Université de Liège, Faculté des Sciences
Département de Mathématiques

Mémoire en vue de l'obtention du diplôme de
Master en sciences mathématiques

Promoteur : Georges Hansoul

Les théorèmes d'incomplétude

Renaud Hoyoux

Liège, le 6 septembre 2011

Remerciements

La réalisation de ce mémoire de fin d'étude fût un travail de longue haleine ainsi qu'un investissement conséquent en temps et en énergie.
Je voudrais ainsi remercier toutes les personnes m'ayant soutenu lors de son écriture :

- Ma maman, Annie Pierret, pour ses nombreuses relectures orthographiques.
- Monsieur Hansoul pour m'avoir permis de réaliser ce mémoire ainsi que pour son aide précieuse.
- Et, pour conclure, Thomas Kleyntssens et Isabelle Mainz pour leur soutien et leur affection tout au long de l'année.

Préambule

Depuis la Grèce Antique, la notion de preuve exacte a toujours été au centre des mathématiques.
Si l'on estime qu'auparavant une des raisons du développement de certaines formules géométriques était de rétablir les limites des champs recouverts lors des crues annuelles du Nil, il n'était pas encore question de preuve exacte de ces formules.
Chaque calcul était effectué en suivant une règle que l'on appliquait aveuglément pour la simple raison qu'elle donnait toujours une réponse correcte. Par conséquent, personne ne se souciait de contester ces règles. Les formules étaient correctes, les raisons pour lesquelles elles l'étaient n'avaient aucun intérêt.
Mais ces expériences n'étaient que des arguments empiriques et non pas une preuve absolue de la véracité de ces formules. En effet, même si une formule se vérifie pour un grand nombre de cas, nous ne pouvons aucunement affirmer de celle-ci qu'elle est toujours valide. Il est a priori possible qu'un cas particulier, fusse-t-il unique, infirme la véracité de la formule.

Ce n'est qu'à partir de la Grèce Antique que, sous l'impulsion des philosophes, les mathématiques quittent le domaine de l'utilitaire pour entrer dans celui de l'abstraction. Il n'est plus question d'appliquer aveuglément des formules parfois considérées comme magiques ; il faut, de manière analogue à l'argumentation philosophique, prouver et convaincre. Pour la première fois, on ressent le besoin de disposer d'une preuve formelle attestant que la formule donnera le bon résultat et ce, pour tous les problèmes respectant les hypothèses posées initialement.
Par la suite, les mathématiques se sont entièrement basées sur cette notion de rigueur.

Au tout début du XXème siècle, le paradoxe de Russel ébranle toute la structure mathématique en exhibant une contradiction dans la théorie naïve des ensembles. Si la théorie moderne des ensembles permit de résoudre ce paradoxe, le doute s'insinua : les mathématiques sont-elles exprimées formellement ? Sont-elles, dans leur état actuel, cohérentes, c'est-à-dire, n'existera-t-il jamais de contradiction à l'intérieur des mathématiques ?

En 1920, David Hilbert, après avoir formellement axiomatisé la géométrie, propose explicitement un programme de recherche dont un des éléments est la démonstration de la cohérence de l'arithmétique. Il souhaite que les mathématiques soient solidement et complètement formulées en s'appuyant sur la logique. Convaincu de la puissance des mathématiques, de leur validité et de leur capacité à tout démontrer, Hilbert prononce la célèbre

phrase « *Wir müssen wissen, wir werden wissen*» [1] exprimant le fait que l'*ignorabimus* [2] pourrait être éliminé des mathématiques. Peu après, en 1931, un logicien autrichien, Kurt Gödel, démontra que tout système formel non-contradictoire et suffisamment complet pour inclure l'arithmétique, ne peut démontrer sa cohérence en s'appuyant sur ses axiomes.
Non seulement il est impossible de déterminer si les mathématiques sont ou non cohérentes mais elles sont également incomplètes, c'est-à-dire qu'elles ne peuvent tout démontrer vu que leur cohérence est indémontrable.

Cependant, les travaux de Gödel ne furent pas considérés comme réellement importants par la communauté mathématique. En effet, beaucoup de mathématiciens estimèrent que les énoncés indémontrables présentés par Gödel étaient trop "artificiels" que pour avoir une "réelle" signification dans les mathématiques usuelles.
Il fallut attendre les travaux de Paul Cohen, Jeffrey Paris et de Leo Harrington pour avoir des énoncés indémontrables ayant pourtant un réel intérêt mathématique.
Grâce à cela, l'indémontrabilité devint un sujet véritablement inquiétant. Il n'était plus seulement question de se demander si certaines conjectures célèbres étaient vraies ou fausses ; il fallait également s'interroger sur leur possible indémontrabilité.

Par la suite, d'autres mathématiciens se sont posé la question « Combien existe-t-il d'énoncés indémontrables ? » ou d'une autre façon « A quel point les mathématiques sont-elles incomplètes ? ».
La théorie algorithmique de l'information, par l'intermédiaire de Christian Calude, Gregory Chaitin et de ses célèbres nombres Oméga, apporta les réponses à ces questions : il en existe une infinité "plus grande" que celle des énoncés démontrables. De plus, Robert Solovay, grâce aux travaux de Chaitin, exhiba une suite infinie d'énoncés indécidables.

Le grand rêve de Hilbert et de bon nombre de mathématiciens de son époque est dorénavant caduc.

Ce mémoire sera ordonné en quatre chapitres : le premier rappellera les bases nécéssaires en logique formelle et en théorie des fonctions récursives. Nous poserons et établirons également les définitions et résultats de base concernant les ensembles récursivement énumérables.

1. "*Nous devons savoir, nous saurons*"
2. Abréviation du latin *ignoramus et ignorabimus* signifiant « Nous ne savons pas et ne saurons jamais », exprimant le pessimisme apparu au XIXème siècle à propos des limites de la connaissance scientifique.

Le deuxième chapitre sera consacré aux théorèmes d'incomplétude de Gödel. Plus précisément, la première partie de ce chapitre établira la notion de fonctions représentables et le lien avec les fonctions récursives avant de décrire explicitement la gödelisation d'une théorie du premier ordre et de finir par le premier théorème d'incomplétude de Gödel.
La deuxième partie de ce chapitre, quant à elle, contiendra l'internalisation des théorèmes de logique formelle relatifs aux théories du premier ordre dans l'arithmétique de Peano, ce qui nous permettra de démontrer le deuxième théorème d'incomplétude de Gödel avant de clôturer par le théorème de non-définissabilité de Tarski et le théorème de Löb.

Le troisième chapitre racontera l'histoire du deuxième des Douze travaux d'Héraclès et présentera un exemple détaillé d'un énoncé aisé à comprendre mais néanmoins indémontrable dans l'arithmétique de Peano, prouvant que l'indémontrabilité ne concerne pas uniquement des énoncés "artificiels", construits dans le but d'être indémontrables, ou méta-mathématiques comme la consistance mais peut également concerner des théorèmes d'arithmétique "pure".

Le quatrième et dernier chapitre tentera de répondre à la question : « "Combien" existe-t-il d'énoncés indémontrables ? » en utilisant des arguments topologiques avant d'introduire brièvement la notion de complexité de Chaitin, et se concluera en exhibant une suite infinie d'énoncés indémontrables construite à partir d'un nombre Ω de Chaitin.

Table des matières

« All truth passes through three stages.
First, it is ridiculed.
Second, it is violently opposed.
Third, it is accepted as being self-evident. »

Arthur Schopenhauer

1 Rappels

Les théorèmes d'incomplétude, de par leur caractère surprenant et fatal au rêve de Hilbert, sont des résultats hautement symboliques en logique mathématique. Leurs démonstrations exploitent autant les résultats logiques des théories du premier ordre que de la théorie de la calculabilité.
Dans ce chapitre, nous rappelons les outils nécessaires à leurs démonstrations. Ce chapitre résume assez fidèlement [7] et [11]. Bien que notre volonté est d'être aussi complet que possible, certaines démonstrations, présentes dans ces livres de référence, seront omises.

1.1 Logique mathématique

La logique est une branche mathématique qui analyse les méthodes du raisonnement. Bien que les premiers écrits portant sur l'analyse du raisonnement remontent à Aristote, la logique mathématique, vue comme une partie intégrante des mathématiques, est née à la fin du XIXe siècle afin de résoudre la crise des fondements et de rétablir les mathématiques sur des bases qu'on espérait solides.

Rappelons, en toute généralité, ce qu'est un système formel déductif ainsi que les définitions de base en logique mathématique. Nous rappellerons ensuite la définition de la logique propositionnelle classique ainsi que celle d'une théorie du premier ordre.

Définition 1.1.1. Un *système formel déductif* est constitué de quatre composantes :

1) Une liste de symboles ou *signes primitifs* : c'est l'alphabet permettant de composer les phrases du discours. Une suite de signes primitifs est appelée *expression*.

2) Des *règles formatives* permettant de sélectionner parmi les expressions certaines qui sont dites bien formées et qui sont les phrases acceptées comme correctes dans le langage considéré.

3) Une liste d'expressions bien formées considérées comme vraies et appellées *axiomes* du système.

4) Des règles permettant d'obtenir de nouvelles expressions vraies à partir d'anciennes. Ce sont les *règles d'inférences* écrites sous la forme

$$R : \frac{\varphi_1, \ldots, \varphi_n}{\psi}$$

où $\varphi_1, \ldots, \varphi_n$ et ψ sont des expressions bien formées.
Nous dirons que ψ est *infèré* de $\varphi_1, \ldots, \varphi_n$ ou que ψ est une *conséquence directe* de $\varphi_1, \ldots, \varphi_n$ relativement à la règle R.

Définition 1.1.2. Soit S un système formel déductif.
Une *preuve* dans S est une suite d'expressions bien formées qui sont

- soit un axiome.
- soit une conséquence directe d'expressions bien formées précédentes.

Un *théorème* de S est une expression bien formée faisant partie d'une preuve.
Le fait que φ est un théorème de S est noté $S \vdash \varphi$. Nous dirons que φ est *prouvable* dans S.

Définition 1.1.3. Une *extension* S_1 de S est un système formel déductif contenant parmi ses signes primitifs, ses règles formatives, ses axiomes et ses règles d'inférences ceux de S.
Supposons que S_1 soit une extension de S par l'adjonction d'une liste Γ de nouveaux axiomes alors un théorème φ de S_1 est aussi appelé *conséquence* de Γ dans S.
On écrit $\Gamma \vdash \varphi$, le système déductif S étant sous-entendu.
Un système formel est *consistant* s'il existe des énoncés bien formés qui ne sont pas des théorèmes. Dans le cas contraire, le système est dit *inconsistant*.

Définition 1.1.4. La *logique propositionnelle classique* est le système déductif suivant :

1) Les signes primitifs qui se partagent en deux catégories :
   - Une liste $p_1, p_2, \ldots$ de *variables propositionnelles* ou *propositions atomiques*
   - Des symboles logiques :
     - Les connecteurs $\Rightarrow$ et $\neg$
     - et les parenthèses (et).

2) Les expressions bien formées dites *propositions*, régies par les règles formatives suivantes :
   - Toute proposition atomique est une proposition.
   - Si φ est une proposition alors $\neg\varphi$ est une proposition.
   - Si φ et ψ sont des propositions alors $(\varphi \Rightarrow \psi)$ est une proposition.
3) Les axiomes qui sont les propositions de la forme suivante :
   - L_1 : L'affirmation du conséquent $(\varphi \Rightarrow (\psi \Rightarrow \varphi))$
   - L_2 : L'autodistributivité de l'implication $((\varphi \Rightarrow (\psi \Rightarrow \theta)) \Rightarrow ((\varphi \Rightarrow \psi) \Rightarrow (\varphi \Rightarrow \theta)))$
   - L_3 : La contraposée converse $(\neg\varphi \Rightarrow \neg\psi) \Rightarrow (\psi \Rightarrow \varphi)$

   où φ, ψ et θ sont des expressions bien formées.
4) La seule règle d'inférence qui est la règle du *modus ponens* :

$$MP : \frac{\varphi, (\varphi \Rightarrow \psi)}{\psi}$$

pour toute proposition φ et ψ.

Définition 1.1.5. La *longueur* d'une formule F, notée $|F|$ est définie de la façon suivante :
- Si F est une proposition atomique, alors $|F| = 1$.
- Si F est $\neg G$, alors $|F| = 1 + |G|$.
- Si F est $G \Rightarrow H$, alors $|F| = |G| + |H| + 1$.

Les variables propositionnelles étant censées représenter des énoncés pouvant être considérés vrais ou faux, nous allons leur attribuer une valeur de vérité égale à 0 ou 1.

Définition 1.1.6. Une *fonction de vérité* désigne toute application v de l'ensemble des propositions dans $\{0, 1\}$ respectant les règles suivantes :
- $v(\neg\varphi) = 1 - v(\varphi)$
- $v(\varphi \Rightarrow \psi) = 1 - v(\varphi) + v(\varphi).v(\psi)$

Définition 1.1.7. Nous définissons de nouveaux connecteurs :
- La conjonction : $\varphi \wedge \psi \equiv \neg(\varphi \Rightarrow \neg\psi)$
- La disjonction : $\varphi \vee \psi \equiv (\varphi \Rightarrow \psi) \Rightarrow \psi$
- La bi-implication : $\varphi \Leftrightarrow \psi \equiv (\varphi \Rightarrow \psi) \wedge (\psi \Rightarrow \varphi)$.

De manière classique, la conjonction sera associée au OU logique, la disjonction au ET et la bi-implication à l'équivalence.

Définition 1.1.8. Une proposition dont la valeur de vérité est toujours 1, indépendamment des valeurs de vérité des variables propositionnelles qui y figurent, est appellée *tautologie* ; si la valeur de vérité est toujours 0, la proposition est qualifiée de *contradiction*.

Lorsque nous voudrons désigner une proposition F en précisant que les variables propositionnelles ayant au moins une occurence dans F se trouvent parmi $A_1, \ldots, A_n$, nous utiliserons la notation $F[A_1, \ldots, A_n]$.

Définition 1.1.9. Soient une proposition $F[A_1, \ldots, A_n, B_1, \ldots, B_m]$ et n propositions $G_1, \ldots, G_n$.
La *substitution* des propositions $G_1, \ldots, G_n$ aux variables propositionnelles $A_1, \ldots, A_n$, notée $F_{G_1/A_1,\ldots,G_n/A_n}$, est définie par induction de la façon suivante :

- Si F est une variable propositionnelle telle que $F = A_k$ pour $k = 1, \ldots, n$, alors $F_{G_1/A_1,\ldots,G_n/A_n}$ est G_k
- Si F est une variable propositionnelle telle que $F \notin \{A_1, \ldots, A_n\}$, alors $F_{G_1/A_1,\ldots,G_n/A_n}$ est F
- Si F est $\neg G$, alors $F_{G_1/A_1,\ldots,G_n/A_n}$ est $\neg G_{G_1/A_1,\ldots,G_n/A_n}$
- Si F est $G \Rightarrow H$, alors $F_{G_1/A_1,\ldots,G_n/A_n}$ est $G_{G_1/A_1,\ldots,G_n/A_n} \Rightarrow H_{G_1/A_1,\ldots,G_n/A_n}$

Remarquons que $F_{G_1/A_1,\ldots,G_n/A_n}$ reste une proposition.

Théorème 1.1.10. *Soient δ une fonction de vérité, n un entier, $F, G_1, \ldots, G_n$ des propositions et $A_1, \ldots, A_n$ des variables propositionnelles deux à deux distinctes.*
Appelons λ la fonction de vérité définie par :

$$\lambda(X) = \begin{cases} \delta(X) \text{ si } X \notin \{A_1, \ldots, A_n\} \\ \delta(G_i) \text{ si } X = A_i \text{ pour un } i \in \{1, \ldots, n\} \end{cases}$$

pour toute variable propositionnelle X. On a alors $\delta(F_{G_1/A_1,\ldots,G_n/A_n}) = \lambda(F)$.

Démonstration. Par récurrence sur la longueur de $|F|$.

– Cas de base : $|F| = 1$.
F est donc une variable propositionnelle.

- Si $F \notin \{A_1, \ldots, A_n\}$, alors $F_{G_1/A_1,\ldots,G_n/A_n} = F$. Par conséquent,

$$\delta(F_{G_1/A_1,\ldots,G_n/A_n}) = \delta(F) = \lambda(F).$$

- Si $F = A_i$ pour un $i \in \{1, \ldots, n\}$, alors $F_{G_1/A_1,\ldots,G_n/A_n} = G_i$. Par conséquent,

$$\delta(F_{G_1/A_1,\ldots,G_n/A_n}) = \delta(G_i) = \lambda(A_i) = \lambda(F).$$

– Cas d'induction :

- Si F est$\neg G$, alors $\delta(F_{G_1/A_1,\ldots,G_n/A_n}) = \delta(\neg G_{G_1/A_1,\ldots,G_n/A_n}) = 1-\delta(G_{G_1/A_1,\ldots,G_n/A_n})$. Vu que la longueur de G est inférieure à la longueur de F, nous pouvons lui appliquer l'hypothèse de récurrence.
  Nous avons donc

$$1 - \delta(G_{G_1/A_1,\ldots,G_n/A_n}) = 1 - \lambda(G) = \lambda(\neg G) = \lambda(F).$$

- Si F est $G \Rightarrow H$, alors

$$\begin{aligned}\delta(F_{G_1/A_1,\ldots,G_n/A_n}) &= \delta(G_{G_1/A_1,\ldots,G_n/A_n} \Rightarrow H_{G_1/A_1,\ldots,G_n/A_n}) \\ &= 1 - \delta(G_{G_1/A_1,\ldots,G_n/A_n}) + \delta(G_{G_1/A_1,\ldots,G_n/A_n}).\delta(H_{G_1/A_1,\ldots,G_n/A_n}).\end{aligned}$$

  Vu que les longueurs de G et H sont inférieures à la longueur de F, nous pouvons leur appliquer l'hypothèse de récurrence.
  Nous avons donc

$$\delta(G_{G_1/A_1,\ldots,G_n/A_n} \Rightarrow H_{G_1/A_1,\ldots,G_n/A_n}) = 1-\lambda(G)+\lambda(G).\lambda(H) = \lambda(G \Rightarrow H) = \lambda(F).$$

■

Une conséquence immédiate de ce théorème est que, si F est une tautologie, alors $F_{G_1/A_1,\ldots,G_n/A_n}$ est également une tautologie.

Voici enfin la définition des théories du premier ordre pour lesquelles nous allons démontrer les théorèmes d'incomplétude.

Définition 1.1.11. Une *théorie du premier ordre* T est définie par les données suivantes :

1) Les signes primitifs de T qui sont de trois types :
   a) Les variables, données en une liste $x_1, x_2, \ldots$
   b) Les signes logiques :
      - connecteurs : $\neg$ et $\Rightarrow$.
      - quantificateur universel $\forall$.
      - parenthèses (et).

c) Les signes primitifs propres formant le langage L de T, divisés en

- symboles fonctionnels $f_1, f_2, \ldots$ ayant chacun leur arité $n_1, n_2, \ldots$ propre, i.e. un naturel non nul qui leur est attaché.
- symboles relationnels ou prédicats $r_1, r_2, \ldots$ d'arités respectives $n_1^{'}, n_2^{'}, \ldots$
- symboles constants $c_1, c_2, \ldots$ représentant des objets particuliers du discours.

2) Les règles formatives sélectionnent deux catégories d'expressions bien formées :

a) Les termes définit de manière inductive :

- Toute variable, tout symbole constant est un terme.
- Si $t_1, \ldots, t_n$ sont des termes et f un symbole fonctionnel d'arité n, $f(t_1, \ldots, t_n)$ est un terme.

b) Les formules définies à partir des termes :

- Si $t_1, \ldots, t_n$ sont des termes et r un symbole relationnel d'arité n, $r(t_1, \ldots, t_n)$ est une formule appelée formule atomique.
- Si F et G sont des formules alors $\neg F$ et $F \Rightarrow G$ sont des formules.
- Si F est une formule et x est une variable alors $\forall x F$ est une formule.

Pour pouvoir formuler les axiomes, nous devons introduire les concepts de variables libres et liées ainsi que le concept de substitution.

- Toute occurence de x dans $\forall x F$ est dite *liée* par l'occurence considérée du quantificateur universel $\forall$.
- Une occurence d'une variable dans une formule qui n'est dans le domaine d'application d'aucune occurence d'un quantificateur universel est dite *libre*.
- Une variable est dite *libre* (resp *liée*) dans F si elle y possède une occurence libre (resp liée).
- Une *formule fermée* est une formule sans variable libre.
- La *substitution* d'un terme t à une variable x dans un terme v (resp. dans une formule F) noté $v(x := t)$ (resp. $F(x := t)$) est définie par induction sur le nombre de symboles intervenant dans v (resp. dans F) de la façon suivante :
  - Si y est la variable x, $y(x := t)$ est t.
    Si y est une autre variable ou un symbole constant, $y(x := t)$ est y.
  - $(f(t_1, \ldots, t_n))(x := t)$ est $f(t_1(x := t), \ldots, t_n(x := t))$ et $(r(t_1, \ldots, t_n))(x := t)$ est $r(t_1(x := t), \ldots, t_n(x := t))$.

- $(\neg F)(x := t)$ est $\neg(F(x := t))$ et $(F \Rightarrow G)(x := t)$ est $F(x := t) \Rightarrow G(x := t)$.
- Si y est une variable autre que x, $(\forall y F)(x := t)$ est $\forall y(F(x := t))$.
- $(\forall x F)(x := t)$ est $\forall x F$.

Nous constatons par induction que $v(x := t)$ (resp. $F(x := t)$) est un terme (resp. une formule).
Nous pouvons à présent poursuivre la définition d'une théorie du premier ordre.

3) Les axiomes de T sont de deux types :
   a) Les axiomes purement logiques qui sont les schémas d'axiome de la logique propositionnelle classique :

      L_1 : $F \Rightarrow (G \Rightarrow F)$

      L_2 : $(F \Rightarrow (G \Rightarrow H)) \Rightarrow ((F \Rightarrow G) \Rightarrow (F \Rightarrow H))$

      L_3 : $(\neg F \Rightarrow \neg G) \Rightarrow (G \Rightarrow F)$

      et les axiomes de la quantification :

      L_4 : Si t est un terme et si aucune occurence d'une variable de t ne devient liée dans $F(x := t)$ alors $\forall x F \Rightarrow F(x := t)$

      L_5 : Si x n'est pas libre dans F alors $(\forall x(F \Rightarrow G)) \Rightarrow (F \Rightarrow \forall x G)$

   b) Les autres axiomes, appelés axiomes propres de la théorie T.
      Ils sont tout comme le langage L spécifiques à T et eux seuls seront mentionnés lors de la description explicite d'une théorie du premier ordre.
      Si T ne contient pas d'axiomes propres, T est qualifiée de *théorie du calcul des prédicats*.

4) Il y a deux règles d'inférence :
   a) Le Modus Ponens : $\frac{F, F \Rightarrow G}{G}$
   b) La règle de généralisation : $\frac{F}{\forall x F}$

Remarque 1.1.12.
Il est possible de substituer des formules aux variables propositionnelles.
À titre d'exemple, supposons que A et B soient des variables propositionnelles. Soient $J = J[A, B] = ((A \wedge B) \Rightarrow \neg(\neg A \vee \neg B))$ et les deux formules suivantes d'un langage L :

$$F = \forall v_0(v_0 = v_0) \text{ et } G = v_1 \simeq c.$$

Alors en substituant les formules F et G aux variables propositionnelles A et B dans J, nous obtenons la formule

$$(((\forall v_0(v_0 = v_0)) \wedge (v_1 \simeq c)) \Rightarrow \neg(\neg(\forall v_0\ (v_0 = v_0)) \vee \neg(v_1 \simeq c))).$$

Remarquons que le résultat de substitutions de variables propositionnelles par des formules du premier ordre est toujours du premier ordre.

Pour la suite, nous définissons un nouveau quantificateur, le *quantificateur existentiel* comme étant

$$\exists F \equiv \neg\forall\neg F$$

pour toute formule F.

Lemme 1.1.13 (Lemme de déduction)**.** *Supposons que $\Gamma \cup \{F\} \vdash G$ par une preuve dans laquelle aucune généralisation d'une formule dépendant de F ne porte sur une variable ayant une occurence libre dans F. Alors $\Gamma \vdash F \Rightarrow G$.*

Démonstration. Admis. Voir [7] pour une démonstration complète. ■

Définition 1.1.14. Soit $\mathcal{L}$ un langage.
Une *$\mathcal{L}$-structure* $\mathfrak{M}$ est déterminée par les données suivantes :

1) Un ensemble M appelé domaine ou univers de la structure.
2) Pour chaque symbole fonctionnel f d'arité n, une fonction $f^{\mathfrak{M}} : M^n \to M$ associant à tout n-uple $(m_1, \ldots, m_n)$ d'éléments de M un élément de M.
3) Pour chaque symbole relationnel r d'arité n, une relation n-aire $r^{\mathfrak{M}} \subset M^n$ (i.e un ensemble de n-uples d'éléments de M).
4) Pour chaque symbole constant c de L, un élément $c^{\mathfrak{M}}$ de $\mathfrak{M}$.

Chaque $f^{\mathfrak{M}}, c^{\mathfrak{M}}, r^{\mathfrak{M}}$ est l'interprétation du symbole correspondant dans la structure $\mathfrak{M}$.

Définition 1.1.15. Étant donné deux $\mathcal{L}$-structures $\mathfrak{M}$ et $\mathfrak{N}$, $\mathfrak{M}$ est une *extension* de $\mathfrak{N}$ (ou $\mathfrak{N}$ est une *sous structure* de $\mathfrak{M}$) si et seulement si les conditions suivantes sont satisfaites :

- Le domaine N de $\mathfrak{N}$ est un sous-ensemble du domaine M de $\mathfrak{M}$.
- Pour tout symbole de constante c de L, $c^{\mathfrak{N}} = c^{\mathfrak{M}}$
- Pour tout entier naturel $k \geq 1$ et pour tout symbole de fonction k-aire f de L, $f^{\mathfrak{N}} = f^{\mathfrak{M}}\big|_{N^k}$
- pour tout entier naturel $k \geq 1$ et pour tout symbole de relation k-aire r de L, $r^{\mathfrak{N}} = r^{\mathfrak{M}} \cap N^k$.

Définition 1.1.16. Soit $x_1, \ldots, x_n$ une liste de variables éventuellement vide.
Une *interprétation* τ de cette liste dans $\mathfrak{M}$ est une liste correspondante $m_1, \ldots m_n$ d'éléments de M.
On écrit $\tau(x_1) = m_1, \ldots, \tau(x_n) = m_n$.
L'interprétation $t_\tau^{\mathfrak{M}}$ d'un terme t sous une interprétation τ de ses variables est définie comme suit :

a) Si t est une variable x_i, alors $t_\tau^{\mathfrak{M}} = \tau(x_i)$

b) Si t est une constante c alors $t_\tau^{\mathfrak{M}} = c^{\mathfrak{M}}$

c) Si t est $f(t_1, \ldots, t_n)$, alors $t_\tau^{\mathfrak{M}} = f^{\mathfrak{M}}(t_{1,\tau}^{\mathfrak{M}}, \ldots, t_{n,\tau}^{\mathfrak{M}})$

Soit $x_1, \ldots, x_n$ la liste des variables ayant une occurence libre dans une formule F.
Si τ est une interprétation de $x_1, \ldots, x_n$ dans $\mathfrak{M}$, nous définissons l'expression « F est *satisfait* dans $\mathfrak{M}$ sous l'interprétation τ » (noté $\mathfrak{M} \vDash_\tau F$) comme suit :

1) Si F est une formule atomique $r(t_1, \ldots, t_n)$ alors $\mathfrak{M} \vDash_\tau F$ si $(t_1^{\mathfrak{M}}, \ldots, t_n^{\mathfrak{M}}) \in r^{\mathfrak{M}}$.

2) Si F est $\neg G$, alors $\mathfrak{M} \vDash_\tau F$ s'il n'est pas vrai que $\mathfrak{M} \vDash_\tau G$ (noté $\mathfrak{M} \nvDash_\tau G$)

3) Si F est $G \Rightarrow H$, alors $\mathfrak{M} \vDash_\tau F$ si $\mathfrak{M} \nvDash_\tau G$ ou si $\mathfrak{M} \vDash_\tau H$

4) Si F est $\forall x\ G$ alors $\mathfrak{M} \vDash_\tau F$ si $\mathfrak{M} \vDash_{\tau'} G$ quelle que soit l'interprétation τ' de $x_1, \ldots, x_n, x$ étendant τ [1].

Nous dirons que F est *valide dans* $\mathfrak{M}$ (noté $\mathfrak{M} \vDash F$) lorsque $\mathfrak{M} \vDash_\tau F$ quelle que soit l'interprétation τ des variables libres de F.
Nous dirons que F est *logiquement valide* (noté $\vDash F$) si $\mathfrak{M} \vDash F$ quelle que soit la L-structure $\mathfrak{M}$ considérée.

Remarque 1.1.17.

- Par la suite, nous mettrons en évidence les variables libres $x_1, \ldots, x_n$ de F en notant $F \equiv F[x_1, \ldots, x_n]$.
- Soient $m_1, \ldots, m_n$ des éléments de $\mathfrak{M}$ et τ une interprétation telle que $\tau(x_i) = m_i$ pour tout entier i compris entre 1 et n. Nous dirons parfois que le n-uple $(m_1, \ldots, m_n)$ satisfait une formule $F[x_1, \ldots, x_n]$ si $\mathfrak{M} \vDash_\tau F[x_1, \ldots, x_n]$.

Définition 1.1.18. Soit $\mathcal{L}$ un langage du premier ordre.

- Soit une formule $F = F[v_1, v_2, \ldots, v_n]$ du langage $\mathcal{L}$ dans laquelle chacune des variables $v_1, v_2, \ldots, v_n$ a au moins une occurence. La formule close $\forall v_1 \forall v_2 \ldots \forall v_n F$ est appellée la *clôture universelle* de la formule F.

1. Remarquons que x étant lié dans F par le quantificateur universel, il ne figure pas parmi $x_1, \ldots, x_n$

- Une formule close de $\mathcal{L}$ est *contradictoire* si et seulement si sa négation est logiquement valide.
- Etant donné deux formules F et G de $\mathcal{L}$, nous dirons que F est *équivalente* à G (noté $F \sim G$) si et seulement si la formule $F \Leftrightarrow G$ est logiquement valide.
- Une *théorie* de $\mathcal{L}$ est un ensemble de formules closes de $\mathcal{L}$.
- Soient une théorie T de $\mathcal{L}$ et une $\mathcal{L}$-structure $\mathfrak{M}$, $\mathfrak{M}$ est un *modèle* de T (noté $\mathfrak{M} \vDash T$) si et seulement si $\mathfrak{M}$ satisfait chaque formule appartenant à T.
- Une théorie *consistante* (ou *cohérente*) est une théorie qui admet au moins un modèle.

Lemme 1.1.19. *Considérons des variables propositionnelles $A_1, A_2, \ldots, A_k$, une formule propositionnelle $J[A_1, \ldots, A_n]$ et des formules du premier ordre du langage $\mathcal{L}$: $F_1, \ldots, F_k$. Si la formule J est une tautologie, alors la formule du premier ordre $J[F_1, \ldots, F_k]$ est une formule logiquement valide.*

Démonstration. Supposons que les variables libres des formules $F_1, \ldots F_k$ soient parmi $v_0, v_1, \ldots, v_n$. Il en est alors de même des variables libres de la formule $F := J[F_1, F_2, \ldots, F_k]$. Considérons une $\mathcal{L}$-structure $\mathfrak{M}$ et une interprétation τ de $v_0, \ldots, v_n$ dans $\mathfrak{M}$. Définissons une distribution de valeurs de vérité δ sur $\{A_1, A_2, \ldots, A_k\}$ en posant, pour $1 \leq i \leq k$:

$$\delta(A_i) = \begin{cases} 1 \text{ si } \mathfrak{M} \vDash_\tau F_i \\ 0 \text{ si } \mathfrak{M} \nvDash_\tau F_i \end{cases}$$

Montrons par induction sur la longueur de J que la formule F est satisfaite dans $\mathfrak{M}$ sous cette interprétation τ si et seulement si la distribution de valeurs de vérité δ donne la valeur 1 à la formule J.
Cas de base : $|J| = 1$
J est donc une variable propositionnelle A_i. F est donc F_i. Le cas de base est vérifié par définition de δ.
Cas d'induction :
Deux cas sont possibles :

- J est de la forme $J' \Rightarrow J''$.
  Posons $F' := J'[F_1, F_2, \ldots, F_k]$ et $F'' := J''[F_1, F_2, \ldots, F_k]$. Par conséquent, $\delta(J) = 1$ si et seulement si $\delta(J') = 0$ ou $\delta(J'') = 1$.
  Vu que les longueurs de J' et J'' sont inférieures à la longueur de J, nous pouvons

leur appliquer l'hypothèse de récurrence. Nous avons donc

$$\begin{aligned}\delta(J^{'}) = 0 \text{ ou } \delta(J^{''}) = 1 &\text{ si et seulement si } \mathfrak{M} \nvDash_{\tau} F^{'} \text{ ou } \mathfrak{M} \vDash_{\tau} F^{''} \\ &\text{si et seulement si } \mathfrak{M} \vDash_{\tau} F^{'} \Rightarrow F^{''} \\ &\text{si et seulement si } \mathfrak{M} \vDash_{\tau} F.\end{aligned}$$

- J est de la forme $\neg J^{'}$.
  Posons $F^{'} := J[^{'}F_1, F_2, \ldots, F_k]$. Nous avons donc $\delta(J) = 1$ si et seulement si $\delta(J^{'}) = 0$.
  Vu que la longueur de $J^{'}$ est inférieure à la longueur de J, nous pouvons lui appliquer l'hypothèse de récurrence. Nous avons donc

$$\begin{aligned}\delta(J^{'}) = 0 \text{ si et seulement si } \mathfrak{M} \nvDash_{\tau} F^{'} \text{ si et seulement si } \mathfrak{M} \vDash_{\tau} \neg F^{'} \\ \text{si et seulement si } \mathfrak{M} \vDash_{\tau} F.\end{aligned}$$

On en déduit immédiatement que lorsque J est une tautologie

$$\mathfrak{M} \vDash_{\tau} F$$

et ce, quelle que soit l'interprétation τ, ce qui prouve que $\mathfrak{M} \vDash F$. Vu que $\mathfrak{M}$ est quelconque, F est logiquement valide. ■

Les formules logiquement valides obtenues par la méthode qui vient d'être indiquée à partir de tautologies propositionnelles sont appelées *tautologies du calcul des prédicats.*

Le théorème suivant, démontré par Gödel aux alentours de 1930, est primordial pour les théories logiques du premier ordre.
La preuve de ce théorème étant assez longue, nous admettons ce résultat. Le lecteur intéressé trouvera une preuve complète dans [7]

Théorème 1.1.20 (Théorème de complétude)**.** *Dans une théorie T, une formule est prouvable si et seulement si elle est vraie dans chaque modèle. i.e $T \vdash F$ si et seulement si $\mathfrak{M} \vDash F$ et cela quel que soit $\mathfrak{M} \vDash T$*

1.2 Fonctions récursives

La théorie des fonctions récursives dont nous allons rappeller ici les bases est une théorie récente visant à formaliser la notion de fonction calculable.
Intuitivement, une fonction est calculable s'il existe une procédure effective, un algorithme, pouvant la calculer.
Les fonctions calculables furent d'abord modélisées par le λ-calcul, mais d'autres formalismes comme les fonctions récursives ou les machines de Turing ont été proposés par la suite. Il a ensuite été prouvé que toutes ces définitions, bien que fondées sur des idées assez différentes, décrivent exactement le même ensemble de fonctions.
Cependant, vu qu'il est impossible de définir formellement une fonction effectivement calculable, nous ne pourrons jamais être certains du fait que ces théories modélisent parfaitement cette notion intuitive. La thèse de Church (appellée parfois thèse de Church-Turing) affirme que notre notion intuitive de fonction calculable coïncide parfaitement avec la notion formelle.
Le fait que toutes ces tentatives pour formaliser le concept d'algorithme aient conduit à des résultats équivalents est l'argument qui rend crédible la thèse de Church.

Nous allons ici définir les fonctions récursives ainsi que les ensembles récursivement énumérables. Nous allons également démontrer la majorité des résultats basiques de ces deux notions.

Définition 1.2.1. Soit p un entier. Notons $\mathfrak{F}_p$ l'ensemble des applications de $\mathbb{N}^p$ dans $\mathbb{N}$. Dans le cas où $p = 0$, les éléments de $\mathfrak{F}_0$ sont identifiés avec les éléments de $\mathbb{N}$.
Posons également

$$\mathfrak{F} = \bigcup_{p \in \mathbb{N}} \mathfrak{F}_p$$

Nous commençons par introduire des fonctions qui vont nous servir de composants élémentaires dans l'élaboration de nouvelles fonctions. Nous les appellerons dès lors *fonctions initiales*. Elles sont de trois types.

Définition 1.2.2. Soient i et p deux entiers tels que $1 \leq i \leq p$. La *fonction de projection* $\mathcal{P}_{i,p}$ appartenant à $\mathfrak{F}_p$ est définie par

$$\mathcal{P}_{i,p} : \mathbb{N}^p \to \mathbb{N} : (x_1, \ldots, x_p) \mapsto x_i.$$

Par exemple, $\mathcal{P}_{1,1} : \mathbb{N} \to \mathbb{N}$ est la fonction identité.

Définition 1.2.3. La fonction *successeur* σ appartenant à $\mathfrak{F}_1$ est définie par

$$\sigma : \mathbb{N} \to \mathbb{N} : x \mapsto x + 1 :$$

Définition 1.2.4. La *fonction zéro* **0** appartenant à $\mathfrak{F}_0$ est simplement la constante 0. C'est une fonction sans argument qui a toujours la valeur 0.

Nous introduisons à présent deux moyens de construction de nouvelles fonctions à partir de fonctions déjà définies.

Définition 1.2.5. Soient $f_1, \ldots, f_n$ des fonctions de $\mathfrak{F}_p$ et g une fonction de $\mathfrak{F}_n$. La *fonction composée* $h = g(f_1, \ldots, f_n)$ est la fonction de $\mathfrak{F}_p$ définie par

$$h(x_1, \ldots, x_p) = g(f_1(x_1, \ldots, x_p), \ldots, f_n(x_1, \ldots, x_p)).$$

Dans la suite, nous noterons indifféremment la composée $g \circ f$ ou $g(f)$.

Définition 1.2.6. Soient $g \in \mathfrak{F}_p$ et $h \in \mathfrak{F}_{p+2}$. Définissons une fonction $f \in \mathfrak{F}_{p+1}$ telle que pour tous $x_1, \ldots, x_p, n \in \mathbb{N}$

- $f(x_1, \ldots, x_p, 0) = g(x_1, \ldots, x_p)$
- $f(x_1, \ldots, x_p, n+1) = h(x_1, \ldots, x_p, n, f(x_1, \ldots, x_p, n))$.

Nous dirons que f est définie par *récursion primitive* à partir de g et de h.

