. Nous noterons $\boldsymbol{\Phi}^{\mathbb{A}}_{\cdot}$ cette énumération.

Démonstration. Pour ce faire, il nous faut énumérer toutes les formules Δ_0 possibles avec tous les paramètres dans $\mathbb{A}$ possibles. En utilisant la numérotation de Gödel pour les formules, il est possible d'encoder chacune d'entre elle par un entier e qui noterait les positions des arguments de $\mathbb{A}$. Le tuple $\langle a_0, \ldots, a_n \rangle$ des éléments de $\mathbb{A}$ étant lui-même un élément de $\mathbb{A}$ et donc l'image par notre bijection de $\mathbb{A}$, il est possible de lui associer l'ordinal γ, antécédent du tuple par celle-ci. Il est possible d'encoder la formule et ses arguments dans les ordinaux, par exemple par l'ordinal $\gamma.\omega + e$.

□

À partir de là, il est naturel de vouloir considérer l'énumération des ensembles possiblement énumérables par des formules Σ_1 à paramètres dans $\mathbb{A}$. C'est à dire en fait une énumération de la collection des ensembles $\mathbb{A}$-*r.e.*.

Comme ci-dessus, il nous faut énumérer toutes les formules, Σ_1 cette-fois, possibles avec tous les paramètres dans $\mathbb{A}$ possibles. On utilise à nouveau la numérotation de Gödel pour encoder chaque formule par un entier e. Le tuple $\langle a_0, \ldots, a_n \rangle$ des éléments de $\mathbb{A}$ étant lui-même un élément de $\mathbb{A}$, il est énuméré par la bijection canonique à un rang γ. Nous pouvons encore une fois encoder la formule et ses arguments dans les ordinaux, par exemple par l'ordinal $\gamma.\omega + e$.

Définition 4.2.7 **Énumération des ensembles $\mathbb{A}$-énumérables.** $\mathbb{W}^{\mathbb{A}}_{.}$ *désigne l'énumération des ensembles $\mathbb{A}$-énumérable induite par* $\mathbf{\Phi}^{\mathbb{A}}_{.}$.

Comme nous l'avons fait précédemment dans le cas des sous-calculabilités, nous pouvons définir les fonctions partielles à partir de cette énumération d'ensembles.

Théorème 4.2.8

Il existe une énumération des fonctions $\mathbb{A}$-partielles. *Soit $\mathbb{A}$ un ensemble admissible. $\varphi^{\mathbb{A}}_{.}$ désignera cette énumération, où pour tout ordinal γ, la fonction $\varphi^{\mathbb{A}}_{.}$ est celle représentée par l'ensemble image de $\mathbf{\Phi}^{\mathbb{A}}_{.}$.*

Avec cela, nous disposons du même formalisme que pour les sous-calculabilités afin de pouvoir étudier ces calculabilités sur les admissibles. En fait, nous allons prouver de nombreuses propriétés nous rapprochant de la possibilité d'interpréter les calculabilités sur les admissibles comme des sous-calculabilités d'ordre supérieur.

4.2.3 Structure

Nos objets de base étant fixés, nous pouvons maintenant nous intéresser à la structure et vérifier que les propriétés algébriques nous ayant permis dans le cas classique de manipuler les fonctions, de les construire et de les composer sont toujours de mise dans le cas de la calculabilité admissible.

En particulier, les trois propriétés de base que nous voulons vérifier ici sont la paramétrisation, le théorème s^m_n et le théorème de la récursion à la Kleene.

Propriété 4.2.9

Soit $\mathbb{A}$ un ensemble admissible. Il existe une fonction s_n^m pour $\Phi^{\mathbb{A}}_{\cdot}$ et pour $\varphi^{\mathbb{A}}_{\cdot}$.

Démonstration. Les formules Δ_0 ou Σ_1 étant encodées sous forme de numéros de Gödel, une simple fonction récursive primitive est capable de remplacer dans celles-ci certains des arguments de la fonction par des positions pour des arguments de $\mathbb{A}$, qu'il faudra ensuite rajouter au tuple des paramètres. Cette opération étant également Δ_0, notre fonction s_n^m est bien une fonction $\mathbb{A}$-fondamentale.

□

Il est toujours possible d'effectuer de la paramétrisation :

Propriété 4.2.10

Soit $\mathbb{A}$ un ensemble admissible. Il existe une fonction de manipulation d'indice permettant la paramétrisation dans $\Phi^{\mathbb{A}}_{\cdot}$ et dans $\varphi^{\mathbb{A}}_{\cdot}$

Démonstration. Dans le cas de l'énumération $\Phi^{\mathbb{A}}_{\cdot}$, c'est à dire des formules Δ_0, la paramétrisation (*si* $\Phi^{\mathbb{A}}_{\alpha}(x) = 0$ *alors calculer* $\Phi^{\mathbb{A}}_{\beta}$, *sinon calculer* $\Phi^{\mathbb{A}}_{\gamma}$) consiste simplement à modifier la formule Δ_0 en écrivant formellement la condition ci-dessus. Nous obtenons par exemple l'énoncé suivant :

$$\mathbf{param}_{\alpha,\beta,\gamma}(x, y, z) : \left(\Phi^{\mathbb{A}}_{\alpha}(x) = 0 \wedge \Phi^{\mathbb{A}}_{\beta}(y) = z\right) \vee \left(\Phi^{\mathbb{A}}_{\alpha}(x) \neq 0 \wedge \Phi^{\mathbb{A}}_{\gamma}(y) = z\right).$$

Cette opération étant une simple manipulation des numéros de Gödel, elle est une fois de plus Δ_0 et donc $\mathbb{A}$-fondamentale.

Dans le cas de l'énumération $\varphi^{\mathbb{A}}_{\cdot}$, la preuve est tout à fait similaire, si ce n'est que nous manipulons des formules Σ_1. Cependant, il est possible de placer les quantificateurs existentiels en tête de formule et de se rendre compte que la formule obtenue *in fine* est bien Σ_1, et représente donc bien une fonction $\mathbb{A}$-partielle.

□

Enfin, nous en arrivons à la version du second théorème de la récursion pour les calculabilités sur les admissibles [4].

Pour appliquer le raisonnement du point fixe dans le formalisme des admissibles, il nous faut parler de formules Σ_1 dans laquelle est autorisée l'utilisation d'une relation n-aire R. Cette relation sera ensuite substituée par une formule de même arité afin de réaliser le point fixe.

4. Voici ce que Jon Barwise a écrit à son propos :

> [Jon Barwise] The Second Recursion Theorem in ω-recursion theory is a mysterious device for implicitly defining recursive partial functions, or equivalently, *r.e.* predicates. This theorem is equally mysterious and equally useful in our setting.

Proposition 4.2.11

Théorème de la récursion à la Kleene. *Soit* $\mathbb{A}$ *un ensemble admissible et* $\phi(x, y, R)$ *une formule* Σ_1 *sur* $\mathbb{A}$ *avec un symbole de relation* R *dénotant une relation* n*-aire et* x *un* n*-uplet.*
Alors il existe une formule Σ_1 $\psi(x, y)$ *telle que*

$$\mathbb{A} \models \forall y, \forall x, \psi(x, y) \leftrightarrow \phi(x, y, u \mapsto \psi(u, y)).$$

Démonstration. Pour réaliser le point fixe, nous avons besoin d'une formule jouant le rôle de machine universelle pour les formules Σ_1. Nous allons pour cela utiliser une formule que nous appellerons Σ-Sat et qui correspondra au prédicat suivant :

y représente une formule Σ_1 à trois variables libres v_1, v_2 et v_3 et il existe un ensemble transitif a qui soit une structure satisfaisant l'interprétation de y dans laquelle v_1, v_2 et v_3 valent respectivement x_1, x_2 et x_3.

Nous ne détaillerons pas plus Σ-Sat, mais il suffit de voir que c'est une formule Σ_1 vérifiant s'il existe un modèle de la formule encodée par y.

Remplaçons maintenant R dans ϕ et nous obtenons une formule θ Σ_1 :

$$\theta(x, y, z) = \phi(x, y, u \mapsto \Sigma\text{-Sat}(z, u, y, z)).$$

Notre formule θ trouve une représentation dans l'ensemble **HF** des ensembles héréditairement finis. Notons e une telle représentation.

Nous pouvons maintenant refermer la boucle en posant ψ la nouvelle formule Σ_1 suivante :

$$\psi(x, y) = \theta(x, y, e)$$

Il nous suffit alors de vérifier que nous avons bel et bien obtenu le résultat désiré :

$$\begin{aligned} \psi(x, y) &\leftrightarrow \theta(x, y, e) \\ &\leftrightarrow \phi(x, y, \Sigma\text{-Sat}_3(e, x, y, e)) \\ &\leftrightarrow \phi(x, y, \theta(\cdot, y, e)) \\ &\leftrightarrow \phi(x, y, \psi(\cdot, y)) \end{aligned}$$

□

Les preuves effectuées dans les premiers chapitres utilisent explicitement le fait que le domaine $\mathcal{D}$ est ω, elles ne permettent pas d'assurer que les calculabilités sur les admissibles se comportent exactement comme les sous-calculabilités. Cependant, ces résultats nous amènent à penser que c'est le cas, ce qui nous permet de formuler la conjecture suivante :

Conjecture 4.2.12

Les calculabilités sur les admissibles sont des sous-calculabilités avec le formalisme défini ci-dessus.

