Théorèmes limitatifs concernant la théorie de l'arithmétique

Simon Lemal

11 mai 2021

Table des matières

Introduction

Nous allons ici aborder deux problèmes intéressants concernant la logique du premier ordre.

Le premier problème consiste à déterminer si, étant donné un ensemble d'axiomes dans un langage donné, il existe un procédure permettant de déterminer de manière automatique si une phrase est une conséquence des ces axiomes ou non. Nous verrons que la réponse à ce premier problème est négative.

Le second problème auquel nous nous intéressons consiste à trouver, dans le langage de l'arithmétique, un ensemble d'axiomes consistant, ne contenant qu'une quantité finie d'information et telle que toute phrase est soit un théorème, soit la négation d'un théorème. Nous verrons qu'ici aussi, la réponse est négative.

Dans la première section, nous rappelons quelques notions de calculabilité. Ensuite, dans la deuxième section, nous présentons un codage des phrases d'un langage, ce qui permet d'énoncer des théorèmes à propos de phrases au sein même du langage de l'arithmétique. Nous introduisons également la notion de théorie axiomatisable et démontrons quelques résultats immédiats. Dans la troisième section, nous présentons des constructions qui permettent de transformer une fonction calculable en une formule du langage de l'arithmétique. À nouveau, cela permet de formuler des théorèmes à propos de fonctions calculables au sein même du langage de l'arithmétique. Dans la quatrième section, nous nous intéressons aux problèmes évoqués dans le paragraphe précédent. Finalement, dans la dernière section, nous présentons quelques compléments sur les outils développés dans les sections précédentes.

1 Quelques notions de calculabilité

Dans cette section, nous nous intéressons à des fonctions de $\mathbb{N}^p$ dans $\mathbb{N}$. Nous introduisons une définition formelle de la notion de fonction effectivement calculable.

1.1 Fonctions récursives

Intuitivement, une fonction $f\colon \mathbb{N}^p \to \mathbb{N}$ est *calculable* s'il existe un algorithme, un programme, permettant de la calculer effectivement.

Exemple. Voici quelques exemples de fonctions calculables.

- La fonction identité $\mathrm{id}\colon \mathbb{N} \to \mathbb{N}$ est calculable
- La fonction successeur $'\colon \mathbb{N} \to \mathbb{N} \quad x \mapsto x+1$ est calculable.
- La fonction nulle $\mathbf{0}\colon \mathbb{N}^p \to \mathbb{N} \quad x \mapsto 0$ est calculable.
- La fonction $+\colon \mathbb{N}^2 \to N$ est calculable.
- La fonction $\log_2\colon \mathbb{N} \to \mathbb{N} \quad x \mapsto \max\{k \in \mathbb{N} : 2^k \text{ divise } x\}$ est calculable. En effet,

```
int log(int x){
    int k;
    for (k=0; k++; !(x%2)) x = x/2;
    return k;
}
```

  est une procédure (en C) calculant $\log_2$.

Cependant, cette notion de fonction calculable n'est pas formelle. Différentes définitions formelles ont été proposées (e.g. machines de Turing, λ-calcul, fonctions récursives). La définition que nous utilisons ici est due à Gödel et Herbrand.

Définition 1.1. La classe des fonctions *récursives* est la plus petite classe de fonctions contenant

- la fonction nulle $\mathbf{0}\colon \mathbb{N}^p \to \mathbb{N} \quad x \mapsto 0$,
- la fonction successeur $'\colon \mathbb{N} \to \mathbb{N} \quad x \mapsto x+1$,
- les projections $\mathrm{id}^p_k\colon \mathbb{N}^p \to \mathbb{N} \quad (x_1, \dots, x_p) \mapsto x_k$,

et stable pour les opérations

- composition : si $f_1, \dots, f_q\colon \mathbb{N}^p \to \mathbb{N}$ et $g\colon \mathbb{N}^q \to \mathbb{N}$ sont récursifs, alors la fonction