Définition 1.2.7. L'ensemble des fonctions *primitives récursives* est le plus petit sous-ensemble $\mathcal{PR}$ de $\mathfrak{F}$ qui

- contient la fonction **0**
- contient les fonctions de projection $\mathcal{P}_{i,p}$ quels que soient les entiers i et p tels que $1 \leq i \leq p$
- contient la fonction successeur σ
- est stable par composition
- est stable par récursion primitive

Proposition 1.2.8. *Soit p un entier. Les fonctions constantes de $\mathfrak{F}_p$ sont primitives récursives.*

Démonstration. Soient n un naturel et n_p la fonction de $\mathfrak{F}_p$ définie par

$$n_p : \mathbb{N}^p \to \mathbb{N} : (x_1, \ldots, x_n) \mapsto n$$

Procédons par récurrence sur p.
Cas de base : $p = 0$
Une simple récurrence sur n montre que n_0 est primitive récursive.
En effet, $n_0 = \underbrace{\sigma \circ \sigma \ldots \sigma}_{n \text{ fois}} \circ \mathbf{0}$.
Cas d'induction : Nous pouvons définir n_{p+1} par récursion primitive à partir de n_p et de $\mathcal{P}_{p+2,p+2}$.

$$\begin{cases} n_{p+1}(x_1, \ldots, x_p, 0) = n_p(x_1, \ldots, x_p) \\ n_{p+1}(x_1, \ldots, x_p, m+1) = \mathcal{P}_{p+2,p+2}(x_1, \ldots, x_n, m, n_{p+1}(x_1, \ldots, x_p, m)), \end{cases}$$

d'où la conclusion. ■

Proposition 1.2.9. *Les fonctions suivantes :*

- *prédécesseur* $\mathcal{P} : n \mapsto \begin{cases} n-1 \text{ si } n > 0 \\ 0 \text{ sinon} \end{cases}$ *appartenant à* $\mathfrak{F}_1$.
- *monus* $\dot{-} : (x, y) \mapsto \begin{cases} x - y \text{ si } x > y \\ 0 \text{ sinon} \end{cases}$ *appartenant à* $\mathfrak{F}_2$.
- *signe sign* $: n \mapsto \begin{cases} 1 \text{ si } n > 0 \\ 0 \text{ si } n = 0 \end{cases}$ *appartenant à* $\mathfrak{F}_1$

sont primitives récursives.

Démonstration.

- Par récursion primitive

$$\begin{cases} \mathcal{P}(0) = \mathbf{0} \\ \mathcal{P}(n+1) = \mathcal{P}_{1,2}(n, \mathcal{P}(n)) \end{cases}$$

- Par récursion primitive

$$\begin{cases} \dot{-}(x, 0) = \mathcal{P}_{1,1}(x) \\ \dot{-}(x, y+1) = \mathcal{P} \circ \mathcal{P}_{3,3}(x, y, \dot{-}(x, y)) \end{cases}$$

  Dans la suite, nous noterons $x \dot{-} y$ au lieu de $\dot{-}(x, y)$
- C'est immédiat en remarquant que $sign(x) = 1 \dot{-} (1 \dot{-} x)$.

■

Proposition 1.2.10. *Soient un entier* $p \geq 1$ *et une fonction* f *appartenant à* $\mathfrak{F}_{p+1}$. *Les fonctions*

- *addition* $\Sigma_p : (x_1, \ldots, x_p) \mapsto \Sigma_{i=1}^{p} x_i$ *appartenant à* $\mathfrak{F}_p$
- *produit* $\Pi_p : (x_1, \ldots, x_p) \mapsto \Pi_{i=1}^{p} x_i$ *appartenant à* $\mathfrak{F}_p$
- *puissance* $p : (x, n) \mapsto x^n$ *appartenant à* $\mathfrak{F}_2$
- *somme bornée* $g(x_1, \ldots, x_p, n) = \Sigma_{i=0}^{n} f(x_1, \ldots, x_p, i)$ *appartenant à* $\mathfrak{F}_{p+1}$

sont primitives récursives.

Démonstration.

- Par récurrence sur p :
  Pour $p = 1$, on a $\Sigma_1 = \mathcal{P}_{1,1}$.
  Supposons le résultat acquis pour p et vérifions-le pour $p + 1$ en se ramenant au schéma de définition par récursion primitive :

$$\begin{cases} \Sigma_{p+1}(x_1, \ldots, x_p, 0) = \Sigma_p(x_1, \ldots, x_p) \\ \Sigma_{p+1}(x_1, \ldots, x_p, n+1) = \sigma(\mathcal{P}_{p+2,p+2}(x_1, \ldots, x_p, n, \Sigma_{p+1}(x_1, \ldots, x_p, n))) \end{cases}$$

- Par récurrence sur p :
  De manière analogue, nous avons $\Pi_1 = \mathcal{P}_{1,1}$ pour le cas de base et

$$\begin{cases} \Pi_{p+1}(x_1, \ldots, x_p, 0) = 0_p(x_1, \ldots, x_p) \\ \Pi_{p+1}(x_1, \ldots, x_p, n+1) = h(x_1, \ldots, x_p, n, \Pi_{p+1}(x_1, \ldots, x_p, n)) \end{cases}$$

  où $h = \Sigma_2(\Pi_p(\mathcal{P}_{1,p+2}, \ldots, \mathcal{P}_{p,p+2}), \mathcal{P}_{p+2,p+2})$ pour le cas d'induction.
- Nous définissons également la fonction puissance par récursion primitive

$$\begin{cases} p(x, 0) = 1_1(x) \\ p(x, n+1) = \Pi_2(\mathcal{P}_{1,3}, \mathcal{P}_{3,3})(x, n, p(x, n)) \end{cases}$$

- Par récursion primitive, nous avons

$$\begin{cases} g(x_1, \ldots, x_p, 0) = h(x_1, \ldots, x_p) \\ g(x_1, \ldots, x_p, n+1) = k(x_1, \ldots, x_p, n, g(x_1, \ldots, x_p, n)) \end{cases}$$

  où $h = f(\mathcal{P}_{1,p}, \ldots, \mathcal{P}_{p,p}, 0_p)$ et $k = \Sigma_2(\mathcal{P}_{p+2,p+2}, f(\mathcal{P}_{1,p+2}, \ldots, \mathcal{P}_{p,p+2}, \sigma \circ \mathcal{P}_{p+1,p+2}))$.

■

Remarquons que l'addition, le produit et la puissance de fonctions primitives récursives restent primitives récursives. Par exemple, pour l'addition $\Sigma_{i=1}^{p} f_i(x_1, \ldots, x_n) = (\Sigma_p \circ (f_1, \ldots, f_p))(x_1, \ldots, x_n)$.

Définition 1.2.11. Soit E un ensemble.
La *fonction caractéristique* de E est la fonction χ_E définie par

$$\chi_E : x \mapsto \begin{cases} \chi_E(x) = 1 \text{ si } x \in E \\ \chi_E(x) = 0 \text{ sinon} \end{cases}$$

Définition 1.2.12. Un ensemble $E \subset \mathbb{N}^p$ est dit *primitif récursif* si sa fonction caractéristique est primitive récursive.

Proposition 1.2.13. *Pour chaque entier p, l'ensemble des sous-ensembles primitifs récursifs de $\mathbb{N}^p$ est clos pour les opérations booléennes.*

Démonstration. Soient A et B des sous-ensembles primitifs récursifs de $\mathbb{N}^p$. Les fonctions caractéristiques suivantes :

- $\chi_{A\cup B} = sign(\chi_A + \chi_B)$
- $\chi_{A\cap B} = \chi_A \cdot \chi_B$
- $\chi_{\mathbb{N}^p \setminus A} = 1 \dot{-} \chi_A$

sont bien primitives récursives.
Et, vu que $A \setminus B = A \cap (\mathbb{N}^p \setminus B)$, $A \setminus B$ est également primitif récursif ■

Définition 1.2.14. Soit P un prédicat d'arité p. Ce prédicat sera dit *primitif récursif* si son ensemble caractéristique $\delta_P = \{(x_1, \ldots, x_p) | P(x_1, \ldots, x_p) \text{ est vrai}\} \subset \mathbb{N}^p$ est primitif récursif.

Exemple 1.2.15. Le prédicat « x est pair » est primitif récursif.
En effet, la fonction caractéristique peut se définir comme suit :

- $\chi(0) = 1$
- $\chi(n+1) = 1 \dot{-} \chi(n)$

Cette fonction est effectivement primitive récursive.

Proposition 1.2.16. *Les prédicats binaires $<, \leq, =, \geq, >$ sont primitifs récursifs.*

Démonstration. En effet, $\chi_<(x, y) = sign(y \dot{-} x)$.
Les autres prédicats s'obtiennent immédiatement en appliquant la proposition 1.2.13. ■

Proposition 1.2.17. *Soient f et g deux fonctions primitives récursives de $\mathfrak{F}_p$. Les ensembles suivants :*

- $\{(x_1, \ldots, x_p) \in \mathbb{N}^p \,|\, f(x_1, \ldots, x_p) < g(x_1, \ldots, x_p)\}$
- $\{(x_1, \ldots, x_p) \in \mathbb{N}^p \,|\, f(x_1, \ldots, x_p) \leq g(x_1, \ldots, x_p)\}$

- $\{(x_1, \ldots, x_p) \in \mathbb{N}^p \,|\, f(x_1, \ldots, x_p) = g(x_1, \ldots, x_p)\}$
- $\{(x_1, \ldots, x_p) \in \mathbb{N}^p \,|\, f(x_1, \ldots, x_p) \geq g(x_1, \ldots, x_p)\}$
- $\{(x_1, \ldots, x_p) \in \mathbb{N}^p \,|\, f(x_1, \ldots, x_p) < g(x_1, \ldots, x_p)\}$

sont primitifs récursifs.

Démonstration. Démontrons pour $<$.
La fonction caractéristique de l'ensemble est $\chi_<(f, g)$ qui est une composée de fonctions primitives récursives.
Les autres cas sont analogues. ■

Proposition 1.2.18 (Quantification bornée)**.** *Soit P un prédicat primitif récursif d'arité $p+1$. Les prédicats*

- $A(x_1, \ldots, x_p, m) \equiv \forall n \leq m \ P(x_1, \ldots, x_p, n)$
- $B(x_1, \ldots, x_p, m) \equiv \exists n \leq m \ P(x_1, \ldots, x_p, n)$

sont primitifs récursifs.

Démonstration. Nous avons

$$\chi_A(x_1, \ldots, x_p, m) = \Pi_{i=0}^m \chi_P(x_1, \ldots, x_p, i)$$

et

$$\chi_B(x_1, \ldots, x_p, m) = sign(\Sigma_{i=0}^m \chi_P(x_1, \ldots, x_p, i))$$

■

Proposition 1.2.19 (Schéma de définition par cas)**.** *Soient $f_1, f_2, \ldots, f_{n+1} \in \mathfrak{F}_p$ des fonctions primitives récursives et $A_1, A_2, \ldots, A_n$ des sous-ensembles primitifs récursifs de $\mathbb{N}^p$; alors la fonction g définie par*

$$\begin{cases} g(x_1, \ldots, x_p) = f_1(x_1, \ldots, x_p) \ si \ (x_1, \ldots, x_p) \in A_1 \\ g(x_1, \ldots, x_p) = f_2(x_1, \ldots, x_p) \ si \ (x_1, \ldots, x_p) \notin A_1 \ et \ (x_1, \ldots, x_p) \in A_2, \\ \vdots \\ g(x_1, \ldots, x_p) = f_n(x_1, \ldots, x_p) \ si \ (x_1, \ldots, x_p) \notin \bigcup_{i=1}^{n-1} A_i \ et \ (x_1, \ldots, x_p) \in A_n \\ g(x_1, \ldots, x_p) = f_{n+1}(x_1, \ldots, x_p) \ si \ (x_1, \ldots, x_p) \notin \bigcup_{i=1}^{n} A_i \end{cases}$$

est primitive récursive.

Démonstration. Nous avons que

$$g = f_1.\chi(A_1) + f_2.\chi(A_2 \setminus A_1) + \cdots + f_n.\chi(A_n \setminus \bigcup_{i=1}^{n-1} A_i) + f_{n+1}.\chi(\mathbb{N}^p \setminus \bigcup_{i=1}^{n} A_i)$$

Vu que g est une somme finie de fonctions primitives récursives, elle est primitive récursive.

■

Proposition 1.2.20 (Schéma μ borné)**.** *Soit A un sous-ensemble primitif récursif de $\mathbb{N}^{p+1}$. Alors la fonction f de $\mathfrak{F}_{p+1}$ définie comme suit est primitive récursive.*

- *$f(x_1, \ldots, x_p, z) = 0$ s'il n'existe pas d'entier $t \leq z$ tel que $(x_1, \ldots, x_p, t) \in A$*
- *$f(x_1, \ldots, x_p, z)$ est égal au plus petit des entiers t tels que $(x_1, \ldots, x_p, t) \in A$ sinon.*

Démonstration. Nous définissons f par récursion primitive

$$\begin{cases} f(x_1, \ldots, x_p, 0) = 0 \\ f(x_1, \ldots, x_p, z+1) = h(x_1, \ldots, x_p, z, f(x_1, \ldots, x_p, z)) \end{cases}$$

Nous appliquons le schéma de définition par cas à la fonction h.

$$h = \begin{cases} \mathcal{P}_{p+2,p+2} \text{ si } \sum_{y=0}^{z} \chi_A(x_1, \ldots, x_p, y) \geq 1 \\ \sigma(\mathcal{P}_{p+1,p+2}) \text{ sinon et si } (x_1, \ldots, x_p, z+1) \in A \\ 0 \text{ dans les autres cas} \end{cases}$$

En effet, si $\sum_{y=0}^{z} \chi_A(x_1, \ldots, x_p, y) \geq 1$, cela signifie qu'il existe un entier $t < z+1$ tel que $(x_1, \ldots, x_p, t) \in A$. Par conséquent, $f(x_1, \ldots, x_p, z+1) = f(x_1, \ldots, x_p, z)$

■

Proposition 1.2.21. *Pour tout entier p, il existe des fonctions primitives récursives $\alpha_p \in \mathfrak{F}_p$ et $\beta_1^p, \ldots, \beta_p^p \in \mathfrak{F}_1$ telles que*

$$\beta_i^p(\alpha_p(x_1, \ldots, x_p)) = x_i \quad \forall i \in \{1, \ldots, p\} \quad (x_1, \ldots, x_p) \in \mathbb{N}^p \,.$$

En d'autres termes, la fonction α_p fait correspondre à tout point de $\mathbb{N}^p$ un élément de $\mathbb{N}$ et, réciproquement, on retrouve les composantes du p-uple initial grâce aux fonctions β_i^p.

Démonstration.
Nous nous contentons de donner les fonctions, définies par récurrence sur p :
Cas de base : $p = 2$

$$\alpha_2(i, j) = \frac{1}{2}(i+j)(i+j+1) + j$$

$$\beta_2^1(n) = \mu_{t \leq n}(\exists j \leq n : \alpha_2(t, j) = n) \text{ et } \beta_2^2(n) = \mu_{t \leq n}(\exists i \leq n : \alpha_2(i, t) = n).$$

Cas d'induction :

$$\alpha_{p+1}(x_1 \ldots, x_{p+1}) = \alpha_2(x_1, \alpha_p(x_2, \ldots, x_{p+1})).$$

$$\beta^1_{p+1}(n) = \beta^1_2(n) \text{ et } \beta^i_{p+1}(n) = \beta^i_p(n) \circ \beta^2_2(n) \quad \forall i \in \{2, \ldots, p+1\}.$$

Pour une démonstration complète, le lecteur intéressé pourra consulter [11].
Nous remarquons que $\alpha_{p+1}(x_1 \ldots, x_{p+1}) > x_i \ \forall i \in \{1, \ldots, p+1\}$. ■

Proposition 1.2.22. *Les éléments suivants :*

- *La fonction $q(x, y)$ qui est égale à la partie entière de $\frac{x}{y}$ si y est non nul et à 0 si y est nul*
- *La fonction* $\mod(x, y)$*, notée* $x \mod y$*, qui est égale au reste de la division entière de x par y*
- *L'ensemble $D = \{(x, y)\ ;\ y \text{ divise } x\ \}$*
- *L'ensemble $P = \{x\ ;\ x \text{ est premier}\ \}$*

sont primitifs récursifs

Démonstration.

- $q(x, y) = \mu_{t \leq x}((t+1).y > x)$
- $\mod(x, y) = x \dot{-} (y.q(x, y))$
- $\chi_D(x, y) = 1 \dot{-} sign(x \dot{-} y.q(x, y))$
- $x \in P$ si et seulement si $x \geq 2$ et $\forall y \leq x (y \leq 1$ ou y ne divise pas x ou $y = x)$.
  Le prédicat P est donc primitif récursif vu la proposition 1.2.18.

■

Malheureusement, l'espoir de modéliser toutes les fonctions effectivement calculables par les fonctions primitives récursives fut un échec. En 1926, Wilhelm Ackermann trouva une fonction calculable au sens intuitif mais non au sens formel de l'époque, c'est-à-dire, primitive récursive.
La définition suivante va nous permettre de définir l'ensemble des fonctions récursives qui contient strictement l'ensemble des fonctions primitives récursives. En effet, la fonction d'Ackermann est récursive mais n'est pas primitive récursive. Le lecteur intéressé par ce résultat pourra consulter [6].

Définition 1.2.23 (Schéma de minimisation)**.** Soit $A \subset \mathbb{N}^{p+1}$. On définit la fonction $f \in \mathfrak{F}_p$ comme suit :

$$f(x_1, \ldots, x_p) \text{ est égal au plus petit des entiers } t \text{ tels que } (x_1, \ldots, x_p, t) \in A.$$

On note cette fonction $f(x_1, \ldots, x_p) = \mu_t((x_1, \ldots, x_p, t) \in A)$.

Si la minimisation bornée appliquée à des ensembles primitifs récursifs fournit des fonctions primitives récursives, cela n'est pas toujours vrai pour la minimisation non bornée. En effet, en utilisant ce schéma, il est possible de construire des fonctions non calculables. S'il n'existe pas d'entier t tel que $(x_1, \ldots, x_p, t)$ appartienne à A, alors la procédure effective calculant la fonction $\mu_t((x_1, \ldots, x_p, t) \in A)$ ne s'arrêtera jamais.
C'est pour cette raison que nous introduisons la définition suivante.

Définition 1.2.24. Une partie $A \subset \mathbb{N}^{p+1}$ est dite sûre si

$$\forall (x_1, \ldots, x_p) \in \mathbb{N}^p; \exists t \in \mathbb{N} : (x_1, \ldots, x_p, t) \in A.$$

Dans la première moitié de ce mémoire, nous considérerons uniquement des parties ou des prédicats sûrs. De cette manière, le schéma de minimisation (non borné) fournira toujours des fonctions calculables. Pour cette raison, on parle parfois du schéma μ total.

Définition 1.2.25. L'ensemble des fonctions *récursives* est le plus petit sous-ensemble de $\mathfrak{F}$ qui

- contient les fonctions primitives récursives de base,
- est stable par composition,
- est stable par récursion primitive,
- est stable par la minimisation (non bornée de parties sûres).

Remarque 1.2.26. Nous pouvons définir de manière équivalente un schéma de minimisation pour les fonctions.
En effet, soit une fonction $g : \mathbb{N}^{p+1} \to \mathbb{N}$. Nous pouvons définir la fonction $f \in \mathfrak{F}_p$ comme

$$f(x_1, \ldots, x_p) \text{ est égal au plus petit des entiers } t \text{ tels que } g(x_1, \ldots, x_p, t) = 0.$$

On note cette fonction $f(x_1, \ldots, x_p) = \mu_t(g(x_1, \ldots, x_p, t) = 0)$.
L'équivalence entre les deux notions se démontre en considérant la fonction $1 \dot{-} \chi_A$ où A est l'ensemble considéré.

1.3 Ensembles récursivement énumérables

Nous allons introduire les ensembles récursivement énumérables d'une façon particulière. S'il est d'usage de définir ces ensembles avec les fonctions partielles (i.e., dont le domaine de définition est un sous-ensemble strict de $\mathbb{N}^p$) ou avec la théorie des machines de Turing, notre approche, ressemblant à celle de [12] et de [15], consiste à définir ces ensembles comme images de fonctions totales récursives. Cela présente l'avantage d'éviter d'introduire de nouveaux concepts.

Définition 1.3.1. Une fonction $f : \mathbb{N}^p \to \mathbb{N}^m$ sera dite *primitive récursive généralisée* (resp. *récursive généralisée*) s'il existe des fonctions primitives récursives (resp. récursives) $f_1, \ldots, f_m \in \mathfrak{F}_p$ telles que

$$f(x_1, \ldots, x_p) = (f_1(x_1, \ldots, x_p), \ldots, f_m(x_1, \ldots, x_p)) \text{ pour tout } (x_1, \ldots, x_p) \in \mathbb{N}^p$$

Par la suite, nous ne préciserons pas si une fonction est primitive récursive généralisée (resp. récursive généralisée) ou simplement primitive récursive (resp. récursive).

Définition 1.3.2. Un ensemble $A \subset \mathbb{N}^p$ est *récursivement énumérable* s'il est vide ou s'il existe une fonction récursive f telle que $A = \mathrm{Im}(f)$ [2]. Dans ce cas, nous dirons que f *énumère* l'ensemble A.

Proposition 1.3.3. *Tout ensemble récursif est récursivement énumérable.*

Démonstration. Soit un ensemble récursif A non vide.
Soient les fonctions f_i définies de la manière suivante :

$$f_i(n_1, \ldots, n_p) = \begin{cases} n_i \text{ si } \chi_A(n_1, \ldots, n_p) = 1 \\ \beta_p^i(\mu_t((\beta_p^1(t), \ldots, \beta_p^p(t)) \in A)) \text{ sinon} \end{cases}$$

Posons $f(n_1, \ldots, n_p) = (f_1(n_1, \ldots, n_p) \ldots, f_p(n_1, \ldots, n_p))$.
Par le schéma de définition par cas, nous avons bien que les fonctions f_i sont récursives pour tout $i \leq p$ et vu la définition de f, nous avons $\mathrm{Im}(f) = A$. ■

Proposition 1.3.4. *L'union et l'intersection de sous-ensembles récursivement énumérables sont récursivement énumérables.*

Démonstration. Soient $A, B \subset \mathbb{N}^p$ deux ensembles récursivement énumérables.
Soient $f = (f_1, \ldots, f_p)$ (resp. $g = (g_1, \ldots, g_p)$) la fonction récursive énumérant l'ensemble

2. Notons que dans le cas où $p > 1$, f est une fonction récursive généralisée et Im(f) =Im $(f_1) \times \ldots$ Im (f_p) où $f_1, \ldots, f_p \in \mathfrak{F}$ sont des fonctions récursives.

A (resp. B) où $f_1, \ldots, f_p, g_1, \ldots, g_p \in \mathfrak{F}$ et A_i (resp. B_i) les ensembles énumérés par f_i (resp. g_i).
Supposons $A \cap B$ non vide. Nous voulons montrer que cet ensemble est récursivement énumérable, c'est-à-dire qu'il existe une fonction h énumérant $A \cap B$.
Nous avons que $x \in A \cap B$ si et seulement si il existe m, n tels que $x = f(m) = g(n)$.
La fonction h définie comme suit va donc énumérer tous les éléments étant l'image de la fonction f et de la fonction g.
Posons

$$h_i(n_1, \ldots, n_p) = \begin{cases} f_i(\beta_2^1(n_i)) \text{ si } f_i(\beta_2^1(n_i)) = g_i(\beta_2^2(n_i)) \\ f_i(\beta_2^1(\mu_t(f_i(\beta_2^1(t)) = g_i(\beta_2^2(t))))) \text{ sinon} \end{cases}$$

On définit la fonction h comme étant $h(n_1, \ldots, n_p) = (h_1(n_1, \ldots, n_p), \ldots, h_p(n_1, \ldots, n_p))$.
Vu que f,g,β_2^1 et β_2^2 sont des fonctions récursives, h est une fonction récursive.
Il est évident que $Imh_i \subset A_i \cap B_i$ vu que tout élément de Imh_i satisfait une égalité entre f_i et g_i.
Montrons que $A_i \cap B_i \subset Imh_i$.
Soit $x \in A_i \cap B_i$. Donc, il existe m, n tels que $x = f_i(m) = g_i(n)$. Soit $r = \alpha_2(m, n)$. On a bien que $h_i(r) = x$.
Pour démontrer que l'union est récursivement énumérable, posons

$$H_i(n_1, \ldots, n_p) = \begin{cases} f_{(\frac{n_i}{2})} \text{ si } n_i \text{ pair} \\ g(\frac{n_i - 1}{2}) \text{ sinon} \end{cases}$$

et $H(n_1, \ldots, n_p) = (H_1(n_1, \ldots, n_p), \ldots, H_p(n_1, \ldots, n_p))$.
Ces fonctions se réécrivent sous la forme

$$H_i(n_1, \ldots, n_p) = \begin{cases} f_i(q(n_i, 2)) \text{ si } n_i \text{ pair} \\ g_i(q(n_i, 2)) \text{ sinon} \end{cases}$$

où $q(x, y)$ est la fonction qui est égale à la partie entière de $\frac{x}{y}$ si y est non nul et à 0 si y est nul.
Le schéma de définition par cas montre que H est récursive car la fonction $q(x, y)$ ainsi que le prédicat "n est pair" sont primitifs récursifs.
Vu la définition de H_i, nous avons que $\mathrm{Im}(H_i) \subset A_i \cup B_i$
Montrons que $A_i \cup B_i \subset Im(H_i)$. Soit $x \in A_i$, il existe un entier m tel que $x = f_i(m)$. Nous avons donc que $H_i(2m) = f_i(m) = x$. La conclusion en découle en remarquant que le raisonnement est analogue pour le cas $x \in B_i$. ■

Proposition 1.3.5. *Soit un ensemble $A \subset \mathbb{N}^p$. A est récursif si et seulement si A et $\mathbb{N}^p \setminus A$*

sont récursivement énumérables.

Démonstration. Supposons que A soit un ensemble récursif.
Nous savons donc que le complémentaire est également récursif et que A et son complémentaire sont récursivement énumérables.
A présent, supposons que A et $\mathbb{N}^p - A$ sont récursivement énumérables.
Soient f et g les fonctions énumérant respectivement les ensembles A et $\mathbb{N}^p - A$.
Montrons que la fonction caractéristique de A est récursive.
Posons $H_i(n_1, \ldots, n_p) = \mu_t(f_i(t) = n_i \text{ ou } g_i(t) = n_i)$.
La fonction H_i est récursive vu que f_i et g_i le sont.
Nous avons donc que

$$\chi_A(n_1, \ldots, n_p) = \begin{cases} 1 \text{ si } (n_1, \ldots, n_p) = (f_1(H_1(n_1, \ldots, n_p)), \ldots, f_p(H_p(n_1, \ldots, n_p))) \\ 0 \text{ sinon} \end{cases}$$

est récursive. ■

Proposition 1.3.6. *La projection d'un ensemble récursif est récursivement énumérable.*

Démonstration. Soit $B \subset \mathbb{N}^{p+1}$ un ensemble récursif non vide.
Posons $\alpha_{p+1}(B) = \{\alpha_{p+1}(x_1, x_2, \ldots, x_{p+1}) : (x_1, x_2, \ldots, x_{p+1}) \in B\}$.
Nous avons que $\alpha_{p+1}(B)$ est primitif récursif vu que

$$\chi_{\alpha_{p+1}(B)}(n) = \chi_B(\beta^1_{p+1}(n), \beta^2_{p+1}(n), \ldots, \beta^{p+1}_{p+1}(n))$$

est primitif récursif car composée de fonctions primitives récursives.
Soit A la projection de B suivante :

$$A = \{(x_1, \ldots, x_p) : \exists x_{p+1} : (x_1, x_2, \ldots, x_{p+1}) \in B\} = \{(\beta^1_{p+1}(z), \ldots, \beta^p_{p+1}(z)) : z \in \alpha_{p+1}(B)\}.$$

Vu que B est non vide, $\alpha_{p+1}(B)$ l'est également. Par conséquent, $\mu_t(t \in \alpha_{p+1}(B))$ existe bien.
Posons

$$f_i(n) = \begin{cases} \beta^i_{p+1}(n) \text{ si } n \in \alpha_{p+1}(B) \\ \beta^i_{p+1}(\mu_t(t \in \alpha_{p+1}(B))) \text{ sinon} \end{cases}$$

et $f(n) = (f_1(n), \ldots, f_p(n))$.
f est donc bien récursive. Nous avons $\text{Im}(f) \subset A$ et le caractère bijectif de la fonction β assure que $\text{Im}(f) = A$. ■

2
Les théorèmes d'incomplétude de Gödel

En 1900, lors du deuxième congrès international des mathématiciens, David Hilbert présenta une liste de vingt-trois problèmes qui devaient, selon lui, être d'une importance capitale dans le développement des mathématiques du XXe siècle et dont le deuxième problème avait pour but la démonstration de la cohérence de l'arithmétique.

Dans ce chapitre, nous allons démontrer les deux théorèmes d'incomplétude de Gödel qui furent un élément de réponse de la deuxième des vingt-trois questions de Hilbert.
L'idée de la démonstration du premier théorème de Gödel est de créer une formule auto-référente exprimant sa propre indémontrabilité. Afin de pouvoir obtenir cette auto-référence dans une formule logique, Gödel eut l'idée d'associer à chaque élément d'une théorie donnée un code numérique. Tout codage des éléments d'une théorie sera par la suite appelée *gödelisation*.

Le deuxième théorème, quant à lui, nécessite d'internaliser les théorèmes principaux de logique mathématique dans l'arithmétique grâce aux codages définis dans le cadre du premier théorème d'incomplétude.
Pour ce faire, nous allons devoir établir préalablement un grand nombre de propriétés portant sur l'arithmétique de Peano, les fonctions représentables et les fonctions récursives avant même de définir les codages

2.1 Les axiomes de Peano

La majorité des principales théories mathématiques sont des théories du premier ordre. Dans cette section, nous allons introduire l'une d'entre elles : l'arithmétique de Peano et nous démontrerons des résultats sur les modèles de cette théorie.

Définition 2.1.1. Le langage $\mathcal{L}_0$ est un langage qui contient les quatre symboles suivants :

- un symbole constant $\underline{0}$
- un symbole de fonction unaire $\underline{S}$
- deux symboles de fonctions binaires $\underline{+}$ et $\underline{\times}$
- un symbole relationnel unaire $\simeq$

Dorénavant, $\mathbb{N}$ sera vue comme une $\mathcal{L}_0$-structure dont l'ensemble de base est l'ensemble des entiers naturels et où $\underline{0}$ est interprété par 0, $\underline{S}$ par la fonction successeur $f(n) = n+1$, $\underline{+}$ par l'addition, $\underline{\times}$ par la multiplication et $\simeq$ par l'égalité.

Définition 2.1.2. Les axiomes de Peano, dont l'ensemble sera noté $\mathcal{P}$, sont les sept axiomes ci-dessous ainsi qu'une infinité d'axiomes : le *schéma d'induction* que l'on note *SI*.

- $A_1 : \forall v_0\ \neg(\underline{S}v_0 \simeq \underline{0})$
- $A_2 : \forall v_0\ \exists v_1(\neg v_0 \simeq \underline{0} \Rightarrow \underline{S}v_1 \simeq v_0)$
- $A_3 : \forall v_0\ \forall v_1(\underline{S}v_0 \simeq \underline{S}v_1 \Rightarrow v_0 \simeq v_1)$
- $A_4 : \forall v_0\ v_0 \underline{+} \underline{0} \simeq v_0$
- $A_5 : \forall v_0\ \forall v_1\ v_0 \underline{+} \underline{S}v_1 \simeq \underline{S}(v_0 \underline{+} v_1)$
- $A_6 : \forall v_0\ v_0 \underline{\times} \underline{0} \simeq \underline{0}$
- $A_7 : \forall v_0\ \forall v_1\ v_0 \underline{\times} \underline{S}v_1 \simeq (v_0 \underline{\times} v_1) \underline{+} v_0$

Le schéma d'induction est l'ensemble de toutes les formules de $\mathcal{L}_0$ de la forme

$$\forall v_1 \ldots \forall v_n((F[\underline{0}, v_1, \ldots, v_n] \wedge \forall v_0(F[v_0, \ldots, v_n] \Rightarrow F[\underline{S}v_0, v_1, \ldots, v_n])) \Rightarrow \forall v_0 F[v_0, \ldots, v_n])$$

où n est un entier et $F[v_0, \ldots, v_n]$ est une formule quelconque de $\mathcal{L}_0$ ne comportant aucune autre variable libre que $v_0, \ldots, v_n$.

Définition 2.1.3. Pour tout entier n, le terme $\underline{n}$ est défini comme étant le terme composé de n occurences du symbole $\underline{S}$ suivies du symbole $\underline{0}$, i.e. $\underline{n} = \underline{S} \ldots \underline{S}\ \underline{0}$.
Dans une $\mathcal{L}_0$-structure, nous dirons qu'un élément est *standard* s'il est l'interprétation d'un terme de la forme $\underline{n}$ avec $n \in \mathbb{N}$

Notations. Nous utiliserons la notation $v_0 \leq v_1$ comme abréviation de la formule

$$\exists v_2(v_2 \underline{+} v_0 \simeq v_1)$$

et $v_0 < v_1$ comme abréviation de la formule $(v_0 \leq v_1 \wedge \neg(v_0 \simeq v_1))$.
$\mathcal{P}_0$ sera la notation désignant la théorie constituée des axiomes A_1 jusque A_7 et ne contiendra donc pas le schéma d'induction.

Définition 2.1.4. Soient $\mathfrak{M}$ et $\mathfrak{N}$ deux modèles de $\mathcal{P}_0$. Supposons que $\mathfrak{N}$ soit une sous-structure de $\mathfrak{M}$.
Nous dirons que $\mathfrak{N}$ est un *segment initial* de $\mathfrak{M}$ (ou que $\mathfrak{M}$ est une *extension finale* de $\mathfrak{N}$) si pour tous points a appartenant à $\mathfrak{N}$ et b appartenant à $\mathfrak{M}$,
1) Si $\mathfrak{M} \vDash b \leq a$ alors b appartient à $\mathfrak{N}$
2) Si $b \notin \mathfrak{N}$, alors $\mathfrak{M} \vDash a \leq b$

Théorème 2.1.5. *Soit $\mathfrak{M}$ un modèle de $\mathcal{P}_0$; alors le sous-ensemble de $\mathfrak{M}$ suivant :*

$$\mathfrak{N} := \{a \text{ ; } \textit{il existe un entier } n \textit{ tel que } a \textit{ soit l'interprétation de } \underline{n} \textit{ dans } \mathfrak{M}\}$$

est une sous-structure de $\mathfrak{M}$ qui en est un segment initial et qui est isomorphe à $\mathbb{N}$

Nous avons besoin de résultats intermédiaires.

Lemme 2.1.6. $\mathcal{P}_0 \vdash \forall v_0 \forall v_1(\neg(v_1 \simeq \underline{0}) \Rightarrow \neg(v_0 \underline{+} v_1 \simeq \underline{0}))$

Démonstration. Par A_2, nous avons,

$$\exists v_2(\neg(v_1 \simeq \underline{0}) \Rightarrow (v_1 \simeq \underline{S}v_2)).$$

Par A_5, nous avons ensuite $v_0 \underline{+} v_1 \simeq v_0 \underline{+} \underline{S}v_2 \simeq \underline{S}(v_0 \underline{+} v_2)$.
L'axiome A_1 nous donne $\neg \underline{S}(v_0 \underline{+} v_2) \simeq \underline{0}$.
Vu que $(v_1 \simeq \underline{S}v_2) \Rightarrow v_0 \underline{+} v_1 \simeq \underline{S}(v_0 \underline{+} v_2)$, nous avons

$$(v_1 \simeq \underline{S}v_2) \Rightarrow (v_0 \underline{+} v_1 \simeq 0 \Rightarrow \underline{S}(v_0 \underline{+} v_2) \simeq 0).$$

Et donc,

$$\neg(v_1 \simeq \underline{0}) \Rightarrow (v_0 \underline{+} v_1 \simeq 0 \Rightarrow \underline{S}(v_0 \underline{+} v_2) \simeq 0).$$

Par contraposée de $(v_0 \underline{+} v_1 \simeq 0 \Rightarrow \underline{S}(v_0 \underline{+} v_2) \simeq 0)$, on a

$$\neg(v_1 \simeq \underline{0}) \Rightarrow (\neg(\underline{S}(v_0 \underline{+} v_2) \simeq 0) \Rightarrow \neg(v_0 \underline{+} v_1 \simeq 0)),$$

d'où la conclusion. ■

Lemme 2.1.7. $\mathcal{P}_0 \vdash \forall v_0 \forall v_1 (v_0 \underline{+} v_1 \simeq \underline{0} \Rightarrow (v_0 \simeq \underline{0} \wedge v_1 \simeq \underline{0}))$

Démonstration. La contraposée du lemme précédent nous donne

$$\mathcal{P}_0 \vdash \forall v_0 \forall v_1 (v_0 \underline{+} v_1 \simeq \underline{0} \Rightarrow v_1 \simeq \underline{0})$$

Vu A_4, nous avons que $v_0 \underline{+} v_1 \simeq \underline{0}$ et $v_0 \underline{+} v_1 \simeq v_0$ impliquent que $v_0 \simeq 0$. ■

Proposition 2.1.8. *Pour tout entier n, nous avons*
$\mathcal{P}_0 \vdash \forall v_0 (v_0 \leq \underline{n} \Rightarrow (v_0 \simeq \underline{0} \vee v_0 \simeq \underline{1} \vee \cdots \vee v_0 \simeq \underline{n}))$

Démonstration. Par récurrence sur n [1] :

- Si $n = 0$, $v_0 \leq \underline{0}$ est l'abréviation de $\exists v_1 (v_0 \underline{+} v_1 \simeq \underline{0})$
  Il suffit donc de montrer, de manière plus générale, que
  $\mathcal{P}_0 \vdash \forall v_0 \forall v_1 (v_1 \underline{+} v_0 \simeq 0 \Rightarrow v_0 \simeq \underline{0})$.
  Ce qui est le cas vu le lemme précédent.
- Supposons la propriété vérifiée pour n et démontrons-la pour $n+1$.
  Soit $\mathfrak{M}$ un modèle de $\mathcal{P}_0$ et a un point de $\mathfrak{M}$ tel que $\mathfrak{M} \vDash a \leq \underline{n+1}$.
  Il suffit de montrer qu'il existe $p \in \mathbb{N}$ tel que $p \leq n+1$ et $\mathfrak{M} \vDash a \simeq \underline{p}$.
  Par définition de $\leq$, il existe un point b de $\mathfrak{M}$ tel que $\mathfrak{M} \vDash b \underline{+} a \simeq \underline{S}\ \underline{n}$
  Si $a = \underline{0}$, $p = 0$ convient.
  Sinon, par A_2, il existe un point c de $\mathfrak{M}$ tel que $\mathfrak{M} \vDash a \simeq \underline{S}c$.
  Par A_3 et A_5, nous avons que $\mathfrak{M} \vDash b \underline{+} c \simeq \underline{n}$ donc $\mathfrak{M} \vDash c \leq \underline{n}$.
  En utilisant l'hypothèse de récurrence, nous voyons qu'il existe $m \leq n$ tel que $\mathfrak{M} \vDash c \simeq \underline{m}$ d'où $\mathfrak{M} \vDash \underline{S}c \simeq \underline{S}\ \underline{m}$ c'est-à-dire $\mathfrak{M} \vDash a \simeq \underline{m+1}$.
  Vu que $p \leq n+1$, $\mathfrak{M} \vDash a \simeq \underline{0}$ ou $\mathfrak{M} \vDash a \simeq \underline{1}$ ou ... ou $\mathfrak{M} \vDash a \simeq \underline{n+1}$.
  La conclusion découle du théorème de complétude.