5 Perspectives

Le nombre de questions posées par cette étude et non résolues par notre travail demeure, malgré nos efforts, important, notamment parce qu'à chaque propriété que nous avons vérifiée, une nouvelle piste s'offrait à nous. Nous allons dans ce paragraphe rappeler les problématiques qui nous ont guidés dans notre recherche et donner les objectifs que nous souhaiterions poursuivre dans le futur.

Complexité de Kolmogorov relativisée

Comment restreindre la notion d'aléatoire et de compressibilité aux classes de sous-calculabilités ? Les résultats obtenus permettraient-ils de retrouver les inégalités usuelles de la théorie de l'information algorithmique usuelle ?

Des travaux d'Antoine Taveneaux [Tav13] nous indiquent qu'il n'est pas possible de faire une théorie axiomatique de Kolmogorov basée sur les fonctions récursives primitives. Mais est-il possible de considérer les fonctions c-partielles ?

Étude des ordinaux c-récursifs

Les ordinaux récursifs, représentables par le $\mathcal{O}$ de Kleene forment un ensemble qui, bien que très étudié, reste toujours hors d'atteinte. Quelque part, il est la définition même des choses que l'on ne peut atteindre, vers lesquelles aucun calcul ne peut tendre. L'imprédicativité trouve rapidement ses limites et il est nécessaire de *croire* en l'existence d'un ordinal non-dénombrable (ω_1) pour espérer s'approcher un peu plus de ω_1^{CK}.

Les versions relativisées de $\mathcal{O}$: les $\mathcal{O}_{\mathsf{c}}$ sont infiniment plus simples. Nous avons conjecturé que l'ordinal type de $\mathcal{O}_{\mathsf{p}}$ était ω^ω et que s'ensuivra une certaine hiérarchie d'ordinaux types pour chaque sous-calculabilités dont les fondations sont les limites de la précédente. La hiérarchie ainsi formée ressemble beaucoup aux bases d'un ordre visant à atteindre ω_1^{CK} sans jamais y parvenir. Son étude en parallèle avec l'étude des ordinaux admissibles et des ordinaux réflectifs nous permettra d'obtenir une meilleure compréhension de ω_1^{CK} et d'en isoler des familles d'ordinaux dont les propriétés de régularité et d'inaccessibilité reflètent fortement celles des cardinaux.

Les ensembles hyperarithmétiques sous-jacents permettront alors d'identifier une structure fine des degrés c-Turing englobant tous les objets définis dans ce document, du degré zéro de la récursivité primitive aux ordinaux admissibles limites d'admissibles, avec la granularité permise par les sous-calculabilités.

Les limites des EORS

L'obtention d'un EORS pour un ordinal donné α n'est pas une opération *automatique* très simple. De plus, nous ne savons pas jusqu'où il est possible d'utiliser des EORS ni à quel point augmenter la théorie de base (**ERA**) permettrait de repousser les éventuelles limites.

La théorie de la récursion sur les admissibles possède, presque axiomatiquement, une énumération pour chacune de ses classes de fonctions. Si l'on trouve un moyen de projeter cette structure et cette énumération sur des ordinaux récursifs, en utilisant par exemple les travaux de Wolfram Pohlers[Poh08], alors nous obtiendrons un remplacement avantageux aux EORS, voire un moyen de les générer au besoin pour toute une classe d'ordinaux récursifs, qu'il conviendra de caractériser.

Vers une structure fine des calculabilités

Nous pensons que ce n'est pas par hasard si nous retrouvons dans nos travaux sur les sous-calculabilités ces objets existants au niveau de la récursion sur les admissibles. Dans le cas des sous-calculabilités, nous avons affaibli le schéma de récursion, et nous ne sommes plus dans un mode aussi *parfaitement* clos que la calculabilité classique. Dans le cas de la récursion sur les admissibles, nous avons augmenté le schéma de récursion, et les mêmes symptômes surgissent. En décidant de passer à un ordre supérieur, la calculabilité sur les admissibles a augmenté la puissance du modèle de calcul. Ce faisant, nous n'avons pas résolu les interrogations que nous avions sur le fonctionnement de la calculabilité. Nous n'avons pas rendu plus simple la résolution des questions qui nous préoccupent aujourd'hui. Existe-t-il un degré *r.e.* intermédiaire *naturel*? À quoi correspond un degré *r.e.low*? Nous avons simplement transposés ces problèmes à un ordre supérieur. Avec les sous-calculabilités, nous avons souhaité montrer qu'il était possible de transposer ces problématiques à un ordre inférieur, dans des strates encore plus basses de l'arithmétique.

Maintenant, il semblerait que les sous-calculabilités sur les admissibles et les sous-calculabilités sur les ordinaux récursifs forment deux familles de calculabilités dont la seconde est simplement une projection de la première. Si nous arrivons à confirmer cela, nous pourrons montrer que les *petits* problèmes de la *petite* arithmétique comptent en fait parmi eux les projections des propriétés les plus complexes sur la structure des ordinaux admissibles, voire des grands ordinaux, des cardinaux réguliers, ou même des grands cardinaux.

Des résultats de calculabilité plus intuitifs

Un ensemble *low r.e.* est un ensemble *r.e.* non récursif dont le saut est de même degré que le problème de l'arrêt. De très élégantes constructions basées sur la méthode de priorité présentée initialement et indépendamment par Friedberg et Muchnik permettent de construire un tel objet. Cependant, si nous regardons la méthode de priorité, si nous déroulons le fonctionnement de cette énumération, nous y voyons une fonction récursive qui fait un travail d'apparence très peu naturelle et très complexe. Pour autant, le degré obtenu est un des degrés les moins *complexes* imaginables, au sens de la réduction Turing.

Présenter un tel degré comme l'effondrement d'une structure appartenant à une classe extrêmement complexe, mais dans laquelle elle représente un objet des plus simples serait quelque part rassurant et réconcilierait les points de vue *complexité de description* et *puissance de calcul* de celui-ci.

En utilisant la réflexivité ou même l'absoluité, nous avons espoir de pouvoir transférer en calculabilité classique des résultats dont les preuves s'effectuent de manière plus claire à des niveaux plus élevés où les difficultés d'encodages et autres lourdeurs techniques sont *absorbées* par la complexité des objets en jeu.

Les travaux de [Maa82] et [Joc84] sur le *forcing* dans les degrés *r.e.* nous laisse penser que de tels résultats de transfert sont possibles, et que les preuves des résultats classiques peuvent être rendues bien plus élémentaires qu'elles ne le sont aujourd'hui, en permettant à la méthode de preuve de s'exprimer dans un langage de plus haut niveau.

A Séquences fondamentales

Calculer des fonctions complexes, utiliser des schémas récursifs et calculer des fonctions croissantes sont des opérations relevant de la même difficulté. Au cœur de celle-ci se trouvent les schémas de récursion qui nous permettent de définir nos fonctions. Et au cœur des schémas de récursion se trouvent les ordinaux qui nous permettent de définir les schémas eux-même, d'assurer leur correction et de prouver la terminaison des fonctions.

La première difficulté soulevée par un ordinal est bien souvent de prouver qu'il existe et qu'il s'agit bien d'un ordinal. Cette problématique s'illustre particulièrement en théorie de la preuve, dans les travaux cherchant à effectuer l'analyse ordinale d'une théorie. Très souvent, la définition de la *puissance* d'une théorie adoptée est le plus petit ordinal de $\mathbb{V}$, l'univers de la théorie des ensembles, que la théorie ne puisse pas reconnaître comme ordinal. Cette question ne sera pas traitée dans notre étude.

La seconde difficulté qui émerge assez rapidement dans l'étude des ordinaux, et en particulier des ordinaux calculables, est la question de leur représentation. Étant donné un ordinal limite calculable, nous voulons pouvoir l'atteindre *par dessous*, c'est à dire construire des séquences croissantes d'ordinaux, de longueur ω dont la limite soit cet ordinal. Ces séquences sont appelées séquences fondamentales et elles vont nous être très utiles par la suite pour construire des fonctions croissantes très rapides.