$$g \circ (f_1, \dots, f_q)\colon \mathbb{N}^p \to \mathbb{N} \quad (x_1, \dots, x_p) \mapsto g(f_1(x_1, \dots, x_p), \dots, f_q(x_1, \dots, x_p))$$

  est récursive,
- récursion : si $f\colon \mathbb{N}^p \to \mathbb{N}$ est récursif et $g\colon \mathbb{N}^{p+2} \to \mathbb{N}$ est récursif, la fonction $\rho(f,g)\colon \mathbb{N}^{p+1} \to \mathbb{N}$, définie par

$$\begin{aligned} \rho(f,g)(0, x_1, \dots, x_p) &= f(x_1, \dots, x_p) \\ \rho(f,g)(y', x_1, \dots, x_p) &= g(y, \rho(f,g)(y, x_1, \dots, x_p), x_1, \dots, x_p), \end{aligned}$$

  est récursive,
- minimisation : si $f\colon \mathbb{N}^{p+1} \to \mathbb{N}$ est récursif, alors la fonction $\mu(f)\colon \mathbb{N}^p \to \mathbb{N}$, définie par

$$\begin{aligned} \mu(f)(x_1, \dots, x_p) = y \text{ ssi } & f(y, x_1, \dots, x_p) = 0 \\ & \text{et } f(z, x_1, \dots, x_p) > 0 \text{ pour tout } z < y, \end{aligned}$$

  est récursive (s'il n'existe pas de tel y ou si $f(z, x_1, \dots, x_p)$ n'est pas défini, alors $\mu(f)(x_1, \dots, x_n)$ n'est pas défini).

Exemple. La fonction $+\colon (x,y) \mapsto x+y$ est récursive. En effet, comme $0+y=y$ et $x'+y=(x+y)'$, on a $+ = \rho(\mathrm{id}^1_1, ' \circ \mathrm{id}^3_2)$.

Nous pouvons donner une définition alternative de la classe des fonctions récursives.

Proposition 1.2. *Une fonction f est récursive si et seulement s'il existe une suite de fonctions $f_0, f_1, \ldots, f_n$ telle que $f_n = f$ et pour tout $i \leq n$, soit f_i est la fonction nulle, la fonction successeur ou une des projections, soit f_i est obtenu à partir des f_j pour $j < i$ par composition, récursion ou minimisation.*

Démonstration. Il est évident que la classe $\mathcal{C}$ des fonctions f pour lesquelles il existe une suite $f_0, \ldots, f_n$ vérifiant les conditions de l'énoncé est incluse dans la classe des fonctions récursives. De plus, $\mathcal{C}$ contient les fonctions nulle, successeur et projections et est stable par composition, récursion et minimisation. Comme la classe des fonctions récursives est la plus petite classe à avoir cette propriété, elle est égale à $\mathcal{C}$. □

Corollaire 1.3. *La classe des fonctions récursives est dénombrable.*

Démonstration. Tout fonction récursive peut s'écrire sous la forme d'une chaîne de caractère sur l'alphabet dénombrable $\{\mathbf{0}, ', \circ, \rho, \mu, (, , ,), \mathrm{id}_1^1, \mathrm{id}_1^2, \mathrm{id}_2^2, \ldots\}$. Or l'ensemble des mots finis sur un alphabet dénombrable est dénombrable. □

Enfin, nous considérons le postulat suivant.

Thèse 1.4 (Church)**.** *Une fonction est calculable si et seulement si elle est récursive.*

Cette thèse ne peut bien évidemment pas être démontrée étant donnée que la notion de fonction calculable n'est pas clairement définie, ni même fixe dans le temps. Cependant, il existe des arguments en faveur de cette thèse.

D'abord, il est évident qu'une fonction récursive est calculable. En effet, les fonctions nulle, successeur et projections sont calculables et les opérateurs $\circ, \rho$ et μ appliqués à des fonctions calculables donnent des fonctions calculables. Ainsi, la décomposition d'une fonction récursive en fonctions élémentaires fournit une procédure effective pour calculer cette fonction.