■

Proposition 2.1.9. *Pour tout entier n,*

$$\mathcal{P}_0 \vdash \forall v_0 (v_0 \leq \underline{n} \vee \underline{n} \leq v_0)$$

Démonstration. Par récurrence sur n :

- Si $n = 0$:
  $\underline{0} \leq v_0$ est l'abréviation de $\exists v_1 (\underline{0} \underline{+} v_1 \simeq v_0)$.
  C'est immédiat vu A_4.

1. Rappelons que $\mathcal{P}_0$ ne contient pas *SI*. La récurrence que nous utilisons ici est une *méta-récurrence*.

– Supposons vrai pour n et montrons cela pour $n+1$.
Soit un modèle $\mathfrak{M}$ de $\mathcal{P}_0$ et un point $a \in \mathfrak{M}$.
Montrons que $\mathfrak{M} \vDash a \leq \underline{n+1}$ ou $\mathfrak{M} \vDash \underline{n+1} \leq a$.
Si $a = 0$, c'est évident.
Sinon, il existe b appartenant à $\mathfrak{M}$ tel que $\mathfrak{M} \vDash a \simeq \underline{S}b$.
Par hypothèse de récurrence, nous avons $\mathfrak{M} \vDash b \leq \underline{n}$ ou $\mathfrak{M} \vDash \underline{n} \leq b$.
Dans le premier cas, il existe c appartenant à $\mathfrak{M}$ tel que $\mathfrak{M} \vDash c + b \simeq n$ donc par A_5, $\mathfrak{M} \vDash c\underline{+}a \simeq \underline{n+1}$, d'où $\mathfrak{M} \vDash a \leq n+1$
Dans le deuxième cas, il existe d appartenant à $\mathfrak{M}$ tel que $\mathfrak{M} \vDash d + n \simeq b$, donc $\mathfrak{M} \vDash d + n + 1 \simeq a$, d'où $\mathfrak{M} \vDash n + 1 \leq a$

■

Nous pouvons maintenant démontrer le théorème 2.1.5

Démonstration. Soit $\varphi : \mathbb{N} \to \mathfrak{N}$ qui à $n \in \mathbb{N}$ fait correspondre l'interprétation de $\underline{n}$ dans $\mathfrak{M}$.
Montrons que φ est un homomorphisme.
Pour cela, montrons que pour tous entiers m et n,

$$\mathcal{P}_0 \vdash \underline{m}\underline{+}\underline{n} \simeq \underline{m+n}$$

et

$$\mathcal{P}_0 \vdash \underline{m}\underline{\times}\underline{n} \simeq \underline{m.n}$$

On procède par récurrence sur n :

- Pour $n = 0$, nous avons
  $\mathcal{P}_0 \vdash \underline{m}\underline{+}\underline{0} \simeq \underline{m}$ (par A_4)
- Supposons le résultat vrai pour n et démontrons-le pour $n+1$. Nous avons [2]
  $\mathcal{P}_0 \vdash \underline{m+n+1} \simeq \underline{S}(\underline{m+n})$ et
  $\mathcal{P}_0 \vdash \underline{m}\underline{+}\underline{Sn} \simeq \underline{S}(\underline{m}\underline{+}\underline{n})$ (par A_5)
  en utilisant l'hypothèse de récurrence, nous avons
  $\mathcal{P}_0 \vdash \underline{m}\underline{+}\underline{n} \simeq \underline{m+n}$ et donc $\mathcal{P}_0 \vdash \underline{S}(\underline{m}\underline{+}\underline{n}) \simeq \underline{S}(\underline{m+n})$
  Par conséquent
  $\mathcal{P}_0 \vdash \underline{m}\underline{+}\ \underline{n+1} \simeq \underline{m+n+1}$.

La démonstration est analogue pour la multiplication : nous utilisons A_6 pour démontrer le cas de base et A_7 ainsi que le résultat obtenu pour l'addition pour démontrer le cas

2. Remarquons que pour tout entier n, on a $\mathcal{P}_0 \vdash \underline{n+1} \simeq \underline{S}\ \underline{n}$ car $\underline{n+1}$ et $\underline{S}\ \underline{n}$ représentent le même terme

d'induction.
Montrons que φ est injectif.

- Pour tout entier n non nul, nous avons $\mathcal{P}_0 \vdash \neg(\underline{n} \simeq \underline{0})$.
  En effet, soit $m = n - 1$. Nous avons $\mathcal{P}_0 \vdash \underline{n} \simeq \underline{S}\ \underline{m}$.
  La conclusion découle de A_1.
- Pour tous entiers m et n distincts, nous avons $\mathcal{P}_0 \vdash \neg(\underline{m} \simeq \underline{n})$.
  Montrons cela par récurrence sur $\inf(m, n)$.
  Si $\inf(m, n) = 0$, nous sommes ramenés au cas précédent.
  Sinon, $\mathcal{P}_0 \vdash \underline{m} \simeq \underline{n} \Rightarrow \underline{S}\ \underline{m-1} \simeq \underline{S}\ \underline{n-1}$.
  En utilisant A_3, nous obtenons
  $\mathcal{P}_0 \vdash \underline{m} \simeq \underline{n} \Rightarrow \underline{m-1} \simeq \underline{n-1}$
  Par contraposition, $\mathcal{P}_0 \vdash \neg(\underline{m-1} \simeq \underline{n-1}) \Rightarrow \neg(\underline{m} \simeq \underline{n})$
  Et, en utilisant l'hypothèse de récurrence $\mathcal{P}_0 \vdash \neg(\underline{m-1} \simeq \underline{n-1})$, la conclusion est immédiate.

L'isomorphisme est évident.
Montrons que l'image de φ est un segment initial de $\mathfrak{M}$.
Soit un point a appartenant à $\mathfrak{N}$ et b appartenant à $\mathfrak{M}$.
1) Si $\mathfrak{M} \vDash b \leq a$ alors b appartient à $\mathfrak{N}$, vu 2.1.8.
2) Si $b \notin \mathfrak{N}$, alors, vu 2.1.9, soit $b \leq a$ et donc b appartient à $\mathfrak{M}$ ce qui aboutit à une contradiction, soit $a \leq b$ ce qui entraîne que $\mathfrak{M} \vDash a \leq b$.
Et $\mathfrak{N}$ est bien un segment initial de $\mathfrak{M}$. ■

Une conséquence immédiate de cette proposition est qu'un entier non standard est nécessairement supérieur à un entier standard. En effet, soit b un élément non standard d'un modèle non standard $\mathfrak{M}$ de $\mathbb{N}$ et soit a un élément standard.
Vu que l'ensemble $\mathfrak{N} := \{a\,;$ il existe un entier n tel que a soit l'interprétation de $\underline{n}$ dans $\mathfrak{M}\}$ est un segment initial de $\mathfrak{M}$, on a que si $\mathfrak{M} \vDash b \leq a$ alors $b \in \mathfrak{N}$.

2.2 Fonctions représentables

Dans cette section, nous allons définir la notion de fonction représentable par une formule logique du premier ordre et démontrer que toute fonction récursive est représentable.

Définition 2.2.1. Soient $f \in \mathfrak{F}_p$ et $F[v_0, v_1, \ldots, v_p]$ une formule de $\mathcal{L}_0$ qui n'a pas de variables libres en dehors de $v_0, \ldots, v_p$.
Nous dirons que $F[v_0, v_1, \ldots, v_p]$ *représente* f si, pour tout p-uple d'entiers $(n_1, n_2, \ldots, n_p)$,

$$\mathcal{P}_0 \vdash \forall v_0(F[v_0, \underline{n_1}, \ldots, \underline{n_p}] \Leftrightarrow v_0 \simeq \underline{f(n_1, \ldots, n_p)})$$

La fonction f est dite *représentable* s'il existe une formule qui la représente.

Nous avons donc que le fait qu'une formule F représente f exprime que pour tout modèle $\mathfrak{M}$ de $\mathcal{P}_0$ et pour toute suite d'entiers $(n_1, \ldots, n_p)$, il existe un unique élément standard x de $\mathfrak{M}$ satisfaisant $F[v_0, \underline{n_1}, \ldots, \underline{n_p}]$ et qui est l'interprétation du terme $\underline{f(n_1, \ldots, n_p)}$.

Définition 2.2.2. Soient $A \subset \mathbb{N}^p$ et $F[v_1, \ldots, v_p]$ une formule qui n'a pas de variable libre en dehors de $v_1, \ldots, v_p$.
Nous dirons que F *représente* A si pour tout p-uple d'entiers $(n_1, \ldots n_p)$,

- $\mathcal{P}_0 \vdash F[\underline{n_1}, \ldots, \underline{n_p}]$ si $(n_1, \ldots, n_p) \in A$
- $\mathcal{P}_0 \vdash \neg F[\underline{n_1}, \ldots, \underline{n_p}]$ sinon

L'ensemble A est dit *représentable* s'il existe une formule qui le représente.

Proposition 2.2.3. *Un ensemble est représentable si et seulement si sa fonction caractéristique l'est.*

Démonstration. Soit un ensemble représentable $A \subset \mathbb{N}^p$.
Si $F[v_1, \ldots, v_p]$ représente A, alors la formule

$$F'[v_0, \ldots, v_p] \equiv (F[v_1, \ldots, v_p] \wedge v_0 \simeq 1) \vee (\neg F[v_1, \ldots, v_p] \wedge v_0 \simeq 0)$$

représente la fonction caractéristique de A.
A présent, supposons que la fonction caractéristique de A est représentable.
Si $G[v_0, \ldots, v_p]$ représente la fonction caractéristique de A, alors la formule $G'[v_1, \ldots, v_p] \equiv G[\underline{1}, v_1, \ldots, v_p]$ représente A. ■

Théorème 2.2.4 (Premier théorème de représentation)**.** *Toute fonction récursive est représentable.*

Pour obtenir ce résultat, nous avons besoin de quatre lemmes intermédiaires.

Lemme 2.2.5. *L'ensemble des fonctions représentables est clos par composition*

Démonstration. Soient $f_1, f_2, \ldots, f_m \in \mathfrak{F}_p$ et $g \in \mathfrak{F}_m$ et supposons que, pour tout i tel que $1 \leq i \leq m$, f_i est représenté par $F_i[v_0, \ldots, v_p]$ et que g est représenté par $G[v_0, \ldots, v_m]$. Nous avons donc pour tout m-uple $(n_1, \ldots, n_m)$,

$$\mathcal{P}_0 \vdash \forall v_0(G[v_0, \underline{n_1}, \ldots, \underline{n_m}] \Leftrightarrow v_0 \simeq \underline{g(n_1, \ldots, n_m)})$$

et pour tout p-uple $(n_1, \ldots, n_p)$

$$\mathcal{P}_0 \vdash \forall v_0(F_i[v_0, \underline{n_1}, \ldots, \underline{n_p}] \Leftrightarrow v_0 \simeq \underline{f_i(n_1, \ldots, n_p)}).$$

Nous avons donc que si, pour tout i, $\exists w_i$ tel que $(F_i[w_i, \underline{n_1}, \ldots, \underline{n_p}] \Leftrightarrow w_i \simeq \underline{f_i(n_1, \ldots, n_p)})$, alors

$$\mathcal{P}_0 \vdash \forall v_0(G[v_0, w_1, \ldots, w_m] \Leftrightarrow v_0 \simeq \underline{g(f_1(n_1, \ldots, n_p)), \ldots, f_m(n_1, \ldots, n_p)})$$

Nous avons donc que $g(f_1, \ldots, f_m)$ est représenté par

$$\exists w_1 \ldots \exists w_m(G[v_0, w_1, \ldots, w_m] \wedge \bigwedge_{1 \leq i \leq p} F_i[w_i, v_1, \ldots, v_p])$$

■

Lemme 2.2.6. *Soit $g : \mathbb{N}^{p+1} \to \mathbb{N}$ une fonction représentable telle que la fonction*

$$f(x_1, \ldots, x_p) = \mu_y(g(x_1, \ldots, x_p, y) = 0)$$

soit totale alors f est représentable.
En d'autres termes, le schéma μ de minimisation non bornée est représentable.

Démonstration. Soit $G[v_0, \ldots, v_{p+1}]$ une formule représentant g, alors la formule

$$F[v_0, v_1, \ldots, v_p] := G[\underline{0}, v_1, \ldots, v_p, v_0] \wedge \forall(w < v_0)\ \neg G[\underline{0}, v_1, \ldots, v_p, w])$$

représente f.
En effet, soient $\mathfrak{M}$ un modèle de $\mathcal{P}_0$, $n_1, \ldots, n_p$ des entiers et b l'interprétation de l'entier $\underline{f(n_1, \ldots, n_p)}$ dans $\mathfrak{M}$.
Montrons que b est le seul élément de $\mathfrak{M}$ qui satisfait la formule $F[v_0, \underline{n_1}, \ldots, \underline{n_p}]$.
Premièrement, puisque G représente g, nous avons $\mathcal{P}_0 \vdash G[0, \underline{n_1}, \ldots, \underline{n_p}, b]$ et puisque $\mathfrak{M}$

est un modèle de $\mathcal{P}_0$, b satisfait $G[0, \underline{n_1}, \ldots, \underline{n_p}, v_0]$ dans $\mathfrak{M}$.
De plus, si c est un élément de $\mathfrak{M}$ qui est inférieur à b alors, vu le théorème 2.1.5, c est l'interprétation d'un entier et par définition de f, il ne satisfait pas $G[0, \underline{n_1}, \ldots, \underline{n_p}, v_0]$.
Donc b satisfait $F[v_0, \underline{n_1}, \ldots, \underline{n_p}]$. Montrons qu'il est unique.
Soit d un élément de $\mathfrak{M}$ qui satisfait $F[v_0, \underline{n_1}, \ldots, \underline{n_p}]$.
Vu la restriction imposée par le quantificateur universel, nous ne pouvons alors avoir dans $\mathfrak{M}$ ni $d < b$ ni $b < d$. Or, vu que b est l'interprétation d'un entier et vu la proposition 2.1.9, nous avons $b \leq d$ ou $d \leq b$.
Par conséquent, $b = d$ et l'unicité est démontrée. ■

Lemme 2.2.7. *Il existe une fonction β à trois variables, qui est primitive récursive et représentable, telle que, pour tout $p \in \mathbb{N}$ et toute suite $(\alpha_1, \ldots, \alpha_p) \in \mathbb{N}^p$, il existe des entiers a et b tels que, pour tout i compris entre 1 et p, on ait $\beta(i, a, b) = \alpha_i$.*

Démonstration. Définissons $\beta(i, a, b)$ comme étant le reste de la division euclidienne de b par $a(i+1)+1$.
C'est-à-dire que pour tout entier b, a, i, il existe un entier k tel que

$$\beta(i, a, b) + k.(a(i+1)+1) = b$$

Montrons que cette fonction possède les propriétés voulues.
Nous voyons que β est représenté par la formule suivante :

$$B[v_0, v_1, v_2, v_3] = \exists(v_4 < v_3)(v_3 \simeq (v_4 \underline{\times} \underline{S}(v_2 \times \underline{S}(v_1))) + v_0) \wedge v_0 < \underline{S}(v_2 \times \underline{S}(v_1))$$

Soit un n-uple d'entiers $(\alpha_0, \ldots, \alpha_n)$ et un entier $m > n+1$ tel que $m! \geq \alpha_i$ pour tout entier i compris entre 0 et n.
Nous avons que les $m!(i+1)+1$ $\forall i \in \{0, \ldots, n\}$ sont 2 à 2 premiers entre eux.

En effet, raisonnons par l'absurde et supposons qu'il existe $i < j$ tel qu'il existe c diviseur commun de $m!(i+1)+1$ et $m!(j+1)+1$.
Dans ce cas, c divise $m!(j+1)+1-(m!(i+1)+1) = m!(j-i)$.
Vu que $(j-i) \leq m$, c divise $m!$. Par conséquent, c divise $m!(i+1)$ et ne divise donc pas $m!(i+1)+1$. Les $m!(i+1)+1$ sont donc bien 2 à 2 premiers entre eux.

Par le théorème des restes chinois [3], il existe b tel que

$$\forall i \in \{0, \ldots, n\}, b \equiv \alpha_i \mod (m!(i+1)+1).$$

Vu que $\alpha_i \leq m! < m!(i+1)+1$, en posant $a = m!$, nous avons bien que $\beta(i, a, b) = \alpha_i$.
De plus, vu que β est la fonction $\mod (\mathcal{P}_{3,3}, \sigma(\Pi_2(\mathcal{P}_{2,3}, \mathcal{P}_{1,3})))$, nous avons que la fonction β est récursive. ■

Lemme 2.2.8. *Soient $g \in \mathfrak{F}_p$ et $h \in \mathfrak{F}_{p+2}$ deux fonctions représentables ; alors la fonction f définie par récursion primitive à partir de g et h par :*
- $f(x_1, \ldots, x_p, 0) = g(x_1, \ldots, x_p)$
- $f(x_1, \ldots, x_p, x_{p+1}+1) = h(x_1, \ldots, x_{p+1}, f(x_1, \ldots, x_{p+1}))$

est représentable.

Démonstration. Soient $G[v_0, \ldots, v_p]$ une formule représentant g et $H[v_0, \ldots, v_{p+2}]$ une formule représentant h.
Soit $B[v_0, v_1, v_2, v_3]$ une formule représentant β. Nous pouvons aussi représenter β par la formule B' suivante :
$B'[v_0, v_1, v_2, v_3] := B[v_0, v_1, v_2, v_3] \wedge (\forall v_4 < v_0) \neg B[v_4, v_1, v_2, v_3]$
L'avantage de cette formule est que si x est un élément standard d'un modèle quelconque $\mathfrak{M}$ de $\mathcal{P}_0$, si a, b, c sont 3 points de $\mathfrak{M}$ tels que $\mathfrak{M} \vDash B'[x, a, b, c]$ alors il n'existe dans $\mathfrak{M}$ aucun autre point (standard ou non) satisfaisant $B'[v_0, a, b, c]$.
Montrons que la formule $F[v_0, \ldots, v_{p+1}]$ qui suit

$$\exists w_1 \exists w_2 (\exists w_0 (B'[w_0, \underline{1}, w_1, w_2] \wedge G[w_0, v_1, \ldots, v_p]) \wedge B'[v_0, v_{p+1} \underline{+} \underline{1}, w_1, w_2]$$
$$\wedge \forall (w_3 \leq v_{p+1}) \exists w_4 \exists w_5 (B'[w_4, \underline{S} w_3, w_1, w_2] \wedge B'[w_5, \underline{SS} w_3, w_1, w_2] \wedge H[w_5, v_1, \ldots, v_p, w_3, w_4]))$$

représente f.
Soient $n_1, \ldots, n_{p+1}$ des entiers, $\mathfrak{M}$ un modèle de $\mathcal{P}_0$ et c un point de $\mathfrak{M}$.
Montrons que $\mathfrak{M} \vDash c \simeq \underline{f(n_1, \ldots, n_{p+1})}$ si et seulement si $\mathfrak{M} \vDash F[c, \underline{n_1}, \ldots, \underline{n_{p+1}}]$.

Supposons que $\mathfrak{M} \vDash c \simeq \underline{f(n_1, \ldots, n_{p+1})}$.
Soient a et b les deux entiers codant la suite d'entiers
$(f(n_1, \ldots, n_p, 0), f(n_1, \ldots, n_p, 1), \ldots, f(n_1, \ldots, n_p, n_{p+1}))$.
L'existence de ces deux entiers est assurée par le lemme précédent. Nous pouvons donc poser $w_1 = \underline{a}$ et $w_2 = \underline{b}$.
Notre cas de base prouve l'existence de w_0 qui prendra la valeur $\underline{g(n_1, \ldots, n_p)}$.

3. Soient $(\alpha_0, \ldots, \alpha_n)$ et $(b_0, \ldots, b_n)$ deux suites d'éléments de $\mathbb{N}$ telles que les b_i soient 2 à 2 premiers entre eux. Alors il existe $a \in \mathbb{N}$ tel que pour tout i compris entre 0 et n on ait : $a \equiv \alpha_i \mod b_i$

w_3 sera l'interprétation d'un entier i tel que $0 \leq i < n_{p+1}$, w_4 devra prendre la valeur $\underline{f(n_1, \ldots, n_p, i)}$ et w_5 devra prendre la valeur $\underline{f(n_1, \ldots, n_p, i+1)}$.
De cette manière, nous avons $\mathfrak{M} \vDash F[c, \underline{n_1}, \ldots, \underline{n_{p+1}}]$.

Supposons que $\mathfrak{M} \vDash F[c, \underline{n_1}, \ldots, \underline{n_{p+1}}]$.
Par conséquent, il existe trois éléments a, b, d appartenant à $\mathfrak{M}$ tels que

$$\mathfrak{M} \vDash B'[d, \underline{1}, a, b] \wedge G[d, \underline{n_1}, \ldots, \underline{n_p}] \wedge B'[c, \underline{n_{p+1}} \underline{+1}, a, b]$$

et pour tout entier i tel que $0 \leq i < n_{p+1}$, il existe des éléments r_i et s_i dans $\mathfrak{M}$ tels que

$$\mathfrak{M} \vDash B'[r_i, \underline{i+1}, a, b] \wedge B'[s_i, \underline{i+2}, a, b] \wedge H[s_i, \underline{n_1}, \ldots, \underline{n_p}, \underline{i}, r_i]$$

Vu que G représente g, nous avons $\mathfrak{M} \vDash d \simeq \underline{g(n_1, \ldots, n_p)}$ et vu que B' a été choisie telle qu'il y ait au plus un point satisfaisant $B'[v_0, x, y, z]$, nous avons que $d = r_0$ et que pour tout entier i tel que $0 \leq i < n_{p+1}$, $r_{i+1} = s_i$. Nous avons donc $c = r_{n_{p+1}} = s_{n_{p+1}-1}$.
Démontrons par récurrence sur $i \leq n_{p+1}$ que $\mathfrak{M} \vDash r_i \simeq \underline{f(n_1, \ldots, n_p, i)}$.

- Cas de base $i = 0$:
  $\mathfrak{M} \vDash r_0 = d \simeq \underline{g(n_1, \ldots, n_p)} \simeq \underline{f(n_1, \ldots, n_p, 0)}$
- Supposons le résultat vérifié pour i. Vérifions le pour $i+1$.
  $\mathfrak{M} \vDash r_{i+1} = s_i \simeq \underline{h(n_1 \ldots, n_p, i, r_i)} \simeq \underline{h(n_1 \ldots, n_p, i, f(n_1, \ldots, n_p, i))} \simeq \underline{f(n_1, \ldots, n_p, i+1)}$

et par conséquent, $c \simeq \underline{f(n_1, \ldots, n_{p+1})}$. ■

Nous pouvons à présent démontrer 2.2.4.

Démonstration. Vu la définition de l'ensemble des fonctions récursives, il nous suffit de démontrer que les fonctions initiales sont représentables pour obtenir que l'ensemble des fonctions récursives est un sous-ensemble de l'ensemble des fonctions représentables et par conséquent que toute fonction récursive est représentable.

- La fonction **0** est représentée par la formule $F[v_0] := v_0 \simeq \underline{0}$.
- La fonction successeur σ est représentée par la formule $F[v_0, v_1] := v_0 \simeq \underline{S} v_1$.
- La fonction projection $\mathcal{P}_{i,p}$ est représentée par la formule

$$F[v_0, \ldots, v_p] := (v_0 \simeq v_i) \wedge (v_1 \simeq v_1) \wedge \cdots \wedge (v_{i-1} \simeq v_{i-1}) \wedge (v_{i+1} \simeq v_{i+1}) \wedge \cdots \wedge (v_p \simeq v_p).$$

■

Remarque 2.2.9. En fait, ces deux ensembles sont égaux et toute fonction représentable est récursive. Le lecteur intéressé pourra consulter [6] pour plus de détails.

Corollaire 2.2.10. *Tout ensemble récursif est représentable*

Démonstration. Soit A un ensemble récursif.
Nous avons que la fonction caractéristique de A est récursive et par conséquent représentable.
Vu la propriété 2.2.3, A est représentable. ■

Remarquons que pour toute théorie $\mathcal{P}'$ contenant $\mathcal{P}_0$, si $f \in \mathfrak{F}_p$ est représentable par la formule F et si $(n_1, \ldots, n_p)$ est une suite finie d'entiers, alors

$$\mathcal{P}' \vdash \forall v_0 F[v_0, \underline{n_1}, \ldots, \underline{n_p}] \Leftrightarrow v_0 \simeq \underline{f(n_1, \ldots, n_p)}$$

Voici une deuxième forme du théorème de représentation.

Définition 2.2.11. L'ensemble Σ est le plus petit ensemble de formules du langage $\mathcal{L}_0$ qui

- contient toutes les formules sans quantificateur
- est clos par conjonction et disjonction
- est clos par quantification existentielle
- est clos par quantification universelle bornée

Si une formule appartient à Σ, nous dirons que la formule est Σ.

Théorème 2.2.12 (Deuxième théorème de représentation)**.** *Toute fonction récursive est représentable par une formule* Σ*.*

Démonstration. La preuve est identique à celle du premier théorème de représentation, excepté pour le schéma de minimisation non borné dont voici la démonstration. ■

Démonstration. Soit g une fonction représentable par une formule Σ que nous noterons G.
Posons $G'[v_0, v_1, \ldots, v_{p+1}]$ comme étant la formule

$$\exists y G[y, v_1, \ldots, v_{p+1}] \wedge \neg(y \simeq v_0)$$

La formule G' est bien une formule Σ vu que G est une formule Σ et que $\neg(y \simeq v_0)$ est une formule sans quantificateur donc également une formule Σ.
Nous voyons donc que en posant

$$F[v_0, v_1, \ldots, v_p] := G[\underline{0}, v_1, \ldots, v_p, v_0] \wedge \forall(w < v_0)\ G'[\underline{0}, v_1, \ldots, v_p, w],$$

on a

$$\mathcal{P}_0 \vdash \forall v_0 (F[v_0, \underline{n_1}, \ldots, \underline{n_p}] \Leftrightarrow (v_0 \simeq \underline{f(n_1, \ldots, n_p)}))$$

où $f(x_1, \ldots, x_p)$ est la fonction $\mu_y(g(x_1, \ldots, x_p, y) = 0)$. ■

2.3 Arithmétisation de la syntaxe

Dans cette section, nous allons coder les termes, formules, théorèmes et démonstrations par des nombres entiers pour pouvoir y appliquer la théorie des fonctions récursives.
Nous montrerons le caractère récursif d'un certain nombre d'ensembles ; ces démonstrations utiliseront principalement le schéma de définition par cas, ainsi que les fonctions et les lemmes qui suivent. Par conséquent, nous omettrons parfois de les citer.

Proposition 2.3.1. *Soit $\mathcal{S}$ l'ensemble des suites finies d'entiers. Les éléments suivants :*

- *La fonction Ω de $\mathcal{S}$ dans $\mathbb{N}$ définie comme suit*

$$\Omega((x_0, \ldots, x_p)) = \pi(0)^{x_0+1}.\pi(1)^{x_1+1} \ldots \pi(p)^{x_p+1}$$

  où π est la fonction qui à l'entier n fait correspondre le $n+1$ème nombre premier.[4] *Si s est la suite vide, nous posons $\Omega(s) = 1$.*

- *La fonction δ de $\mathfrak{F}_2$ définie comme suit :*

$$\delta(i, x) = \mu_{z \leq x}(x \text{ n'est pas divisible par } \pi(i)^{z+2}).$$

  $\delta(i, x)+1$ est donc l'exposant de $\pi(i)$ dans la décomposition de x en facteurs premiers. Remarquons que $\delta(i, x) < x$ pour tout $x \geq 1$ et pour tout $i \in \mathbb{N}$.

- *La fonction lg de $\mathfrak{F}_1$ définie comme suit :*

$$lg(x) = \mu_{n \leq x}(x \text{ n'est pas divisible par } \pi(n)).$$

- *La fonction de concaténation de $\mathfrak{F}_2$ définie comme*

$$a * b = a. \prod_{j=0}^{lg(b)} \pi(lg(a) + j)^{\delta(j,b)+1}$$

sont primitifs récursifs.

Démonstration. En nous basant sur les résultats de la proposition 1.2.22, nous avons que δ et lg sont primitifs récursifs et nous pouvons définir π par récursion primitive :

- $\pi(0) = 2$
- $\pi(n+1) = \mu_{z \leq (\pi(n)!+1)}(z > \pi(n) \text{ et } z \in P)$

4. Rappelons que le premier nombre premier est l'entier 2.

où P est l'ensemble des nombres premiers.
La fonction de concaténation, étant une composition de fonctions récursives, est primitive récursive. ■

Proposition 2.3.2. *L'ensemble des codes des suites finies d'entiers, dénoté K, est primitif récursif.*

Démonstration. Il suffit de remarquer que

$$x \in K \text{ si et seulement si } x = \prod_{i=0}^{lg(x)-1} \pi(i)^{\delta(i,x)+1}.$$

En effet, si x n'était pas un produit de nombres premiers consécutifs, nous aurions, à la place de l'égalité, une inégalité stricte. ■

Lors de la démonstration du caractère récursif d'une fonction, nous sommes souvent amenés à utiliser la récursion primitive.
Or, comme nous pouvons le remarquer, la définition 1.2.6 de la récursion primitive nous permet de définir la valeur d'une fonction f évaluée en un entier $n+1$ uniquement à partir de la valeur de f évaluée en l'entier n prédécesseur de $n+1$.
Le lemme suivant va nous permettre d'élargir nos possibilités.

Lemme 2.3.3. *Soient p et n des entiers, $k_1, \ldots, k_n \in \mathfrak{F}_1, g \in \mathfrak{F}_p$ et $h \in \mathfrak{F}_{n+p+1}$ des fonctions primitives récursives. Supposons que pour tous $y > 0$ et i compris entre 1 et n, $k_i(y) < y$.*
Alors l'unique fonction déterminée par les conditions suivantes :

- $f(x_1, \ldots, x_p, 0) = g(x_1, \ldots, x_p)$;
- $f(x_1, \ldots, x_p, y) = h(x_1, \ldots, x_p, y, f(x_1, \ldots, x_p, k_1(y)), \ldots, f(x_1, \ldots, x_p, k_n(y)))$,

où $y > 0$, est primitive récursive.

Démonstration. Définissons la fonction φ par la récursion primitive suivante :

- $\varphi(x_1, \ldots, x_p, 0) = 2^{g(x_1,\ldots,x_p)+1}$
- $\varphi(x_1, \ldots, x_p, y+1) = H(x_1, \ldots, x_p, y, \varphi(x_1, \ldots, x_p, y)) := \varphi(x_1, \ldots, x_p, y).\Pi(y+1)^{\gamma+1}$

où

$$\gamma = h(x_1, \ldots, x_p, y+1, \delta(k_1(y+1), \varphi(x_1, \ldots, x_p, y)), \ldots, \delta(k_n(y+1), \varphi(x_1, \ldots, x_p, y))).$$

Vu que la fonction

$$H(x_1, \ldots, x_p, y, w) = w.\Pi(y+1)^{h(x_1,\ldots,x_p,y+1,\delta(k_1(y+1),w),\ldots,\delta(k_n(y+1),w))+1}$$

est le résultat de compositions de fonctions primitives récursives, elle est primitive récursive.
Par conséquent, φ est primitive récursive.
Montrons par récurrence sur y que

$$\varphi(x_1, \ldots, x_p, y) = \Omega(f(x_1, \ldots, x_p, 0), f(x_1, \ldots, x_p, 1), \ldots, f(x_1, \ldots, x_p, y))$$

- Cas de base :

$$\varphi(x_1, \ldots, x_p, 0) = 2^{g(x_1,\ldots,x_p)+1} = \Pi(0)^{f(x_1,\ldots,x_p,0)+1} = \Omega(f(x_1, \ldots, x_p, 0))$$

- Supposons l'égalité vraie pour y et montrons qu'elle est vraie pour $y+1$.
  En utilisant l'hypothèse de récurrence, constatons que [5]

$$\begin{aligned}\delta(n, \varphi(x_1, \ldots, x_p, y)) &= \delta(n, \Omega(f(x_1, \ldots, x_p, 0), f(x_1, \ldots, x_p, 1), \ldots, f(x_1, \ldots, x_p, y))) \\ &= f(x_1, \ldots, x_p, n)\end{aligned}$$

  pour $n \leq y$, et donc que

$$\begin{aligned}\gamma &= h(x_1, \ldots, x_p, y+1, \delta(k_1(y+1), \varphi(x_1, \ldots, x_p, y)), \ldots, \delta(k_n(y+1), \varphi(x_1, \ldots, x_p, y))) \\ &= h(x_1, \ldots, x_p, y+1, f(x_1, \ldots, x_p, k_1(y+1)), \ldots, f(x_1, \ldots, x_p, k_n(y+1)))\end{aligned}$$

  Et vu les hypothèses,

$$\gamma = f(x_1, \ldots, x_p, y+1)$$

  Nous avons donc

$$\begin{aligned}\varphi(x_1, \ldots, x_p, y+1) &= \varphi(x_1, \ldots, x_p, y).\Pi(y+1)^{\gamma+1} \\ &= \Omega(f(x_1, \ldots, x_p, 0), f(x_1, \ldots, x_p, 1), \ldots, f(x_1, \ldots, x_p, y)).\Pi(y+1)^{\gamma+1} \\ &= \Pi(0)^{f(x_1,\ldots,x_p,0)} \ldots \Pi(y)^{f(x_1,\ldots,x_p,y)}.\Pi(y+1)^{f(x_1,\ldots,x_p,y+1)+1}. \\ &= \Omega(f(x_1, \ldots, x_p, 0), \ldots, f(x_1, \ldots, x_p, y), f(x_1, \ldots, x_p, y+1))\end{aligned}$$

De plus, la fonction f est primitive récursive car

$$f(x_1, \ldots, x_p, y) = \delta(y, \varphi(x_1, \ldots, x_p, y))$$

5. Nous rappelons que, par définition de δ et de Ω,

$$\delta(i, \Omega(x_0, \ldots, x_p)) = \delta(i, \pi(0)^{x_0}, \ldots, \pi(p)^{x_p}) = x_i$$

pour tout $i \leq p$.

■

Nous avons donc démontré que nous pouvons définir, de manière récursive, une valeur $f(x)$ en fonction de n valeurs précédentes.
Cependant, il est parfois nécessaire d'exploiter un nombre variable de valeurs précédentes, voire toutes ces valeurs. Le lemme suivant va répondre à nos attentes.

Définition 2.3.4. Soit une fonction totale f, on définit la fonction f^* de la façon suivante :

$$f^*(n_1, \ldots, n_k, 0) = 1 \text{ et } f^*(n_1, \ldots, n_k, n+1) = f^*(n_1, \ldots, n_k, n) * 2^{f(n_1,\ldots,n_k,n)+1}.$$

On a que

$$f^*(n_1, \ldots, n_k, n) = \Omega(f(n_1, \ldots, n_k, 0), \ldots, f(n_1, \ldots, n_k, n-1)).$$

Lemme 2.3.5. *Si $h \in \mathfrak{F}_{k+2}$ est une fonction primitive récursive, alors la fonction f définie de la façon suivante*

$$f(n_1, \ldots, n_k, n) = h(f^*(n_1, \ldots, n_k, n), n_1, \ldots, n_k, n)$$

est primitive récursive.

Démonstration. On a que la fonction f^* est primitive récursive vu qu'elle peut-être définie de la façon suivante :

$$f^*(n_1, \ldots, n_k, 0) = 1 \text{ et } f^*(n_1, \ldots, n_k, n+1) = f^*(n_1, \ldots, n_k, n) * 2^{h(f^*(n_1,\ldots,n_k,n),n_1,\ldots,n_k,n)+1}$$

et, par conséquent, f est primitive récursive. En effet, il suffit de remarquer qu'elle peut être définie comme suit :

$$f(n_1, \ldots, n_k, n) = \delta(n, f^*(n_1, \ldots, n_k, n))$$

qui est une composition de fonctions primitives récursives. ■

Corollaire 2.3.6. *Soit une fonction primitive récursive $k \in \mathfrak{F}_2$ telle que $\forall (i,x) \in \mathbb{N}^2$, $k(i,x) < x$. Alors la fonction f définie comme*

$$f(x) = \prod_{i=1}^{k(0,x)} f(k(i,x))$$

est primitive récursive.

Démonstration. Soit la fonction primitive récursive δ' définie de la façon suivante :

$$\begin{cases} \delta'(i,x) = 1 \text{ si } i \geq lg(x) \\ \delta'(i,x) = \delta(i,x) \text{ sinon} \end{cases}.$$

On utilise le lemme précédent où

$$h(x, n_1, \ldots, n_k, n) = \prod_{i=1}^{k(0,x)} \delta'(k(i,n), x).$$

■

Codage des formules

Passons maintenant aux codages proprement dits.
Nous allons coder les éléments d'une théorie finie[6] du premier ordre T sur un langage $\mathcal{L}$ de la façon suivante :
Soit un symbole relationnel r_i. Le nombre associé est $\alpha_3(i, n_i, 0)$ où n_i est l'arité de r_i.
Soit un symbole fonctionnel f_i. Le nombre associé est $\alpha_3(i, m_i, 1)$ où m_i est l'arité de f_i.
Les nombres associés aux symboles constants et aux variables (dénotés respectivement c_i et v_i où $i \in \mathbb{N}$)sont donnés dans la définition suivante.
Par la suite, nous poserons *Rel* (respectivement $Fonc$, Var et C) l' ensemble des nombres codant les symboles relationnels (respectivement symboles fonctionnels, variables, et symboles constants).
Vu que ces ensembles sont finis, ils sont primitifs récursifs. Il suffirait en effet d'appliquer le schéma de généralisation par cas pour passer en revue tous les entiers codant leurs éléments.

Définition 2.3.7. Nous définissons par induction sur le terme t un entier noté $\#t$ appellé *le numéro de Gödel du terme* t, par les conditions suivantes :

- Si $t = c_m$, alors $\#t = \alpha_3(m, 0, 0)$;
- Si $t = v_n$, alors $\#t = \alpha_3(n, 0, 1)$;
- Si $t = f(t_1, t_2, \ldots, t_n)$, alors $\#t = \alpha_3(\Omega(\#t_1, \#t_2, \ldots, \#t_n), \#f, 2)$.

Lemme 2.3.8. *L'ensemble Term*$= \{\#t;\ t$ *est un terme de* $\mathcal{L}\}$ *est primitif récursif.*

Démonstration. Soit la fonction caractéristique g de Term définie de la façon suivante : $g(0) = 1$ et si $x > 0$

6. Cela signifie que l'ensemble des symboles relationnels, fonctionnels et constants est un ensemble fini.

- Si $\beta_3^3(x) = 0$ et $\beta_3^2(x) = 0$ alors $g(x) = 1$.
- Si $\beta_3^3(x) = 0$ et $\beta_3^2(x) \neq 0$ alors $g(x) = 0$.
- Si $\beta_3^3(x) = 1$ et $\beta_3^2(x) = 0$ alors $g(x) = 1$.
- Si $\beta_3^3(x) = 1$ et $\beta_3^2(x) \neq 0$ alors $g(x) = 0$.
- Si $\beta_3^3(x) = 2$ et $\beta_3^2(x) \in Fonc$ et $\beta_3^2(\beta_3^2(x)) = lg(\beta_3^1(x))$ alors

$$g(x) = \prod_{i=0}^{lg(\beta_3^1(x))-1} g(\delta(i, \beta_3^1(x)))$$

- $g(x) = 0$ sinon.