Dans la suite, on rencontrera ω_1 qui est le premier ordinal non-dénombrable, ou dit autrement : l'ensemble des ordinaux dénombrables. Parmi les ordinaux non-dénombrables, tous ne sont pas limite d'une séquence croissante d'ordinaux de longueur ω, certains nécessitent que cette suite soit de longueur plus grande, parfois même plus grande que ω_1. Par exemple, ω_1^ω est clairement limite d'une séquence de longueur ω : $n \mapsto \omega_1^n$, en revanche, ce n'est pas le cas pour ω_1. On appelle *cofinalité* de α un ordinal limite (souvent notée $\mathrm{cf}(\alpha)$) le plus petit ordinal θ tel qu'il existe une séquence d'ordinaux strictement croissante, de longueur θ, qui convergent vers α.

L'outil permettant de donner une consistance davantage constructible à ces séquences d'ordinaux est appelé séquence fondamentale :

Définition A.0.13 **Séquence fondamentale.** *Soit α un ordinal limite. Une* séquence fondamentale pour α *est une séquence d'ordinaux $\{\alpha_n\}_{n\in\mathrm{cf}\,\alpha}$ strictement* *croissante et de limite α.*

Nous allons maintenant voir comment construire ces séquences fondamentales.

A.1 Séquences fondamentales jusqu'à ϵ_0

ϵ_0 est le plus petit point fixe de l'exponentiation de base ω pour les ordinaux. Grâce à la décomposition des ordinaux inférieurs à ϵ_0 sous la forme normale de Cantor, nous savons très facilement construire des séquences fondamentales jusqu'à cet ordinal.

Définition A.1.1 **Séquences fondamentales jusqu'à ϵ_0.** *La décomposition de Cantor nous permet de vérifier que chaque ordinal limite inférieur à ϵ_0 s'écrit d'une des manières suivantes :*

$$\alpha = \begin{cases} \omega^{\gamma+1} & \\ \omega^{\gamma} & \text{avec } \gamma \text{ limite,} \\ \beta + \gamma & \text{avec } \beta \text{ et } \gamma \text{ limites et } \alpha > \beta > \gamma. \end{cases}$$

On associe alors à chacun de ces cas la séquence fondamentale suivante, définie par induction :

$$\alpha_n = \begin{cases} \omega^{\gamma} \times n & \text{si } \alpha = \omega^{\gamma+1}, \\ \omega^{\gamma_n} & \text{si } \alpha = \omega^{\gamma} \text{ avec } \gamma \text{ limite,} \\ \beta + \gamma_n & \text{si } \alpha = \beta + \gamma \text{ avec } \gamma \text{ limite.} \end{cases}$$

γ étant dans tous les cas nécessairement inférieur à ϵ_0. On pose également $(\epsilon_0)_0 = \omega$ et $(\epsilon_0)_{n+1} = \omega^{(\epsilon_0)_n}$.

A.2 Séquences fondamentales jusqu'à ϵ_{ω_1+1}

Les séquences fondamentales que nous allons construire mettent en œuvre des ordinaux inférieurs à ϵ_{ω_1+1} (qui est lui-même compris entre ω_1 et ω_2.) Il est important de noter pour la suite que tous les ordinaux de ce segment initial sont de cofinalité inférieure ou égale à ω_1 (voir [Odi99].)

On peut continuer le même genre de raisonnement que dans le paragraphe précédent, mais en se basant cette fois sur la décomposabilité des ordinaux inférieurs à ϵ_{ω_1+1} en forme normale de Cantor de base ω_1, nous allons étendre les séquences fondamentales définies jusqu'à ϵ_0 à quelques ordinaux non-dénombrables.

Définition A.2.1 **Séquences fondamentales jusqu'à ϵ_{ω_1+1}.** *Soit $\alpha \in \epsilon_{\omega_1+1}$ un* *ordinal limite.*

Rappelons que tout ordinal limite $\alpha \in \epsilon_{\omega_1+1}$ est de cofinalité ω ou ω_1. De plus, il se décompose en base ω_1 dans une généralisation de la forme normale de Cantor. Il est donc possible de définir par induction les séquences fondamentales, suivant la disjonction suivante :

si α est de cofinalité ω, alors les séquences fondamentales sont indicées par des éléments $n \in \omega$:

$$\alpha_n = \begin{cases} \omega_1^\gamma \times \delta_n & \textit{si } \alpha = \omega_1^\gamma \times \delta, \\ \omega_1^{\gamma_n} & \textit{si } \alpha = \omega_1^\gamma \textit{ avec } \gamma \textit{ limite et de cofinalité } \omega, \\ \beta + \gamma_n & \textit{si } \alpha = \beta + \gamma \textit{ avec } \gamma \textit{ limite et de cofinalité } \omega\,; \end{cases}$$

si α est de cofinalité ω_1, alors les séquences fondamentales sont indicées par des ordinaux $\lambda \in \omega_1$:

$$\alpha_\lambda = \begin{cases} \omega_1^\gamma \times \lambda & \textit{si } \alpha = \omega_1^{\gamma+1}\,; \\ \omega_1^{\gamma_\lambda} & \textit{si } \alpha = \omega_1^\gamma \textit{ avec } \gamma \textit{ limite et de cofinalité } \omega_1, \\ \beta + \gamma_\lambda & \textit{si } \alpha = \beta + \gamma \textit{ avec } \gamma \textit{ limite et de cofinalité } \omega_1. \end{cases}$$

Attention cependant, cette définition généralise les séquences fondamentales définies sur les ordinaux dénombrables. Or nous ne disposons pas de séquence fondamentale pour chacun d'entre eux. Par conséquent, nous ne disposons pas non plus de séquences fondamentales pour l'ensemble des ordinaux inférieurs à ϵ_{ω_1+1} mais seulement pour certains d'entre eux.

Cependant, ces séquences fondamentales vont au moyen de la hiérarchie de Veblen-Bachmann nous permettre de définir des séquences fondamentales vers des ordinaux dénombrables (récursifs) bien plus grands que ϵ_0.

A.3 Hiérarchie de Veblen-Bachmann

Au-delà, la machinerie à mettre en œuvre pour exhiber les séquences fondamentales devient plus importante. Pour essayer d'énumérer de grands ordinaux, une notion clé pratiquement inévitable est celle de *point fixe*. L'itération de l'opération $\alpha \mapsto 1 + \alpha$ à partir de 0, nous donne ω comme premier point fixe, puis $\omega \times 2$, puis $\omega \times 3$, *etc.* Le point fixe définit donc une nouvelle opération : $\alpha \mapsto \omega + \alpha$. Mais cette opération possède également des points fixes, le premier étant ω^2, donnant la nouvelle opération $\alpha \mapsto \omega^2 + \alpha$, nous conduisant à ω^3. Remarquons alors que le fait de prendre le point fixe d'une opération, et de l'itérer pour obtenir un nouveau point fixe ainsi qu'une nouvelle opération est en soit une nouvelle opération, qui nous mène à un nouveau point fixe ω^ω, le point fixe de l'opération de calcul de point fixe des opérations initiales.

Ce raisonnement par itération de points fixes a été formalisé en 1908 par Veblen dans [Veb08] et il a ainsi défini une hiérarchie de fonctions permettant de calculer les ordinaux jusqu'à ϵ_0. Cette hiérarchie a été étendue par Bachmann en 1950 de sorte à obtenir la définition qui va suivre par induction sur des ordinaux pas nécessairement dénombrables. Ces résultats sont notamment discutés dans [Poh08].

Définition A.3.1 (Veblen, Bachmann) Hiérarchie de Veblen-Bachmann. *La famille de fonctions $(\phi_\alpha)_\alpha$ est définie par induction sur α comme suit :*

$$\begin{array}{lcll} \phi_0(\gamma) & = & \omega^\gamma\,; & \\ \phi_{\alpha+1}(\gamma) & = & \textit{le } \gamma\textit{-ème point fixe de } \phi_\alpha\,; & \\ \phi_\alpha(\gamma) & = & \sup\{\phi_{\alpha_n}(\gamma) : n \in \omega\} & \textit{si } \alpha \textit{ est limite et de cofinalité } \omega\,; \\ \phi_\alpha(\gamma) & = & \phi_{\alpha_\gamma+1}(0) & \textit{si } \alpha \textit{ est limite et de cofinalité } \omega_1. \end{array}$$

Note A.3.2 Domaine de définition. *On notera que le domaine de définition de cette famille dépend de l'existence de séquences fondamentales.*

Nous remarquerons que

$$\phi_1(0) = \sup\{\text{le plus petit point fixe de } \phi_0 = \alpha \mapsto \omega^\alpha\} = \epsilon_0.$$

De même, il est possible d'aller plus loin en notant, pour tout ordinal α, $\epsilon_{\alpha+1}$ le plus petit point fixe de l'exponentiation de base ω supérieur à ϵ_α. Nous noterons alors par exemple que $\epsilon_\alpha = \phi_1(\alpha)$.