Ensuite, à ce jour, aucune fonction calculable mais non récursive n'a été trouvée. On peut trouver dans la littérature des démonstrations que telle ou telle fonction calculable est récursive. C'est par exemple le cas de $\log_2$ (la démonstration se trouve dans [2]).

Finalement, certains logiciens, tels que Turing ou Church, ont proposé d'autres définitions pour formaliser la notion de fonction calculable. Ils ont ensuite montré que ces différentes définitions étaient équivalentes à celle de fonctions récursives.

Dans la suite, nous supposerons que le thèse de Church est vérifiée et ne feront plus la distinction entre les fonctions calculables et les fonctions récursives. Cela permet de raccourcir certaines démonstrations qui autrement sont fort laborieuses.

1.2 Ensembles décidables

On peut naturellement étendre la notion de récursivité aux ensembles et relations sur les naturels.

Définition 1.5. Un ensemble $S \subset \mathbb{N}^p$ est *décidable* si sa fonction caractéristique est récursive.

Une relation sur $\mathbb{N}$ n'étant qu'une sous-ensemble de $\mathbb{N}^p$, on parle également de relations *récursives*.

Définition 1.6. Un ensemble $S \subset \mathbb{N}^p$ est *semirécursif* s'il existe une relation récursive $R \subset \mathbb{N}^{p+1}$ telle que $(x_1, \ldots, x_p) \in S$ si et seulement si $\exists y\, R(y, x_1, \ldots, x_p)$.

Remarque. La notion d'ensemble semirécursif peut s'interpréter. Un ensemble S est semirécursif s'il existe un procédure qui, appliquée au p-uple $(x_1, \ldots, x_p)$, renvoie "vrai" en un temps fini si et seulement si $(x_1, \ldots, x_p) \in S$. Par contre, si $(x_1, \ldots, x_p) \notin S$, la procédure peut ne jamais se terminer.

Remarque. Au vu du corollaire 1.3, il n'existe qu'une infinité dénombrable de sous-ensembles décidables de $\mathbb{N}$. Comme il existe une infinité non dénombrable de sous-ensemble de $\mathbb{N}$, on en déduit l'existence d'ensembles non décidables.

Pour définir la notion de décidabilité pour d'autres ensembles que les sous-ensembles de $\mathbb{N}$, nous introduisons la notion de codage.

Définition 1.7. Soit N un ensemble dénombrable. Un *codage* est une injection $j\colon N \to \mathbb{N}$ telle que $j(N)$ est décidable.

Exemple. La fonction $j\colon \mathbb{N}^2 \to \mathbb{N} \quad (x,y) \mapsto 2^x 3^y$ est un codage. La fonction définie sur l'ensemble des suites finies de $\mathbb{N}$ qui à $(x_i)_{i\leq n}$ associe

$$2^{x_0+1}3^{x_1+1}\dots p_n^{x_n+1}$$

est un codage. La fonction qui à une chaîne de caractère associe son code ASCII est un codage.

Définition 1.8. Soit N un ensemble dénombrable et j un codage pour cet ensemble. Un ensemble $S \subset N$ est *décidable* si l'ensemble de ses codes, i.e. $j(S)$, est décidable.

Exemple. Par exemple, si Σ est un alphabet et que l'on dispose d'un codage de Σ^*, tout langage régulier est décidable.

2 Arithmétisation de la logique

La première étape de la démonstration des théorèmes limitatifs consiste à fournir un codage pour les formules de la logique du premier ordre.

2.1 Codage de Gödel

Pour coder les formules de la logique, on encode les différents symboles utilisés pour écrire ces formules selon la table suivantes. Nous démontrons ensuite quelques théorèmes utiles.

	symboles logiques				variables	prédicats				fonctions			
	(	¬	∃	=	v_0	P_0^0	P_0^1	P_0^2	…	f_0^0	f_0^1	f_0^2	…
symboles	)	∨	∀		v_1	P_1^0	P_1^1	P_1^2	…	f_1^0	f_1^1	f_1^2	…
	,	∧			⋮	⋮	⋮	⋮		⋮	⋮	⋮	
	1	2	3	4	5	6	68	688	…	7	78	788	…
codes	19	29	39		59	69	689	6889	…	79	789	7889	…
	199	299			⋮	⋮	⋮	⋮		⋮	⋮	⋮	

TABLE 1 – Symboles et leurs codes

Définition 2.1. À toute formule ϕ on associe le nombre $\ulcorner\phi\urcorner$ obtenu en concaténant les codes des symboles de ϕ.