Vu que $\beta_3^i(x) < x$ pour tout entier $x > 1$, en posant $k(0, x) = lg(\beta_3^1(x))+1$ et $k(i, x) = \delta(i-1, \beta_3^1(x))$ pour tout $i \geq 1$, le corollaire 2.3.6 et le schéma de définition par cas démontrent que g est primitive récursive. ■

Proposition 2.3.9. *Le codage est injectif*

Démonstration. Par récurrence sur la longueur des termes.
Soient t et t' des termes tels que $\#t = \#t'$.
Cas de base :
Si t' est une constante alors $\exists m, n$ tels que $t = c_n, t' = c_m$.
On a donc $\#t = \alpha_3(n, 0, 0)$ et $\#t' = \alpha_3(m, 0, 0)$.
On a donc $m = n$ et donc $t = t'$.
Le raisonnement est similaire si t' est une variable.
Cas d'induction :
Il suffit de se rappeler que

$$\beta_3^i(\#t) = \beta_3^i(\#t')\forall i \in \{1, 2, 3\}$$

■

Nous appellerons *terme fermé* un terme qui est soit une constante soit un terme de la forme $f(t_1, \ldots, t_n)$ où f est un symbole fonctionnel et où $t_1, \ldots, t_n$ sont des termes fermés.

Lemme 2.3.10. *L'ensemble* $\overline{Term} = \{\#t;\ t$ *est un terme fermé de* $\mathcal{L}\}$ *est primitif récursif.*

Démonstration. Soit la fonction caractéristique f de $\overline{Term}$ définie de la façon suivante : $g(0) = 1$ et si $x > 0$

- Si $\beta_3^3(x) = 0$ et $\beta_3^2(x) = 0$ alors $g(x) = 1$.

- Si $\beta_3^3(x) = 0$ et $\beta_3^2(x) \neq 0$ alors $g(x) = 0$.
- Si $\beta_3^3(x) = 1$ alors $g(x) = 0$.
- Si $\beta_3^3(x) = 2$ et $\beta_3^2(x) \in Fonc$ et $\beta_3^2(\beta_3^2(x)) = lg(\beta_3^1(x))$ alors

$$g(x) = \prod_{i=0}^{lg(\beta_3^1(x))-1} g(\delta(i, \beta_3^1(x)))$$

- $g(x) = 0$ sinon.

■

Définition 2.3.11. De manière analogue, nous définissons un entier noté $\#F$ appellé *le numéro de Gödel de la formule* F, par les conditions suivantes :

- Si $F = r(t_1, t_2, \ldots, t_n)$, alors $\#F = \alpha_3(\Omega(\#t_1, \#t_2, \ldots, \#t_n), \#r, 0)$;
- Si $t = \neg F_1$, alors $\#F = \alpha_3(\#F_1, 0, 1)$;
- Si $F = F_1 \wedge F_2$, alors $\#F = \alpha_3(\#F_1, \#F_2, 2)$;
- Si $F = F_1 \vee F_2$, alors $\#F = \alpha_3(\#F_1, \#F_2, 3)$;
- Si $F = F_1 \Rightarrow F_2$, alors $\#F = \alpha_3(\#F_1, \#F_2, 4)$;
- Si $F = F_1 \Leftrightarrow F_2$, alors $\#F = \alpha_3(\#F_1, \#F_2, 5)$;
- Si $F = \forall v_n F_1$, alors $\#F = \alpha_3(\#F_1, n, 6)$;
- Si $F = \exists v_n F_1$, alors $\#F = \alpha_3(\#F_1, n, 7)$;

Vu que $\alpha_3(x_1, x_2, x_3) > x_i$ où $i \in \{1, 2, 3\}$ et que $\Omega(x_1, x_2, \ldots, x_n) > x_j$, pour tout $j \leq n$, on remarque que l'entier codant une formule est strictement supérieur aux entiers codant les termes composant cette dernière.

Lemme 2.3.12. *L'ensemble* $\mathcal{F}orm = \{\#F;\ F$ *est une formule de* $\mathcal{L}\}$ *est primitif récursif.*

Démonstration. La démonstration est analogue à celle du lemme 2.3.8.
Soit h la fonction caractéristique de $\mathcal{F}orm$ définie de la façon suivante :

- Si $\beta_3^3(x) = 0$ et $\beta_3^2(x) \in Rel$ et $\beta_3^2((\beta_3^2(x))) = lg(\beta_3^1(x))$, alors

$$h(x) = \prod_{i=1}^{lg(\beta_3^1(x))-1} g(\delta(i, \beta_3^1(x))).$$

- Si $\beta_3^3(x) = 1$ et $\beta_3^2(x) \neq 0$ alors $h(x) = 0$.
- Si $\beta_3^3(x) = 1$ et $\beta_3^2(x) = 0$ alors $h(x) = h(\beta_3^1(x))$.

- Si $\beta_3^3(x) = 2, 3, 4$ ou 5 alors $h(x) = h(\beta_3^1(x)).h(\beta_3^2(x))$.
- Si $\beta_3^3(x) = 6$ ou 7 alors $h(x) = h(\beta_3^1(x))$
- $h(x) = 0$ sinon.

■

Nous pouvons remarquer que le codage des formules est également injectif.

Lemme 2.3.13. *Les ensembles suivants :*
$\Theta_0 = \{(\#t, n); t$ *est un terme dans lequel* v_n *n'a pas d'occurence*$\}$,
$\Theta_1 = \{(\#t, n); t$ *est un terme dans lequel* v_n *a au moins une occurence*$\}$,
$\Theta_2 = \{(\#t, n, m); t$ *est un terme dans lequel* v_n *a exactement* m *occurences*$\}$,
$\Phi_0 = \{(\#F, n); F$ *est une formule dans laquelle* v_n *n'a pas d'occurence*$\}$,
$\Phi_1 = \{(\#F, n); F$ *est une formule dans laquelle* v_n *n'a pas d'occurence libre*$\}$,
$\Phi_2 = \{(\#F, n); F$ *est une formule dans laquelle* v_n *n'a pas d'occurence liée*$\}$,
$\Phi_3 = \{\#F; F$ *est une formule close*$\}$,
$\Phi_4 = \{(\#F, n); F$ *est une formule dans laquelle* v_n *a au moins une occurence libre*$\}$,
$\Phi_5 = \{(\#F, n); F$ *est une formule dans laquelle* v_n *a au moins une occurence liée*$\}$,
$\Phi_6 = \{(\#F, n, m);$ *aucune occurence libre de* v_m *dans* F *ne se trouve dans le champ d'un quantificateur de* $v_n\}$,
sont primitifs récursifs.

Démonstration.

- Θ_0 : Soit g_0 la fonction caractéristique de l'ensemble Θ_0 définie de la façon suivante :
  - Si $\beta_3^3(x) = 0$ alors $g_0(x, y) = 1$ si et seulement si $\beta_3^2(x) = 0$ et $\beta_3^1(x) \neq y + 1$.
  - Si $\beta_3^3(x) = 1$ et $\beta_3^2(x) \neq 0$ alors $g_0(x, y) = 0$.
  - Si $\beta_3^3(x) = 1$ et $\beta_3^2(x) = 0$ alors $g_0(x, y) = g_0(\beta_3^1(x), y)$.
  - Si $\beta_3^3(x) = 2$ alors $g_0(x, y) = \prod_{i=0}^{lg(\beta_3^1(x))-1} g_0(\delta(i, \beta_3^1(x)), y)$.
  - Si $\beta_3^3(x) > 2$ alors $g_0(x, y) = 0$
- Θ_1 : Soit g_1 la fonction caractéristique de l'ensemble Θ_1. Nous remarquons que $g_1(x, y) = 1 - g_0(x, y)$ et donc que g_1 est une fonction primitive récursive.
- Θ_2 : Soit f une fonction définie de la façon suivante :
  - Si $\beta_3^3(x) = 0$ alors $f(x, y) = 1$ si et seulement si $\beta_3^2(x) = 0$ et $\beta_3^1(x) = y + 1$.
  - Si $\beta_3^3(x) = 1$ et $\beta_3^2(x) \neq 0$ alors $f(x, y) = 0$.
  - Si $\beta_3^3(x) = 1$ et $\beta_3^2(x) = 0$ alors $f(x, y) = f(\beta_3^1(x), y)$.

- Si $\beta_3^3(x) = 2$ alors $f(x,y) = \sum_{i=0}^{lg(\beta_3^1(x))-1} f(\delta(i, \beta_3^1(x)), y)$.
- Si $\beta_3^3(x) > 2$ alors $f(x,y) = 0$

On définit g_2, la fonction caractéristique de l'ensemble Θ_2, de la façon suivante : $g(x,y,m) = 1$ si et seulement si $m = f(x,y)$.
Vu que f est primitive récursive, on a la conclusion.

– Φ_0 : En remarquant que $\Phi_0 = \Phi_1 \cup \Phi_2$, il nous suffit de démontrer que Φ_1 et Φ_2 sont primitifs récursifs.

– Φ_1 : Soit h_1 la fonction caractéristique de l'ensemble Φ_1 définie de la façon suivante :
- Si $\beta_3^3(x) = 0$ alors $h_1(x,y) = \prod_{i=0}^{lg(\beta_3^1(x))-1} g_0(\delta(i, \beta_3^1(x)), y)$.
- Si $\beta_3^3(x) = 1$ et $\beta_3^2(x) \neq 0$ alors $h_1(x,y) = 0$.
- Si $\beta_3^3(x) = 1$ et $\beta_3^2(x) = 0$ alors $h_1(x,y) = h_1(\beta_3^1(x), y)$.
- Si $\beta_3^3(x) = 2, 3, 4$ ou 5 alors $h_1(x,y) = h_1(\beta_3^1(x), y).h_1(\beta_3^2(x), y)$.
- Si $\beta_3^3(x) = 6$ ou 7 et $\beta_3^2(x) \neq y$ alors $h_1(x,y) = h_1(\beta_3^1(x), y)$.
- Si $\beta_3^3(x) = 6$ ou 7 et $\beta_3^2(x) = y$ alors $h_1(x,y) = h(\beta_3^1(x))$.
- Si $\beta_3^3(x) > 7$ alors $h_1(x,y) = 0$

– Φ_2 : Soit h_2 la fonction caractéristique de l'ensemble Φ_2 définie de la façon suivante :
- Si $\beta_3^3(x) = 0$ alors $h_2(x,y) = 1$.
- Si $\beta_3^3(x) = 6$ ou 7 et $\beta_3^2(x) \neq y$ alors $h_2(x,y) = h_2(\beta_3^1(x), y)$.
- Si $\beta_3^3(x) = 6$ ou 7 et $\beta_3^2(x) = y$ alors $h_2(x,y) = 0$.
- Si $\beta_3^3(x) > 7$ alors $h_2(x,y) = 0$

Les cas $\beta_3^3(x) = 1, 2, 3, 4$ et 5 étant identiques à h_1.

– Φ_3 : Soit h_3 la fonction caractéristique de l'ensemble Φ_3 définie de la façon suivante :
- Si $\exists n \leq x$ tel que $h_1(x,n) = 1$ alors $h_3(x) = 0$
- Sinon, $h_3(x) = 1$.

– Φ_4 : Soit h_4 la fonction caractéristique de l'ensemble Φ_4. Nous remarquons que $h_4(x,y) = 1 - h_1(x,y)$ et donc que h_4 est une fonction primitive récursive.

– Φ_5 : Soit h_5 la fonction caractéristique de l'ensemble Φ_5. Nous remarquons que $h_5(x,y) = 1 - h_2(x,y)$ et donc que h_5 est une fonction primitive récursive.

– Φ_6 : Soit h_6 la fonction caractéristique de l'ensemble Φ_6 définie de la façon suivante :

- Si $x \notin \mathcal{F}orm$: $g(x,n,m) = 0$
- Si $x \in \mathcal{F}orm$:
  - Si $\beta_3^1(x) = 0$ alors $g(x,n,m) = 1$
  - Si $\beta_3^1(x) = 1$ alors $g(x,n,m) = g(\beta_3^1(x),n,m)$
  - Si $\beta_3^1(x) = 2,3,4$ ou 5 alors $g(x,n,m) = g(\beta_3^1(x),n,m).g(\beta_3^2(x),n,m)$
  - Si $\beta_3^1(x) = 6$ ou $7, n = \beta_3^2(x)$ et si $(\beta_3^1(x),n) \in \Phi_4$ alors $g(x,n,m) = 0$
  - Si $\beta_3^1(x) = 6$ ou 7 alors $g(x,n,m) = g(\beta_3^1(x),n,m)$
  - Si $\beta_3^1(x) > 7$ alors $g(x,n,m) = 0$

■

Lemme 2.3.14. *Il existe deux fonctions récursives $Subst_t$ et $Subst_f$ à trois variables telles que, si t et u sont des termes et si F est une formule, alors, pour tout entier n :*
$Subst_t(n,\#t,\#u) = \#u_{v_n:=t}$
$Subst_f(n,\#t,\#F) = \#F_{v_n:=t}$

Démonstration. Nous avons pour $Subst_t$:

- Si $\beta_3^3(x) = 0$ et si $x \neq \alpha_3(n+1,0,0)$ alors $Subst_t(n,y,x) = x$.
- Si $\beta_3^3(x) = 0$ et si $x = \alpha_3(n+1,0,0)$ alors $Subst_t(n,y,x) = y$.
- Si $\beta_3^3(x) = 1$ et $\beta_3^2(x) = 0$ alors $Subst_t(n,y,x) = \alpha_3(Subst_t(n,y,\beta_3^1(x)),0,1)$.
- Si $\beta_3^3(x) = 2$ alors

$$Subst_t(n,y,x) = \alpha_3\left(\prod_{i=0}^{lg(\beta_3^1(x))-1} \pi(i)^{Subst_t(n,y,\delta(i,\beta_3^1(x)))+1}, \beta_3^2(x), \beta_3^3(x)\right).$$

- Dans les autres cas, $Subst_t(n,y,x) = x$.

Montrons que $Subst_t$ est primitif récursif en appliquant le lemme 2.3.5.
Nous avons que la fonction

$$h(x,y,t,z) = \alpha_3\left(\prod_{i=0}^{lg(\beta_3^1(z))-1} \pi(i)^{\delta(\delta(i,\beta_3^1(z)),x)+1}, \beta_3^2(z), \beta_3^3(z)\right)$$

est primitive récursive.
A présent, vérifions que $Subst_t(n,y,x) = h(Subst_t^*(n,y,x),n,y,x)$.
On a que

$$h(Subst_t^*(n,y,x),n,y,x) = \alpha_3\left(\prod_{i=0}^{lg(\beta_3^1(x))-1} \pi(i)^{\delta(\delta(i,\beta_3^1(x)),Subst_t^*(n,y,x))+1}, \beta_3^2(x), \beta_3^3(x)\right)$$

Or, vu que $\delta(\delta(i,\beta_3^1(x)),Subst_t^*(n,y,x)) = Subst_t(n,y,\delta(i,\beta_3^1(x)))$, on a bien que

$$h(Subst_t^*(n,y,x),n,y,x) = \alpha_3\left(\prod_{i=0}^{lg(\beta_3^1(x))-1}\pi(i)^{Subst_t(n,y,\delta(i,\beta_3^1(x)))+1},\beta_3^2(x),\beta_3^3(x)\right)$$
$$= Subst_t(n,y,x).$$

Par conséquent, on a bien que $Subst_t$ est primitif récursif. De même, pour $Subst_f$ nous avons :

- Si $\beta_3^3(x) = 0$ alors

$$Subst_f(n,y,x) = \alpha_3\left(\prod_{i=1}^{lg(\beta_3^1(x))-1}\pi(i)^{Subst_t(n,y,\delta(i,\beta_3^1(x)))+1},\beta_3^2(x),\beta_3^3(x)\right).$$

- Si $\beta_3^3(x) = 1$ et $\beta_3^2(x) = 0$ alors $Subst_f(n,y,x) = \alpha_3(Subst_f(n,y,\beta_3^1(x)),0,1)$.
- Si $\beta_3^3(x) = 2,3,4$ ou 5 alors

$$Subst_f(n,y,x) = \alpha_3(Subst_f(n,y,\beta_3^1(x)),Subst_f(n,y,\beta_3^2(x)),\beta_3^3(x)).$$

- Si $\beta_3^3(x) = 6$ ou 7 et si $\beta_3^2(x) = n$ alors $Subst_f(n,y,x) = x$.
- Si $\beta_3^3(x) = 6$ ou 7 et si $\beta_3^2(x) \neq n$ alors

$$Subst_f(n,y,x) = \alpha_3(Subst_f(n,y,\beta_3^1(x)),\beta_3^2(x),\beta_3^3(x)).$$

- Dans les autres cas, $Subst_f(n,y,x) = x$.

■

Codage des démonstrations

Nous allons maintenant nous intéresser aux objets de la logique propositionnelle classique. Considérons $A_1,\ldots,A_n$ comme étant les variables propositionnelles de base.

Définition 2.3.15. Nous définissons un entier noté $\#P$ appellé *le numéro de Gödel de la proposition* P, par les conditions suivantes :

- Si $P = A_n$, alors $\#P = \alpha_3(n,0,0)$;
- Si $P = \neg P_1$, alors $\#P = \alpha_3(\#P_1,0,1)$;
- Si $P = P_1 \wedge P_2$, alors $\#P = \alpha_3(\#P_1,\#P_2,2)$;

- Si $P = P_1 \vee P_2$, alors $\#P = \alpha_3(\#P_1, \#P_2, 3)$;
- Si $P = P_1 \Rightarrow P_2$, alors $\#P = \alpha_3(\#P_1, \#P_2, 4)$;
- Si $P = P_1 \Leftrightarrow P_2$, alors $\#P = \alpha_3(\#P_1, \#P_2, 5)$;

Lemme 2.3.16. *L'ensemble Prop* $= \{\#P : P$ *est une proposition* $\}$ *est primitif récursif*

Démonstration. Soit f la fonction caractéristique de l'ensemble *Prop* définie de la façon suivante :

- Si $\beta_3^3(x) = 0$ alors $f(x) = 1$ si et seulement si $\beta_3^2(x) = 0$.
- Si $\beta_3^3(x) = 1$ et $\beta_3^2(x) = 0$ alors $f(x) = f(\beta_3^1(x))$.
- Si $\beta_3^3(x) = 1$ et $\beta_3^2(x) \neq 0$ alors $f(x) = 0$.
- Si $\beta_3^3(x) = 2, 3, 4$ ou 5 alors $f(x) = f(\beta_3^1(x)).f(\beta_3^2(x))$.
- Si $\beta_3^3(x) > 5$ alors $f(x) = 0$

■

A chaque entier k, on fait correspondre la valeur de vérité λ_k définie de la façon suivante :

$$\begin{cases} \lambda_k(A_n) = 1 \text{ si } \pi(n) \text{ divise } k; \\ \lambda_k(A_n) = 0 \text{ sinon} \end{cases}$$

Remarque 2.3.17. Soit λ une distribution de valeur de vérité et c un entier.
On trouve facilement un entier k tel que pour tout $i \leq c$, $\lambda_k(A_i) = \lambda(A_i)$.
En effet, il suffit de prendre

$$k = \prod_{i=0}^{c} \pi(i)^{\lambda(A_i)}$$

Lemme 2.3.18. *La fonction E définie de la façon suivante :*
• *Si x n'est pas le numéro de Gödel d'une proposition alors* $E(k, x) = 0$
• *Si x est le numéro de Gödel d'une proposition P alors* $E(k, x) = \lambda_k(P)$
est récursive primitive

Démonstration.
Si $x \notin$ Prop, alors $E(k, x) = 0$.
Sinon :

- Si $\beta_3^3(x) = 0$ alors
  - Si $\pi(\beta_3^3(x))$ divise k, alors $E(k, x) = 1$
  - Si $\pi(\beta_3^3(x))$ ne divise pas k, alors $E(k, x) = 0$

- Si $\beta_3^3(x) = 1$ alors $E(k,x) = 1 - E(k, \beta_3^1(x))$
- Si $\beta_3^3(x) = 2$ alors $E(k,x) = E(k, \beta_3^1(x)).E(k, \beta_3^2(x))$
- Si $\beta_3^3(x) = 3$ alors $E(k,x) = sign(E(k, \beta_3^1(x)) + E(k, \beta_3^2(x)))$
- Si $\beta_3^3(x) = 4$ alors $E(k,x) = sign(1 - E(k, \beta_3^1(x)) + E(k, \beta_3^2(x)))$
- Si $\beta_3^3(x) = 5$ et $E(k, \beta_3^1(x)) = E(k, \beta_3^2(x))$ alors $E(k,x) = 1$
- Si $\beta_3^3(x) = 5$ et $E(k, \beta_3^1(x)) \neq E(k, \beta_3^2(x))$ alors $E(k,x) = 0$

■

Théorème 2.3.19. *L'ensemble* $\mathcal{T} = \{\#P : P \text{ est une tautologie }\}$ *est primitif récursif*

Démonstration. Soit P une formule propositionnelle.
Il est évident que si A_n est une variable propositionnelle apparaissant dans P alors $n \leq \#P$
Vu la remarque 2.3.17, P est une tautologie si et seulement si pour tout entier k inférieur ou égal à $\pi(\#P)!$, $\lambda_k(P) = 1$
Par conséquent, nous avons que $x \in \mathcal{T}$ si et seulement si $\forall k \leq \pi(x)!\ E(k,x) = 1$ ■

Théorème 2.3.20. *L'ensemble*

$$Taut = \{\#F : F \text{ est une formule et une tautologie du calcul des prédicats }\}$$

est primitif récursif

Démonstration. A chaque formule F, on associe une proposition P_F obtenue de la façon suivante :
On écrit F sous la forme $P[F_1, \ldots, F_k]$ où P est une proposition dont les variables propositionnelles sont $A_1, \ldots A_k$ et les formules $F_1, \ldots F_k$ ne peuvent pas être de nouveau décomposées à l'aide des connecteurs propositionnels, i.e. chaque formule F_i est soit une formule atomique soit une formule commençant par un quantificateur.
Posons pour chaque i, $c_i := \#F_i$ et $P_F = P[A_{c_1}, \ldots, A_{c_k}]$.
Montrons que F est une tautologie du calcul des prédicats si et seulement si P_F est une tautologie.
Supposons que F est de la forme $J[G_1, \ldots, G_m]$ où les $G_1, \ldots, G_m$ sont des formules du langage $\mathcal{L}$ et où $J[B_1, \ldots, B_m]$ est une formule propositionnelle qui est une tautologie.
Remarquons que P_F s'obtient à partir de J en y substituant aux variables propositionnelles $B_1, \ldots, B_m$ des formules propositionnelles adéquates contruites avec les variables $A_{c_1}, \ldots, A_{c_k}$.
Vu que tous les F_i sont des formules atomiques ou des formules commençant par un quantificateur, nous remarquons que P_F représente la décomposition propositionnelle maximale

de F et que cette décomposition est unique.
En effet, si la décomposition n'était pas maximale, il existerait des entiers i, j, k tels que A_{c_i} serait $\neg A_{c_j}$ ou $A_{c_j} \Rightarrow A_{c_k}$. Par conséquent, F_i serait $\neg F_j$ ou $F_j \Rightarrow F_k$. Ce qui contredit nos hypothèses.
De plus, J est un stade intermédiaire de cette décomposition et, vu que J est une tautologie du calcul propositionnel, alors par 1.1.10, P_F en est également une.
L'autre implication est immédiate vu le lemme 1.1.19.

Il suffit donc de construire une fonction primitive récursive γ telle que, pour toute formule F, $\gamma(\#F) = \#P_F$. On aura alors

$$x \in Taut \text{ si et seulement si } x \in \mathcal{F}orm \text{ et } \gamma(x) \in \mathcal{T}$$

On applique le lemme 2.3.3 en définissant γ de la façon suivante :

- Si $\beta_3^3(x) = 0, 6$ ou 7 alors x est le numéro d'une variable propositionnelle de base et donc $\gamma(x) = \alpha_3(x, 0, 0)$
- Si $\beta_3^3(x) = 1$ alors $\gamma(x) = \alpha_3(\gamma(\beta_3^1(x)), 0, 1)$
- Si $\beta_3^3(x) = 2, 3, 4$ ou 5 alors $\gamma(x) = \alpha_3(\gamma(\beta_3^1(x)), \gamma(\beta_3^2(x)), \beta_3^3(x))$
- Si $\beta_3^3(x) > 7$ alors $\gamma(x) = 0$

■

Nous allons maintenant montrer que les axiomes logiques forment un ensemble primitif récursif.

Théorème 2.3.21. *L'ensemble* $Ax = \{\#F\ ;\ F$ *est un axiome logique*$\}$ *est primitif récursif*

Démontrons tout d'abord les 5 lemmes suivants.

Lemme 2.3.22. *L'ensemble* $Ax_1 = \{\#(F \Rightarrow (G \Rightarrow F))\ ;\ F$ *et* G *sont des formules*$\}$ *est primitif récursif*

Démonstration. Nous avons que $\#(F \Rightarrow (G \Rightarrow F)) = \alpha_3(\#F, \alpha_3(\#G, \#F, 4), 4)$.
Par conséquent, $x \in Ax_1$ si et seulement si $\exists y \leq x, \exists z \leq x$ tels que $y, z \in \mathcal{F}orm$ et $x = \alpha_3(y, \alpha_3(z, y, 4), 4)$ ■

Les démonstrations des 2 lemmes suivants sont analogues à celle du lemme précédent.

Lemme 2.3.23. *L'ensemble*

$$Ax_2 = \{\#((F \Rightarrow (G \Rightarrow H)) \Rightarrow ((F \Rightarrow G) \Rightarrow (F \Rightarrow H)))\,;\, F, G \text{ et } H \text{ sont des formules } \}$$

est primitif récursif

Lemme 2.3.24. *L'ensemble*

$$Ax_3 = \{\#((\neg F \Rightarrow \neg G) \Rightarrow (G \Rightarrow F)); F \text{ et } G \text{ sont des formules } \}$$

est primitif récursif

Lemme 2.3.25. *L'ensemble* $Ax_4 = \{\#(\forall vF \Rightarrow F_{v:=t})$ *; v est une variable, F est une formule, t est un terme et, dans F aucune occurence libre de v ne se trouve dans le champ d'un quantificateur liant une variable de t*$\}$ *est primitif récursif*

Démonstration. Nous avons que

$$\#(\forall v_m F \Rightarrow F_{v_m:=t}) = \alpha_3(\alpha_3(\#F, m, 6), Subst_f(m, \#t, \#F), 4).$$

Par conséquent, $x \in Ax_4$ si et seulement s'il existe y, z et m inférieurs à x tel que $y \in \mathcal{F}orm$, $z \in Term$, et telles que pour tout $n < z$,$((z, n) \in \Phi_0$ ou $(y, n, m) \in \Phi_6)$ et $x = \alpha_3(\alpha_3(y, m, 6), Subst_f(m, z, y), 4)$ ■

Lemme 2.3.26. *L'ensemble* $Ax_5 = \{\#(\forall v(F \Rightarrow G) \Rightarrow (F \Rightarrow \forall vG))$ *; F et G sont des formules et v est une variable qui n'a pas d'occurence libre dans F*$\}$ *est primitif récursif*

Démonstration. De la même façon, on a

$$\#(\forall v_n(F \Rightarrow G) \Rightarrow (F \Rightarrow \forall v_n G)) = \alpha_3(\alpha_3(\alpha_3(\#F, \#G, 4), n, 6), \alpha_3(\#F, \alpha_3(\#G, n, 6), 4), 4).$$

On a donc que $x \in Ax_2$ si et seulement s'il existe y, z et n inférieurs à x telles que $(y, n) \in \Phi_1$, $z \in \mathcal{F}orm$ et $x = \alpha_3(\alpha_3(\alpha_3(y, z, 4), n, 6), \alpha_3(y, \alpha_3(z, n, 6), 4), 4)$ ■

Nous pouvons maintenant démontrer le théorème 2.3.21.

Démonstration. Évident car $Ax = Ax_1 \cup Ax_2 \cup Ax_3 \cup Ax_4 \cup Ax_5$. ■

Définition 2.3.27.

1) Soit T une théorie ; on dit que T est *récursive* si l'ensemble $\#T = \{\#F; F \in T\}$ est récursif.
2) On notera $Th(T) = \{\#F$; F est une formule close et $T \vdash F\}$ l'ensemble des numéros de Gödel des théorèmes de T.
3) On dit que T est *décidable* si $Th(T)$ est récursif. Sinon, on dit que T est *indécidable*.

Soit $d = (F_0, \ldots, F_k)$ une suite de formules du langage $\mathcal{L}$; par $\#\#d$, on désignera l'entier

$$\#\#d = \Omega((\#F_0, \#F_1, \ldots, \#F_k)).$$

On appellera *numéro de Gödel* de la suite de formules d l'entier $\#\#d$.

Proposition 2.3.28. *Soit T une théorie récursive ; alors l'ensemble $Dem(T) = \{(n,m); n = \#F, m = \#\#d$, où F est une formule et d est une démonstration de F dans $T\}$ est primitif récursif.*

Démonstration. Posons les prédicats suivants :

- $A(i,m) \equiv \delta(i,m) \in \mathcal{F}orm$.
- $B(n,m) \equiv \delta(lg(m)-1,m) = n$.
- $C_1(i,m) \equiv \delta(i,m) \in Ax \cup \#T$
- $C_2(i,m) \equiv \exists(j<i)\exists(p<m)\delta(i,m) = \alpha_3(\delta(j,m),p,6)$
- $C_3(i,m) \equiv \exists(j<i)\exists(k<i)\delta(j,m) = \alpha_3(\delta(k,m),\delta(i,m),4)$

On a $(n,m) \in$ Dem si et seulement si le prédicat

$$B(n,m) \cap \forall(i < lg(m))(A(i,m) \cap (C_1(i,m) \cup C_2(i,m) \cup C_3(i,m)))$$

est vrai. La conclusion résultant du fait que ce prédicat est primitif récursif ■

Grâce à cela, nous avons la réciproque du théorème 2.2.4.

Corollaire 2.3.29. *Toute fonction $f : \mathbb{N}^p \to \mathbb{N}$ totale et représentable est récursive.*

Démonstration. Si F est la formule représentant f, on a

$$x = f(n_1,\ldots,n_p) \text{ si et seulement si } \mathcal{P}_0 \vdash F[\underline{x},\underline{n_1},\ldots,\underline{n_p}]$$
$$\text{si et seulement si } \exists y \text{ tel que } (\#F[\underline{x},\underline{n_1},\ldots,\underline{n_p}],y) \in Dem_0.$$

Il nous suffit donc de poser la fonction g définie comme

$$g(n_1,\ldots,n_p) = \mu_y((\#F[\beta_2^1(y),n_1\ldots,n_p],\beta_2^2(y)) \in Dem_0).$$

Cette fonction est totale, récursive et l'on a $f(n_1,\ldots,n_p) = \beta_2^1(g(n_1,\ldots,n_p))$.
La conclusion en découle. ■

Corollaire 2.3.30. *Soit T une théorie récursive : alors $Th(T)$ est récursivement énumérable. En particulier, les ensembles suivants sont récursivement énumérables.*
$\{\#F$; *F est une formule close valide* $\}$
$\{\#F$; *F est un théorème de $\mathcal{P}_0$*$\}$
$\{\#F$; *F est un théorème de $\mathcal{P}$*$\}$

Démonstration. On a que $n \in Th(T)$ si et seulement si $n \in \Phi_3$ et il existe un entier m tel que $(n, m) \in Dem(T)$.
Vu le lemme 1.3.6, la projection de Dem est récursivement énumérable. Φ_3 est récursif donc récursivement énumérable donc Th est l'intersection de deux ensembles récursivement énumérables et est donc récursivement énumérable. ■

Corollaire 2.3.31. *Si T est une théorie complète et récursive alors elle est décidable*

Démonstration. On sait que $Th(T)$ est récursivement énumérable. Montrons que son complémentaire l'est, on aura alors par la proposition 1.3.5 que $Th(T)$ est récursif et donc T est décidable.
Vu que T est complète, si F est une formule close qui n'est pas un théorème de T, alors $\neg F$ est un théorème de T, ce qui se traduit par

$$m \notin Th(T) \text{ si et seulement si } m \notin \Phi_3 \text{ ou } \alpha_3(m, 0, 1) \in Th(T)$$

■

2.4 Le premier théorème d'incomplétude

Cette brève section démontrera le premier théorème d'incomplétude de Gödel en se basant sur le caractère récursif des ensembles définis dans la section précédente.

Définition 2.4.1. Soit T une théorie sur un langage $\mathcal{L}(T)$.
Supposons que $\mathcal{L}(T)$ est une extension finie de $\mathcal{L}_0$ et soit $N[x]$ une formule de $\mathcal{L}(T)$.
La *relativisation F^N d'une formule F* par rapport à N est obtenue en remplaçant toutes les sous-formules de F de la forme $\forall x F_1$ par $\forall x(N[x] \Rightarrow F_1)$.
On dit que T *contient* $\mathcal{P}_0$ s'il existe une formule $N[x]$ telle que les assertions suivantes soient des théorèmes de T :

1. $N[0]$
2. $N[1]$
3. $\forall x \forall y(N[x] \wedge N[y] \Rightarrow N[x + y])$
4. $\forall x \forall y(N[x] \wedge N[y] \Rightarrow N[x.y])$

et si $\mathcal{P}_0^N = \{F^N : F \in \mathcal{P}_0\} \subset T$ i.e. tout axiome de $\mathcal{P}$ relativement à $N[x]$ est dans T.

Remarque 2.4.2. Prenons la théorie des ensembles $T := ZFC$. Nous avons que les constantes et les symboles fonctionnels peuvent être définis dans ZFC, que les axiomes de $\mathcal{P}$ peuvent être démontrés dans ZFC et qu'il existe une formule $N[x]$ de ZFC définissant

les nombres naturels (voir [7]).
Nous pouvons donc supposer que $\mathcal{L}(\mathcal{P}) \subset \mathcal{L}(ZFC)$ et nous avons que ZFC contient bien $\mathcal{P}$.

Théorème 2.4.3. *Si T est une théorie consistante contenant $\mathcal{P}_0$; alors T est indécidable*

Démonstration. Par l'absurde. Supposons que T est décidable.
Considérons l'ensemble $\Theta = \{(m,n); m$ est le numéro de Gödel d'une formule $F[v_0]$ dont v_0 est la seule variable libre éventuelle et $T \vdash F[\underline{n}]\}$.
Montrons que si T est décidable alors Θ est récursif :
Soit $A = \{\#F; F$ est une formule dont v_0 est la seule variable libre éventuelle$\}$.
Cet ensemble est récursif car $m \in A$ si et seulement si pour tout p compris entre 1 et m, $(m,p) \notin \Phi_4$ (i.e s'il n'y a pas de v_p libre dans m).
On a alors que $(m,n) \in \Theta$ si et seulement si $m \in A$ et $Subst_f(0, \#\underline{n}, m) \in Th(T)$.
L'ensemble Θ est donc récursif et il en résulte que l'ensemble $B = \{n \in \mathbb{N}; (n,n) \notin \Theta\}$ est récursif.
D'après le théorème de représentation, il existe une formule $G[v_0]$ qui le représente.
On a donc pour tout $n \in \mathbb{N}$,

$$n \in B \text{ si et seulement si } \mathcal{P}_0 \vdash G[\underline{n}] \text{ donc } T \vdash G[\underline{n}] \tag{2.1}$$

$$n \notin B \text{ si et seulement si } \mathcal{P}_0 \vdash \neg G[\underline{n}] \text{ donc } T \vdash \neg G[\underline{n}] \tag{2.2}$$

On a que $\#G[v_0]$ est un élément de A.
Or, $\#G[v_0] \in B$ si et seulement si $(\#G[v_0], \#G[v_0]) \notin \Theta$ et par définition de Θ, il est faux que $T \vdash G[\underline{\#G[v_0]}]$.
Or, en utilisant (2.1), $\#G[v_0] \in B$ implique que $T \vdash G[\underline{\#G[v_0]}]$.
Si $\#G[v_0] \notin B$, le même raisonnement implique que T n'est pas consistante.

■

Remarquons que la formule $G[\underline{\#G[v_0]}]$ exprime sa propre improuvabilité dans la théorie T. En cela, une analogie peut être faite avec le célèbre *paradoxe du menteur*[7].
Le corollaire suivant montre l'indécidabilité du calcul des prédicats.

Corollaire 2.4.4 (Théorème de Church)**.** $T_0 = \{F; F$ *est une formule close de* $\mathcal{L}$ *universellement valide* $\}$ *n'est pas récursif*

7. Paradoxe attribué à Épiménide le Crétois (VIIe siècle av. J.-C.) exprimant le propos « Tous les Crétois sont des menteurs ».

Démonstration. Soit G la conjonction de tous les axiomes de $\mathcal{P}_0$.
Il est clair que toute formule close de F de $\mathcal{L}_0$, $\mathcal{P}_0 \vdash F$ si et seulement si $(G \Rightarrow F) \in T_0$
On voit que si T_0 était récursive, on aurait que $\mathcal{P}_0$ serait décidable, ce qui n'est pas vrai vu le théorème précédent. ■

Nous voici enfin au premier théorème d'incomplétude de Gödel.

Théorème 2.4.5 (Théorème d'incomplétude de Gödel-Rosser)**.** *Soit T une théorie récursive et consistante contenant $\mathcal{P}_0$.*
Alors T n'est pas complète. En particulier $\mathcal{P}$ n'est pas complète.

Démonstration. Par l'absurde, supposons que T est complète, vu le corollaire 2.3.31, on a que T est décidable. Or, vu le théorème 2.4.3, T est indécidable. Contradiction. ■

Il reste cependant un point que nous n'avons pas démontré : le caractère récursif de $\mathcal{P}$!

Proposition 2.4.6. *L'arithmétique de Peano est récursive.*

Démonstration. Vu que les axiomes $A_1, \ldots, A_7$ sont de simples formules, la seule difficulté concerne le schéma d'induction.
Grâce au codage des éléments de $\mathcal{P}$, on peut voir que l'ensemble des codes des formules de la forme

$$(F[\underline{0}, v_1, \ldots, v_n] \wedge \forall v_0(F[v_0, \ldots, v_n] \Rightarrow F[\underline{S}v_0, v_1, \ldots, v_n])) \Rightarrow \forall v_0 F[v_0, \ldots, v_n]$$

est récursif. Il nous reste à déterminer de manière récursive combien de variable libre possède F pour connaitre le nombre exact de quantificateurs devant la formule précédente.
Il suffit de procéder par récurrence sur la longueur de F : si c'est une formule atomique, la réponse est l'arité du symbole relationnel, sinon seul le cas où F est de la forme $\forall x G$ mérite d'être un minimum soigneux.