Mais cette hiérarchie de fonction permet en fait d'obtenir des ordinaux et des séquences fondamentales inimaginablement plus grandes. Par exemple, l'ordinal de Bachmann-Howard [How72] est représenté par $\zeta = \phi_{\epsilon_{\omega_1+1}}(0)$ et l'ordinal de Feferman-Schütte, Γ_0, est le plus petit ordinal α dénombrable tel que $\phi_\alpha(0) = \alpha$.

A.4 Séquences fondamentales jusqu'à $\phi_{\epsilon_{\omega_1+1}+1}(0)$

En utilisant les fonctions de la hiérarchie de Veblen-Bachmann, il est alors possible de généraliser les séquences fondamentales dénombrables pour toute une famille d'ordinaux dénombrables rendus accessibles grâce à celles-ci.

Définition A.4.1 Séquences fondamentales jusqu'à $\phi_{\epsilon_{\omega_1+1}+1}(0)$. *Soit un ordinal limite $\alpha \in \phi_{\epsilon_{\omega_1+1}+1}(0)$. Il est alors possible de le décomposer*[1] *à l'aide de la hiérarchie de fonctions de Veblen-Bachmann :*

1. La justification de la décomposition est donnée dans [Odi99].

$$\alpha_n = \begin{cases} \omega^\gamma \times n & \textit{si } \alpha = \phi_0(\gamma+1)\,; \\ \omega^{\gamma_n} & \textit{si } \alpha = \phi_0(\gamma) \textit{ pour } \gamma \textit{ ordinal limite}\,; \\ \phi_\beta^{(n)}(0) & \textit{si } \alpha = \phi_{\beta+1}(0)\,; \\ \phi_\beta^{(n)}(\phi_{\beta+1}(\gamma)+1) & \textit{si } \alpha = \phi_{\beta+1}(\gamma+1)\,; \\ \phi_{\beta+1}(\gamma_n) & \textit{si } \alpha = \phi_{\beta+1}(\gamma), \textit{ avec } \gamma \textit{ un ordinal limite}\,; \\ \phi_{\beta_n}(\gamma) & \textit{si } \alpha = \phi_\beta(\gamma), \textit{ avec } \beta \textit{ un ordinal limite de cofinalité } \omega\,; \\ \phi_{\beta_\gamma}^{(n)}(0) & \textit{si } \alpha = \phi_\beta(\gamma), \textit{ avec } \beta \textit{ un ordinal limite de cofinalité } \omega_1\,; \\ \beta+\gamma_n & \textit{si } \alpha = \beta+\gamma, \textit{ avec } \gamma \textit{ un ordinal limite.} \end{cases}$$

La construction des séquences fondamentales, en plus de nous fournir un outil clé pour la construction de fonctions à croissance rapide, est un bon exercice pour comprendre la récursivité. En effet, on retrouve celle-ci dans les définitions inductives et l'utilisation des séquences fondamentales, qui doivent dans le contexte de la récursivité impérativement rester calculables.

B

Théorie des ensembles KP

Notions de base de théorie des ensembles de Kripke-Platek

Sommaire

B.1 Théorie des ensembles de Kripke-Platek

La théorie des ensembles de Kripke-Platek, est une version de la théorie des ensembles dont l'axiomatique est plus faible que celle de **ZF**. Alors que les principes de séparation et de collection sont assurés pour toutes les formules ϕ quelle que soit leur complexité dans **ZFC**, l'axiomatique de **KP** requiert seulement que ces propriétés soient vérifiées pour les formules Δ_0, c'est à dire pour les formules sans quantificateurs non-borné. Ainsi, l'existence d'un ensemble, et surtout sa preuve d'existence, est liée à un procédé de construction mécanisable, dans le sens où il fonctionne principalement par recherche bornée d'éléments, voire pour le cas des objets Σ_1, par une recherche non-bornée d'éléments. Nous distinguons donc deux des schémas classiques vus au premier chapitre, la recherche bornée et la recherche non-bornée, bien que cette fois-ci elle ne s'effectue plus seulement sur des entiers.

B.1.1 Axiomatique et propriétés élémentaires

L'axiomatique de la théorie **KP** est la suivante :

Définition B.1.1 **Modèle de Kripke-Platek.** *Une structure est un modèle de la théorie de Kripke-Platek si elle satisfait l'ensemble, infini mais récursif, d'axiomes suivant :*

1. *Extentionalité : Soit* a *et* b *deux éléments*
$$\forall x, (x \in a \leftrightarrow x \in b) \rightarrow a = b$$
2. *Fondation : Soit* $\phi(\cdot)$ *une formule ne contenant pas* y *comme variable libre,*
$$\exists x, \phi(x) \rightarrow \exists x, (\phi(x) \wedge \forall y \in x, \neg\phi(x))$$
3. *Paire : Soit* x *et* y *deux éléments*
$$\exists a, (x \in a \wedge y \in a)$$
4. *Union : Soit* a *un élément,*
$$\exists b, \forall y \in a, \forall x \in y, x \in b$$
5. Δ_0*-séparation : Soit* a *un élément et* $\phi(\cdot)$ *une formule* Δ_0 *ne contenant pas* b *comme la variable libre,*
$$\exists b, \forall x, (x \in b \leftrightarrow x \in a \wedge \phi(x))$$
6. Δ_0*-collection : Soit* a *un élément et* $\phi(\cdot,\cdot)$ *une formule* Δ_0 *ne contenant pas* b *comme variable libre.*
$$\forall x \in a, \exists y, \phi(x,y) \rightarrow \exists b, \forall x \in a, \exists y \in b, \phi(x,y)$$
7. *Non-vacuité :*
$$\exists a, (a = a)$$

Nous vérifions facilement en utilisant l'extentionalité et quelques autres axiomes que les objets de cette théorie sont bien manipulables comme des ensembles au sens intuitif que nous avons d'eux :

Propriété B.1.2

Quelques propriétés sur les ensembles de **KP** *:*
- *Il existe un unique ensemble vide* $\varnothing$*,*
- *étant donné un ensemble* a*, il existe un unique ensemble* $b = \cup_{y \in a} y$*,*
- *étant donnés* a *et* b*, il existe un unique ensemble* $c = a \cup b$*,*
- *étant donnés* a *et* b*, il existe un unique ensemble* $c = a \cap b$*,*

De plus, il est possible de définir les paires d'éléments de la même manière qu'en théorie des ensembles. Ceci nous permettra de définir des relations, des graphes, et bien sûr des fonctions.

Définition B.1.3 **Paires ordonnées.** *La paire ordonnée* $\langle x, y \rangle$ *est l'ensemble* $\{\{x\}, \{x, y\}\}$

Ces paires d'éléments se comportent bien elles aussi comme les paires classiques. De plus, il est possible de construire le produit cartésien de deux ensembles, ce qui n'était pas donné d'avance sachant que l'ensemble des parties d'un ensemble n'est, lui, pas constructible en général.

Propriété B.1.4

Il y a (existence et) unicité des ensembles représentant la paire et le produit cartésien :
- *Soit* w, x, y *et* z *des éléments, alors* $\langle w, x\rangle = \langle y, z\rangle \leftrightarrow w = x \wedge y = z$,
- *étant donnés* a *et* b, *il existe un ensemble* $c = a \times b$,

Toutes ces propriétés nous indiquent qu'il s'agit bel et bien ici de la notion d'ensembles intuitive que l'on manipule habituellement en théorie des ensembles.

B.1.2 Ensembles admissibles

Les énoncés de **KP** sont formulés dans un langage **L** du premier ordre comportant l'égalité, des symboles pour les relations, les fonctions et les constantes auxquels nous rajoutons un symbole d'appartenance $\in$.

Définition B.1.5 **Ensembles admissibles.** *Un ensemble admissible* $\mathbb{A}$ *est une structure* $\mathbb{A} = (A, E, \dots)$ *sur* **L**, *composée en particulier d'un ensemble non-vide* A, *qui est l'ensemble des ensembles de* $\mathbb{A}$ *et d'une relation* $E \subseteq A \times A$ *qui permet d'interpréter le symbole d'appartenance* $\in$, *telle que cette structure satisfait* **KP**.

Autrement dit, ces ensembles sont des structures modélisant la théorie de **KP**. Ces structures sont donc en particulier closes par toutes les opérations de définition d'ensemble présentes dans l'axiomatique de **KP**.

B.1.3 Ordinaux

Une autre structure de données qui va nous être fondamentale et que nous avons jusqu'à présent utilisée sans vraiment l'expliciter est la structure d'ordinaux. Les ordinaux représentent des ordres totaux bien fondés. La bonne fondation[1] est une propriété systématiquement utilisée pour prouver la terminaison d'une fonction. De plus, comme nous l'avons vu dans les premiers chapitres, la définition de fonctions via des schémas de récursion ordinale est un outil de choix pour mesurer une sorte de complexité de ces fonctions.

1. Une relation d'ordre est dite *bien fondée* si elle ne contient pas de chaîne infinie décroissante. Pour les ordinaux, l'ordre bien fondé qui leur est associé est l'appartenance : $\in$.