Exemple. Si ϕ est la formule $\forall x \exists y\, y \neq x$, qui s'écrit formellement sous la forme $\forall x \exists y \neg = (y, x)$, on a, en supposant que x et y sont les deux premières variables,

$$\ulcorner\phi\urcorner = 39\ 5\ 3\ 59\ 2\ 4\ 1\ 59\ 199\ 5\ 19.$$

On se convainc sans problème que $\ulcorner\cdot\urcorner$ est un codage. De plus, on peut montrer que l'ensemble des formules est décidable.

Proposition 2.2. *L'ensemble des codes de formules (fermées) est décidable.*

Idée de la preuve. Les formules étant définie de manière récursive, on se convainc facilement qu'une procédure récursive permet de déterminer les codes de formules (fermées). Il est également possible d'avoir un raisonnement plus formel et de montrer que le langage des codes de formules est décidable par une machine de Turing. Il faut ensuite montrer que les ensembles décidables par une machine de Turing sont récursifs. □

On peut également montrer que les fonctions qui au code d'une formule associe le code de sa négation ou qui aux codes de deux formules associe le code de leur conjonction, etc. sont récursives.

Lemme 2.3. *Soient*

$$\begin{aligned}
\text{not}\colon &\ \mathbb{N} \to \mathbb{N} && \ulcorner\phi\urcorner \mapsto \ulcorner\neg\phi\urcorner,\\
\text{or}\colon &\ \mathbb{N}^2 \to \mathbb{N} && (\ulcorner\phi\urcorner, \ulcorner\psi\urcorner) \mapsto \ulcorner(\phi \vee \psi)\urcorner,\\
\text{and}\colon &\ \mathbb{N}^2 \to \mathbb{N} && (\ulcorner\phi\urcorner, \ulcorner\psi\urcorner) \mapsto \ulcorner(\phi \wedge \psi)\urcorner,\\
\text{exists}\colon &\ \mathbb{N}^2 \to \mathbb{N} && (\ulcorner\phi\urcorner, \ulcorner x\urcorner) \mapsto \ulcorner\exists x\, \phi\urcorner,\\
\text{forall}\colon &\ \mathbb{N}^2 \to \mathbb{N} && (\ulcorner\phi\urcorner, \ulcorner x\urcorner) \mapsto \ulcorner\forall x\, \phi\urcorner.
\end{aligned}$$

Les fonctions not, or, and, exists, forall *sont récursives.*

Idée de la preuve. On a

$$\begin{aligned}
\text{not}(n) &= 2 * n,\\
\text{or}(m, n) &= 1 * m * 29 * n * 19,\\
\text{exists}(m, n) &= 3 * n * m,\\
&\vdots
\end{aligned}$$

où $*$ est l'opération de concaténation. Il suffit donc de montrer que $*$ est récursif. Nous n'en donnons pas une preuve formelle, mais la thèse de Church permet de s'en convaincre facilement. □

2.2 Théories axiomatisables

Pour formuler nos premiers résultats, nous avons besoin de la définition de preuve. Nous ne donnons pas ici une définition complète.

Définition 2.4. Sans rentrer dans les détails, si ϕ est une phrase et Γ un ensemble de phrases, une *preuve* de ϕ à partir de Γ est une suite de formules $\phi_1 \dots \phi_n$ telle que ϕ_n n'est autre que ϕ et telles que, pour tout $i \leq n$, on a

— ϕ_i est un axiome, i.e. une formule qui respecte un certain schéma,

— $\phi_i \in \Gamma$ ou

— il existe $j, k < i$ tels que ϕ_k n'est autre que $\phi_j \to \phi_i$.