De cette manière, pour un code de F donné, nous pouvons calculer récursivement le code de

$$\forall v_1 \ldots \forall v_n((F[\underline{0}, v_1, \ldots, v_n] \wedge \forall v_0(F[v_0, \ldots, v_n] \Rightarrow F[\underline{S}v_0, v_1, \ldots, v_n])) \Rightarrow \forall v_0 F[v_0, \ldots, v_n])$$

et inversément, pour un entier codant une formule de cette forme, il est possible de retrouver le code de F, d'où la conclusion. ■

2.5 Vers le deuxième théorème

Dans cette section, nous allons continuer l'utilisation des codages pour pouvoir démontrer, à l'intérieur d'une théorie T, les théorèmes de logique classique comme le lemme de déduction, le principe de démonstration par l'absurde ainsi que le théorème de complétude en nous basant sur [1].
Ce travail exigeant un grand nombre de vérifications fastidieuses est, pour cette raison, absent de la majorité des ouvrages traitant des théorèmes de Gödel sous prétexte qu'il n'est pas difficile de se convaincre du caractère récursif de ces démonstrations.

Par la suite, nous allons souvent confondre volontairement un ensemble récursif et la formule le représentant. À titre d'exemple, lors de la lecture de l'expression $\mathcal{F}orm[x]$, il faudra comprendre $x \in \mathcal{F}orm$.
Nous appliquerons cet abus de langage à la majorité des ensembles récursifs rencontrés précédemment.

Soit T une théorie récursive contenant $\mathcal{P}$, soient les deux ensembles récursifs Dem et Dem_0 définis par :

- $Dem_T = \{(a,b)\,;\, a$ est le numéro de Gödel d'une formule close F et b est le numéro de Gödel d'une démonstration de F dans $T\}$
- $Dem_0 = \{(a,b)\,;\, a$ est le numéro de Gödel d'une formule close F et b est le numéro de Gödel d'une démonstration de F dans $\mathcal{P}_0\}$

Vu le théorème de représentation, il existe deux formules de $\mathcal{L}_0$, à deux variables libres qui représentent ces ensembles. Nous les noterons $\mathfrak{Dem}$ et $\mathfrak{Dem}_0$. Soit la fonction primitive récursive Neg définie de la façon suivante :
Si n est le numéro de Gödel d'une formule close F, alors $Neg(n)$ est le numéro de Gödel de $\neg F$ sinon $Neg(n) = 0$. C'est-à-dire,

$$Neg(n) = \begin{cases} \alpha_3(n,0,1) \text{ si } n \text{ est le numéro de Gödel d'une formule close } F. \\ 0 \text{ sinon.} \end{cases}$$

Vu que l'ensemble des formules closes est primitif récursif, cette fonction est bien primitive récursive. Elle est de plus représentable. Soit $\mathfrak{Neg}[v_0, v_1]$ une formule de $\mathcal{L}_0$ représentant cette fonction.

Nous écrirons $\mathcal{W}_T[\#F]$ la formule $\exists v_1 \mathfrak{Dem}_T[\#F, v_1]$.
Avant toute chose, voici quelques propriétés qui nous seront utiles par la suite

Proposition 2.5.1. *Les formules*

$$F \Rightarrow [G \Rightarrow \neg(F \Rightarrow \neg G)]$$

et

$$(\neg F \Rightarrow G) \Rightarrow [(\neg F \Rightarrow \neg G) \Rightarrow F]$$

sont des théorèmes du calcul des prédicats.

Démonstration. Pour la première formule, étant donné que $\{F\} \cup \{F \Rightarrow \neg G\} \vdash \neg G$, il suffit d'utiliser, dans cet ordre, le lemme de déduction, la contraposée et encore une fois le lemme de déduction.
La deuxième formule est une conséquence de la première ; en effet, on a :

$$\{\neg F\} \cup \{G\} \vdash (\neg(\neg F \Rightarrow \neg G)).$$

Or, on sait aussi que $\{\neg F\} \cup \{\neg F \Rightarrow G\} \vdash G$, d'où $\{\neg F\} \cup \{\neg F \Rightarrow G\} \vdash \neg(\neg F \Rightarrow \neg G)$. En utilisant le lemme de déduction et la contraposée, on obtient

$$\{\neg F \Rightarrow G\} \vdash (\neg F \Rightarrow \neg G) \Rightarrow F$$

On conclut en appliquant une dernière fois le lemme de déduction. ■

Remarque 2.5.2. Vu que $F \wedge G$ est $\neg(F \Rightarrow \neg G)$, la première formule peut se réécrire sous la forme

$$\{F\} \cup \{G\} \vdash F \wedge G.$$

Nous allons à présent commencer l'internalisation de la logique dans l'arithmétique.

Proposition 2.5.3.

$$\mathcal{P} \vdash (Ax[\#F] \vee T[\#F]) \Rightarrow \mathcal{W}_T[\#F]$$

Démonstration. Évident, il suffit de constater que $\Omega((\#F))$ est une démonstration de F dans T. ■

Proposition 2.5.4.

$$\mathcal{P} \vdash (\mathcal{W}_T[\#F] \wedge \mathcal{W}_T[\#(F \Rightarrow G)]) \Rightarrow \mathcal{W}_T[\#G]$$

Démonstration. En se rappelant du codage des démonstrations et du caractère récursif de la fonction de concaténation, 2.3.1, on remarque qu'une démonstration possible de G est la concaténation de la démonstration de F et de $F \Rightarrow G$. ■

Soit S une formule à une variable libre. Posons Γ_S la formule

$$\forall F, G((\mathcal{F}orm[\#F] \wedge \mathcal{F}orm[\#G]) \Rightarrow ((S[\#F] \wedge S[\#(F \Rightarrow G)]) \Rightarrow S[\#G]))$$

et Δ_S la formule

$$\forall F(\mathcal{F}orm[\#F] \Rightarrow (S[\#F] \Rightarrow S[\#(\forall v_n F)]))$$

On en tire les équivalences suivantes

$\mathfrak{M} \vDash \Gamma_S$ si et seulement si $S^M = \{a \in M : \mathfrak{M} \vDash S[a]\}$ est fermé par passage au modus ponens

et

$\mathfrak{M} \vDash \Delta_S$ si et seulement si S^M est fermé par passage à la règle de généralisation.

La proposition suivante nous sera fort utile par la suite.

Proposition 2.5.5.

$$\mathcal{P} \vdash (\Gamma_S \wedge \Delta_S \wedge \forall x((T[x] \vee Ax[x]) \Rightarrow S[x])) \Rightarrow \forall x(\mathcal{W}_T[x] \Rightarrow S[x])$$

Démonstration. Tout d'abord, montrons que

$$\mathcal{P} \cup \{\Gamma_S \wedge \Delta_S \wedge \forall x[(T[x] \vee Ax[x]) \Rightarrow S[x]]\} \cup \{\mathcal{W}_T[\#F]\} \vdash S[\#F]$$

Nous supposons donc une preuve de F dans T. Soient $\mathfrak{M}$ un modèle de T et $d \in M$ cette preuve de F. Considérons l'ensemble suivant

$$\{i \in M : i <^M lg^M(d) \Rightarrow \delta(i, d) \in S^M\}$$

Vu que lg et δ sont des fonctions représentables, cet ensemble est également représentable par une formule que nous appellerons $A[v_0]$.
A présent, utilisons SI sur $A[i]$.
Cas de base :
On a que $\delta(0, d) \in T \cup Ax$, ce qui suffit.
Cas d'inférence :
Supposons que $\delta(i, d) \in S^M$ pour $i < lg(d) - 1$ et montrons que $\delta(i + 1, d) \in S^M$.
Vu le théorème 2.3.28, on a que $\delta(i+1, d)$ est soit dans T, soit dans Ax, soit une implication, soit une généralisation. Ce qui suffit vu les formules Γ_S et Δ_S.

En utilisant le lemme de déduction, on a donc que

$$\mathcal{P} \cup (\Gamma_S \wedge \Delta_S \wedge \forall x[(T(x) \vee Ax(x)) \Rightarrow S(x)]) \vdash \mathcal{W}_T(\#F) \Rightarrow S(\#F)$$

Il nous suffit d'appliquer la règle d'inférence de généralisation suivie du lemme de déduction pour obtenir la conclusion souhaitée. ■

Posons $T_z[x]$ comme étant la formule $T[x] \wedge (x \leq z)$. Nous obtenons le *théorème de compacité dans Peano.*

Proposition 2.5.6. *La formule*

$$\mathcal{W}_T[\#F] \Rightarrow \exists z \mathcal{W}_{T_z}[\#F]$$

est démontrable dans $\mathcal{P}$.

Démonstration. Si d est une preuve, alors $\delta(i, d) < d$ pour tout $i < lg(d)$.
Et on a bien que d est une preuve dans T_d. ■

Soit $(T; y)[x]$ la formule $T[x] \vee (x = y)$. Nous obtenons le *lemme de déduction dans Peano.*

Proposition 2.5.7. *La formule*

$$\mathcal{W}_{(T;\#F)}[\#G] \Leftrightarrow \mathcal{W}_T[\#(F \Rightarrow G)]$$

est démontrable dans $\mathcal{P}$.

Démonstration. Tout d'abord, montrons que

$$\mathcal{P} \vdash [\mathcal{W}_T[\#(F \Rightarrow G)] \Rightarrow \mathcal{W}_{(T;\#F)}[\#G]]$$

en utilisant le lemme de déduction.
Une démonstration valide dans T restant évidemment valide dans $(T; \#F)$, on a

$$\mathcal{P} \cup \{\mathcal{W}_T[\#(F \Rightarrow G)]\} \vdash \mathcal{W}_{(T;\#F)}[\#(F \Rightarrow G)]$$

On a évidemment que $\mathcal{P} \vdash \mathcal{W}_{(T;\#F)}[\#F]$ et donc que $\mathcal{P} \cup \{\mathcal{W}_T[\#(F \Rightarrow G)]\} \vdash \mathcal{W}_{(T;\#F)}[\#F]$.
On obtient enfin,

$$\mathcal{P} \cup \{\mathcal{W}_T[\#(F \Rightarrow G)]\} \vdash \mathcal{W}_{(T;\#F)}[\#G]$$

par modus ponens.
A présent, montrons que

$$\mathcal{P} \vdash \mathcal{W}_{(T;\#F)}[\#G] \Rightarrow \mathcal{W}_T[\#(F \Rightarrow G)]$$

On utilise la proposition 2.5.5 avec la formule $\mathcal{W}_T[\#(F \Rightarrow x)]$ et la théorie $(T;\#F)$.
Montrons donc que

$$\mathcal{P} \vdash \Gamma_{\mathcal{W}_T[\#(F\Rightarrow x)]} \wedge \Delta_{\mathcal{W}_T[\#(F\Rightarrow x)]} \wedge \forall x((T;\#F)[x] \vee Ax[x] \Rightarrow \mathcal{W}_T[\#(F \Rightarrow x)])$$

On commence par montrer que

$$\mathcal{P} \vdash \Gamma_{\mathcal{W}_T[\#(F\Rightarrow x)]}$$

i.e.

$$\mathcal{P} \vdash \forall H, G((\mathcal{F}orm[\#H] \wedge \mathcal{F}orm[\#G]) \Rightarrow ((\mathcal{W}_T[\#(F \Rightarrow H)] \wedge \mathcal{W}_T[\#(F \Rightarrow (H \Rightarrow G))] \Rightarrow \mathcal{W}_T[\#(F \Rightarrow G)]))).$$

C'est le cas vu que la formule $(F \Rightarrow (H \Rightarrow G)) \Rightarrow ((F \Rightarrow H) \Rightarrow (F \Rightarrow G))$ est l'axiome L_2. Il suffit donc d'appliquer deux fois le modus ponens pour avoir le résultat souhaité.
Ensuite, montrons que

$$\mathcal{P} \vdash \Delta_{\mathcal{W}_T[\#(F\Rightarrow x)]}$$

Il faut donc montrer que

$$\forall G(\mathcal{F}orm[\#G] \Rightarrow (\mathcal{W}_T[\#(F \Rightarrow G)] \Rightarrow \mathcal{W}_T[\#(F \Rightarrow \forall yG)]))$$

C'est bien le cas par L_5.
A présent, montrons que

$$\mathcal{P} \vdash \forall x((T;\#F)[x] \vee Ax[x] \Rightarrow \mathcal{W}_T[\#(F \Rightarrow x)])$$

Considérons les deux cas suivants :

- Soit x appartient à l'ensemble des axiomes ou à l'ensemble T. Dans ce cas, $\Omega((x, \#L_1))$ est le code d'une démonstration de $F \Rightarrow x$ dans T.

- Soit x est F. Dans ce cas, $F \Rightarrow F$ est bien une tautologie de T.

■

Nous constatons donc que, pour la majorité des preuves, il suffit de réécrire dans $\mathcal{P}$ les preuves faites de manière générale en logique du premier ordre.

Posons $Cons(T)$ comme étant la formule exprimant la *consistance* de la théorie T , c'est-à-dire, qu'il n'existe pas de formule F telle que T démontre F et sa négation.
Nous posons donc la formule $Cons(T)$ comme étant

$$\forall v_0 \forall v_1 \neg(\mathcal{W}_T[v_0] \wedge \mathcal{W}_T[v_1] \wedge \mathfrak{Neg}[v_0, v_1])$$

et $\perp$ la formule $F \wedge \neg F$ où F est une formule quelconque.
On en tire que

Proposition 2.5.8.

$$\mathcal{P} \vdash \neg Cons(T) \Leftrightarrow \forall v_0(\mathcal{F}orm[v_0] \Rightarrow \mathcal{W}_T[v_0])$$

Démonstration. On a trivialement que

$$\mathcal{P} \vdash \mathcal{W}_{(\{\neg F\} \cup \{\neg G\})}[\#\neg F]$$

Par le lemme de déduction dans Peano, on a que

$$\mathcal{P} \vdash \mathcal{W}_{\{\neg F\}}[\#(\neg G \Rightarrow \neg F)]$$

Par contraposée converse, on a donc que

$$\mathcal{P} \vdash \mathcal{W}_{\{\neg F\}}[\#(F \Rightarrow G)]$$

Et en appliquant de nouveau le lemme de déduction dans Peano, on obtient

$$\mathcal{P} \vdash \mathcal{W}[\#(\neg F \Rightarrow (F \Rightarrow G))]$$

A présent, supposons avoir

$$\mathcal{P} \vdash \neg Cons(T)$$

Par conséquent,

$$\mathcal{P} \vdash \exists \#F(\mathcal{W}_T[\#F] \wedge \mathcal{W}_T[\#\neg F]).$$

On en tire donc que

$$\mathcal{P} \vdash \mathcal{W}_T[\#G]$$

quelle que soit la formule G, d'où la conclusion, l'autre implication étant évidente. ■

Proposition 2.5.9.

$$\mathcal{P} \vdash Cons(T) \Leftrightarrow \neg \mathcal{W}_T[\#(F_0 \wedge \neg F_0)]$$

où F_0 est une formule quelconque de T.

Démonstration. Démontrons cela par contraposées :
Si $\mathcal{P} \vdash \mathcal{W}_T[\#(F \wedge \neg F)]$, on a $\mathcal{P} \vdash \mathcal{W}_T[\#F] \wedge \mathcal{W}_T[\#\neg F]$.
Par conséquent, $\mathcal{P} \vdash \neg Cons(T)$.
Réciproquement, si $\mathcal{P} \vdash \neg Cons(T)$, on conclut par le théorème précédent. ■

À présent, nous allons obtenir le *théorème de démonstration par l'absurde dans Peano.*

Proposition 2.5.10. *La formule*

$$\neg Cons(T; \#F) \Leftrightarrow \mathcal{W}_T[\#\neg F]$$

est démontrable dans $\mathcal{P}$

Démonstration. Montrons que

$$\mathcal{P} \vdash [\mathcal{W}_T[\#\neg F] \Rightarrow \neg Cons(T; \#F)]$$

On utilise le lemme de déduction. On veut donc montrer que

$$\mathcal{P} \cup \{\mathcal{W}_T[\#\neg F]\} \vdash\Rightarrow \neg Cons(T; \#F)$$

Vu qu'une démonstration de $\neg F$ dans T reste une démonstration de $\neg F$ dans $T \cup \{F\}$ et que $T \cup \{F\}$ démontre trivialement F, on a

$$\mathcal{P} \cup \{\mathcal{W}_T[\#\neg F]\} \vdash \mathcal{W}_{(T;\#F)}[\#\neg F] \text{ et } \mathcal{P} \cup \{\mathcal{W}_T[\#\neg F]\} \vdash \mathcal{W}_{(T;\#F)}[\#F]$$

On en tire que

$$\mathcal{P} \cup \{\mathcal{W}_T[\#\neg F]\} \vdash \neg Cons(T; \#F)$$

A présent, montrons que

$$\mathcal{P} \vdash \neg Cons(T; \#F) \Rightarrow \mathcal{W}_T[\#\neg F]$$

Vu le théorème 2.5.8, on a

$$\mathcal{P} \vdash \neg Cons(T; \#F) \Leftrightarrow \forall v_0 \ (\mathcal{F}orm[v_0] \Rightarrow \mathcal{W}_{(T;\#F)}[v_0])$$

Donc, on a

$$\mathcal{P} \cup \{\neg Cons(T; \#F)\} \vdash \mathcal{W}_{(T;\#F)}[\#H] \text{ et } \mathcal{P} \cup \{\neg Cons(T; \#F)\} \vdash \mathcal{W}_{(T;\#F)}[\#\neg H]$$

où H est une formule de $(T;\#F)$ et, en utilisant le lemme de déduction dans Peano, 2.5.7, on a

$$\mathcal{P} \cup \{\neg Cons(T;\#F)\} \vdash \mathcal{W}_T[\#(F \Rightarrow H)] \text{ et } \mathcal{P} \cup \{\neg Cons(T;\#F)\} \vdash \mathcal{W}_T[\#(F \Rightarrow \neg H)]$$

Il n'est pas difficile de se convaincre que les démonstrations de la proposition 2.5.1 sont transposables dans Peano. Par conséquent, on obtient

$$\mathcal{P} \cup \{\neg Cons(T;\#F)\} \vdash \mathcal{W}_T[\#\neg F]$$

d'où la conclusion par le lemme de déduction. ■

Proposition 2.5.11.

$$\mathcal{P} \vdash [Cons(T) \wedge \mathcal{W}_T[\#F] \Rightarrow Cons(T;\#F)]$$

Démonstration. Par l'absurde et en utilisant le théorème précédent, on a

$$\mathcal{P} \cup \{Cons(T) \wedge \mathcal{W}_T[\#F]\} \cup \{\neg Cons(T;\#F)\} \vdash \mathcal{W}_T[\#\neg F]$$

Ce qui entraîne une contradiction. ■

Proposition 2.5.12.

$$\mathcal{P} \vdash [Cons(T) \Rightarrow \forall F(Cons(T;\#F) \vee Cons(T;\#\neg F))]$$

Démonstration. Supposons que T est consistante ; on a donc que, pour toute formule F, soit F soit $\neg F$ n'est pas démontrable dans T. Vu le théorème 2.5.10, on a la conclusion. ■

Nous allons maintenant obtenir le *théorème du changement de variables liées dans Peano*.

Proposition 2.5.13.

$$\mathcal{P} \vdash (Var[\#z] \wedge (\#F < \#z) \wedge (\#z \neq \#x) \Rightarrow (\mathcal{W}_T[\#\forall xF] \Leftrightarrow \mathcal{W}_T[\#\forall zF(x := z)]))$$

Démonstration. Montrons que

$$\mathcal{P} \cup \{Var[\#z] \wedge (\#F < \#z) \wedge (\#z \neq \#x)\} \vdash (\mathcal{W}_T[\#\forall xF] \Rightarrow \mathcal{W}_T[\#\forall zF(x := z)])$$

La condition $\#F < \#z$ assurant qu'aucune occurence d'une variable de code z ne deviendra liée dans $F(x := z)$, il suffit d'utiliser L_4 suivi de la règle d'inférence de généralisation.

Montrons à présent que

$$\mathcal{P} \cup \{Var[\#z] \wedge (\#F < \#z) \wedge (\#z \neq \#x)\} \vdash (\mathcal{W}_T[\#\forall z F(x := z)]) \Rightarrow \mathcal{W}_T[\#\forall x F]$$

Si l'on substitue x à z dans la formule $F(x := z)$, on retrouve la formule F.
Par conséquent, en partant de $\forall z F(x := z)$, on applique L_4 puis la règle d'inférence de généralisation pour conclure. On peut effectivement appliquer L_4 car la précédente substitution a eu pour conséquence que x n'appartient pas à l'ensemble des variables libres de $\forall z F(x := z)$. ■

Avant de prouver le théorème de complétude dans Peano, nous introduisons *l'axiome d'Henkin*.
Soient la fonction totale $Z(x, \#F) = \alpha_3(\alpha_2(x, \#F), 0, 1)$ et la formule $h[\#F]$ définie par

$$\exists x, z, \#G \leq \#F[z = Z(x, \#G) \wedge \mathcal{F}orm[\#G] \wedge \#F = imp(Neg(gen(x, Neg(\#G))), \#G(x := z))]$$

déterminant l'ensemble des formules de la forme $\neg\forall\neg G \Rightarrow G(x := z)$, où la variable z est choisie de telle sorte qu'elle n'apparaît dans aucune paire $(y, \#H)$ où $y \leq x$ et $\#H \leq \#G$.
Soit Th la formule $T \vee h$. On a alors

Théorème 2.5.14.

$$\mathcal{P} \vdash Cons(T) \Rightarrow Cons(Th)$$

pour toute théorie T.

Démonstration. Soit $\{H_1, H_2, \ldots\}$ l'ensemble des formules satisfaisant la formule h.
Par l'absurde, supposons que Th est inconsistante.

$$\mathcal{P} \cup Cons(T) \vdash \neg Cons(Th)$$

i.e., vu la proposition 2.5.9,

$$\mathcal{P} \cup Cons(T) \vdash \mathcal{W}_{Th}[\#\bot]$$

Par le théorème de compacité dans Peano, 2.5.6, il existe un sous-ensemble fini T_m tel que $T_m \vdash \bot$.

$$\mathcal{P} \cup Cons(T) \vdash \exists m \mathcal{W}_{Th_m}[\#\bot]$$

où Th_m est de la forme $Th_m = T' \cup \{H_1, \ldots H_l\}$ où $T' \subset T$.

Posons $T_n^* = T \cup \{H_1, \ldots H_n\}$ pour tout n. On a donc que $Th_m \subset T_l^*$ et donc T_l^* est aussi inconsistant.

Afin d'obtenir une contradiction, montrons que T_n^* est consistant pour tout n.

Cas de base : $n = 0$
Vu que $T_0^* = T$, on a trivialement

$$\mathcal{P} \cup Cons(T) \vdash Cons(T_0^*)$$

Cas d'induction : Supposons que T_n^* est consistant et montrons que T_{n+1}^* l'est également.
Montrons que

$$\mathcal{P} \cup Cons(T) \vdash Cons(T_n^*) \Rightarrow Cons(T_{n+1}^*)$$

Par l'absurde, supposons que T_{n+1}^* est inconsistant.

$$\mathcal{P} \cup Cons(T) \cup Cons(T_n^*) \vdash \neg Cons(T_{n+1}^*)$$

Or, $T_{n+1}^* = T_n^* \cup H_{n+1}$. Vu le théorème de démonstration par l'absurde dans Peano, 2.5.10, on a

$$\mathcal{P} \cup Cons(T) \cup Cons(T_n^*) \vdash \mathcal{W}_{T_n^*}[\#\neg H_{n+1}]$$

Dans ce cas, $\mathcal{P} \cup Cons(T) \cup Cons(T_n^*) \vdash \mathcal{W}_{T_n^*}[\#(\neg(\neg\forall x \neg F \Rightarrow F(x := c)))]$.
On a donc que

$\mathcal{P} \cup Cons(T) \cup Cons(T_n^*) \vdash \mathcal{W}_{T_n^*}[\#\neg\forall x \neg F]$ et $\mathcal{P} \cup Cons(T) \cup Cons(T_n^*) \vdash \mathcal{W}_{T_n^*}[\#\neg(F(x := c))]$

Remarquons que, par construction, $\#c > \#F$, c'est-à-dire que c n'apparaît pas dans la formule F. Par la règle de généralisation, on a que

$$\mathcal{P} \cup Cons(T) \cup Cons(T_n^*) \vdash \mathcal{W}_{T_n^*}[\#\forall c \neg F(x := c)]$$

Ce qui implique par le théorème précédent 2.5.13

$$\mathcal{P} \cup Cons(T) \cup Cons(T_n^*) \vdash \mathcal{W}_{T_n^*}[\#\forall x \neg F].$$

Par conséquent, on a

$\mathcal{P} \cup Cons(T) \cup Cons(T_n^*) \vdash \mathcal{W}_{T_n^*}[\#\neg\forall x \neg F]$ et $\mathcal{P} \cup Cons(T) \cup Cons(T_n^*) \vdash \mathcal{W}_{T_n^*}[\#\forall x \neg F]$.

Ce qui implique, vu la proposition 2.5.9, que $\mathcal{P} \cup Cons(T) \cup Cons(T_n^*) \vdash \neg Cons(T_n^*)$.
Ce qui est absurde. Par conséquent, tous les T_n^* sont consistant et donc Th est consistant.
■

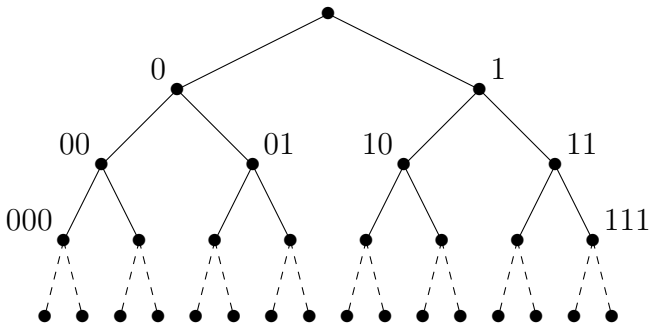
Il nous reste à déterminer, de manière arithmétique, une extension maximale consistante de la théorie Th.
Soit $K_0 \subset K$, où K est l'ensemble des codes des suites finies d'entiers (voir 2.3.2), l'ensemble des codes des suites composées uniquement de 0 et de 1.

$$a \in K_0 \text{ si et seulement si } (a \in K \wedge \forall i < lg(a)[\delta(i,a) = 0 \vee \delta(i,a) = 1])$$

Cet ensemble est représenté par $K_0[x]$ tel que l'ensemble $K_0^M := \{a \in M : \mathfrak{M} \vDash K_0[a]\}$ consiste en l'ensemble de toutes les séquences $a \in M$ dont tous les termes sont 0 ou 1.
K_0^M est partiellement ordonné par la relation

$$a \leq_0 b \text{ si et seulement si } (lg(a) \leq lg(b) \wedge \forall i < l(a)[\delta(i,a) = \delta(i,b)])$$

Le couple $< K_0, \leq_0 >$ est donc un arbre binaire et les $a \in K_0$ sont les sommets de cet arbre.



Soit la fonction f définie telle que :

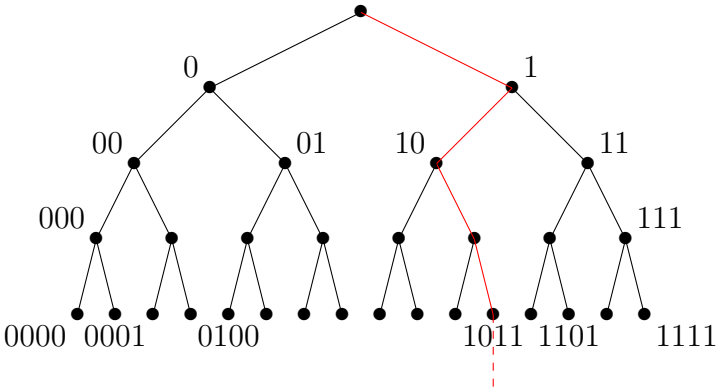
$$\begin{cases} f(0) = \min \mathcal{F}orm \\ f(n+1) = \min\{F \leq Neg(f(n)) : F \in \mathcal{F}orm \wedge (F > f(n))\} \end{cases}$$

est totale, croissante et récursive. De plus, on a que, pour tout modèle $\mathfrak{M}$, f énumère toutes les formules i.e. $\mathcal{F}orm^M = \{f^M(m) : m \in M, m > 0\}$.
D'où, la fonction g, définie comme

$$\begin{cases} g(a) = 1 \text{ si } a = 1. \\ g(a) = f(l(a)) \text{ si } \delta(lg(a) - 1, a) = 0. \\ g(a) = Neg(f(l(a))) \text{ si } \delta(lg(a) - 1, a) = 1. \\ g(a) = 0 \text{ si } a \notin K_0. \end{cases}$$

est totale récursive.

Définition 2.5.15. Une formule $\Gamma[x]$ est appellée *branche* pour un modèle $\mathfrak{M}$ si l'ensemble $\Gamma^M = \{a : \mathfrak{M} \vDash \Gamma[a]\}$ est un sous-ensemble ordonné linéaire de K_0^M par $\leq_0$ tel que pour tout naturel n, il existe une séquence $a \in \Gamma^M$ de longueur n.
Une branche $\Gamma[x]$ est consistante (selon $\mathfrak{M}$) si l'ensemble $S_\Gamma^M = \{g^M(a) : a \in \Gamma^M \wedge a > 1\}$ est consistant.



La suite $1, 10, 101, 1011, \ldots$ est une branche.

Proposition 2.5.16. *Si l'ensemble S_Γ^M est consistant, alors il est maximal consistant.*

Démonstration. Soit une formule F n'appartenant pas à S_Γ^M, montrons que S_Γ^M contient $\neg F$.

Vu que f énumère l'ensemble des formules, il existe un naturel c tel que $f(c) = F$. Or, par définition, Γ contient des suites de longueur c.
Soit a une suite de longueur c appartenant à K_0, elle finit donc par 0 ou par 1.
Si elle finit par 0, on a $g(a) = f(l(a)) = f(c) = F$.
On en conclut que Γ ne contient aucune suite de longueur c finissant par 0.

Par conséquent, toutes les suites de longueur c dans Γ finissent par 1 et on a

$$g(a) = Negf(l(a)) = Negf(c) = \neg F.$$

■

S_Γ^M est représenté dans $\mathfrak{M}$ par la formule $S_\Gamma[x] \equiv \exists a(\Gamma[a] \wedge \Gamma_g[a, x])$. Où Γ_g représente g dans $\mathcal{P}$.

Lemme 2.5.17. *Pour toute formule $T[x]$, il existe une formule Γ telle que si $\mathfrak{M} \vDash Cons(T)$ alors Γ est une branche consistante dans $\mathfrak{M}$ et la formule S_Γ représente dans $\mathfrak{M}$ une extension maximale consistante de la théorie Th^M.*

Démonstration. Soit la fonction

$$\bigwedge(a) := \bigwedge_{i<l(a)} \delta(i,a) = \delta(0,a) \wedge \cdots \wedge \delta(lg(a)-1,a)$$

qui à une séquence de formules associe la conjonction de ces dernières. Elle est récursive et totale dans $\mathcal{P}$.
On pose $\Gamma[a]$ la formule [8] telle que

$$K_0[a] \wedge \{a = 1 \vee [a > 1 \wedge \neg W_{Th}(\neg \bigwedge_{i<lg(a)} g(a|(i+1)))$$
$$\wedge \forall j < lg(a)[\neg W_{Th}(\bigwedge_{i<j} g(a|(i+1)) \Rightarrow \neg f(j)) \Leftrightarrow \delta(j,a) = 0]]\}$$

D'où l'ensemble Γ^M consiste en les séquences $a \in K_0^M$ pour lesquelles la conjonction des formules correspondantes aux restrictions $a|i$ est consistante avec Th^M et, de plus, pour tout $j < l(a)$, nous posons $\delta(j,a) = 0$ si et seulement si la formule $\#F = f(j)$ est consistante avec $Th^M \cup \{\bigwedge_{i<j} g(a|(i+1))\}$.
En effet, par le théorème 2.5.10 et le lemme de déduction 2.5.7, on a que $\#F = f(j)$ est consistante avec $Th^M \cup \{\bigwedge_{i<j} g(a|(i+1))\}$ si et seulement si $\neg W_{Th}(\bigwedge_{i<j} g(a|(i+1)) \Rightarrow \neg F)$.

Supposons que $\mathfrak{M} \vDash Cons(T)$ et montrons que Γ est une branche consistante dans $\mathbb{M}$.
Tout d'abord, montrons que si $a \in \Gamma^M$ alors soit $a*(0) \in \Gamma^M$ soit $a*(1) \in \Gamma^M$.
Supposons que $\Gamma[a]$, posons $k \in \{0,1\}$ et montrons qu'on a $\Gamma[a*(k)]$.
On a bien $a*(k) \in K_0$, $a*(k) > 1$ et $lg(a*(k)) = lg(a)+1$.
Il faut donc montrer que

$$\{\neg W_{Th}(\neg \bigwedge_{i<l(a)+1} g(a*(k)|(i+1))) \tag{2.3}$$
$$\wedge \forall j < l(a)+1[\neg W_{Th}(\bigwedge_{i<j} g(a*(k)|(i+1)) \Rightarrow \neg f(j)) \Leftrightarrow \delta(j,a*(k)) = 0]\} \tag{2.4}$$

Vu que a est dans Γ, on a que $Th \cup \bigwedge_{i<l(a)} g(a|(i+1))$ est consistant.
Vu la proposition 2.5.12, on a que

- Soit $Th \cup \bigwedge_{i<l(a)} g(a|(i+1)) \cup \{f(l(a)+1)\}$ est consistante.
- Soit $Th \cup \bigwedge_{i<l(a)} g(a|(i+1)) \cup \{Negf(l(a)+1)\}$ est consistante.

Donc on a soit $k = 1$ soit $k = 0$.

8. Pour être formel, nous devrons utiliser la formule du premier ordre représentant la fonction g, cependant nous nous sommes abstenu dans un souci de clarté

Pour le second terme (3.2), on a que $f(l(a))$ est consistante avec le reste si et seulement si il n'existe pas de démonstration dans Th de

$$\bigwedge_{i<lg(a)-1} g(a*(k)|(i+1)) \wedge g(a*(k)|(lg(a))) \Rightarrow \neg f(lg(a)).$$

Ce qui est le cas si et seulement si $g(a*(k)|(lg(a))) = f(lg(a))$ i.e. si et seulement si $\delta(lg(a), a*(k)) = 0$.

D'où, par le théorème 2.5.14 et par induction, on a

$$PA \cup Cons(T) \vdash \forall x \exists a[l(a) = x \wedge \Gamma(a)]$$

En effet, soit $F[y]$ définie comme $\exists a[l(a) = y \wedge \Gamma(a)]$.

Cas de base : $F[1] \equiv \exists a[l(a) = 1 \wedge \Gamma(a)]$.
En utilisant la définition de Gamma et l'axiome d'Henkin, on voit que $a = 1$ convient.

Cas d'induction : Supposons que cela soit vérifié pour $F[n]$, montrons-le pour $F[n+1]$.
C'est évident vu que si $a \in \Gamma$ alors soit $a*(0) \in \Gamma$ soit $a*(1) \in \Gamma$.
Vu précédemment, on a que pour une longueur donnée, la suite correspondante est univoquement déterminée et que si $a, b \in \Gamma$ et $l(a) \leq^M l(b)$, alors $a \leq_0^M b$.
On a donc que Γ est bien une branche, et vu la définition de Γ, on a qu'elle est consistante.
Il nous reste à montrer que $Th^M \subset S_\Gamma^M$ pour tout modèle $\mathfrak{M}$ tel que $\mathfrak{M} \vDash Cons(T)$.

Soit $F \in Th$. Il existe donc un k tel que $F = f(k)$ et, comme vu avant, il existe un a tel que $lg(a) = k$ et $a \in \Gamma$. Donc $g(a)$ est dans S_Γ et $g(a)$ est soit $f(k)$ soit $Negf(k)$ mais vu la définition de Γ, la conjonction des $g(a|(i+1))$ doit être consistante avec Th donc $g(a) = f(k)$. Par conséquent, toute formule de Th appartient à S_Γ. ■

Définition 2.5.18. Soit un ensemble de formules $\mathcal{H} =< H_U, \{H_r\}_{r\in Rel}, \{H_f\}_{f\in Fonc}, \{H_c\}_{c\in C} >$, on dit qu'une structure $\mathbb{U} =< U, \{r^U\}_{r\in Rel}, \{f^U\}_{f\in Fonc}, \{c^U\}_{c\in C} >$ est représentable dans le modèle $\mathfrak{M}$ de $\mathcal{P}$ si

- l'univers U est l'ensemble $\{a \in M : \mathfrak{M} \vDash H_U[a]\}$,
- chaque fonction f^U est représentée par une formule H_f,
- chaque relation r^U est représentée par une formule H_r,
- chaque constante c^U est représentée par une formule H_c.

Théorème 2.5.19. *Soit T une théorie récursive consistante dans un langage fini $\mathcal{L}(T)$. Alors il existe un système de formules*

$$\mathcal{H} =< H_U, \{H_r\}_{r\in Rel}, \{H_f\}_{f\in Fonc}, \{H_c\}_{c\in C} >$$

telles que pour tout modèle $\mathfrak{M}$ de $\mathcal{P}$ satisfaisant $Cons(T)$, $\mathcal{H}$ représente un modèle $\mathbb{U}$ de T

Démonstration. Si $\mathfrak{M} \vDash Cons(T)$ alors vu le lemme, on a une formule S représentant dans $\mathfrak{M}$ une extension maximale consistante de la théorie Th^M.

On définit un modèle de termes fermés de la façon suivante :
Soit $H_U[x]$ la formule représentant l'ensemble des termes fermés. Cet ensemble étant primitif récursif, vu 2.3.10, il est bien représentable par une formule $\overline{Term}[x]$.

On a donc que

$$U = \{m \in M : \mathfrak{M} \vDash H_U[m]\}$$

On définit les relations r^U comme suit :

$$(z_1, \ldots, z_n) \in r^U \text{ si et seulement si } \alpha_3(\Omega(z_1, \ldots, z_n), r, 0) \in S^M \text{ pour } z_1, \ldots, z_n \in U$$

A présent, définissons les opérations f^U comme suit

$$f^U(z_1, \ldots, z_n) = \alpha_3(\Omega(z_1, \ldots, z_n), f, 2) \text{ pour } z_1, \ldots, z_n \in U$$

Enfin, les constantes c^U sont définies par $c^U = \#c$.
Nous remarquons que chaques fonctions f, relations r et constantes c peuvent être représentées par des formules H_r, H_f, H_c construites à partir de S.
On a le système

$$\mathbb{U} =< U, \{r^U\}_{r\in Rel}, \{f^U\}_{f\in Fonc}, \{c^U\}_{c\in C} >$$

Prouvons l'équivalence

$$\mathbb{U} \vDash F \text{ si et seulement si } F \in S^M \text{ pour des formules fermées } F$$

On vérifie par induction sur les formules.

a. $\mathbb{U} \vDash r(t_1, \ldots, t_n)$ ssi $(t_1, \ldots, t_n) \in r^U$ si et seulement si $\alpha_3(\Omega(t_1, \ldots, t_n), r, 0) \in S^M$.
b. Vu que S est maximal consistant, on a que $\neg F \in S^M$ si et seulement si $F \notin S^M$. Par conséquent, $\mathbb{U} \vDash \neg F$ si et seulement si $\mathbb{U} \nvDash F$ [9] si et seulement si $F \notin S^M$ par hypothèse de récurrence. Et $F \notin S^M$ si et seulement si $\neg F \in S^M$.