Ce qui était vrai en calculabilité classique le reste dans le théorie de la récursion d'ordre supérieur. En particulier, nous étudierons les liens entre les niveaux ordinaux des constructibles de Gödel et les ensembles admissibles dans les prochains paragraphes.

Définition B.1.6 **Ensembles transitifs.** *Un ensemble est transitif s'il contient tous ses éléments :*

$$\begin{aligned} \mathrm{Trans}(a) &\leftrightarrow \forall x \in a, a \subset x \\ &\leftrightarrow \forall x \in a, \forall y \in x, y \in a \end{aligned}$$

Remarquons qu'il s'agit là d'une propriété Δ_0.

Un ordinal est un ensemble transitif dont tous les éléments sont des ensembles transitifs, ceci est également une propriété Δ_0 *:*

$$\mathrm{Ord}(a) \leftrightarrow \mathrm{Trans}(a) \wedge \forall x \in a, \mathrm{Trans}(x)$$

Le plus petit ordinal est $\varnothing$, habituellement noté 0 quand il nous intéresse en tant qu'ordinal. Le successeur d'un ordinal est ensuite défini par une opération sur les ensembles.

Définition B.1.7 **Successeurs.** *Le successeur d'un ensemble suivant l'encodage des ordinaux et des entiers de von Neumann est défini comme suit :*

$$\mathcal{S}(a) = a \cup \{a\}$$

Ainsi, 1, 2, 3 et tous les nombres entiers naturels non nuls sont des ordinaux successeurs. Un ordinal limite est un ordinal non nul qui n'est le successeur d'aucun ordinal. Par exemple, $\omega = \{0, 1, 2, \dots\}$, le plus petit ordinal infini, qui est bien transitif, et dont tous les éléments sont biens transitifs puisque ce sont des ordinaux.

Note B.1.8

Pour un ordinal α*, la notation* $\alpha + 1$ *désignera l'ordinal successeur* $\mathcal{S}(\alpha)$.

Il est important de remarquer que, bien que les ordinaux soient toujours présents dans l'étude de la calculabilité, ils ne le sont pas nécessairement de la même manière. En calculabilité classique, les ordinaux servent à prouver la terminaison des fonctions, mais n'apparaissent pas vraiment dans les objets de base de la théorie. Ils sont encodés par des fonctions, mais la preuve de leur ordinalité reste à faire et finalement se trouve être faite dans l'étude de ces fonctions. Ainsi, d'une certaine manière, les ordinaux sont cantonnés à la théorie de la preuve de la calculabilité.

Dans les sous-calculabilités, les ordinaux sont présents dès la définition des fonctions, en particulier lorsque les classes fondamentales considérées sont des classes α-récursives. Cependant, ils peuvent également apparaître comme objets de base dans le cas où le domaine $\mathcal{D}$ n'est pas ω mais un autre ordinal, tel ω^ω. Sur un tel domaine, une fonction récursive primitive est capable d'effectuer des opérations qui nécessiteraient un schéma de récursion ω^ω-récursif si le domaine était ω.

Dans le cas de la théorie de la récursion sur les admissibles, les ordinaux sont, comme nous venons de le voir, des objets de base. Aussi, la notion de récursion sur un ordinal ou grâce à l'axiome de fondation sur n'importe quel ensemble de la théorie, est un des outils élémentaires qui permet de construire des fonctions totales, en fait des prédicats Δ_0, de la même manière que les fonctions α-récursives permettaient d'établir une fondation aux sous-calculabilités.

B.1.4 Classes de formules

Nous avons abordé à plusieurs reprises différentes classes de formules basées sur différentes hiérarchies à ne pas confondre.

Lorsque nous travaillions sur des formules de l'arithmétique du premier ordre, nous considérions la *hiérarchie arithmétique* dont la base est constituée des formules Δ^0_0 qui sont les formules sur les entiers, sans quantificateur non borné.

Définition B.1.9 **Hiérarchie arithmétique.** *Les formules Δ_0 sont les formules sur les entiers utilisant un nombre fini de connecteurs logiques (implication, négation, conjonction, disjonction) et uniquement des quantificateurs bornés (par un entier).*

Posons $\Delta^0_0 = \Sigma^0_0 = \Pi^0_0$.

Pour tout n,

- *une formule est Σ^0_{n+1} si elle s'écrit de la forme $\exists a, \phi$ où ϕ est une formule Π^0_n ;*
- *une formule est Π^0_{n+1} si elle s'écrit de la forme $\forall a, \phi$ où ϕ est une formule Σ^0_n ;*
- *une formule est Δ^0_n si elle est à la fois Σ^0_n et Π^0_n.*

Cette hiérarchie est étendue aux indices ordinaux par la hiérarchie hyperarithmétique. Enfin, son deuxième niveau (quand l'exposant est 1) est défini de manière similaire, mais sur les ensembles d'entiers.

Cette distinction *entiers*, *ensembles d'entiers* convient pour classer les énoncés d'une théorie arithmétique, mais dans le cas de **KP**, c'est simplement le nombre d'alternance des quantificateurs qui nous intéresse.

Définition B.1.10 **Hiérarchie d'alternance des quantificateurs.** *Les formules Δ_0 sont les formules sans quantificateur non-bornés et dont les constantes sont des symboles du langage représentant des éléments de la théorie. Pour tout n, les classes Σ_n et Δ_n sont définies de la même manière que pour la hiérarchie arithmétique, si ce n'est que les quantificateurs portent sur tout l'univers et non pas sur les entiers uniquement.*

B.2 Constructibles de Gödel et admissibles

Dans le but de construire un modèle minimal de la théorie des ensembles **ZF**, Gödel a identifié un petit lot d'opérations élémentaires sur les ensembles. Par itération de ces opérations un nombre ordinal α de fois, nous obtenons une collection croissante d'ensembles $\mathbb{L}_\alpha$ qui sont de fait *constructibles* et donc nécessairement présents dans tout univers $\mathbb{V}$ de la théorie des ensembles [2].

Chaque niveau $\mathbb{L}_\alpha$ est alors un préfixe d'une classe $\mathbb{L}$ satisfaisant **ZF**.

Définition B.2.1 **Constructibles de Gödel.** *Les niveaux des constructibles de Gödel sont définis par induction transfinie.*

— $\mathbb{L}_0 = \varnothing$
— $\mathbb{L}_{\alpha+1} = \mathrm{Def}(\mathbb{L}_\alpha)$
— $\mathbb{L}_\gamma = \cup\{\mathbb{L}_\alpha : \alpha < \gamma\}$

Où $\mathrm{Def}(X)$ *désigne la classe des ensembles définissables dans X par une formule du premier ordre ϕ à paramètres dans X. Autrement dit,*

$$\mathrm{Def}(X) = \{\{x : X \models \phi(x)\} : \phi \textit{ formule du premier ordre à paramètres dans } X\}$$

Si $\mathbb{L}$ satisfait tout **ZF**, des théories plus faibles que **ZF** sont satisfaites par des niveaux inférieurs de cette hiérarchie. En effet, lorsque l'ordinal α utilisé pour construire $\mathbb{L}_\alpha$ par induction transfinie possède de fortes propriétés de clôture, ces propriétés se transmettent à la structure $\mathbb{L}_\alpha$ obtenue.

Par exemple, et pour revenir au cas de **KP**, il existe une collection d'ordinaux α, ne serait-ce que dans les ordinaux dénombrables, tel que $\mathbb{L}_\alpha \models$ **KP**. Ces ordinaux vérifiant que $\mathbb{L}_\alpha$ soit admissible sont appelés les ordinaux admissibles.

Définition B.2.2 **Ordinaux admissibles.** *Un ordinal α est admissible si $\mathbb{L}_\alpha$ est un modèle pour* **KP**.

Intéressons-nous un instant aux propriétés induites sur un ordinal α du fait de son admissibilité. De la même manière que la théorie **KP** force certaines propriétés de clôtures sur tous les modèles satisfaisant ses axiomes, nous savons d'un ordinal admissible qu'il sera *inatteignable* par les fonctions Σ_1 sur ses prédécesseurs. Cette propriété est en fait une caractérisation des ordinaux admissibles :

Propriété B.2.3

α est admissible si et seulement si pour aucun ordinal $\gamma \in \alpha$ il n'existe de surjection Σ_1 définissable sur $\mathbb{L}_\alpha$ de γ sur α.

2. À cet univers $\mathbb{V}$ est également associée une hiérarchie $\mathbb{V}_\alpha$, dite *hiérarchie cumulative des ensembles*, voir [Jec03]. La différence principale d'avec la hiérarchie des constructibles est l'utilisation de l'axiome du *Power Set* : $\forall A, \exists P, \forall X \subset A, X \in P$.