Remarque. Nous avons vu dans un des exemples de la première section comment coder les suites finies de $\mathbb{N}$. Maintenant que nous savons coder des formules, nous pouvons également coder les preuves, qui ne sont que des suites finies de formules.

Les deux résultats suivants ne sont pas démontrés rigoureusement ici. Cependant, le corollaire est important pour la suite.

Proposition 2.5. *Si Γ est un ensemble de phrases récursif, alors la relation "$\phi_1 \dots \phi_n$ est une preuve de ϕ à partir de Γ" est récursive.*

Idée de la preuve. Étant donné une preuve $\phi_1 \dots \phi_n$, il suffit de vérifier que chaque formule est soit dans Γ, soit un axiome, soit une conséquence des formules précédentes. Comme Γ est récursif, la première vérification peut être faite efficacement. La deuxième peut également se faire efficacement si les axiomes sont bien choisis (c'est en général le cas, on prend pour axiomes des formules vérifiant un schéma particulier). La troisième vérification peut également se faire efficacement car il suffit de parcourir les formules précédentes. □

Corollaire 2.6. *L'ensemble des formules déductibles d'un ensemble récursif de formules est semirécursif.*

Idée de la preuve. Si l'ensemble Γ est récursif, il est possible d'énumérer efficacement toutes les preuves à partir de Γ. Ainsi, pour vérifier si une formule ϕ est conséquence de Γ, il suffit de parcourir toutes les preuves et de vérifier que l'une d'elle est une preuve de ϕ. □

Nous donnons maintenant les définitions de théorie et de théorie axiomatisable.

Définition 2.7. Une *théorie* est un ensemble de phrases T telle que si ϕ est un phrase prouvable à partir de T, alors $\phi \in T$.

Définition 2.8. Une théorie T est *axiomatisable* s'il existe un ensemble récursif Γ de phrases, appelées *axiomes* tel que T est l'ensemble des formules qui sont conséquences de Γ.

Les théories utilisables en pratique sont axiomatisables. En effet, les seules théories que nous pouvons décrire doivent contenir une quantité finie d'information. En pratique, les théories utilisées sont définies par un ensemble fini d'axiomes et un ensemble fini de schémas d'axiomes.

Le corollaire précédent peut se reformuler comme ceci : toute théorie axiomatisable est récursive.

Nous introduisons également les notions de théories complètes et consistantes.

Définition 2.9. Une théorie est *consistante* si elle ne contient pas toutes les formules, i.e. si pour toute formule ϕ, les formules ϕ et $\neg\phi$ ne sont jamais vrais simultanément.

Une théorie T est *complète* si pour toute formule ϕ, on a $\phi \in T$ ou $\neg\phi \in T$.

Remarque. Ces deux notions sont complémentaires. On a notamment que si $T' \subset T$ sont deux théories avec T' complet et T consistant, alors $T' = T$. En effet, si $\phi \in T$, alors $\neg\phi \notin T$ car T est consistant. Alors $\neg\phi \notin T'$ donc $\phi \in T'$ car T' est complet. En particulier, une théorie consistante et complète est maximale parmi les théories consistantes et minimale parmi les théories complètes.

Le théorème d'incomplétude de Gödel affirme que si on se restreint aux théories axiomatisables qui permettent de faire de l'arithmétique, il n'existe pas de théorie consistante et complète.

Remarque. Si T est une théorie consistante et complète, il existe un modèle $\mathcal{M}$ tel que les phrases de T sont exactement les phrases vraies dans $\mathcal{M}$. En effet, comme la théorie est consistante, par le théorème de complétude, il existe un modèle $\mathcal{M}$ tel que $T \subset \mathrm{Th}(\mathcal{M})$ où

$$\mathrm{Th}(\mathcal{M}) = \{\phi \text{ est une phrase vraie dans } \mathcal{M}\}.$$

Cependant T est complet et $\mathrm{Th}(\mathcal{M})$ est consistant donc $T = \mathrm{Th}(\mathcal{M})$.

Réciproquement, si $\mathcal{M}$ est un modèle, la théorie $\mathrm{Th}(\mathcal{M})$ est consistante et complète.