9. vu que F est une formule fermée

c. $\mathbb{U} \vDash F \Rightarrow G$ si et seulement si $\mathbb{U} \nvDash F$ ou $\mathbb{U} \vDash G$ si et seulement si $F \notin S^M$ ou $G \in S^M$ si et seulement si $(F \Rightarrow G) \in S^M$.

d. Il nous reste à traiter le cas du quantificateur universel :
$\mathbb{U} \nvDash \forall x F$ si et seulement si $\mathbb{U} \vDash \exists x \neg F$ si et seulement si $\mathbb{U} \vDash_\tau \neg F$ où τ est une interprétation de x.
Soit $z = x_\tau$, on a $\mathbb{U} \vDash_\tau \neg F$ si et seulement si $\mathbb{U} \vDash \neg F(x := z)$ si et seulement si $\mathbb{U} \nvDash F(x := z)$ si et seulement si $F(x := z) \notin S^M$.
Or, vu que S est une extension de T, on a que $\forall x F \in S^M$ implique que $F(x := z) \in S^M$.
Par contraposée, on a bien que $F(x := z) \notin S^M$ implique que $\forall x F \notin S^M$.
Réciproquement, si $\forall x F \notin S^M$ alors $\neg \forall x F \in S^M$ implique que $\neg F(x := z) \in S^M$ par l'axiome d'Henkin.
On a donc bien que $\mathbb{U} \nvDash \forall x F$ si et seulement si $\forall x F \notin S^M$, d'où la conclusion.

■

2.6 Le deuxième théorème d'incomplétude

A présent que toutes les difficultés techniques sont passées, nous allons arriver véritablement à l'énoncé du deuxième théorème d'incomplétude de Gödel.
Il nous reste cependant quelques résultats préliminaires à établir.

Lemme 2.6.1. *Soient $\mathfrak{M}_1, \mathfrak{M}_2$ deux modèles de $\mathcal{P}$.*
Si $\mathfrak{M}_2$ est représentable dans $\mathfrak{M}_1$ alors $\mathfrak{M}_1$ est plongeable dans $\mathfrak{M}_2$ comme un segment initial de $\mathfrak{M}_2$

Démonstration. Vu que $\mathfrak{M}_2$ est représentable dans $\mathfrak{M}_1$, on a que $0^{M_2}, 1^{M_2}, +^{M_2}, \times^{M_2}$ sont représentables dans $\mathfrak{M}_1$. Soit une fonction représentable $f : \mathfrak{M}_1 \to \mathfrak{M}_2$ telle que

$$f(0^{M_1}) = 0^{M_2} \text{ et } f(a +^{M_1} 1) = f(a) +^{M_2} 1^{M_2} \text{ pour } a \in \mathfrak{M}_1.$$

Montrons que f est monotone :
En effet, pour tout $a \in \mathfrak{M}_1$, considérons l'ensemble

$$Z(a) = \{c \in M_1 : f(a) <_{M_2} f(a + c + 1)\}$$

Vu que $\mathfrak{M}_2, <$ et f sont représentables dans $\mathfrak{M}_1$, l'ensemble $Z(a)$ est également représentable dans $\mathfrak{M}_1$. De plus, il contient 0^{M_1}.

En effet, $f(a) <_{M_2} f(a+1)$ si et seulement si

$$\exists d \in M_2 \text{ tel que } f(a) +^{M_2} d = f(a+1) = f(a) +^{M_2} 1^{M_2}.$$

On a que $d = 1^{M_2}$ convient.
Supposons que $c \in Z(a)$ et montrons que $c+1 \in Z(a)$.
En effet, $f(a) <_{M_2} f(a+c+1) <_{M_2} f(a+c+1+1)$ où la deuxième inégalité est due au fait que $0 \in Z(a+c+1)$.
On en conclut bien que f est monotone en appliquant le *SI* sur la formule représentant cet ensemble.
Vu que f est monotone, elle est injective.

Montrons à présent que f préserve l'addition.
Soit l'ensemble

$$Z'(a) = \{b \in M_1 : f(a +^{M_1} b) = f(a) +^{M_2} f(b)\}.$$

Cet ensemble contient de nouveau 0^{M_1} vu que $f(0 +^{M_1} b) = f(b) = 0^{M_2} +^{M_2} f(b)$ et que $\mathfrak{M}_2$ est un modèle de $\mathcal{P}$.
Supposons que $b \in Z'(a)$ et montrons que $b+1 \in Z'(a)$.
Il suffit de remarquer que $f(a+b+1) = f(a+b)+1 = f(a)+f(b)+1 = f(a)+f(b+1)$.
En utilisant *SI*, on conclut que f préserve l'addition.

Montrons que f préserve la multiplication. Soit l'ensemble

$$Z''(a) = \{b \in M_1 : f(a \times^{M_1} b) = f(a) \times^{M_2} f(b)\}.$$

Les mêmes arguments que pour l'ensemble précédent conviennent en remarquant que

$$\begin{aligned} f(a \times^{M_1} (b +^{M_1} 1)) &= f(a \times^{M_1} b +^{M_1} a) = f(a \times^{M_1} b) +^{M_2} f(a) \\ &= f(a) \times^{M_2} f(b) +^{M_2} f(a) \\ &= f(a) \times^{M_2} (f(b) +^{M_2} 1^{M_2}) \\ &= f(a) \times^{M_2} (f(b) +^{M_2} f(1^{M_1})) \\ &= f(a) \times^{M_2} f(b +^{M_1} 1). \end{aligned}$$

On a donc que f préserve la multiplication.

Pour finir, montrons que $f(\mathfrak{M}_1)$ est un segment initial i.e. montrons que si $b \in M_2$ et

$b \leq_{M_2} f(a)$ pour $a \in M_1$ alors $b \in f(M_1)$. Pour un b répondant aux hypothèses, posons

$$Z'''(b) = \{a \in M_1 : b \leq^{M_2} f(a)\} \text{ est représentable.}$$

Soit $a_0 = \min Z'''(b)$. Deux cas sont possibles :
Si $a_0 = 0$ alors $b \leq^{M_2} f(0^{M_1}) = 0^{M_2}$ et donc $b = 0^{M_2} = f(0^{M_1})$.
Si $a_0 > 0$; alors $f(a_0 - 1) < b \leq f(a_0)$ et donc, $b = f(a_0)$.
La conclusion en découle. ■

Dans la suite, nous noterons $\mathfrak{M}(\mathcal{H})$ le modèle représenté dans $\mathfrak{M}$ par le système de formules $\mathcal{H} =< H_U, \{H_r\}_{r \in Rel}, \{H_f\}_{f \in Fonc}, \{H_c\}_{c \in C} >$.

Tous ces résultats techniques nous permettent, à présent, d'obtenir le résultat suivant :

$$\text{Si } F \text{ est une formule close } \Sigma, \text{ alors } \mathcal{P} \vdash F \Rightarrow \exists v_1 \mathfrak{Dem}_0[\#F, v_1].$$

qui est en réalité une des *trois conditions d'Hilbert-Bernays-Löb* [10].
Établissons tout d'abord un résultat général sur les formules Σ.

Lemme 2.6.2. *Soit $\mathfrak{N}$ une $\mathcal{L}_0$-structure, $\mathfrak{M}$ une extension finale de $\mathfrak{N}$, $F[v_1, \ldots, v_p]$ une formule Σ et $a_1, a_2, \ldots, a_p$ des points de $\mathfrak{N}$. Alors, $\mathfrak{N} \vDash F[a_1, \ldots, a_p]$ implique $\mathfrak{M} \vDash F[a_1, \ldots, a_p]$*

Démonstration. Soit $\mathfrak{A}$ l'ensemble des formules G qui sont telles que, pour p le nombre de variables libres de G, pour tout $a_1, \ldots, a_p$ de $\mathfrak{N}$, $\mathfrak{N} \vDash F[a_1, \ldots, a_p]$ implique $\mathfrak{M} \vDash F[a_1, \ldots, a_p]$.
Montrons, par récurrence sur le nombre de connecteurs, que cet ensemble contient toutes les formules Σ.

- Vu les définitions 1.1.15 et 1.1.16, les formules atomiques appartiennent bien à $\mathfrak{A}$.
- Clos par l'implication. Soient $H, H' \in \mathfrak{A}$ et montrons que $H \Rightarrow H' \in \mathfrak{A}$, on a que $\mathfrak{N} \vDash H \Rightarrow H'$ si et seulement si $\mathfrak{N} \nvDash H$ ou $\mathfrak{N} \vDash H'$.
  Or, $\mathfrak{N} \vDash H'$ implique $\mathfrak{M} \vDash H'$, ce qui implique $\mathfrak{M} \vDash H \Rightarrow H'$.

10. Les conditions de prouvabilité d'Hilbert-Bernays, sont un ensemble de conditions, portant sur les formules exprimant la prouvabilité dans des théories formelles de l'arithmétique, utilisé dans la majorité des preuves du second théorème d'incomplétude.
Les conditions de prouvabilité d'Hilbert-Bernays-Löb sont, pour toutes formules F et G :
1. Si T prouve F alors T prouve $\mathcal{W}_T[\#F]$.
2. $T \vdash \mathcal{W}_T[\#(F \Rightarrow G)] \Rightarrow (\mathcal{W}_T[\#F] \Rightarrow \mathcal{W}_T[\#G])$.
3. $T \vdash \mathcal{W}_T[\#F] \Rightarrow \mathcal{W}_T[\#(\mathcal{W}_T[\#F])]$.

- Clos par conjonction et disjonction. Montrons pour la disjonction, nous remarquerons que c'est analogue pour la conjonction. Soient $H, H^{'} \in \mathfrak{A}$, on a
  $\mathfrak{N} \vDash H \vee H^{'}$ si et seulement si $\mathfrak{N} \vDash H$ ou $\mathfrak{N} \vDash H^{'}$. Ce qui implique $\mathfrak{M} \vDash H$ ou $\mathfrak{M} \vDash H^{'}$ qui est équivalent à $\mathfrak{M} \vDash H \vee H^{'}$.
  Par conséquent, $H \vee H^{'} \in \mathfrak{A}$.
- Clos par quantificateur existentiel car $\mathfrak{N}$ est une sous-structure de $\mathfrak{M}$.
  En effet, soit $H \in \mathfrak{A}$ on a
  $\mathfrak{N} \vDash_{\tau} \exists x H$ si et seulement si $\mathfrak{N} \vDash_{\tau'} H$. Ce qui implique $\mathfrak{M} \vDash_{\tau'} H$ qui est équivalent à $\mathfrak{M} \vDash_{\tau} \exists x H$ car $x \in N \subset M$.
  où $\tau^{'}$ est une interprétation $v_1, \ldots v_p, x$ étendant τ.
- Clos par quantificateur universel borné :
  Supposons que $H \in \mathfrak{A}$ et que $\mathfrak{N} \vDash_{\tau} \forall (x < y) H$.
  On a $\mathfrak{M} \vDash_{\tau} \forall (x < y) H$ si et seulement si $\mathfrak{M} \vDash_{\tau'} H$ pour toute interprétation $\tau^{'}$ de $v_1, \ldots v_n, x$ étendant τ. Or, $\mathfrak{M}$ est une extension finale de $\mathfrak{N}$ et $y \in N$.
  Donc, $x < y$ entraîne que $x \in N$.
  La conclusion en découle.

■

Proposition 2.6.3. *Soit F une formule close Σ de $\mathcal{L}_0$. Alors*

$$\mathbb{N} \vDash F \Rightarrow \exists v_1 \mathfrak{Dem}_0[\underline{\#F}, v_1]$$

En particulier, si F est une formule close Σ, $\mathbb{N} \vDash F$ si et seulement si $\mathcal{P} \vdash F$.

Démonstration. Si F est fausse dans $\mathbb{N}$ alors la propriété est vraie.
Supposons que F est vraie dans $\mathbb{N}$ et montrons qu'elle est démontrable dans $\mathcal{P}_0$. Par le théorème de complétude, il suffit de montrer qu'elle est vraie dans tout modèle.
Mais tout modèle de $\mathcal{P}_0$ est extension finale de $\mathbb{N}$. Vu le lemme précédent, on a la conclusion.

■

Posons $\mathcal{P}_1 := \mathcal{P}_0 \cup \{F \Rightarrow \exists v_1 \mathfrak{Dem}_0(\underline{\#F}, v_1);\ F \text{ est une formule close } \Sigma\}$.
Vu la propriété précédente, $\mathbb{N}$ est modèle de $\mathcal{P}_1$. Nous pouvons remarquer que $\mathcal{P}_1$ est une théorie récursive.

Lemme 2.6.4. *Soit F une formule close Σ de $\mathcal{L}_0$. Alors*

$$\mathcal{P} \vdash F \Rightarrow \exists v_1 \mathfrak{Dem}_0[\#F, v_1]$$

Démonstration.
Utilisons le théorème de complétude classique i.e. montrons que $F \Rightarrow \exists v_1 \mathfrak{Dem}_0[\#F, v_1]$ est

valide dans tout modèle de $\mathcal{P}$.

Soit un modèle $\mathfrak{M}$ de $\mathcal{P}$.
Si $\exists v_1 \mathfrak{Dem}_0[\#F, v_1]$ est vraie dans $\mathfrak{M}$ alors $F \Rightarrow \exists v_1 \mathfrak{Dem}_0[\#F, v_1]$ y est également vraie.
Sinon, soit G la conjonction de $\neg F$ et de toutes les formules de $\mathcal{P}_0$. Vu le théorème de la démonstration par l'absurde dans Peano, nous avons que $\neg\exists v_1 \mathfrak{Dem}_0[\#F, v_1]$ est équivalent à $Cons(G)$.
Par conséquent,

$$\mathfrak{M} \vDash Cons(G)$$

On applique le théorème de complétude de Peano à G.
Il existe donc un système de formules $\mathcal{H}$ représentant un modèle $\mathfrak{M}(\mathcal{H})$ tel que

$$\mathfrak{M}(\mathcal{H}) \vDash G$$

Vu le lemme 2.6.1, on a que la structure $\mathfrak{M}(\mathcal{H})$ est une extension finale d'une structure isomorphe à $\mathfrak{M}$. Or $\mathfrak{M}(\mathcal{H}) \nvDash F$, et, vu le lemme 2.6.2, $\mathfrak{M} \nvDash F$.
Par conséquent,

$$\mathfrak{M} \vDash F \Rightarrow \exists v_1 \mathfrak{Dem}_0[\#F, v_1]$$

et cela pour tout modèle de $\mathcal{P}$. ■

Lemme 2.6.5. *Soit T une théorie récursive, consistante qui démontre toutes les formules de $\mathcal{P}_1$.*
Alors T ne démontre pas $Cons(T)$.

Démonstration. Soit la fonction g de $\mathbb{N}$ dans $\mathbb{N}$ définie par :

- Si n est le numéro de Gödel d'une formule $F[v_0]$ à une variable libre alors $g(n)$ est le numéro de Gödel de la formule $F[\underline{n}]$.
- Sinon $g(n) = 0$.

La fonction g est donc primitive récursive, c'est en fait la fonction $Subst_f(0, n, n)$. Vu le deuxième théorème de représentation, il existe donc $G[v_0, v_1]$ une formule Σ qui représente g. On a donc pour tout n,

$$\mathcal{P}_0 \vdash \forall v_0(G[v_0, \underline{n}] \Leftrightarrow v_0 \simeq \underline{g(n)}) \tag{2.5}$$

Soit $\varepsilon[v_0]$ la formule Σ définie par

$$\exists v_1 \exists v_2(\mathfrak{Dem}[v_2, v_1] \wedge G[v_2, v_0])$$

Vu la définition de ε et de G, on a que si n est le numéro de Gödel d'une formule $F[v_0]$ à une variable libre alors

$$\mathbb{N} \vDash \varepsilon[\underline{n}] \Leftrightarrow F[\underline{n}] \text{ est dénombrable}$$

Soient a le numéro de Gödel de la formule $\neg\varepsilon[v_0]$ et $b = g(a)$ le numéro de Gödel de $\neg\varepsilon[a]$. Vu la définition de ε et de 2.5, on a

$$\mathcal{P}_0 \vdash \varepsilon[\underline{a}] \Leftrightarrow \exists v_1 \mathfrak{Dem}[\underline{b}, v_1] \tag{2.6}$$

Montrons que T ne démontre pas $\neg\varepsilon[\underline{a}]$.
Si on suppose le contraire, cela signifie qu'il existe un entier c code d'une démonstration de $\neg\varepsilon[\underline{a}]$ dans T i.e.

$$\mathcal{P}_0 \vdash \mathfrak{Dem}[\underline{b}, \underline{c}]$$

Dans ce cas, vu 2.6, T démontre $\neg\varepsilon[\underline{a}]$ et $\varepsilon[\underline{a}]$. T sera donc inconsistante ce qui est en contradiction avec nos hypothèses.
Montrons que $T \vdash Cons(T) \Rightarrow \neg\varepsilon[\underline{a}]$.
Soit $T_1 = \mathcal{P}_1 \cup \varepsilon[\underline{a}]$. De 2.6, on obtient

$$T_1 \vdash \exists v_1 \mathfrak{Dem}[\underline{b}, v_1]$$

Or, $\varepsilon[\underline{a}]$ est une formule close Σ.
Notons d le numéro de Gödel de $\varepsilon[\underline{a}]$. Alors

$$(\varepsilon[\underline{a}] \Rightarrow \exists v_2 \mathfrak{Dem}_0[\underline{d}, v_2]) \in \mathcal{P}_1 \text{ et } T_1 \vdash \exists v_2 \mathfrak{Dem}_0[\underline{d}, v_2].$$

Or, on a que $\forall v_0 \forall v_1 (\mathfrak{Dem}_0[v_0, v_1] \Rightarrow \mathfrak{Dem}[v_0, v_1])$ et donc

$$T_1 \vdash \exists v_1 \mathfrak{Dem}[\underline{b}, v_1] \wedge \exists v_2 \mathfrak{Dem}[\underline{d}, v_2]$$

On a donc que

$$T_1 \vdash \neg Cons(T)$$

et vu le lemme de déduction

$$\mathcal{P}_1 \vdash \varepsilon[a] \Rightarrow \neg Cons(T)$$

On a donc que $T \vdash Cons(T) \Rightarrow \neg\varepsilon[\underline{a}]$ par contraposée.
Par conséquent, si $Cons(T)$ était démontrable, on aurait $T \vdash \neg\varepsilon[\underline{a}]$ et on a déjà montré que cela est impossible, d'où la conclusion. ■

Théorème 2.6.6 (Deuxième théorème d'incomplétude de Gödel)**.** *Soit T une théorie*

consistante, récursive et contenant $\mathcal{P}$. Alors T ne démontre pas $Cons(T)$.

Démonstration. Par le lemme 2.6.4, toute formule de $\mathcal{P}_1$ est conséquence de $\mathcal{P}$.
Par conséquent, $\mathcal{P}_1 = \mathcal{P}$.
La conclusion découle du lemme précédent. ■

Proposition 2.6.7. *Les formules $\neg\exists v_0 \mathfrak{Dem}[\#(\underline{0} \simeq \underline{1}), v_0]$ et $Cons(T)$ sont équivalentes.*

Démonstration. Supposons que $\mathcal{P} \vdash \neg Cons(T)$.
Vu la propriété 2.5.8, on a que

$$\mathcal{P} \vdash \neg Cons(T) \Rightarrow \mathcal{W}_T[\#(\underline{0} \simeq \underline{1})].$$

Réciproquement, supposons que $\mathcal{P} \vdash Cons(T)$.
Par définition de la formule $Cons(T)$, on a donc

$$\mathcal{P} \vdash \forall v_0 \forall v_1 (\neg \mathcal{W}_T[v_0] \vee \neg \mathcal{W}_T[v_1] \vee \neg \mathfrak{Neg}[v_0, v_1]).$$

En particulier, on a que $\mathcal{P} \vdash \neg(\underline{0} \simeq \underline{1})$.
Par conséquent, $\mathcal{P} \vdash \neg \mathcal{W}_T[\#(\underline{0} \simeq \underline{1})]$, d'où la conclusion. ■

De cette manière, nous obtenons de manière évidente cette formulation originale du second théorème d'incomplétude.

Corollaire 2.6.8. *Si $0 = 1$ est indémontrable dans T alors "$0 = 1$ est indémontrable dans T" est indémontrable dans T.*

2.7 Théorèmes de Löb et de Tarski

Cette brève section clôturera ce deuxième chapitre en démontrant les théorèmes de Löb et de non-définissabilité de Tarski.

Par le théorème de complétude, nous savons qu'une formule F est prouvable dans $\mathcal{P}$ si et seulement si elle est vraie. En particulier, si une formule est prouvable alors elle est vraie.
Le théorème de Löb démontre que cette assertion, une fois exprimée dans Peano, peut être lourde de conséquences.

Théorème 2.7.1 (Théorème de Löb)**.** *Si F est une formule close et si*

$$\mathcal{P} \vdash \exists v_0 \mathfrak{Dem}[\underline{\#F}, v_0] \Rightarrow F$$

alors $\mathcal{P} \vdash F$

Démonstration. Supposons que $\mathcal{P} \vdash \mathcal{W}_T[\underline{\#F}] \Rightarrow F$.
Par contraposée et par le lemme de déduction, nous avons,

$$\mathcal{P} \cup \{\neg F\} \vdash \neg \mathcal{W}_T[\underline{\#F}, v_0].$$

Or, vu le théorème de démonstration par l'absurde dans Peano, 2.5.10, on a donc que

$$\mathcal{P} \cup \{\neg F\} \vdash Cons(\mathcal{P}; \neg F).$$

Or, dans ce cas, vu le second théorème de Gödel, $\mathcal{P} \cup \{\neg F\}$ est inconsistante.
Donc, $\mathcal{P} \vdash F$ ■

Prenons à présent le cas particulier $0 = 1$. L'assertion « S'il existe une démonstration de $0 = 1$, alors $0 = 1$ » nous parait logiquement valide et pourtant, si cette dernière est un théorème de $\mathcal{P}$, cela signifie que $0 = 1$ est vrai dans Peano !

Nous allons à présent donner la démontration du théorème de Tarski énonçant qu'on ne peut définir la notion de vérité à l'intérieur d'une théorie T.

Lemme 2.7.2 (Lemme diagonal)**.** *Soient T une théorie récursive contenant $\mathcal{P}_0$ et $G[v_0]$ une formule ayant au plus une variable libre. Alors, il existe une formule close F telle que*

$$T \vdash F \Leftrightarrow G[\#F]$$

Démonstration. Soit g la fonction définie dans le lemme 2.6.5.
Vu le théorème de représentation, il existe une formule $\Gamma_g[x, y]$ représentant g.
Soit $H[x]$ la formule $\exists v_0(\Gamma_g[v_0, x] \wedge G[v_0])$.
On a que si v_0, code d'une formule V, satisfait la formule H, cela signifie que $\#V[v_0]$ satisfait la formule G.
Posons F comme étant la formule $H[\#H]$.
Vu que $\#F = \#H[\#H] = g(\#H)$, on a que

$$(1) \quad T \vdash \Gamma_g[\#F, \#H].$$

Ensuite, vu (1) et vu que F est la formule $\exists v_0(\Gamma_g[v_0, \#H] \wedge G[v_0])$, on a,

$$T \cup \{F\} \vdash G[\#F].$$

Par le même raisonnement, on obtient que

$$T \cup \{G[\#F]\} \vdash F.$$

Et on conclut par le lemme de déduction. ■

Théorème 2.7.3 (Théorème de Tarski)**.** *Si T est une théorie récursive contenant $\mathcal{P}_0$ et $\mathfrak{M}$ est un modèle de T, alors l'ensemble $T(\mathfrak{M}) = \{F : \mathfrak{M} \vDash F\}$ des formules valides dans $\mathfrak{M}$ n'est pas représentable dans $\mathfrak{M}$*

Démonstration. Par l'absurde.
Supposons qu'il existe $V[v_0]$ une formule à une variable libre représentant l'ensemble $T(\mathfrak{M})$ i.e. pour toute formule F, on a

$$T \vdash V[\#F] \text{ si et seulement si } \mathfrak{M} \vDash F$$

Par le théorème de complétude, nous avons donc que si $\mathfrak{M} \vDash F$ alors $\mathfrak{M} \vDash V[\#F]$. Utilisons le lemme diagonal avec la formule $\neg V[x]$. Par conséquent, il existe une formule F telle que

$$T \vdash F \Leftrightarrow \neg V[\#F]$$

Par conséquent, on obtient

$$\mathfrak{M} \vDash F \text{ si et seulement si } \mathfrak{M} \vDash \neg V[\#F].$$

Contradiction avec le fait que V représente la validité dans $\mathfrak{M}$. ■

Cependant, nous pouvons constater que nous pouvons définir la "vérité" pour les formules Σ.
En effet, l'ensemble des formules close Σ valides dans un modèle peut être représenté par la formule $\mathcal{W}_T[v_0]$ vu la proposition 2.6.3.

3 Théorèmes indémontrables

Après avoir démontré l'existence de théorèmes indémontrables, nous sommes en droit de nous demander si ces théorèmes "ont un sens", une réelle signification mathématique ou s'ils consistent en une construction abstraite auto-référente et inatteignable par des mathématiques "classiques".

Lors de la publication des théorèmes d'incomplétude de Gödel, ce fut l'avis de la majorité des mathématiciens. Ils pensaient que ces théorèmes indémontrables formaient une zone à part, totalement invisible et inatteignable des mathématiques usuelles.
Cependant, en 1982, les mathématiciens Laurence Kirby et Jeffrey Paris démontrèrent qu'un théorème de l'arithmétique ordinaire était indémontrable dans l'arithmétique de Peano : le théorème de Goodstein.

Par voie de conséquence, ils montrèrent qu'un théorème, syntaxiquement simple, et n'utilisant que des objets mathématiques usuels comme l'exponentiation pouvait se révéler indémontrable.
Dans cette section, nous allons donner un exemple de théorème indémontrable en nous intéressant à un autre résultat d'indémontrabilité obtenu par Kirby et Paris : le combat entre Héraclès et l'Hydre de Lerne.

3.1 L'Hydre de Lerne

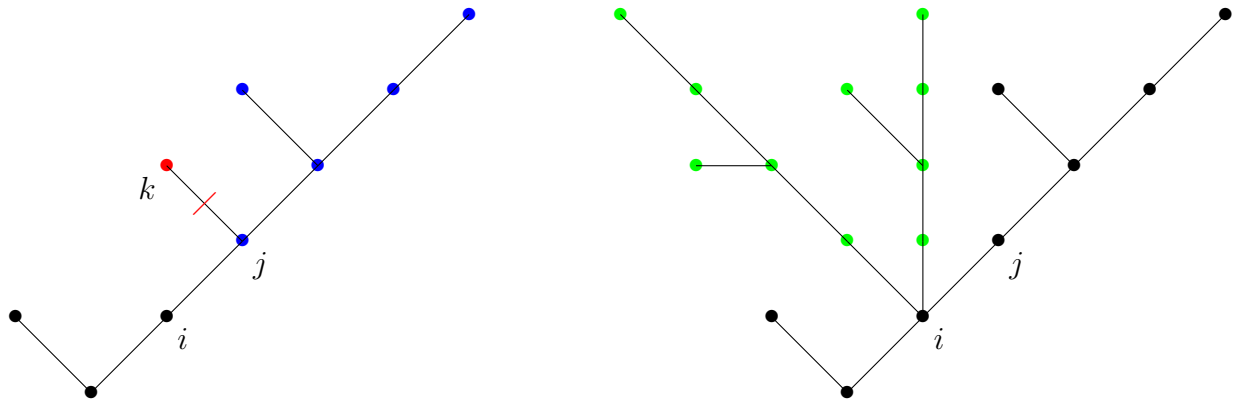
Selon la mythologie grecque, l'Hydre de Lerne était une créature immortelle possédant plusieurs têtes qui se régénéraient doublement lorsqu'elles étaient tranchées, et dont la

défaite constituait le deuxième des Douze Travaux d'Héraclès.
Nous allons nous intéresser ici au combat d'Héraclès contre l'Hydre. Nous verrons que l'Hydre est condamnée quelle que soit la stratégie suivie par Héraclès et que ce fait est indémontrable dans l'arithmétique de Peano.

Définition 3.1.1. Dans la suite, nous définirons une *hydre* comme un arbre fini.
La racine d'une hydre est appelée son *corps*.
Une feuille d'une hydre est quant à elle appellée *tête*.
Nous continuerons à appeller *noeud* un sommet de l'hydre.

Le combat entre l'Hydre et Héraclès se déroule comme suit : à l'étape n, $(n \geq 1)$, Héraclès sectionne une tête de l'Hydre. En réaction, l'Hydre génère n nouvelles têtes de la façon suivante :
Considérons l'Hydre comme un graphe orienté de la racine jusqu'aux feuilles.
Soit k la tête venant d'être sectionnée[1], soit j le prédecesseur de k et i le prédécesseur de j. On considère le sous-arbre ayant pour racine j (qui n'a donc plus l'arête (j, k)) et on le duplique n fois, chaque racine de ces sous-arbres étant ensuite reliée à i.
Si la tête sectionnée est un successeur du corps, l'Hydre ne génère pas de nouvelles têtes.

Exemple 3.1.2. Ci dessous, un exemple de reproduction après décapitation de la tête k.



La question que l'on est en droit de se poser est :

"Héraclès a-t-il une chance de vaincre l'Hydre et, si oui, comment ?"

Précisons tout d'abord qu'Héraclès sera considéré comme vainqueur si, après un nombre fini d'étapes, et sans aide divine, il ne laisse à l'Hydre que son corps qui ne générera donc plus aucune tête.

1. dé-k-pitée en quelque sorte ...

Définition 3.1.3. Une *stratégie* est une fonction qui, à une étape donnée, détermine quelle tête Héraclès doit couper.

Quelles sont donc les stratégies gagnantes pour Héraclès ? Une petite surprise est le résultat suivant :

Théorème 3.1.4. *Toute stratégie est une stratégie gagnante (pour Héraclès).*

Avant de donner une démonstration de ce théorème, nous allons rappeller quelques définitions et propriétés concernant ces dernières.

Définition 3.1.5. Un *ordinal* α est un ensemble tel que :

1. La relation d'appartenance est transitive i.e. si $x \in y \in \alpha$ alors $x \in \alpha$
2. Le couple $(\alpha, \in)$ est un bon ordre [2].

Une des principales propriétés est la suivante :

Théorème 3.1.6. *Toute suite strictement décroissante d'ordinaux est finie.*

Démonstration. Supposons avoir une suite infinie strictement décroissante d'ordinaux $\{\alpha_1, \alpha_2, \ldots\}$.
Par définition des ordinaux, cet ensemble doit avoir un minimum.
Ce qui mène à une contradiction. ■

Rappelons que les ordinaux sont de trois types : zéro, successeur ou limite.

Définition 3.1.7. Définissons des opérations arithmétiques que l'on va appliquer sur les ordinaux.

- La somme ordinale est définie récursivement de la façon suivante :
  - $\alpha + 0 = \alpha$
  - $\alpha + (\beta + 1) = (\alpha + \beta) + 1$
  - $\alpha + \beta = \bigcup_{\gamma \in \beta} (\alpha + \gamma)$ si β est un ordinal limite.
- La multiplication ordinale est définie récursivement de la façon suivante
  - $\alpha.0 = 0$
  - $\alpha.(\beta + 1) = (\alpha.\beta) + \alpha$
  - $\alpha.\beta = \bigcup_{\gamma \in \beta} (\alpha.\gamma)$ si β est un ordinal limite.

2. Cela signifie que toute partie non vide de α admet un minimum pour $\in$

– L'exponentiation ordinale est définie récursivement de la façon suivante
  – $\alpha^0 = 1$
  – $\alpha^{(\beta+1)} = \alpha^\beta.\alpha$
  – $\alpha^\beta = \bigcup_{\gamma\in\beta} \alpha^\gamma$ si β est un ordinal limite.

Maintenant que les bases sont posées, nous pouvons revenir à notre Hydre sans ambiguïté.

Définition 3.1.8. On associe à chaque noeud d'une Hydre donnée un ordinal de la manière récursive suivante :

– A chaque tête, on associe l'ordinal 0.
– Soit un noeud x n'étant pas une tête, soit $H_1, \ldots, H_m$ les sous-arbres lui étant rattachés et α_i l'ordinal associé au sous-arbre H_i pour tout $i \leq m$, alors l'ordinal associé à x est $\omega^{\alpha_1} + \omega^{\alpha_2} + \cdots + \omega^{\alpha_m}$ où $\alpha_1 > \alpha_2 > \cdots > \alpha_m$.

L'ordinal *associé à une Hydre* est l'ordinal de son corps.

Remarque 3.1.9. On remarque que vu qu'une Hydre est un arbre *fini*, l'ordinal associé est toujours inférieur à l'ordinal $\varepsilon_0 = \omega^{\omega^{\omega^{\cdots}}}$ défini comme étant le plus petit ordinal solution de l'équation $x = \omega^x$.

Pour toute stratégie $\mathcal{S}$, on peut définir une opération $[\alpha]_\mathcal{S}(n)$ qui, à partir de l'ordinal α de l'Hydre après l'étape $n-1$, rend l'ordinal de l'Hydre après l'étape n où $\mathcal{S}$ est la stratégie adoptée.

Rappelons les propriétés suivantes :

Proposition 3.1.10. *Soit α, β, γ trois nombres ordinaux.*

– *Si $\alpha < \beta$, alors $\gamma + \alpha < \gamma + \beta$*
– *Si $\alpha \leq \beta$, alors $\alpha + \gamma \leq \beta + \gamma$*
– *Si $\alpha < \beta$ et $\gamma > 0$, alors $\gamma.\alpha < \gamma.\beta$*
– *Si $\alpha < \beta$ et $\gamma > 1$, alors $\gamma^\alpha < \gamma^\beta$*

Proposition 3.1.11. *Soit β un nombre ordinal et m, n deux entiers naturels.*
On a $\omega^{\beta+n}.m < \omega^{\beta+n+1}$

Démonstration. Vu que $m < \omega$ et $\omega^{\beta+n} > 0$, on a $\omega^{\beta+n}.m < \omega^{\beta+n}.\omega = \omega^{\beta+n+1}$. ■

Proposition 3.1.12. *Soit α, β deux nombres ordinaux et m un entier naturel.*
Si $\alpha < \beta$, alors $\omega^\alpha.m < \omega^\beta$

Démonstration. Si $\alpha < \beta$, on a $\omega^\alpha.m < \omega^\beta$.
De même, si $\alpha < \beta$, on a $\alpha + 1 \leq \beta$.
Par conséquent, $\omega^\alpha.m < \omega^{\alpha+1} \leq \omega^\beta$. ■

Démontrons à présent le théorème 3.1.4

Démonstration. Montrons que pour toute stratégie $\mathcal{S}$, tout ordinal α tel que $0 < \alpha < \varepsilon_0$ et tout $n \in \mathbb{N}$,

$$[\alpha]_{\mathcal{S}}(n) < \alpha$$

Rappelons que l'on ne peut couper que des têtes.
Soient une Hydre, une stratégie $\mathcal{S}$ et T la tête donnée par $\mathcal{S}$.
L'ordinal associé à T est donc de 0 et, par conséquent, l'ordinal du parent est de la forme $\beta + n$ où n est un entier naturel non nul.
Après décapitation de la tête T, l'ordinal associé au parent P devient $\beta + (n-1)$.
Supposons être à l'étape k, on réalise $k+1$ copies du sous-hydre ayant P comme racine (sans la tête T).
L'ordinal associé au parent de P qui était $\gamma + \omega^{\beta+n} + \delta$ devient, après décapitation et copie, $\gamma + \omega^{\beta+(n-1)}.(k+1) + \delta$ à l'étape k.

Il nous faut donc montrer que

$$\gamma + \omega^{\beta+n}.m + \delta < \gamma + \omega^{\beta+n+1} + \delta$$

quels que soient n, m entiers.

Vu la proposition 3.1.11, on a $\omega^{\beta+n}.m < \omega^{\beta+n+1}$
Ensuite, vu nos conditions sur l'écriture de l'ordinal associé à l'Hydre, on a

$$\omega^{\beta+n}.m + \delta < \omega^{\beta+n+1} + \delta.$$

En effet, δ s'écrit sous la forme $\sum_{i=1}^{n} \omega^{\lambda_i}$ où $\beta + n + 1 > \lambda_1 \geq \cdots \geq \lambda_n$.
Par conséquent, on a $\omega^{\lambda_i} \leq \omega^{\beta+n}$ pour tout i.

On obtient

$$\begin{aligned}\omega^{\beta+n}.m+\delta &= \omega^{\beta+n}.m+\sum_{i=1}^{n}\omega^{\lambda_i}\\ &\leq \omega^{\beta+n}.m+\sum_{i=1}^{n}\omega^{\beta+n}=\omega^{\beta+n}.(m+n)\\ &< \omega^{\beta+n+1}.\\ &\leq \omega^{\beta+n+1}+\sum_{i=1}^{n}\omega^{\lambda_i}.\\ &= \omega^{\beta+n+1}+\delta.\end{aligned}$$

où la dernière inégalité est obtenue par la proposition 3.1.10 en exploitant le fait que $0 \leq \sum_{i=1}^{n}\omega^{\lambda_i}$.
Donc, $\omega^{\beta+n}.m+\delta < \omega^{\beta+n+1}+\delta$.
Par la même proposition, on a bien $\gamma+\omega^{\beta+n}.m+\delta < \gamma+\omega^{\beta+n+1}+\delta$.

Montrons à présent que l'ordinal de l'Hydre décroît.
On a démontré que l'ordinal du parent de P a diminué après que toutes les nouvelles branches lui aient été rattachées.
Pour prouver que l'ordinal de l'Hydre décroît, il suffit de montrer que si P est sur la branche i d'un noeud X, l'ordinal de X décroît, c'est-à-dire, que si $\omega^{\alpha_i'} < \omega^{\alpha_i}$, alors

$$\omega^{\alpha_1}+\cdots+\omega^{\alpha_i'}+\cdots+\omega^{\alpha_n} < \omega^{\alpha_1}+\cdots+\omega^{\alpha_i}+\cdots+\omega^{\alpha_n}.$$

Remarquons que α_i' n'occupe pas forcément la même position dans la somme que α_i. En effet, à tout moment, nous écrivons l'expression pour qu'elle conserve la forme $\sum_{i=1}^{n}\omega^{\beta_i}$ où $\beta_1 \geq \cdots \geq \beta_n$.
Supposons que α_i' n'occupe pas la même position dans la somme que α_i. Le cas où α_i' occupe la même position dans la somme que α_i se règle de façon semblable.
On doit donc montrer que,

$$\omega^{\alpha_{i+1}}+\cdots+\omega^{\alpha_i'}+\cdots+\omega^{\alpha_n} < \omega^{\alpha_i}+\omega^{\alpha_{i+1}}+\cdots+\omega^{\alpha_n}.$$

C'est le cas car,

$$\omega^{\alpha_{i+1}} + \cdots + \omega^{\alpha_i'} + \cdots + \omega^{\alpha_n} \leq \omega^{\alpha_{i+1}}.(n-i+1) < \omega^{\alpha_i} \leq \omega^{\alpha_i} + \omega^{\alpha_{i+1}} + \cdots + \omega^{\alpha_n}$$

où $n-i+1$ est un nombre naturel et où la majoration $\omega^{\alpha_{i+1}}.(n-i+1) < \omega^{\alpha_i}$ est obtenue par la proposition 3.1.12.
On conclut avec la proposition 3.1.10.