Cette caractérisation est en fait celle permettant de définir l'ordinal de Church-Kleene (ω_1^{CK}) dans le modèle $\mathbb{H}_\omega (= \mathbb{L}_\omega)$ des ensembles héréditairement finis. En effet, l'ordinal de Church-Kleene est le plus petit ordinal non récursif. En d'autres termes ω_1^{CK} est le plus petit ordinal tel qu'aucune fonction récursive, c'est à dire aucune fonction (définie par une formule) Σ_1^0 ou de manière équivalente Σ_1 sur $\mathbb{H}_\omega$, ne peut être la fonction caractéristique d'une relation d'ordre isomorphe à ω_1^{CK}.

Ainsi, si nous itérons ce procédé un nombre ordinal dénombrable α de fois, nous obtenons un nouvel ordinal $\omega_\alpha^{\mathrm{CK}}$ nécessairement admissible.

Définition B.2.4 Généralisations de l'ordinal de Church-Kleene ($\omega_\alpha^{\mathrm{CK}}$).

L'ordinal de Church-Kleene (ω_1^{CK}) est le plus petit ordinal qui n'est pas représentable par une relation récursive. De manière équivalente, c'est l'ensemble des ordinaux récursifs.

La généralisation des ordinaux de Church-Kleene est définie par induction transfinie sur les ordinaux dénombrables.

— $\omega_{\alpha+1}$, *pour* α *un ordinal dénombrable, est le plus petit ordinal récursif en* ω_α.

— ω_γ, *pour* γ *un ordinal limite dénombrable, est l'union des* ω_α *pour* $\alpha^2\gamma$.

Intuitivement, ces ordinaux sont tels qu'ils ne sont pas accessibles par les fonctions Σ_1 sur les modèles d'ordinal type inférieur. Il est possible de faire le rapprochement avec de nombreuses notions de théories des ensembles. La plus équivoque est peut-être celle de cardinal régulier. Un cardinal κ est régulier si sa cofinalité est égale à lui-même. C'est à dire qu'aucune suite croissante de cardinaux ne peut tendre vers κ à moins d'être de longueur au moins κ.

En pratique, de nombreuses propriétés sur les cardinaux ou les ordinaux se reflètent ensuite dans les modèles inférieurs, souvent fruits de projections de cardinaux et autres objets complexes aux propriétés de clôtures similaires. Nous reviendrons sur ce sujet plus tard.

Pour conclure sur la famille des ordinaux $\omega_\alpha^{\mathrm{CK}}$, un résultat de Gerald E. Sacks nous indique que les $\omega_\alpha^{\mathrm{CK}}$ sont exactement tous les ordinaux admissibles dénombrables.

Théorème B.2.5 **Sacks**

Les ordinaux admissibles dénombrables sont exactement les ω_α^{CK}.

Il y a donc un lien très fort entre le fait, pour un niveau α des constructibles, d'être un ensemble admissible, et les propriétés de clôtures de l'ordinal α. En pratique, ce lien se matérialise même par l'existence d'une bijection entre l'ordinal et le modèle des constructibles basé sur lui. Cette bijection a la très agréable propriété de pouvoir être vue comme une énumération du modèle des constructibles, et d'être partie intégrante du modèle, en tant que fonction Δ_0.

Propriété B.2.6

Si α est un ordinal admissible, alors il existe une bijection primitive récursive entre α et $\mathbb{L}_\alpha$.

Avec cette énumération de $\mathbb{L}_\alpha$ et la particularité des fonctions Σ_1, nous commençons à voir apparaître plusieurs propriétés familières nous rappelant la calculabilité, et, comme nous le verrons plus tard, les sous-calculabilités.

C

E-récursion

Sommaire

La généralisation des sous-calculabilités aux calculabilités sur les admissibles semble très bien fonctionner et permet d'élargir leur champ de vision vers des horizons infiniment plus grands, au sens propre du terme. Cependant, la récursion sur les admissibles manque parfois de finesse. Nous pouvons lui faire le même reproche qu'aux constructibles de Gödels : tous ces *passages à la limite* sont des unions non maîtrisées, pas si constructibles que ça. Ou du moins, pas d'une manière assez mécanisable pour nous. La E-récursion est un peu le point de vue développé pour répondre à ce reproche. Nous allons présenter quelques notions ayant été étudiées en E-récursion et que nous voudrions voir apparaître dans les sous-calculabilités. Notre espoir est que la E-récursion réécrite dans le formalisme des sous-calculabilités permette de mettre en exergue une structure raffinant les sous-calculabilités d'ordre supérieur définies dans le chapitre précédent.

C.1 Théorie de la E-récursion

Dans le paragraphe précédent, nous avons exprimé la théorie de la récursion d'ordre supérieur dans le formalisme des sous-calculabilités. Nous avons vu que cette théorie prenait comme définition d'énumérabilité la Σ_1-définissabilité. Or, lorsque nous imaginons ce qu'est une notion de calcul, nous nous attendons souvent à trouver un procédé *efficace* au sens de *constructif*, un procédé où la puissance de calcul n'est pas un oracle permettant de parcourir l'univers.

La E-récursion repose très largement sur l'induction. Théorie sur les ensembles, la E-récursion introduite par Dag Normann en 1978 dans [FGS78] et largement étudiée par Gerald Enoch Sacks et dont l'ouvrage [Sac90] fait référence en la matière, permet de construire ceux-ci soit à partir de rien, par des moyens finitistes, soit à partir d'ensembles déjà construits. Mais en aucun cas une fonction ne peut parcourir tout l'univers à la recherche d'un objet ayant certaines propriétés.

C.1.1 Définitions et tour d'horizon

Dans le modèle de calcul de la E-récursion, les fonctions sont encodées par des ensembles interprétés comme des tuples particuliers. Chacun de ses tuples est associé à un schéma de construction. Le jeu de schémas utilisés est appelé « Schémas de Normann » et permet de construire toutes les fonctions E-récursives.

Définition C.1.1 **(Normann) Schéma de construction des fonctions E-récursives.**

(1) **Projection :**

$$\{e\}(x_1,\ldots,x_n) = x_i \quad si\ e = \langle 1,n,i\rangle$$

(2) **Différence :**

$$\{e\}(x_1,\ldots,x_n) = x_i - x_j \quad si\ e = \langle 2,n,i,j\rangle$$

(3) **Pairage :**

$$\{e\}(x_1,\ldots,x_n) = \{x_i, x_j\} \quad si\ e = \langle 3,n,i,j\rangle$$

(4) **Union :**

$$\{e\}(x_1,\ldots,x_n) = \bigcup\{\{c\}(y,x_2,\ldots,x_n) : y \in x_1\} \quad si\ e = \langle 4,n,c\rangle$$

(5) **Composition :**

$$\{e\}(x_1,\ldots,x_n) = \{c\}(\{d_1\}(x_1,\ldots,x_n),\ldots,\{d_m\}(x_1,\ldots,x_n)) \quad si\ e = \langle 5,n,m,c,d_1,\ldots,d_m\rangle$$

(6) **Énumération :**

$$\{e\}(c,x_1,\ldots,x_n,y_1,\ldots,y_m) = \{c\}(x_1,\ldots,x_n) \quad si\ e = \langle 6,n,m\rangle$$

*Où la règle d'*union *demande à ce que $\{e\}(x_1,\ldots,x_n)$ soit défini pour tout $y \in x_1$.*

Comme il est possible de créer tous les entiers avec ces schémas, il est possible de faire de la récursion dessus, c'est à dire de la récursion primitive. En fait, tous les prédicats Δ_0 sont E-récursifs, ce qui reflète bien le fait que la recherche bornée est possible.

Propriété C.1.2

Soit P un prédicat Δ_0 de **KP***, alors P est E-récursif.*

Cherchons maintenant à comprendre à quoi ressemble une classe de fonctions E-récursives close par les schémas de construction de Normann. Une telle classe n'est pas vide puisqu'elle contient au moins $\langle 1,1,1\rangle$ (l'identité.) La clôture de la classe par toutes ces opérations donne un ensemble appelé structure E-close.

Définition C.1.3 **Structures E-closes.** *Soit A un ensemble transitif. A est dit E-clos si pour toute fonction partielle E-récursive f,*

$$\forall x, (x \in A \wedge f(x) \downarrow) \Rightarrow f(x) \in A$$

$E(x)$ désigne le plus petit ensemble y E-clos tel que la clôture transitive de $\{x\}$: $\mathrm{TC}(\{\mathrm{x}\})$ soit incluse dans y.

Comme nous l'avons annoncé, la construction se base fortement sur l'induction. Aussi, la règle de clôture pour un ensemble A est très simple : tout élément obtensible à partir de A et des règles de construction est dans A.