On a donc montré que pour toute stratégie $\mathcal{S}$, tout ordinal α tel que $0 < \alpha < \varepsilon_0$ et tout $n \in \mathbb{N}$,

$$[\alpha]_{\mathcal{S}}(n) < \alpha$$

De cette façon, l'historique du combat entre Héraclès et l'Hydre serait la suite $(\alpha_n)_{n\in\mathbb{N}}$ définie de la façon suivante :

$$\alpha_0 = \alpha$$
$$\alpha_i = [\alpha_{i-1}]_{\mathcal{T}}(i)$$

où $\mathcal{T}$ est la stratégie adoptée par Héraclès.
Vu ce que nous venons de démontrer, on a $\alpha_{n+1} < \alpha_n$ pour tout n et vu qu'il n'existe pas de suite infinie strictement décroissante d'ordinaux, il résulte que cette suite finit par tomber sur 0. ■

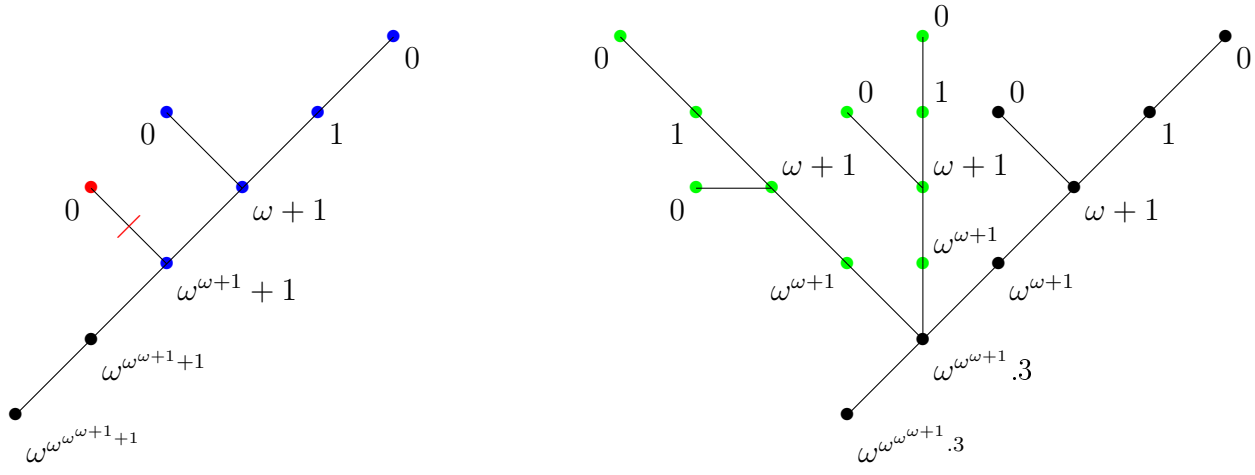


FIGURE 3.1 – Exemple de décroissance de l'ordinal associé à une Hydre.

Remarque 3.1.13. En analysant la démonstration ci-dessus, nous constatons que nous n'exploitons en aucun cas le fait que $k+1$ têtes repoussent après la décapitation de l'Hydre

à l'étape k. Par conséquent, nous pourrions définir une fonction f quelconque et une Hydre telles que $f(k)$ têtes repoussent après la décapitation de l'étape k. Quelle que soit cette fonction f, la démonstration ci-dessus nous montre que cette Hydre est condamnée.

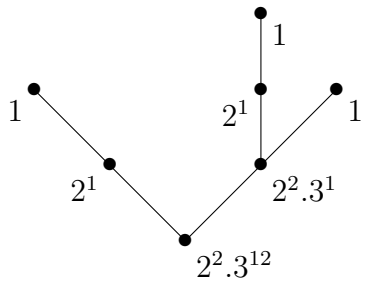
Avant de pouvoir démontrer que ce théorème est indémontrable dans Peano, nous devons prouver que nous pouvons effectivement l'énoncer, cela dans le langage de l'arithmétique du premier ordre. Pour cela, il nous faut trouver un codage des Hydres.

Définition 3.1.14. Les noeuds de l'Hydre seront codés récursivement de la façon suivante :

- Si le noeud N est une tête, on lui associe l'entier 1.
- Sinon, ce noeud possède n sous-arbres $H_1, \ldots, H_n$ qui lui sont attachés.
  Supposons que, pour $i \in \{1, \ldots, n\}$, $\#H_i$ soit l'entier codant H_i. On pose l'entier codant le noeud N comme étant $\pi(1)^{\#H_1} \ldots \pi(n)^{\#H_n}$ où $\pi(n)$ est le n-ième nombre premier.

L'entier associé à une Hydre, sera l'entier codant le corps de l'Hydre.

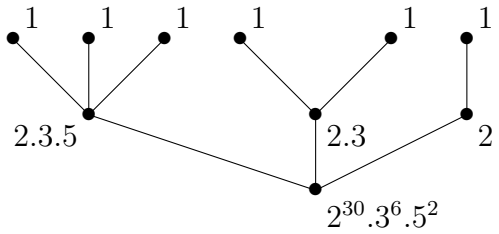
Exemple 3.1.15.



Exemple de codage par des nombres naturels

De même, le décodage se fait sans ambiguïté. A titre d'exemple, l'Hydre associée au nombre 19568944742400 est la suivante

Exemple 3.1.16. Hydre associée au nombre 19568944742400.



Proposition 3.1.17. *L'ensemble des entiers codant une hydre est récursif.*

Démonstration. Il suffit de montrer qu'un nombre est code d'une hydre si ce nombre ainsi que chaque exposant est produit de nombres premiers consécutifs et ainsi de suite.
Soient h la fonction caractéristique de cet ensemble et k la fonction caractéristique de l'ensemble K. La fonction h peut se définir de la façon suivante :

- $h(1) = 1$.
- $h(x) = k(x).\prod_{i=0}^{lg(x)-1} h(\delta(i,x)+1)$.

qui est bien récursive. ■

Nous verrons plus loin que l'on peut également définir des stratégies de manière récursives.

Définition 3.1.18. Soit l'opération $\{\alpha\}(n)$ pour $\alpha < \varepsilon_0$ et $n \in \mathbb{N}$ définie par induction sur α

$$\{0\}(n) = 0, \{\beta+1\}(n) = \beta, \{\omega^{\gamma+1}(\beta+1)\}(n) = \omega^{\gamma+1}\beta + \omega^{\gamma}n$$

et pour un ordinal limite δ,

$$\{\omega^{\delta}(\beta+1)\}(n) = \omega^{\delta}\beta + \omega^{\{\delta\}(n)}$$

Si l'on se restreint aux stratégies récursives, on peut exprimer le théorème précédent dans le langage de l'arithmétique et on a même :

Théorème 3.1.19. *L'assertion "Toute stratégie récursive est une stratégie gagnante" est un énoncé indémontrable dans $\mathcal{P}$.*

Par l'absurde, supposons que l'assertion "toute stratégie récursive est une stratégie gagnante" est prouvable dans $\mathcal{P}$ et montrons qu'il existe une stratégie récursive τ telle que "La stratégie τ est gagnante" est indémontrable dans $\mathcal{P}$.
Soit la stratégie $\mathcal{T}$ exprimée par l'algorithme suivant : on commence à la racine et on passe d'un noeud à un de ses successeurs de la façon suivante : une fois sur un noeud, on choisit, parmi tous les noeuds fils, celui ayant l'ordinal minimal associé. S'il existe plusieurs noeuds ayant l'ordinal minimal associé, on prend celui le plus à droite.
Enfin, lorsqu'on atteint une tête, on la coupe.
On a donc

Proposition 3.1.20.

- $[0]_{\mathcal{T}}(n) = 0$,

- $[\beta+1]_{\mathcal{T}}(n) = \beta$,
- $[\omega^{\gamma+1}(\beta+1)]_{\mathcal{T}}(n) = \omega^{\gamma+1}\beta + \omega^{\gamma}(n+1)$,
- $[\omega^{\delta}(\beta+1)]_{\mathcal{T}}(n) = \omega^{\delta}(\beta) + \omega^{[\delta]_{\mathcal{T}}(n)}$.

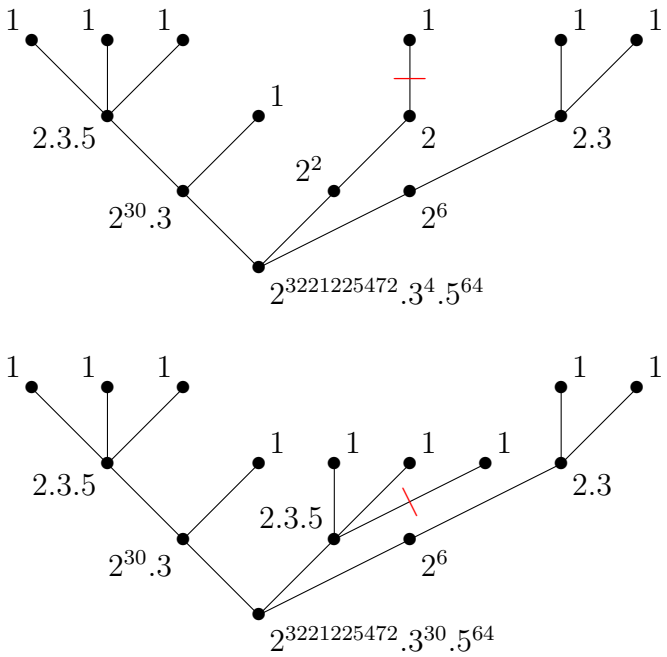
En particulier,

$$[\alpha]_{\tau}(n) = \{\alpha\}(n+1)$$

Avant d'exprimer de façon récursive la stratégie $\mathcal{T}$, remarquons que l'ordinal minimal correspond à la branche "la plus courte".

Exemple 3.1.21.

Décapitation à l'étape 2.



Proposition 3.1.22. *Les fonctions suivantes sont récursives :*

- *la fonction f qui à partir du code d'une Hydre rend le nombre de têtes de l'Hydre*
- *la fonction $g \in \mathfrak{F}_2$ qui à partir d'un nombre m et d'un nombre x dont la décomposition en nombres premiers est $x_0^{a_0} \dots x_n^{a_n}$ rend $\pi(0)^{a_0} \dots \pi(n)^{a_m}$ si $m \leq n$ et rend $\pi(0)^{a_0} \dots \pi(n)^{a_n}$ sinon.*
- *La fonction $g'(\pi(0)^{a_0} \dots \pi(m)^{a_m}) \mapsto \min(a_0, \dots, a_m)$*
- *La fonction h qui prend une Hydre et rend la longueur du plus petit chemin de la racine vers une feuille*

- *La fonction $path(x)$ donnant le numéro de la branche par laquelle passe le plus petit chemin (et donnant le plus grand de ces numéros s'il en existe plusieurs).*
- *La fonction $F(x,m)$ qui prend une Hydre x à l'étape $m-1$ et qui renvoie l'Hydre à l'étape m où la stratégie $\mathcal{T}$ a été suivie.*

Démonstration.

- Il suffit de remarquer que cette fonction est définie comme

$$f(x) = f(a_0) + \cdots + f(a_n) \text{ si } x \text{ est de la forme } \pi(0)^{a_0} \ldots \pi(n)^{a_n}.$$
$$f(x) = 0 \text{ sinon}$$

  ce qui se réécrit $f(x) = \sum_{i=0}^{lg(x)-1} f(\delta(i,x))$ si $x \in K$.
- Définissons la fonction intermédiaire $\mathcal{G} \in \mathfrak{F}_3$ comme suit :

$$\mathcal{G}(1,n,n') = 1$$
$$\mathcal{G}(x,n,0) = 1$$
$$\mathcal{G}(x,n,n') = \begin{cases} \pi(n)^{\delta(n,x)}.\mathcal{G}(x',n+1,n'-1) \text{ si } \pi(n) \text{ divise } x \\ \pi(n)^{\delta(z,x)}.\mathcal{G}(x'',n+1,n'-1) \text{ sinon} \end{cases}$$

où $x' = \dfrac{x}{\pi(n)^{\delta(n,x)}}$, $z = \mu_{y \leq x}(y > n \text{ et } \pi(z) \text{ divise } x)$ et $x'' = \dfrac{x}{\pi(z)^{\delta(z,x)}}$.

  Il nous suffit de poser
$$g(x,m) = \mathcal{G}(x,0,m).$$
- Il suffit de constater que

$$g'(x) = \min(\delta(0,x), \ldots, \delta(lg(x)-1,x))$$

- Vu que x est une Hydre, x est de la forme $\pi(0)^{a_0} \ldots \pi(n)^{a_n}$. On a

$$\begin{cases} h(1) = 0. \\ h(x) = h(\pi(0)^{a_0} \ldots \pi(n)^{a_n}) \mapsto 1 + \min(h(a_0), \ldots, h(a_n)) \end{cases}$$

  où l'on sait que $n = lg(x) - 1$.
- Définissons la fonction intermédiaire $pathaux \in \mathfrak{F}_2$ comme suit :

$$\begin{cases} pathaux(x,i) \mapsto i+1 \text{ si } min(h(a_1), \ldots, h(a_n)) = h(a_i). \\ pathaux(x,i) \mapsto 0 \text{ sinon.} \end{cases}$$

  On pose ensuite $path(x) = \mu_{z<lg(x)}(pathaux(x, lg(x)-1-z) > 0)$.

- Vu tout ce qui précède, F peut se définir comme

$$\begin{cases} F(1,m) = 0 \\ F(x,m) = g(x_0^{a_0} \dots x_i^{F(a_i,m)} \dots x_n^{a_n} x_{n+1}^{F(a_i,m)} \dots x_{n+m}^{F(a_i,m)}) \text{ si } i = path(x) - 1 \text{ et si } h(x) = 2. \\ F(x,m) = g(x_0^{a_0} \dots x_i^{F(a_i,m)} \dots x_n^{a_n}) \text{ si } i = path(x) - 1 \text{ et si } h(x) \neq 2 \end{cases}$$

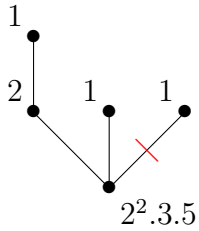
■

Définition 3.1.23. Soit x le code d'une Hydre. On définit la suite hydrique $(\mathcal{H}_n(x))_{n\in\mathbb{N}}$ comme suit :

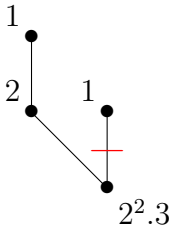
$$\mathcal{H}_0(x) = x$$
$$\mathcal{H}_{i+1}(x) = F(\mathcal{H}_i(x), i+1)$$

Cela consiste simplement en la suite numérique des codes d'une Hydre x de départ tout au long de sa bataille contre Héraclès, ce dernier suivant la stratégie $\mathcal{T}$.

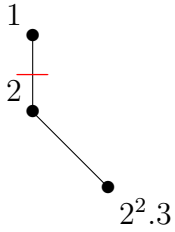
Exemple 3.1.24. Considérons la suite hydrique débutant à 60.



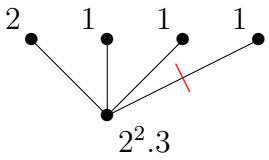
On a que $h(x) = 1$ et $path(x) = 3$.
Par conséquent, on a que
$F(2^2.3.5, 1) = g(2^2.3.5^{F(1,1)}) = 2^2.3.5^0 = 2^2.3$



On a que $h(x) = 1$ et $path(x) = 2$.
Par conséquent, on a que
$F(2^2.3, 2) = g(2^2.3^{F(1,2)}) = g(2^2.3^0) = 2^2$



On a que $h(x) = 2$ et $path(x) = 1$.
Par conséquent, on a que
$F(2^2, 3) = g(2^{F(2,3)}.3^{F(2,3)}.5^{F(2,3)}.7^{F(2,3)})$.
Vu que $h(2) = 1, path(2) = 1$ et $F(2,3) = 1$,
on a $F(2^2, 3) = g(2^1.3^1.5^1.7^1) = 2.3.5.7$.



Par la suite, on aura toujours $h(x) = 1$ et l'on aura

$$F(2.3.5.7, 4) = 2.3.5$$
$$F(2.3.5, 5) = 2.3$$
$$F(2.3, 6) = 2$$
$$F(2, 7) = 1.$$

La suite de 60 est donc

$$\mathcal{H}_0(60) = 60, \mathcal{H}_1(60) = 12, \mathcal{H}_2(60) = 4,$$
$$\mathcal{H}_3(60) = 210, \mathcal{H}_4(60) = 30, \mathcal{H}_5(60) = 6,$$
$$\mathcal{H}_6(60) = 2, \mathcal{H}_7(60) = 1, \mathcal{H}_8(60) = 0$$

et se termine au 8ème terme.

A présent que l'on a démontré que la stratégie $\mathcal{T}$ est récursive, montrons qu'elle est indémontrable.

Définition 3.1.25. Soit $X = \{X_0, X_1, \ldots, X_{|X|-1}\}$ un sous-ensemble fini de $\mathbb{N}$ dont les élements sont classés par ordre croissant.
Nous définissons la notion d'ensemble fini α-large (où $\alpha < \varepsilon_0$) de la manière récursive suivante :
X est 1-large si et seulement si $|X| \geq 2$;
X est α-large si et seulement si $X \setminus \{X_0\}$ est $\{\alpha\}(X_1)$-large

Par la suite, nous écrirons $\{\alpha\}(n_1, \ldots, n_k)$ pour $\{...\{\{\alpha\}(n_1)\}(n_2)\ldots\}(n_k)$.

Proposition 3.1.26. *L'ensemble $X = \{X_0, X_1, \ldots, X_{|X|-1}$ est α-large si et seulement si*

$$\{\alpha\}(X_1, X_2, \ldots, X_{|X|-1}) = 0.$$

Démonstration. Par induction sur α.

- Si $\alpha = 1$:
  X est 1-large si et seulement si $|X| - 1 \geq 1$.
  Ce qui est le cas si et seulement si $\{1\}(X_1, \ldots, X_{|X|-1}) = 0$.
- Si α est un ordinal successeur : On a donc $\alpha = \beta + 1$.
  On a que X est $(\beta + 1)$-large si et seulement si $X \setminus \{X_0\}$ est $\{\beta + 1\}(X_1)$-large i.e. β-large. Ce qui est le cas si et seulement si $\{\beta\}(X_2, \ldots, X_{|X|-1}) = 0$ [3].
  Vu que $\{\beta + 1\}(X_1) = \beta$, on a

  $$\{\beta\}(X_2, \ldots, X_{|X|-1}) = \{\beta+1\}(X_1)(X_2, \ldots, X_{|X|-1}) = \{\beta+1\}(X_1, X_2, \ldots, X_{|X|-1}) = 0.$$

3. $X \setminus \{X_0\}$ peut être vu comme un nouvel ensemble X' tel que $X'_0 = X_1, X'_1 = X_2, \ldots, X'_{|X'|-1} = X_{|X|-1}$

– Si α est un ordinal limite :
X est α-large si et seulement si $X \setminus \{X_0\}$ est $\{\alpha\}(X_1)$-large.
Par induction,

$$X \setminus \{X_0\} \text{ est } \{\alpha\}(X_1)\text{-large si et seulement si } \{\{\alpha\}(X_1)\}(X_2, \ldots, X_{|X|-1}) = 0.$$

Or, $\{\{\alpha\}(X_1)\}(X_2, \ldots, X_{|X|-1}) = \{\alpha\}(X_1, X_2, \ldots, X_{|X|-1})$.

La conclusion en résulte. ■

Remarque 3.1.27. Soit α l'ordinal associé à une Hydre et m le nombre d'étapes nécessaires à Héraclès pour le vaincre.
Vu que

$$0 = [[\ldots[\alpha]_{\mathcal{T}}(1)]_{\mathcal{T}}(2)\ldots]_{\mathcal{T}}(m) = \{\{\ldots\{\alpha\}(2)\}(3)\ldots\}(m+1) = \{\alpha\}(2, \ldots, m+1),$$

on a que l'ensemble $\{1, 2, \ldots, m+1\}$ est α-large.

Posons $\omega_0 = \omega, \omega_{n+1} = \omega^{\omega_n}$ et dénotons $[1, n]$ comme étant l'ensemble $\{1, 2, \ldots, n\}$.

Théorème 3.1.28. *Il existe un modèle non standard $\mathfrak{M}$ tel que $\mathfrak{M} \vDash \mathcal{P}$ et un entier non standard c tel que*

$$\mathfrak{M} \vDash \neg(\exists n\ ([1, n]\ \textit{est}\ \omega_c - \textit{large}))$$

Démonstration. Admis. Voir [9] pour plus de détails. ■

Vu que toute Hydre se termine, on a que pour tout α, il existe n tel que $[1, n]$ est α-large.
Si l'on suppose que $\mathcal{P}$ prouve "La stratégie T est une stratégie gagnante", on a que

$$\mathcal{P} \vdash \text{"pour tout } \alpha\text{, il existe } n \text{ tel que } [1, n] \text{ est } \alpha\text{-large."}.$$

En prenant α de la forme ω_c, vu le théorème, on obtient la conclusion par contradiction.
Il en résulte donc que la stratégie $\mathcal{T}$ est indémontrable dans $\mathcal{P}$ et donc que le théorème 3.1.19 est vérifié.

3.2 Autres théorèmes indémontrables

Dans cette section, nous énoncerons, sans les démontrer, d'autres exemples de théorèmes indémontrables.
Comme évoqué précédemment, le premier résultat d'indémontrabilité présenté par Kirby et Paris fut le théorème de Goodstein.
Nous le présenterons brièvement, une introduction ainsi que des démonstrations complètes étant reprises dans [2]

Définition 3.2.1. Soient $g, p \in \mathbb{N}, p > 1$, la *fonction associée à la représentation de g en base pure p* est définie de manière récursive comme suit :

1. $f_{0,p}(x) = 0$.
2. Supposons que les $f_{k,p}$ soient définis pour tout k inférieur à g. Alors, si la représentation de g en base p est :

$$g = p^{k_1}.a_{k_1} + \cdots + p^{k_n}.a_{k_n}$$

où $0 \leq a_i < p, g > k_1 > \cdots > k_n$, on pose

$$f_{g,p}(x) = x^{f_{k_1,p}(x)}.a_{k_1} + \cdots + x^{f_{k_n,p}(x)}.a_{k_n}$$

Lorsque g est mis sous la forme $f_{g,p}(p)$, on dit que g *est représenté en base pure p.*

La suite de Goodstein en base p commençant à g est la suite $g_0, g_1, \ldots$ où $g_0 = g$ et $g_{n+1} = f_{g_n,p+n}(p+n+1) - 1$ si $g_n \neq 0$.
Si $g_n = 0$, on considère que la suite se termine.

Exemple 3.2.2. La suite de Goodstein en base 2 commençant à 266 est [4] :

$$\begin{aligned}
g_0 = g = 266 &= 2^{2^{2+1}} + 2^{2+1} + 2. \\
g_1 = f_{g_0,2}(3) - 1 &= 3^{3^{3+1}} + 3^{3+1} + 3 - 1 \simeq 4,43.10^{38}. \\
g_2 = f_{g_1,3}(4) - 1 &= 4^{4^{4+1}} + 4^{4+1} + 2 - 1 \simeq 3,2317.10^{616}. \\
g_3 = f_{g_2,4}(5) - 1 &= 5^{5^{5+1}} + 5^{5+1} + 1 - 1 \simeq 2,54.10^{10921}. \\
g_4 = f_{g_3,5}(6) - 1 &= 6^{6^{6+1}} + 6^{6+1} - 1. \\
&= 6^{6^{6+1}} + 5.6^6 + 5.6^5 + 5.6^4 + 5.6^3 + 5.6^2 + 5.6 + 6 - 1. \\
&= 6^{6^{6+1}} + 5.6^6 + 5.6^5 + 5.6^4 + 5.6^3 + 5.6^2 + 5.6 + 5 \simeq 3,54.10^{217832}. \\
g_5 = f_{g_4,6}(7) - 1 &= 7^{7^{7+1}} + 5.7^7 + 5.7^5 + 5.7^4 + 5.7^3 + 5.7^2 + 5.7 + 5 - 1 \simeq 1,06.10^{4871822}.
\end{aligned}$$

4. Calculs réalisés par le programme *Mathématica*

Cette suite croît donc de manière extrêmement rapide sur ces premiers termes, et nous pourrions, a priori, spéculer que cette suite est strictement croissante et tend vers l'infini. Cependant, le théorème suivant vient contrer notre a priori.

Théorème 3.2.3. *(Théorème de Goodstein) Soient $g, p \in \mathbb{N}, p > 1$ et g_n la suite de Goodstein en base p commençant à g. Alors, il existe un n tel que $g_n = 0$.*
En d'autres termes, toute suite de Goodstein se termine.

Mais la célébrité de ce théorème ne vient pas du résultat précédent, démontré par Goodstein, mais du suivant démontré par Kirby et Paris dans [9].

Théorème 3.2.4. *Le théorème de Goodstein est indémontrable dans $\mathcal{P}$*

Le théorème suivant "The strengthened finite Ramsey theorem" est un cas particulier du théorème de Ramsey général concernant l'apparition de sous-graphes monochromatiques complets dans toute coloration de graphes "assez grands".

Théorème 3.2.5 (Ramsey, 1930)**.** *Quels que soient les entiers d, c, m, il existe un entier N tel que pour toute fonction de coloriage $f : \{1, \ldots, N\}^d \to \{1, \ldots, c\}$, il existe $H \subset \{1, \ldots, N\}$ t.q. $card(H) \geq m$, $card(H) > inf(H)$ et f est constante sur H^d.*

Le résultat suivant, qui fut l'un des premiers théorèmes indémontrables connus, fut démontré par Paris et Harrington en 1977.

Théorème 3.2.6 (Paris-Harrington)**.** *Le théorème précédent est indémontrable dans $\mathcal{P}$.*

Plus fondamentalement, nous pourrions également citer les théorèmes de non-contradiction de l'Hypothèse du Continu ou de l'Axiome du Choix prouvant l'indémontrabilité de ces théorèmes et de leurs négations dans la théorie des ensembles ZF (et même ZFC pour l'Hypothèse du Continu).

4
L'indémontrabilité est-elle l'exception ?

Dans les chapitres précédents, nous avons vu l'existence de théorèmes indémontrables. Cependant, le nombre de théorèmes indémontrables connus est négligeable comparativement au nombre, sans cesse croissant, des théorèmes démontrés jusqu'à présent.
Nous nous posons donc naturellement les questions suivantes : « Combien en existe-t-il ? En existe t-il un nombre fini ? Et sinon, sont-ils rares parmi l'ensemble des théorèmes démontrables ? ».

Dans cette section, nous allons tenter de répondre à ces questions en considérant des ensembles dénombrables mais non récursivement énumérables et en étudiant brièvement l'ensemble $I(A)$ des éléments de A dont l'appartenance à A est improuvable avant de nous intéresser aux nombres Ω de Chaitin.

4.1 Rappels

Jusqu'à présent, par souci de simplicité, nous avons toujours travaillé sur des fonctions totales récursives en veillant à appliquer l'opérateur de minimisation non borné sur des parties sûres.
Dans la pratique, il existe bien d'autres fonctions calculables non totales dont l'opérateur de minimisation n'est pas appliqué sur des parties sûres ou est appliqué sur des parties dont nous ne pouvons être certains qu'elles soient des parties sûres.
Nous introduisons donc à présent la notion de fonction partielle récursive.

Définition 4.1.1. Une fonction partielle de $\mathbb{N}^p$ dans $\mathbb{N}$ est un couple (A, f) où A est un sous-ensemble de $\mathbb{N}^p$ et f est une application de A dans $\mathbb{N}$; A est appelé le *domaine de définition* de la fonction.

Nous noterons $\mathfrak{F}_p^*$ l'ensemble des fonctions partielles de $\mathbb{N}^p$ dans $\mathbb{N}$ et $\mathfrak{F}^* = \bigcup_{p\in\mathbb{N}} \mathfrak{F}_p^*$.
Notons que les opérations de composition et de récursion primitive de fonctions partielles sont possibles sous certaines contraintes.
En effet, remarquons que pour des fonctions partielles $f_1, \ldots, f_n \in \mathfrak{F}_p^*$ et $g \in \mathfrak{F}_n^*$, la fonction composée $h = g(f_1, \ldots, f_n)$ est définie en $(x_1, \ldots, x_p) \in \mathbb{N}^p$ si et seulement si

1. toutes les fonctions f_i sont définies en $(x_1, \ldots, x_p)$ et
2. si g est définie en $(f_1(x_1, \ldots, x_p), \ldots, f_n(x_1, \ldots, x_p)) \in \mathbb{N}^p$.

De même, lors de la récursion primitive 1.2.6, f est définie en $(x_1, \ldots, x_p, 0)$ si et seulement si g est définie en $(x_1, \ldots, x_p)$ et est définie en $(x_1, \ldots, x_p, y+1)$ si et seulement si h est définie en $(x_1, \ldots, x_p, y, f(x_1, \ldots, x_p, y))$.
On arrive donc à la définition suivante

Définition 4.1.2. L'ensemble des fonctions *partielles récursives* est le plus petit sous-ensemble de $\mathfrak{F}^*$ qui

- contient les fonctions primitives récursives de base,
- est stable par composition,
- est stable par récursion primitive,
- est stable par la minimisation non bornée.

Exemple 4.1.3.

1. Soit la fonction $f(x) = \mu_y(2y = x)$. Vu qu'il n'est pas possible pour tout entier x d'avoir un entier y tel que $x = 2y$, nous appliquons l'opérateur de minimisation sur une partie non-sûre.
   Cette fonction n'est donc pas récursive mais partielle récursive, son domaine correspond aux entiers naturels pairs.
2. Soit la fonction $f(x) = \mu_y(y^2.x < y)$. Cette fonction est une fonction partielle récursive de domaine vide.

Définition 4.1.4.

- Un *alphabet* est un ensemble fini.
- Les éléments d'un alphabet sont appelés *lettres* ou *symboles*.
- Soit X un alphabet. Un *mot* sur X est une suite finie (et ordonnée) de lettres.

- La *longueur* d'un mot w est le nombre de symboles constituant ce mot ; on la note $|w|$.
- L'unique mot de longueur 0 est le mot correspondant à la suite vide. Ce mot s'appelle le *mot vide* et on le note ε.
- L'ensemble des mots sur X est noté X^*.

Proposition 4.1.5. *L'ensemble des mots sur un alphabet fini est dénombrable.*

Démonstration. En effet, on l'énumère par blocs de mots de taille fixe, d'abord le mot vide, puis les mots de taille 1, puis ceux de taille 2, etc. Chaque bloc peut être ordonné lexicographiquement, à partir d'un ordre arbitraire sur l'alphabet de départ. ■

Proposition 4.1.6. *L'ensemble des fonctions récursives est dénombrable.*

Démonstration. Par définition, une fonction récursive est créée à partir de 5 conditions, elle peut donc être caractérisée par une chaîne finie de caractères.
Nous pouvons donc considérer que toute fonction récursive est un mot sur un alphabet fini. On conclut par la propriété précédente. ■

4.2 Ensembles non récursivement énumérables

Définition 4.2.1.

- Nous définissons *l'ordre préfixe* sur X^* comme suit :
  Pour tout $x, y \in X^*$,

$$x \leq_p y \text{ si et seulement si } \exists z \in X^* : y = xz.$$

- De même, nous définissons *l'ordre lexicographique* sur X^* comme suit :
  Soit $\leq_X$ un ordre total sur X. Alors, pour tout $x, y \in X^*$,

$$x \leq_{lex} y \Leftrightarrow x \leq_p y \vee \exists u, v, w \in X^* \exists a, b \in X : (x = uav \wedge y = ubw \wedge a <_X b)$$

- Vu que X est un ensemble fini, il existe une bijection *string* de $\mathbb{N}$ dans X^* où $string(n)$ est le n-ième mot suivant l'ordre lexicographique sur X^*.
- Une fonction partielle $f : X^* \to X^*$ définie sur l'ensemble des mots sur X est *récursive* s'il existe une fonction partielle récursive $g : \mathbb{N} \to \mathbb{N}$ telle que

$$f(x) = string(g(string^{-1}(x)))$$

  pour tout $x \in X^*$

- Soit $\mathcal{R}^*$ l'ensemble de toutes les fonctions partielles récursives appartenant à $\mathfrak{F}_1^*$ de X^* dans X^*. Sans perte de généralité, nous supposons que l'opérateur de minimisation μ est pris avec l'ordre lexicographique.
- Une relation binaire R sur X^* est *récursive* si et seulement si le prédicat binaire "$(x, y) \in R$" est récursif.
- Soit $\equiv$ une relation d'équivalence sur X^*. La classe d'équivalence de x (relative à la relation $\equiv$) sera notée $[x]_{\equiv}$.
- *L'index* d'une relation d'équivalence $\equiv$ est son nombre de classes d'équivalences.

Exemple 4.2.2. La relation d'équivalence longueur définie par : $u \equiv_{lg} v \Leftrightarrow |u| = |v|$ est trivialement une relation d'équivalence récursive d'index infini.

Soit $\mathbb{T}$ la classe de toutes les theories T qui sont récursivement énumérables et même telles que l'ensemble des théorèmes de T est récursivement énumérable, consistant et suffisamment riche que pour contenir l'arithmétique de Peano.

Définition 4.2.3. Soit T une théorie de $\mathbb{T}$.
Un prédicat H sera dit vrai pour x si $H[x]$ est un théorème de T.
Un ensemble $A \subset X^*$ est dit *définissable* dans T s'il existe un prédicat H tel que pour tout x, $x \in A$ si et seulement si $T \vdash H[x]$.

Proposition 4.2.4. *Il existe des ensembles dénombrables qui ne sont pas récursivement énumérables.*

Démonstration. Vu que X^* et l'ensemble des fonctions récursives sont dénombrables, on a une suite de mots $X^* = \{x_0, x_1, \ldots\}$ ainsi qu'une suite de fonctions récursives $f_0, f_1, \ldots$.
On considère alors le langage L suivant :

$$L = \{x_i |\ x_i \text{ n'est pas dans l'image de } f_i\}$$

Montrons que L n'est pas récursivement énumérable. Procédons par l'absurde et supposons que L soit récursivement énumérable. Il est alors l'image d'une fonction récursive.
Il existe donc un entier naturel k tel que la fonction f soit égale à la fonction f_k. Considérons à présent le mot w_k.

- Soit le mot w_k est dans l'image de f_k et donc il n'appartient pas à L par définition. Ceci contredit le fait que l'image de $f = f_k$ soit l'ensemble L.
- Soit le mot w_k n'est pas dans l'image de f et donc il appartient à L par définition. Ceci contredit le fait que l'image de $f = f_k$ soit l'ensemble L.

Par conséquent, le langage L n'est pas récursivement énumérable. ■

Proposition 4.2.5. *Soit $A \subset X^*$ tel que A est définissable dans T mais n'est pas récursivement énumérable.*
Alors il existe au moins un $x \in A$ tel que l'assertion "$x \in A$" est indémontrable dans T.

Démonstration. Soit H la formule définissant A.
Vu que T est récursivement énumérable, on a que l'ensemble $A_T = \{x | x \in X^*, T \vdash H[x]\}$ est également récursivement énumérable.
Par conséquent, si $T \vdash H[x]$ pour tout $x \in A$, alors $A = A_T$.
Or A n'est pas récursivement énumérable par hypothèse. Par conséquent, il existe au moins un $x \in A$ tel que $T \nvdash H[x]$ et donc tel que l'assertion "$x \in A$" est indémontrable dans T. ■

Pour toute théorie $T \in \mathbb{T}$ et tout ensemble $A \subset X^*$ non récursivement énumérable et définissable dans T, posons $I(A) = \{u | u \in A, T \nvdash "u \in A"\}$.
Le résultat précédent nous a prouvé que $I(A)$ est non vide et même infini.
En effet, si l'on suppose que $I(A)$ est fini, nous pourrions démontrer, vu que T est récursivement énumérable, que l'ensemble A l'est également.
Nous allons maintenant essayer de déterminer la "taille" de $I(A)$.

Définition 4.2.6. Un ensemble $A \subset X^*$ sera dit *saturé relativement à une relation d'équivalence* si A est une union de classes d'équivalence.

Théorème 4.2.7. *Supposons $A \subset X^*$ définissable dans T et supposons que $\equiv$ est une relation d'équivalence récursive sur X^* qui sature A.*
Si A n'est pas récursivement énumérable, alors il existe une infinité d'éléments x appartenant à A tels que $[x]_\equiv \subset I(A)$.
De plus, $\equiv$ est d'index infini.

Démonstration. Soit

$$B = \{z | z \in X^*, \exists u \in X^* (u \equiv z \wedge T \vdash "u \in A")\}$$

Alors B est récursivement énumérable vu que T est récursivement énumérable et que $\equiv$ est récursive.
De plus, vu que $\equiv$ sature A, on a $B \subset A$. En effet, soit $z \in B$, il existe donc un $u \in A$ tel que $z \equiv u$ i.e. $z \in [u]_\equiv$ et vu que A est union de classes d'équivalence, $z \in A$.
Vu que A n'est pas récursivement énumérable, il vient que $A \neq B$ et donc, qu'il existe un élément $x \in A \setminus B$.

On a même que $A \setminus B$ est infini.
En effet, si $A \setminus B$ est fini, il est récursivement énumérable. Par conséquent, $(A \setminus B) \cup B = A$ est récursivement énumérable, ce qui est impossible.

Montrons à présent que $[x]_{\equiv} \subset I(A)$ pour tout $x \in A \setminus B$.
Soit $u \in X^*$ tel que $u \in [x]_{\equiv}$. On a que $u \in A$ vu que A est saturé.
Supposons que $T \vdash$ "$u \in A$". Dans ce cas, on aurait que $x \in B$ ce qui contredit notre hypothèse.
Montrons que $\equiv$ est d'index infini. Procédons par l'absurde et supposons qu'il n'a qu'un nombre fini de classes d'équivalence. Vu que la relation récursive $\equiv$ sature l'ensemble A, on a que celui-ci, étant une union fini de classes d'équivalence, est récursivement énumérable. Ce qui contredit nos hypothèses. ■

4.3 Topologie de l'indémontrable

Soit τ une topologie sur X^*, nous notons $\overline{A}_\tau$ l'adhérence de A relativement à la topologie τ.

Définition 4.3.1.

- Un ensemble A de X^* est *dense* si pour tout élément x de X^*, tout voisinage de x contient au moins un point de A ou, de manière équivalente, si son adhérence est X^*.
- Un ensemble $A \subset X^*$ est *rare* relativement à la topologie τ si, pour tout $x \in X^*$ et tout ouvert de voisinage N_x, de x, on a $N_x \not\subset \overline{A}_\tau$.
- Un ensemble $A \subset X^*$ est dit *corare* si son complémentaire est rare.

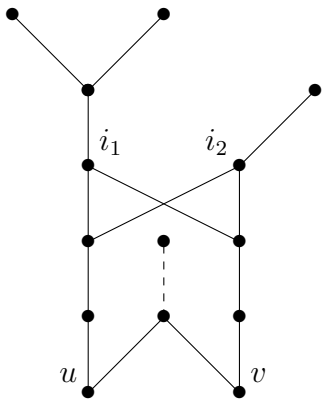
Un ordre partiel $\leq$ sur X^* définit une topologie $\tau_\leq$ comme suit : Pour chaque $u \in X^*$, soit $N_u^\leq = \{v | v \in X^*, u \leq v\}$.