Note C.1.4 *Tout ensemble admissible est E-clos. Ainsi, à chaque calculabilité sur les admissibles est associée un ensemble E-clos. En fait, nous verrons qu'entre deux admissibles, il existe en général une multitude d'ensembles E-clos.*

La définition inductive du codage des fonctions permet de construire pour chaque calcul un arbre représentant le déroulement de celui-ci. La construction de ces arbres de calcul est avantageusement présentée sous la forme d'une relation de *successeur immédiat* du calcul, que nous présentons dans la définition suivante :

Définition C.1.5 **Arbres de calcul et arbre de calcul universel.** *Soit U l'arbre de calcul universel. Chaque nœud de cet arbre correspond à une étape de calcul. Un nœud est fils d'un autre nœud si c'est une étape de calcul lui succédant immédiatement.*

La relation de successeur immédiat est définie par induction sur les étapes de calcul.

1. *Si $e = \langle 1,n,i\rangle$, alors $\langle e, x_1, \ldots, x_n\rangle$ est une feuille de l'arbre et n'a pas de successeur immédiat ;*
2. *si $e = \langle 2,n,i,j\rangle$, idem ;*
3. *si $e = \langle 3,n,i,j\rangle$, idem ;*
4. *si $e = \langle 4,n,c\rangle$, alors pour tout $y \in x_1$, $\langle c, y, x_2, \ldots, x_n\rangle$ est un successeur immédiat de $\langle e, x_1, \ldots, x_n\rangle$;*
5. *si $e = \langle 5,n,m,c,d_1,\ldots,d_m\rangle$, alors $\forall 0 \leqslant i \leqslant m$, $\langle d_i, x_1, \ldots, x_m\rangle$ est un successeur immédiat de $\langle e, x_1, \ldots, x_n\rangle$, de plus, si tous ces calculs convergent en des y_i respectifs, $\langle c, y_1, \ldots, y_m\rangle$ est aussi un successeur immédiat de $\langle e, x_1, \ldots, x_n\rangle$;*
6. *si $e = \langle 6,n,m\rangle$, alors $\langle c, x_1, \ldots, x_n\rangle$ est un successeur immédiat de $\langle e, c, x_1, \ldots, x_n, y_1, \ldots, y_m\rangle$;*

7. *enfin, si e n'est pas un indice satisfaisant aux schémas de Normann, où que le nombre d'argument est incorrect, alors* $\langle e, x_1, \ldots, x_n\rangle$ *est un successeur immédiat de lui-même.*

Étant donné $\langle e, x\rangle$ *un nœud de cet arbre, l'arbre formé de l'ensemble de ses successeurs est appelé le* calcul *de* $\{e\}(x)$ *et est noté* $T_{\langle e,x\rangle}$.

Regardons maintenant ce que nous pouvons dire sur les propriétés de cet arbre de calcul universel. Tout d'abord, U n'est pas E-récursif. Heureusement, car si ça n'était pas le cas, U serait un élément de E, et tous les éléments de E seraient récursifs en E, donc tous les éléments de E seraient récursifs. Cela-dit, U reste E-récursivement énumérable. L'ordre de *succession immédiate* est lui plus complexe.

Propriété C.1.6

U est E-récursivement-énumérable (le domaine d'une fonction E-récursive avec paramètre dans E) et l'ordre de succession clos par transitivité n'est pas E-récursivement énumérable mais simplement Σ_1-définissable.

Considérons maintenant l'arbre de calcul $T_{\langle e,x\rangle}$ associé à un calcul $\langle e, x\rangle$. En utilisant les schémas de Normann, un calcul calcul $\{e\}(x)$ *converge* ($\downarrow$) s'il est bien défini par ces schémas. Cependant, une autre caractérisation bien plus imagée nous est permise par les arbres de calculs.

Propriété C.1.7

$\{e\}(x)$ *converge si et seulement si* $T_{\langle e,x\rangle}$ *est bien fondé.*

En effet, la relation de successeur immédiat a été conçue de manière à ce que si un calcul n'est pas défini, par exemple s'il *boucle*, alors il possède un chemin décroissant infini (une étape de calcul invalide est le successeur immédiat d'elle-même.)

C.2 Ordinaux remarquables

La E-récursion étant basée sur l'induction, il faut s'attendre à voir les ordinaux jouer un rôle clé dans la construction de ses modèles. Et en effet, plus encore que dans la théorie de la récursion sur les admissibles, tout se mesure avec des ordinaux. Que ce soit la profondeur d'un arbre de calcul divergeant, la représentation dans les constructibles d'une classe E-close donnée, ou tout ce qui concerne les notions de réflexivité pour des fragments de modèles de E-récursion, les ordinaux sont véritablement la colonne vertébrale de la théorie.

C.2.1 Ordinaux et clôtures

Dans un premier temps, nous allons voir qu'il est possible de ramener la E-clôture d'un ensemble à un niveau des constructibles de Gödel. Mais avant cela, nous allons tenter de comprendre pourquoi cela n'a rien d'étonnant.

Nous avons indiqué dans un précédent paragraphe qu'en E-récursion, tout élément calculé pouvait être utilisé pour effectuer un nouveau calcul. Si un nouveau calcul $\{e\}(x)$ converge en un élément y, alors que x et e étaient des éléments d'un ensemble E-clos A, alors c'est qu'il est possible de calculer y et donc y est dans A. Cependant, il est également possible de raisonner sur des éléments qui ne sont pas dans A. Prenons r un élément quelconque, et donc potentiellement hors de A, si une fonction E-récursive f permet de calculer un élément $s = \{f\}(r)$, alors s est dit E-récursif en r.

Définition C.2.1 **E-Réduction ($\leqslant_E$).** *y est E-récursif en $x_1, \ldots, x_n$ ($y \leqslant_E x_1, \ldots, x_n$) s'il existe un e tel que $y = \{e\}(x_1, \ldots, x_n)$*

Les arguments se comportent un peu comme des oracles étant donné qu'il est permis de faire des recherches *de facto* bornées dans ceux-ci. Dans ce contexte, il paraît assez clair qu'étant donné un paramètre $x = \alpha$, alors *être E-récursif en x* est une notion parente d'*être α-récursif*. Nous reviendrons sur ce point dans un prochain paragraphe, quand nous aborderons les liens entre la E-récursion et la calculabilité classique.

Si un ordinal permet de faire de la récursion, alors il est naturel de vouloir s'intéresser au plus grand ordinal définissable, étant donné un paramètre-oracle x. Cela revient en fait équivalent à mesurer le *supremum* des hauteurs des arbres de calculs :

Notons cette première mesure κ_0^x :

$$\kappa_0^x = \sup\{\gamma : \gamma \leqslant_E x\} = \sup\left\{\left|T_{\langle e,x\rangle}\right| : \ e \text{ fonction } E\text{-récursive.} \right\}$$

κ_0^x est nécessairement un ordinal limite, étant donné qu'il est toujours possible de prendre le *successeur* d'un ordinal. κ_0^x est également forcément de co-finalité ω puisque limite d'un ensemble dénombrable (l'ensemble des fonctions E-récursives étant lui-même dénombrable).

Cependant, les ordinaux E-récursifs en x ne sont pas tous les ordinaux que obtensibles dans une E-clôture en x $E(x)$.

$$\kappa^x = \sup\{\gamma \in E(x)\}$$

Ces deux mesures sont étroitement liées par la relation suivante :

$$\kappa_0^x = \sup\left\{\left|T_{\left\langle \pi_1^2(n), \pi_2^2(n), x\right\rangle}\right| : n \in \kappa^x\right\}$$

Cette relation signifie que la puissance de calcul de la E-structure engendrée par x est en fait complètement reflétée par le *supremum* des ordinaux présents dans $E(x)$. Une autre hiérarchie où cette constatation se fait est la hiérarchie des constructibles de Gödel. Et pour cause, il y a un lien direct entre les deux :

Proposition C.2.2

La E-clôture d'un élément x correspond à un niveau de la hiérarchie des constructibles avec paramètre.

$$E(x) = \mathbb{L}_{\kappa^x}[\mathrm{TC}(x)]$$

où $\kappa^x = \sup\{\gamma \leqslant_E x, a_0, \ldots, a_n \text{ pour } a_0, \ldots, a_n \in \mathrm{TC}(x)\}$

C.2.2 Liens avec la calculabilité classique

Il est très simple de vérifier que la E-récursion permet bien de retrouver la calculabilité habituelle. La définition des fonctions récursives (partielles) coïncide sur l'ensemble des fonctions de ω dans ω :

Propriété C.2.3

Soit f une fonction partielle de ω dans ω.

$$f \text{ } E\text{-récursive } \leftrightarrow f \text{ récursive}$$

Qui plus est, l'ensemble des ordinaux présents dans la E-clôture de ω est exactement l'ensemble des ordinaux récursifs. D'où les deux résultats suivant par application directe de la proposition précédente :

Propriété C.2.4

$$\kappa^\omega = \omega_1^{CK}$$
$$E(\omega) = \mathbb{L}_{\omega_1^{CK}}$$

C.3 Ordinaux réflexifs et *absoluité*

La E-récursion nous intéresse principalement pour deux raisons. Premièrement, parce que notre objectif initial était d'étudier les questions d'absoluité de certains énoncés de la calculabilité, et deuxièmement parce que le principe de récursion transfinie effective que nous avons rencontré est très présent dans les sous-calculabilités et nous amène sur la voie d'une théorie générale englobant toutes les sous-calculabilités dans des degrés de calcul liés à celle-ci.

Dans ce paragraphe, nous allons présenter les principes de base de l'absoluité pour la E-récursion. La notion clé ici est appelée *réflexivité* parce qu'elle identifie les modèles reflétant un fragment des propriétés d'un modèle de référence.

Définition C.3.1 **Ordinaux x-réflexifs.** *Un ordinal δ est x-réflexif si pour toute formule Σ_1 close ϕ,*

$$(\mathbb{L}_\delta [\mathrm{TC}(\{x\})] \models \phi) \Rightarrow \left(\mathbb{L}_{\kappa_0^x} [\mathrm{TC}(\{x\})] \models \phi\right)$$

Les ordinaux réflexifs nous permettent réellement d'identifier des fragments de modèles dans lesquels les propositions sont le reflet du modèle initial.

Conjecture C.3.2

Projections et réflexivité.

1. *Une sous-calculabilité $\mathtt{C}_\alpha$ est le reflet par la classe de formules du langage de $\mathtt{C}$ d'ordinal α de la calculabilité classique.*
2. *Chaque classe E-close se projette sur une sous-calculabilité et permet d'en déterminer la structure, la limite, l'énumération canonique,* etc.

Les travaux de [Poh08] en théorie de la preuve indiquent comment à partir d'un ordinal admissible et en travaillant sur des classes de formules dont la profondeur de l'arbre de preuve est limitée, il est possible d'identifier de grands ordinaux calculables liés à l'admissible de départ. Nous tentons de voir à quel point ces travaux sont utilisables afin d'identifier une projection entre calculabilités d'ordre supérieur et sous-calculabilités.

Bibliographie

[Ack28] Wilhelm Ackermann : Zum hilbertschen aufbau der reellen zahlen. *Mathematische Annalen*, 99(1) :118–133, 1928.

[Bar75] Jon Barwise : *Admissible Sets and Structures : An Approach to Definability Theory.* Perspectives in Mathematical Logic Series. Springer-Verlag, 1975.

[Con70] Robert Lee Constable : On the size of programs in subrecursive formalisms. *In Proceedings of the second annual ACM symposium on Theory of computing*, pages 1–9. ACM, 1970.

[FGS78] J.E. Fenstad, R.O. Gandy et G.E. Sacks : *Generalised recursion theory II.* North-Holland Publishing Company, 1978.

[Fri57] Richard M. Friedberg : Two recursively enumerable sets of incomparable degrees of unsolvability (solution of Post's problem, 1944). *Proceedings of the National Academy of Sciences of the United States of America*, 43(2) :236, 1957.

[FS95] Harvey Friedman et Michael Sheard : Elementary descent recursion and proof theory. *Annals of Pure and Applied Logic*, 71(1) :1–45, 1995.

[Gen36] Gerhard Gentzen : Die widerspruchsfreiheit der reinen zahlentheorie. *Mathematische Annalen*, 112(1) :493–565, 1936.

[GL11] Fabien Givors et Grégory Lafitte : Sub-computabilities. *In FCT*, pages 322–335, 2011.

[GP81] Jean-Yves Girard et Peter Päppinghaus : A result on implications of Σ_1-sentences and its application to normal form theorems. *Journal of Symbolic Logic*, 46(3) :634–642, 1981.

[How72] William A. Howard : A system of abstract constructive ordinals. *Journal of Symbolic Logic*, pages 355–374, 1972.

[Jec03] Thomas Jech : *Set Theory.* Springer, 2003.

[Joc84] Carl G. Jockush : Genericity for recursively enumerable sets. *In Proceedings, Recursion Theory Week*, 1984.

[Kle36] Stephen Cole Kleene : General recursive functions of natural numbers. *Mathematische Annalen*, 112 :727–742, 1936.

[Kle38] Stephen Cole Kleene : On notation for ordinal numbers. *Journal of Symbolic Logic*, 3(4) :150–155, 1938.

[Koz72] V. V. Koz'minyh : On a presentation of partial recursive functions by compositions. *Algebra i Logika*, 11(3) :270–294, 1972. In Russian.

[Koz80] Dexter Kozen : Indexings of subrecursive classes. *Theoretical Computer Science*, 11 :277–301, 1980.

[Kri96] Lars Kristiansen : *Papers on subrecursion theory*. Thèse de doctorat, Department of Informatics, University of Oslo, 1996.

[Kri98] Lars Kristiansen : A jump operator on honest subrecursive degrees. *Archive for Mathematical Logic*, 37(2) :105–125, 1998.

[Kri99] Lars Kristiansen : Low$_n$, high$_n$, and intermediate subrecursive degrees. *In* Calude et Dinneen, éditeurs : *Combinatorics, computation and logic*, pages 286–300. Springer, Singapore, 1999.

[KSPW11] Lars Kristiansen, Jan-Christoph Schlage-Puchta et Andreas Weiermann : Streamlined subrecursive degree theory. *Annals of Pure and Applied Logic*, 2011. to appear.

[Kuč86] Antonín Kučera : An alternative, priority-free, solution to Post's problem. *Mathematical Foundations of Computer Science 1986*, pages 493–500, 1986.

[Kus06] Boris A Kushner : The constructive mathematics of AA Markov. *The American Mathematical Monthly*, pages 559–566, 2006.

[Lac66] Alistair H Lachlan : Lower bounds for pairs of recursively enumerable degrees. *Proc. London Math. Soc*, 16(3) :537–569, 1966.

[Laf02] Gregory Lafitte : *Calculs et infinis*. Thèse de doctorat, Ecole Normale Supérieure de Lyon, 2002.

[Maa82] Wolfgang Maass : Recursively enumerable generic sets. *Journal of Symbolic Logic*, 47(4) :809–823, December 1982.

[Myh55] John Myhill : Creative sets. *Mathematical Logic Quarterly*, 1(2) :97–108, 1955.

[Nie09] André Nies : *Computability and Randomness*. Oxford University Press, 2009.

[Nor78] Dag Normann : Set recursion. *Generalized recursion theory II*, pages 303–320, 1978.

[Odi88] Piergiorgio Odifreddi : *Classical Recursion Theory*. North Holland - Elsevier, 1988.

[Odi99] Piergiorgio Odifreddi : *Classical Recursion Theory. Volume II*. North Holland - Elsevier, 1999.

[Poh08] Wolfram Pohlers : *Proof Theory : The First Step into Impredicativity*. Springer, 2008.

[Pos03] Emil L. Post : Recursively enumerable sets of positive integers and their decision problems. *Mathematical Logic in the 20th Century*, page 352, 2003.

[Rad62] Tibor Rado : On non-computable functions. *Bell System Tech. J*, 41(3) :877–884, 1962.

[Rat99] Michael Rathjen : The realm of ordinal analysis. *In* S. B. Cooper et J. Truss, éditeurs : *Sets and Proofs*, pages 219–279. Cambridge University Press, 1999.

[Ric53] Henry Gordon Rice : Classes of recursively enumerable sets and their decision problems. *American Mathematical Society*, pages 358–366, 1953.

[RJ58] Hartley Rogers Jr : Gödel numberings of partial recursive functions. *Journal of symbolic logic*, pages 331–341, 1958.

[Rob65] Joel W. Robbin : *Subrecursive hierarchies.* Thèse de doctorat, Princeton., 1965.

[Rog87] Hartley Rogers : *Theory of recursive functions and effective computability.* 1987.

[Ros84] Harvey E. Rose : *Subrecursion : Functions and Hierarchies*, volume 9 de *Oxford Logic Guides.* Oxford University Press, USA, New York, NY, 1984.

[Sac90] Gerald E. Sacks : *Higher Recursion Theory.* Springer-Verlag, 1990.

[Soa87] Robert I. Soare : *Recursively Enumerable Sets and Degrees : A Study of Computable Functions and Computably Generated Sets (Perspectives in Mathematical Logic).* Springer, 1987.

[Tav13] Antoine Taveneaux : Axiomatizing Kolmogorov complexity. *Theory of Computing Systems*, 52(1) :148–161, 2013.

[Veb08] Oswald Veblen : Continuous increasing functions of finite and transfinite ordinals. *American Mathematical Society*, 9(3) :pp. 280–292, 1908.

[VS03] Nikolai Konstantinovich Vereshchagin et Alexander Shen : *Computable functions*, volume 19. American Mathematical Society, 2003.

[Wai72] Stanley S. Wainer : Ordinal recursion, and a refinement of the extended Grzegorczyk hierarchy. *Journal of Symbolic Logic*, 37(2) :281–292, 1972.

Sommaire