Proposition 4.3.2. *L'ensemble* $\mathfrak{V} := \{N_u^\leq | u \in X^*\}$ *est le systeme de base de voisinage pour* $\tau_\leq$.

Démonstration. Il suffit de montrer que X^* est une union d'éléments de $\mathfrak{V}$ et que l'intersection de deux éléments de $\mathfrak{V}$ est une union d'éléments de $\mathfrak{V}$.
Vu que $x \in N_x^\leq$, le premier point est évident. A présent, soient u et v deux éléments de X^*.
On a que $x \in N_u^\leq \cap N_v^\leq$ si et seulement si $x \in X^*$ et si $u \leq x$ et $v \leq x$.
Trois cas sont possibles :

- Soit un tel élément n'existe pas. L'intersection est donc vide.

– Soit un tel élément existe, et dans ce cas :
  – Soit u et v sont comparables. Dans ce cas, si $u \leq v$ (resp. $v \leq u$), on a alors $N_v^{\leq} \subset N_u^{\leq}$ et $N_u^{\leq} \cap N_v^{\leq} = N_v^{\leq}$ (resp. et par le même raisonnement, $N_u^{\leq} \cap N_v^{\leq} = N_u^{\leq}$)
  – Soit u et v ne sont pas comparables. Remarquons que l'on pourrait avoir le cas dégénéré où une infinité d'éléments sont supérieurs à u et v sans que u et v soient comparables.



On a donc dans tous les cas,

$$N_u^{\leq} \cap N_v^{\leq} = \bigcup_{i \in N_u^{\leq} \cap N_v^{\leq}} N_i^{\leq}.$$

qui est bien une union dénombrable. ■

Proposition 4.3.3. *L'adhérence d'un ensemble A est donnée par*

$$\overline{A}_{\tau_{\leq}} = \{u | u \in X^*, \exists v \in A : u \leq v\}$$

pour tout $A \subset X^*$.

Démonstration. Cela découle de la définition de l'adhérence et de notre système de bases de voisinages. ■

Proposition 4.3.4. *Si $\leq_a$ et $\leq_b$ sont des ordres partiels tels que $\leq_a \subset \leq_b$ alors (i.e. $x \leq_a y$ implique $x \leq_b y$ pour tout x, y) tout ensemble dense pour $\tau_{\leq_a}$ est également dense pour $\tau_{\leq_b}$.*

Démonstration. Cela découle de la proposition précédente et de la définition d'ensemble dense. ■

Définition 4.3.5.

- Une topologie τ satisfait la *condition de Calude* pour une relation d'équivalence récursive $\equiv$ sur X^* si pour tout $x \in X^*$ et tout voisinage ouvert N_x de x, l'ensemble $\{y | y \in X^*, N_x \cap [y]_\equiv = \emptyset\}$ est fini.
- Nous dirons qu'un ordre partiel $\leq_a$ sur X^* *préserve la longueur* si, pour tout $x, y \in X^*$, $x \leq_a y$ implique $|x| \leq |y|$.
- La relation d'équivalence $\equiv_{lg}$ est définie comme $u \equiv_{lg} v$ si et seulement si $|u| = |v|$.

Exemple 4.3.6. Parmi les ordre satisfaisant la condition de Calude, nous pouvons citer l'ordre préfixe et lexicographique. L'article original [4] définit d'autres exemples.

Remarque 4.3.7. Tout ordre total préservant la longueur satisfait la condition de Calude relativement à la relation d'équivalence $\equiv_{lg}$.
En effet, on a que $N_x \cap [y]_{\equiv_{lg}} = \emptyset$ si et seulement si $v < x$ i.e. $|v| < |x|$ pour tout $v \in [y]_{\equiv_{lg}}$ et il existe bien un nombre fini d'éléments y satisfaisant cette condition.

Théorème 4.3.8. *Soit $\equiv$ une relation d'équivalence récursive sur X^* et soit $A \subset X^*$ un ensemble qui est définissable dans T, saturé par $\equiv$ et non récursivement énumérable.*
Alors l'ensemble $I(A)$ est dense pour toute topologie τ sur X^* qui satisfait la condition de Calude pour $\equiv$.

Démonstration. Par le théorème 4.2.7, l'ensemble $S = \{x | x \in X^*, [x]_\equiv \subset I(A)\}$ est infini.
Soit un élément $y \in X^*$ et un ouvert de voisinage N_y de y. Vu que la condition de Calude est satisfaite, l'ensemble des éléments $u \in X^*$ tels que $N_y \cap [u]_\equiv = \emptyset$ est fini.
Donc, il existe un élément $u \in S$ tel que $N_y \cap [u]_\equiv \neq \emptyset$ i.e. tel que $[u]_\equiv \subset I(A)$ et $N_y \cap [u]_\equiv \neq \emptyset$. Alors, vu que pour tout $y \in X^*$, tout voisinage de y contient un point de $I(A)$, $I(A)$ est dense dans X^*. ■

4.4 Les nombres Omega de Chaitin

La section précédente nous a montré que, sous certaines conditions, l'ensemble des énoncés indémontrables est "plus grand" que l'ensemble des énoncés démontrables. Par conséquent, il nous paraît normal de nous poser la question « Pourquoi n'a t-on pas plus d'énoncés indémontrables ? Est-il possible de trouver une infinité d'énoncés prouvés indémontrables ? ». La réponse à la dernière question est affirmative et nous allons exposer ici les résultats de Gregory Chaitin allant dans ce sens.

Définition 4.4.1.

(i) Un mot $x \in X^*$ est un *préfixe* d'un mot y si $x \leq_p y$ i.e. s'il existe un mot $z \in X^*$ tel que $y = xz$.

(ii) Un ensemble $S \subset X^*$ est appelé *sans préfixe* si pour tout $x, y \in S$, $x \leq_p y$ implique $x = y$.

Définition 4.4.2.

- Un *ordinateur* est une fonction partielle récursive $\varphi : X^* \times X^* \to X^*$.
- Un *ordinateur de Chaitin* est un ordinateur C tel que pour tout $v \in X^*$, le domaine de C_v est sans préfixe où $C_v : X^* \to X^*$ est la fonction partielle telle que $C_v(x) = C(x, v)$ pour tout $x \in X^*$.

Cette définition d'un ordinateur de Chaitin a pour but de véritablement modéliser un programme informatique.
Nous pouvons voir le couple $(x, v) \in X^* \times X^*$ comme étant un couple "(programme,input)" rendant, si ce couple est défini, l'output correspondant.
Remarquons que si la fonction C_v est définie en $x \in X^*$, il n'existe pas de $y \in X^*, y \leq_p x$ tel que C_v est définie en y. Nous pouvons voir cela comme l'expression du fait qu'un programme informatique possède une syntaxe précise indiquant le début et la fin de la partie exécutable de ce programme. Par conséquent, une chaîne de caractères étant le préfixe de ce programme ne contiendra pas les éléments de syntaxe indiquant la fin du programme et cette chaîne ne sera donc pas valide.

Définition 4.4.3. Un ordinateur (de Chaitin) ψ est *universel* si pour tout ordinateur (de Chaitin) φ, il existe une constante c, dépendant de ψ et φ, avec la propriété suivante : si φ est définie en $(x, v) \in X^* \times X^*$, alors il existe un mot x' tel que $\psi(x', v) = \varphi(x, v)$ et $|x'| \leq |x| + c$.

Théorème 4.4.4. *Il existe un ordinateur universel (de Chaitin).*

Démonstration. Donnons la preuve pour un ordinateur de Chaitin.
Supposons avoir $F : \mathbb{N} \times X^* \times X^* \to X^*$ une fonction partielle récursive universelle pour l'ensemble des fonctions partielles récursives $C : X^* \to X^*$ tel que pour tout $v \in X^*$, l'ensemble $\{u \in X^* \mid C \text{ est définie en } (u, v)\}$ est sans-préfixe. Alors, posons

$$\psi(a_1^i a_2 u, v) = F(i, u, v)$$

Vérifions que ψ est un ordinateur universel. Soit un ordinateur de Chaitin φ, il existe donc un i tel que $\varphi(u, v) = F(i, u, v)$. On a alors que pour tout (x, v) appartenant au domaine de φ,

$$\varphi(u, v) = \psi(a_1^i a_2 u, v)$$

où $a_1, a_2 \in X^*$. De plus, on a que $|a_1^i a_2 u| \leq |u| + i + 1$. Il suffit donc de prendre $c \geq i + 1$. On a également que ψ est un ordinateur de Chaitin grâce à la présence de la lettre a_2.

■

Pour la suite, nous dénoterons C un ordinateur de Chaitin et U un ordinateur universel de Chaitin. De plus, rappelons que ε représente le mot vide.

Définition 4.4.5.

(a) La *complexité absolue de Chaitin* (appellée aussi complexité absolue auto-délimitée) associée à l'ordinateur de Chaitin C est la fonction partielle $H_C : X^* \to \mathbb{N}$,

$$H_C(x) = \min\{|u| \ : u \in X^*, C(u, \varepsilon) = x\}.$$

Dans le cas où $C = U$ on écrit $H(x) = H_U(x)$.

(b) Le *programme canonique* défini en fonction d'un ordinateur universel de Chaitin U est

$$x^* = \min\{u \in X^* \ : U(u, \varepsilon) = x\}$$

où le minimum est considéré relativement à l'ordre standart sur X^*.

Corollaire 4.4.6. *Pour tout ordinateur de Chaitin C, on a*

$$H(x) \leq H_C(x) + O(1).$$

Démonstration. Cela découle de la définition d'un ordinateur universel de Chaitin. ■

Lemme 4.4.7. *Soit A un langage sans préfixe sur un alphabet de longueur Q. Alors, on a que*

$$\sum_{u \in A} Q^{-|u|} \leq 1$$

Démonstration. Soit r_i le nombre de mots de longueur i appartenant à A. Remarquons que :

- Il y a autant de mots de longueur 1 possibles qu'il n'y a d'éléments dans notre alphabet.
  Donc, $r_1 \leq Q$.
- Étant donné que A est un langage sans préfixe, la première lettre d'un mot de longueur 2 appartenant à A est une lettre n'appartenant pas à A. On constate qu'il n'y a que $Q - r_1$ possibilités pour la première lettre.
  Par conséquent, $r_2 \leq (Q - r_1).Q$.
- De même, pour un mot w de longueur 3, on a que $w \in A$ si w n'a pas de préfixe appartenant à A. Or, par le point précédent, on sait que le nombre de mots de 2 lettres pouvant être dans A est $(Q - r_1).Q$. Donc l'ensemble des paires de lettres pouvant constituer w est de taille $(Q - r_1).Q - r_2$.
  Par conséquent, $r_3 \leq ((Q - r_1).Q - r_2)Q$

Le même raisonnement nous donne

$$r_m \leq (\ldots(((Q - r_1).Q - r_2).Q - r_3).Q \cdots - r_{m-1}).Q$$

Il est possible de montrer, par récurrence, que cette expression se réécrit sous la forme

$$r_m \leq Q^m - r_1 Q^{m-1} - \cdots - r_{m-1}Q$$

A présent, considérons la suite $(u_n)_{n\in\mathbb{N}}$ telle que $u_m = \sum_{i=1}^{m} r_i Q^{-i}$ pour tout m. Vu que

$$\begin{aligned}
u_m &= \sum_{i=1}^{m} r_i Q^{-i} = \sum_{i=1}^{m-1} r_i Q^{-i} + r_m Q^{-m} \\
&\leq \sum_{i=1}^{m-1} r_i Q^{-i} + (Q^m - r_1 Q^{m-1} - \cdots - r_{m-1}Q)Q^{-m} \\
&= \sum_{i=1}^{m-1} r_i Q^{-i} + (1 - r_1 Q^{-1} - \cdots - r_{m-1}Q^{-(m-1)}) \\
&= 1
\end{aligned}$$

On a que chaque u_m est borné. De plus, vu que $r_i Q^{-i} > 0$ pour tout i, on a que $(u_n)_{n\in\mathbb{N}}$ est une suite convergente car croissante et majorée.
Enfin, vu que $u_m \leq 1$ pour tout m, on a que $\lim_{n\to+\infty} u_n \leq 1$.

Par conséquent, on a que

$$\sum_{i=1}^{+\infty} r_i Q^{-i} = \sum_{v \in A} Q^{-|v|} \leq 1$$

■

Définition 4.4.8. Soit un ordinateur de Chaitin C, on définit les "probabilités" suivantes :

$$P_C(x) = \sum_{\{u \in X^* | C(u,\varepsilon)=x\}} Q^{-|u|},$$

Dans le cas où C est un ordinateur universel de Chaitin (que nous noterons U), on pose $P(x) = P_U(x)$.
Nous dirons que $P_C(x)$ est la *probabilité algorithmique absolue* de l'ordinateur de Chaitin C avec *output* x.

Nous pouvons voir cela comme la probabilité que l'ordinateur C produise l'output x.

Lemme 4.4.9. *Pour chaque ordinateur de Chaitin C et tout mots x et y on a :*

$$\Omega_C = \sum_{x \in X^*} P_C(x) \leq 1,$$

Démonstration.

$$\sum_{x \in X^*} P_C(x) = \sum_{x \in X^*} \sum_{\{u \in X^* | C(u,\varepsilon)=x\}} Q^{-|u|} = \sum_{u \in dom(C_\varepsilon)} Q^{-|u|} \leq 1$$

où la dernière inégalité est obtenue par le fait que C est un ordinateur de Chaitin ainsi que par le lemme précédent. ■

Remarque 4.4.10. Nous voyons ici apparaître l'utilité de la restriction présente dans la définition d'ordinateur de Chaitin : si le domaine de C_ε n'avait pas été sans préfixe, le nombre Ω_C ne serait pas toujours inférieur à 1.
En effet, prenons un alphabet binaire $\{0, 1\}$, il suffit que les mots 0 et 1 fassent partie de C_ε pour que, pour tout autre mot x, on ait

$$\Omega_C = \sum_{u \in dom(C_\varepsilon)} 2^{-|u|} = \frac{1}{2} + \frac{1}{2} + Q^{-|x|} > 1$$

Soit $\Sigma : \mathbb{N} \to \mathbb{N}$ la fonction définie comme

$$\Sigma(n) = \max_{x \in X^n} H(x)$$

Il est possible de démontrer que $\Sigma(n) = n + H(string(n)) + O(1)$. Cette démonstration est présente dans [3].
Par la suite, un mot de longueur n sera considéré comme étant aléatoire s'il est de complexité maximale parmi l'ensemble des mots de longueur n. Plus précisément,

Définition 4.4.11.

- Un mot $x \in X^*$ est *m-aléatoire* où $m \in \mathbb{N}$ si $H(x) \geq \Sigma(|x|) - m$;
- Un mot $x \in X^*$ est *aléatoire* s'il est 0-aléatoire.

Historiquement, la recherche d'une définition rigoureuse de la notion de suite aléatoire fut une gageure pour les mathématiciens tout au long du XXème siècle. Toutes les définitions proposées entre 1919 et 1965 se révélant inadéquates. Ce n'est qu'en 1965 que Per Martin-Löf donna une définition aujourd'hui acceptée comme valable. Cependant, il existe d'autres définitions équivalentes à celle de Martin-Löf dont celle de Chaitin, présentée ci-dessus, et que nous allons exploiter.

Soient X^ω l'ensemble des suites infinies d'éléments de X et une suite $\mathbf{x} \in X^\omega$.
Pour tout n, nous posons $\mathbf{x}(n)$ comme étant le mot appartenant à X^n constitué des n premières lettres de $\mathbf{x}$ et $\mathbf{x}_j$ comme étant la j-ième lettre du mot $\mathbf{x}$.

Définition 4.4.12 (Chaitin)**.** Une suite $\mathbf{x} \in X^\omega$ est *aléatoire* si et seulement s'il existe un naturel $c > 0$ tel que

$$H(\mathbf{x}(n)) \geq n - c,$$

pour tout naturel $n \geq 1$.

Par la suite, nous considérerons uniquement des ordinateurs dont la deuxième variable ("l'input") est ε. Nous écrirons donc, sans ambiguité possible, U, M, C à la place de $U_\varepsilon, M_\varepsilon, C_\varepsilon$. De même, nous posons $\Omega := \Omega_U$.
Soit $A_Q = \{0, 1, 2, \ldots, Q-1\}$ et $f : \mathbb{N}_+ \to X^*$ une fonction récursive et injective telle que $f(\mathbb{N}_+) = \mathrm{dom}(U_\varepsilon)$ et soit

$$\omega_k = \sum_{i=1}^{k} Q^{-|f(i)|}.$$

Il est clair que la séquence $(\omega_k)_{k \geq 0}$ converge vers Ω par des valeurs inférieures.
Soit

$$\Omega = \Omega_U = 0.\Omega_1\Omega_2 \ldots \Omega_n \ldots$$

Le développement en base Q de Ω et soit

$$\mathbf{r}_Q(\Omega) = \Omega_1\Omega_2 \ldots \Omega_n \ldots \in A_Q^\omega$$

Lemme 4.4.13. *Si $\omega_n \geq \Omega(i)$, alors*

$$\Omega(i) \leq \omega_n < \Omega \leq \Omega(i) + Q^{-i}$$

Démonstration. Remarquons tout d'abord que

$$\sum_{j=i+1}^{+\infty} \Omega_j Q^{-j} \leq \sum_{j=i+1}^{+\infty} (Q-1)Q^{-j} = \sum_{j=i+1}^{+\infty} (Q^{1-j} - Q^{-j}) = Q^{-i}$$

On a donc bien que

1. $\Omega(i) \leq \omega_n$ par hypothèse.
2. $\omega_n < \Omega$ vu que $(\omega_k)_{k\in\mathbb{N}}$ converge vers Ω par des valeurs inférieures.
3. $\Omega = \Omega(i) + \sum_{j=i+1}^{+\infty} \Omega_j Q^{-j} \leq \Omega(i) + Q^{-i}$.

■

Théorème 4.4.14 (Chaitin)**.** *La séquence $\mathbf{r}_Q(\Omega) \in X_Q^\omega$ est aléatoire.*

Démonstration.
Soit un ordinateur de Chaitin M effectuant les opérations suivantes : pour un x donné, il considère $y = U(x)$ ainsi que le plus petit nombre t tel que $\omega_t \geq 0.y$ et rend comme valeur $M(x)$ le premier mot [1] n'appartenant pas à l'ensemble $\{U(f(1)), U(f(2)), \ldots, U(f(t))\}$. On constate que $M(x)$ n'est défini que si $U(x)$ et t existent.

Remarquons que si M est défini en un élément $x \in X^*$ et s'il existe un élément x' tel que $U(x) = U(x')$, alors $M(x) = M(x')$. Par conséquent, en prenant un x tel que $M(x)$ soit défini et le programme canonique $x' := (U(x))^*$ de $U(x)$, on a

$$H_M(M(x)) \leq |x'| = H_U(U(x)).$$

De plus, vu le corollaire 4.4.6, il existe une constante c telle que

$$H_U(M(x)) \leq H_M(M(x)) + c$$

pour tout x tel que $M(x)$ soit défini.
A présent, soient n un entier et x un mot tel que

$$U(x) = \Omega_1\Omega_2\ldots\Omega_n.$$

1. selon l'ordre lexicographique

Soit t le plus petit entier tel que $\omega_t \geq 0.\Omega_1\Omega_2\ldots\Omega_n$. Vu le lemme précédent, on a que

$$\begin{aligned} 0.\Omega_1\Omega_2\ldots\Omega_n + \sum_{s=t+1}^{+\infty} Q^{-|f(s)|} &\leq \omega_t + \sum_{s=t+1}^{+\infty} Q^{-|f(s)|} \\ &= \Omega_U \\ &\leq 0.\Omega_1\Omega_2\ldots\Omega_n + Q^{-n} \end{aligned}$$

Par conséquent, on a

$$\sum_{s=t+1}^{+\infty} Q^{-|f(s)|} \leq Q^{-n}.$$

Cela implique que $|f(s)| \geq n$ pour tout $s \geq t+1$. Par construction, M calcule tout les $U(f(1)), U(f(2)), \ldots, U(f(t))$, on en tire que le programme de taille minimale est $f(r)$ où $r > t$. Par conséquent, $|f(r)| = H_U(M(x)) \geq n$ et on a donc

$$\begin{aligned} n &\leq H_U(M(x)) \\ &\leq H_M(M(x)) + c \\ &\leq H_U(U(x)) + c \\ &= H_U(\Omega_1\ldots\Omega_n) + c. \end{aligned}$$

Cela se réécrit $H_U(\Omega_1\ldots\Omega_n) \geq n - c$ et prouve que $\mathbf{r}_Q(\Omega)$ est aléatoire. ■

Théorème 4.4.15 (Théorème d'incomplétude de la théorie de l'information de Chaitin)**.** *Supposons que l'ensemble des théorèmes d'un système axiomatique formel T est récursivement énumérable. Si T a la propriété que chaque assertion de la forme*

$$\text{"le } n^e \text{ bit de } \Omega_U \text{ est } 0\text{"},$$

$$\text{"le } n^e \text{ bit de } \Omega_U \text{ est } 1\text{"},$$

peut être représentée dans T et qu'une telle assertion est un théorème de T seulement si elle est vraie[2] *alors T nous permet de déterminer les valeurs d'au plus un nombre fini de bits de Ω_U*

Démonstration. Admise. Voir [3] page 326 pour plus de détails. ■

Nous avons donc un nombre infini dénombrable de propositions indémontrables. Cependant, il existe un résultat plus précis de R. Solovay [14] utilisant un ordinateur de Chaitin

2. sous-entendu, dans tout modèle de T

pour lequel tous les bits du nombre Ω associé sont incalculables.
Pour un tel Ω, nous posons $P(n)$ l'assertion "Le n^{e}bit de Ω est 0". Nous obtenons immédiatement le résultat suivant :

Théorème 4.4.16. *Il existe une suite infinie dénombrable $(P(n))_{n\in\mathbb{N}}$ d'énoncés telle que, pour tout n, $P(n)$ soit indémontrable.*

4.5 Conclusion

L'indémontrabilité est un sujet à-part dans les domaines de recherche mathématique. Touchant au plus profond de la logique mathématique et philosophique par ses résultats et exhibant des limites infranchissables au pouvoir du raisonnement formel en mathématique, chacune des découvertes, de son existence par Gödel à ses tentatives de caractérisation par Calude et Chaitin, fut riche en surprises contre-intuitives et en reconsidération de la perception même que nous avions des mathématiques.

Entre le début du XXème siècle où les mathématiques étaient perçues par certains comme "Le langage de Dieu" pouvant tout démontrer et la fin de ce siècle où l'indémontrabilité se révèle être omniprésente, une révolution intellectuelle se produisit dans le rapport unissant les mathématiciens et leur science.
De même, le lien établi par Chaitin entre incomplétude, indémontrabilité et théorie algorithmique de l'information risque de se révéler encore plus fructueux à l'avenir...

Peut-être au point de devoir, une fois de plus, reconsidérer notre relation à cette science que l'on croyait parfaite.

5

Annexes

5.1 Quelle arithmétique ?

Dans le deuxième chapitre, nous avons démontré que l'arithmétique de Peano (et même de Robinson) est incomplète. Cependant, nous pourrions nous demander si l'axiomatisation de Peano est une bonne façon de considérer l'arithmétique et, par voie de conséquence, nous poser la question « L'incomplétude fait-elle partie de l'arithmétique ou est-elle seulement liée à la formalisation de Peano ? ».
Bien entendu, cela dépend de ce que l'on considère comme étant *l'arithmétique*. Dans ce qui suit, nous nous baserons sur l'idée que l'arithmétique est "la" théorie ayant pour modèle $< \mathbb{N}, 0, S, +, \times >$.

Définition 5.1.1. Soient $\mathcal{L}$ un langage fini et $\mathfrak{M}$ une $\mathcal{L}$-structure.
On dit que $\mathfrak{M}$ est *fortement indécidable* si toute théorie sur $\mathcal{L}$ dont $\mathfrak{M}$ est modèle est indécidable.

Le résultat suivant va donner un élément de réponse à notre question initiale.

Théorème 5.1.2. $\mathbb{N}$ *est fortement indécidable*

Démonstration. Soient H la conjonction des axiomes de $\mathcal{P}_0$ et T une théorie ayant $\mathbb{N}$ comme modèle.
Procédons par l'absurde en supposant que T est décidable.
Par définition, nous avons alors que $Th(T)$ est récursif et que l'ensemble

$$\mathcal{X} = \{\#F : \#(H \Rightarrow F) \in Th(T)\}$$

est également récursif. En effet,

$$\chi_X(x) = \chi_{Th(T)}(\alpha_3(\#H, x, 4))$$

et l'on connait le code de $\#H$.

Montrons que $\mathcal{X} = Th(T \cup \mathcal{P}_0)$.

- Soit F un théorème de T. Vu l'axiome L_1, on a $T \vdash F \Rightarrow (H \Rightarrow F)$ et, par conséquent, $T \vdash H \Rightarrow F$. Donc, $\#F \in \mathcal{X}$.
  De plus, on a trivialement que tout théorème de $\mathcal{P}_0$ est dans $\mathcal{X}$.
- Soit $\#F \in \mathcal{X}$. On a donc que $T \vdash H \Rightarrow F$. Or, par le lemme de déduction, on a $T \cup \{H\} \vdash F$ ce qui suffit.

Nous avons donc que $Th(T \cup \mathcal{P}_0)$ est récursif. Or, nous avons que $T \cup \mathcal{P}_\prime$ est une théorie consistante (car ayant $\mathbb{N}$ comme modèle), contenant $\mathcal{P}_0$ et donc, vu le théorème 2.4.3, indécidable.
La contradiction en découle et T est donc indécidable. ■

Nous en concluons

Corollaire 5.1.3. *Toute théorie récursive ayant* $\mathbb{N}$ *comme modèle est incomplète.*

Démonstration. Cela découle du résultat 2.3.31. ■

Nous avons donc que toute axiomatisation de l'arithmétique en une théorie du premier ordre complète et récursive est vouée à l'échec.

Nous avons répondu à notre question initiale dans le cas des théories récursives. Serait-il plus judicieux de considérer des théories non récursives pour espérer leur complétude ?
La question reste ouverte, nous attirons cependant l'attention du lecteur sur le fait que des théories non récursives apparaissent hautement non-naturelles. En effet, peut-on considérer comme valable une axiomatisation de l'arithmétique telle qu'il est impossible de vérifier de manière effective qu'une formule est un axiome de la théorie ou qu'elle est simplement bien formulée ?

Les résultats qui suivent ont pour but de démontrer que la théorie des anneaux est indécidable.

Théorème 5.1.4. *Soient* $\mathcal{L}$ *un langage fini et* $\mathfrak{M}$ *une* $\mathcal{L}$*-structure.*
Si $\mathbb{N}$ *est définissable dans* $\mathfrak{M}$ *alors* $\mathfrak{M}$ *est fortement indécidable.*

Démonstration. Vu que $\mathbb{N}$ est définissable dans $\mathfrak{M}$, on a que toute formule F de $\mathcal{L}_0$ possède une représentation F^* dans $\mathfrak{M}$[1]. De plus, le passage de F à F^* est effectif. Il existe donc une fonction récursive g telle que $g(\#F) = \#F^*$.

Soit T_0 une théorie dont $\mathfrak{M}$ est modèle. Tout d'abord, montrons que toute théorie contenant $T_0 \cup \{H^*\}$ est indécidable (où H est toujours la conjonction des axiomes de Robinson).
Posons $T = T_0 \cup \{H^*\}$ et

$$T^- = \{F; F \text{ formule close de } \mathcal{L}_0 \text{ et } T \vdash F^*\}$$

et montrons que $T^- \vdash G$ si et seulement si $T \vdash G^*$ pour toute formule close G de $\mathcal{L}_0$.

- Supposons que $T^- \vdash G$ et, en vue d'obtenir une contradiction, que $T \cup \{\neg G^*\}$ a un modèle $\mathfrak{A}$.
  Soit $\mathfrak{N}$ la $\mathcal{L}$-structure définissable dans $\mathfrak{A}$. Vu que $\mathfrak{A} \vDash T$, on a que $\mathfrak{N} \vDash T^-$ et donc $\mathfrak{N} \vDash G$.
  Or, $\mathfrak{A} \vDash \neg G^*$ et, par conséquent, $\mathfrak{N} \vDash \neg G$, d'où la contradiction.
- Supposons que $T \vdash G^*$. On a donc que $G \in T^-$ et donc $T^- \vdash G$

Cette équivalence implique que si T était décidable, T^- le serait également vu que le passage de $\#F$ à $\#F^*$ est récursif. Or, T^- est une théorie consistante, contenant $\mathcal{P}_0$. Elle est donc indécidable et T l'est également.

A présent, montrons que T_0 est indécidable.
Il suffit de remarquer que, pour toute formule F de $\mathcal{L}_0$, $T_0 \cup \{H^*\} \vdash F$ si et seulement si $T_0 \vdash H^* \vdash F$ et que le caractère indécidable de $T_0 \cup \{H^*\}$ implique que T_0 est indécidable. Vu que T_0 est une théorie quelconque ayant $\mathfrak{M}$ comme modèle, on a la conclusion. ■

Proposition 5.1.5. $\mathbb{N}$ *est définissable dans* $\mathbb{Z}$

Démonstration. Pour définir les entiers naturels, nous utiliserons un théorème de Lagrange exprimant que tout nombre naturel est égal à une somme de quatre carrés de nombres entiers. Nous posons donc les formules suivantes

- $H_{\mathbb{N}}[v_0] \equiv \exists v_1 \exists v_2 \exists v_3 \exists v_4 (v_0 \simeq ((v_1 \underline{\times} v_1) \underline{+} (v_2 \underline{\times} v_2) \underline{+} (v_3 \underline{\times} v_3) \underline{+} (v_4 \underline{\times} v_4))$.

1. En effet, cela peut se démontrer en montrant que tout terme, toute formule atomique et enfin toute formule est représentable. Et tout cela découle du fait que $\mathbb{N}$ est définissable dans $\mathfrak{M}$ i.e. qu'il existe un système de formule de la forme

$$\mathcal{H} =< H_{\mathbb{N}}, H_0, \{H_S, H_+, H_\times\} > .$$

- $H_0[v_0] \equiv v_0 \simeq \underline{0}$.
- $H_S[v_0, v_1] \equiv H_{\mathbb{N}}[v_0] \wedge H_{\mathbb{N}}[v_1] \wedge \exists v_2 \forall v_3 (v_2 \underline{\times} v_3 \simeq v_3 \wedge v_0 \simeq v_1 \underline{+} v_2)$.
- $H_+[v_0, v_1, v_2] \equiv H_{\mathbb{N}}[v_0] \wedge H_{\mathbb{N}}[v_1] \wedge H_{\mathbb{N}}[v_2] \wedge v_0 \simeq v_1 \underline{+} v_2$.
- $H_\times[v_0, v_1, v_2] \equiv H_{\mathbb{N}}[v_0] \wedge H_{\mathbb{N}}[v_1] \wedge H_{\mathbb{N}}[v_2] \wedge v_0 \simeq v_1 \underline{\times} v_2$.

où l'on a "réutilisé" les symboles 0, + et × de $\mathbb{Z}$. ■

5.2 Les limites de $\mathcal{P}_0$

Toujours dans le deuxième chapitre, nous avons démontré le premier théorème d'incomplétude de Gödel pour l'arithmétique de Robinson, notée $\mathcal{P}_0$, n'acceptant pas le schéma d'induction.
Dans cette section nous allons exhiber quelques limites de $\mathcal{P}_0$ par des théorèmes de $\mathcal{P}$ indémontrable dans Robinson. Cela nous permettra de mieux nous rendre compte du tour de force que constitue le premier théorème de Gödel.

Théorème 5.2.1. *Nous avons les propriétés suivantes :*

1. *L'addition n'est pas commutative dans $\mathcal{P}_0$.*
2. *La multiplication n'est pas associative dans $\mathcal{P}_0$.*
3. $$\mathcal{P}_0 \nvdash \forall v_0 \forall v_1 ((v_0 \leq v_1 \wedge v_1 \leq v_0) \Rightarrow v_0 \simeq v_1).$$
4. $\mathcal{P}_0 \nvdash \forall v_0 (\underline{0} \times v_0 \simeq \underline{0})$.

Démonstration. Nous voyons qu'il suffit de trouver un modèle adéquat pour démontrer ces quatre propriétés.
Tout d'abord, posons le modèle.
Soient X un ensemble non vide et une fonction $f : X \times X \to X$. Nous posons $\mathfrak{M}$ la $\mathcal{L}_0$-structure dont l'univers est $\mathbb{N} \cup (X \times \mathbb{Z})$ et où les symboles $\underline{S}, \underline{+}$ et $\underline{\times}$ sont interprétés par les fonctions S, + et . définies de la façon suivantes :

- Si $a = (x, n) \in M \setminus \mathbb{N}$ alors $S(a) = (x, n + 1)$.
- Si $a = (x, n) \in M \setminus \mathbb{N}$ et $m \in \mathbb{N}$ alors $a + m = m + a = (x, n + m)$.
- Si $a = (x, n) \in M \setminus \mathbb{N}$ et $b = (y, m) \in M \setminus \mathbb{N}$ alors $(x, n) + (y, m) = (x, n + m)$.
- Si $a = (x, n) \in M \setminus \mathbb{N}$ et $m \in \mathbb{N}$ alors $(x, n).m = (x, n.m)$ si $m \neq 0$ et $(x, n).m = 0$ sinon.
- Si $a = (x, n) \in M \setminus \mathbb{N}$ et $m \in \mathbb{N}$ alors $m.(x, n) = (x, n.m)$.

– Si $a = (x, n) \in M \setminus \mathbb{N}$ et $b = (y, m) \in M \setminus \mathbb{N}$ alors $(x, n).(y, m) = (f(x, y), n.m)$.

À présent, montrons que $\mathfrak{M}$ est un modèle de $\mathcal{P}_0$.

Il suffit de vérifier que les sept axiomes sont valides dans $\mathfrak{M}$ ce qui ne présente pas une grande difficulté. À titre d'exemple, nous allons vérifier A_5 et A_7 les deux axiomes les plus difficiles à vérifier.

– A_5 : Montrons que $\mathfrak{M} \vDash \forall a \forall b(a + S(b) = S(a + b))$.
  - Si $a, b \in \mathbb{N}$, c'est évident.
  - Si $a, b \in M \setminus \mathbb{N}$, on a $a + S(b) = (x, n) + (y, m + 1) = (x, n + m + 1) = S((x, n + m)) = S(a + b)$.
  - Si $a \in M \setminus \mathbb{N}$ et $b \in \mathbb{N}$, on a $b + (x, n + 1) = (x, n + b + 1) = S(b + (x, n))$.
  - Sinon, si $a \in \mathbb{N}$ et $b \in M \setminus \mathbb{N}$, le raisonnement est similaire.

– A_7 : Montrons que $\mathfrak{M} \vDash \forall a \forall b(a.S(b) = (a.b) + a)$.
  - Si $a, b \in \mathbb{N}$, c'est évident.
  - Si $a, b \in M \setminus \mathbb{N}$, on a $(x, n).(y, m + 1) = (f(x, y), n.(m + 1)) = (f(x, y), n.m) + (x, n)$.
  - Si $a \in M \setminus \mathbb{N}$ et $b \in \mathbb{N}$, on a $(x, n).(b + 1) = (x, n.(b + 1)) = (x, n.b) + (x, n) = ((x, n).b) + (x, n)$ en tenant compte que b peut valoir 0.
  - Sinon, si $a \in \mathbb{N}$ et $b \in M \setminus \mathbb{N}$, $a.(x, n + 1) = (x, a.(n + 1)) = (x, a.n + a) = (x, a.n) + a = (a.(x, n)) + a$.

Maintenant que nous savons que $\mathfrak{M}$ est un modèle de $\mathcal{P}_0$, nous pouvons démontrer les quatre propriétés voulues en posant $X = \mathbb{N}$ et $f(x, y) = 2y + x$.

1. Soit $x \neq y$. On a $(x, n) + (y, m) = (x, n + m) \neq (y, m) + (x, n) = (y, n + m)$.
2. On a $[(1, 1).(2, 2)].(3, 3) = (2.2 + 1, 2).(3, 3) = (6 + 5, 6) = (11, 6)$.
   Or, $(1, 1).[(2, 2).(3, 3)] = (1, 1).(6 + 2, 6) = (17, 6)$.
3. Soit $m \neq n$. On a donc $(x, n) + (x, m - n) = (x, m)$ et $(x, m) + (x, n - m) = (x, n)$.
   Par conséquent, $(x, n) \leq (x, m)$ et $(x, m) \leq (x, n)$. Or, $(x, n) \neq (x, m)$.
4. $0.(x, n) = (x, 0) \neq 0$.

■

Notons qu'il est également possible, avec un autre modèle de $\mathcal{P}_0$, de montrer que l'addition n'est pas associative dans cette théorie.

Bibliographie

[1] Zofia Adamowicz et Pawel Zbierski, *Logic of Mathematics : a modern course of classical logic*, Wiley-Interscience, 1997.

[2] Guillaume Amand, *Le théorème de Goodstein : un indécidable de PA.*, Thèse de Maîtrise, Faculté des Sciences. Département de mathématiques., 2006–2007.

[3] C. Calude, *Information and Randomness*, 2^{e} éd., Springer, 2002.

[4] C. Calude, H. Jürgensen, et M. Zimand, *Is independence an exception ?*, Appl. Math. Comput. **66** (1994), n^{o} 1, 63–76.

[5] René Cori et Daniel Lascar, *Logique mathématique cours et exercices I. Calcul propositionnel, algèbres de Boole, calcul des prédicats.*, Masson, 1993.

[6] ———, *Logique mathématique cours et exercices II. Fonctions récursives, théorème de Gödel, théorie des ensembles, théorie des modèles.*, Masson, 1993.

[7] Georges Hansoul, *Théorie des ensembles et Logique mathématique*, Université de Liège, Faculté des Sciences, 2009.

[8] David Hilbert et Paul Bernays, *Fondements des Mathématiques*, vol. 2, L'Harmattan, 2001, traduit de l'ouvrage "Grundlagen der Mathematik" 2.

[9] L. Kirby et J. Paris, *Accessible independence results for peano arithmetic*, Bulletin of the London Math. Society **14** (1982), 285–293.

[10] Piergiorgio Odifreddi, *Classical Recursion Theory : the theory of functions and sets of natural numbers*, North Holland, 1999.

[11] Michel Rigo, *Algorithmique et Calculabilité*, Université de Liège, Faculté des Sciences, 2006-2007.

[12] Hartley Rogers, *Theory of recursive functions and effective computability*, McGraw-Hill, 1987.

[13] Raymond Smullyan, *Les théorèmes d'incomplétude de Gödel*, Masson, 1993.

[14] R. Solovay, *A Version of ω for which ZFC can not predict a single bit*, Finite Versus Infinite : Contributions to an Eternal Dilemma (C. Calude et G. Paun, éds), Springer-Verlag, 2000, p. 323–334.

[15] Ann Yasuhara, *Recursive function theory and logic*, Academic Press, New York, 1971.