Remarque. Étant donné un langage, on peut définir une relation d'équivalence sur les phrases du langage par $\phi \sim \psi$ si $\phi \leftrightarrow \psi$ est démontrable. On peut alors munir l'ensemble des classes d'équivalence d'une structure d'algèbre de Boole. Les théories du langage correspondent aux filtres (non vides) sur l'algèbre. Une théorie est consistante si et seulement si le filtre associé est propre et une théorie est complète si et seulement si le filtre associé est premier. Ainsi, les théories consistantes et complètes correspondent exactement aux ultrafiltres.

Le résultat suivant montre que les théories axiomatisables, consistantes sont d'un grand intérêt.

Théorème 2.10. *Si T est une théorie axiomatisable, consistante et complète, alors T est décidable.*

Démonstration. Comme T est axiomatisable, on sait que l'ensemble des phrases vraies est semirécursif, i.e. il existe une relation récursive R telle que ϕ est vrai si et seulement si

$$\exists y\, R(y, \ulcorner \phi \urcorner).$$

Comme T est complet, on a toujours $\phi \vee \neg\phi$ donc

$$\exists y\, (R(y, \ulcorner \phi \urcorner) \vee R(y, \ulcorner \neg\phi \urcorner))$$

est vrai. Une procédure effective pour déterminer si $\phi \in T$ consiste donc à trouver le plus petit $y \in \mathbb{N}$ tel que $R(y, \ulcorner \phi \urcorner)$ ou $R(y, \ulcorner \neg\phi \urcorner)$. On sait alors que $\phi \in T$ si et seulement si $R(y, \ulcorner \phi \urcorner)$ (car T est consistant). □

3 Représentabilité des fonctions récursives

La deuxième étape de la preuve des théorèmes limitatifs consiste à exprimer dans le langage de l'arithmétique des relations du type $f(x) = y$ où f est récursif. Commençons par définir le langage de l'arithmétique.

Définition 3.1. Le langage de l'*arithmétique* est le langage contenant la constante $\mathbf{0}$, la fonction unaire $'$, les fonctions binaires $+$ et $\cdot$, et le prédicat binaire $<$.

Dans la suite, on écrira $\mathbf{m}$ pour $\mathbf{0}^{\prime\cdots\prime}$ (où $'$ apparaît m fois).

3.1 Arithmétique standard et fonctions arithmétiques

Nous introduisons maintenant la théorie standard de l'arithmétique et allons montrer que les fonctions récursives peuvent être représentées dans cette théorie.

Définition 3.2. Le modèle *standard* du langage de l'arithmétique est $\mathbb{N}$, où $\mathbf{0}$ est le 0 de $\mathbb{N}$, $'$ le successeur, $+$ l'addition, $\cdot$ la multiplication et $<$ l'ordre (stricte) des naturels. On désigne par $\mathbf{N}$ la théorie des phrases vraies dans ce modèle.

Une formule est *correcte* si elle est vrai dans l'interprétation standard.

Définition 3.3. Une relation $R \subset \mathbb{N}^p$ est *arithmétique* s'il existe une formule $\phi(x_1, \ldots, x_p)$ telle que $(n_1, \ldots, n_p) \in R$ si et seulement si $\phi(\mathbf{n_1}, \ldots, \mathbf{n_p})$ est correct.

Par extension, une fonction $f\colon \mathbb{N}^p \to \mathbb{N}$ est *arithmétique* si son graphe est arithmétique, i.e. s'il existe une formule $\phi(x_1, \ldots, x_p, y)$ telle que pour tout $n_1, \ldots, n_p, m \in \mathbb{N}$, on a $f(n_1, \ldots, n_p) = m$ si et seulement si $\phi(\mathbf{n_1}, \ldots, \mathbf{n_p}, \mathbf{m})$ est correct. On dit que la formule ϕ définit f.

Exemple. Les fonctions suivantes sont arithmétiques.

- La fonction nulle $\mathbf{0}$ est définie par $y = \mathbf{0}$,
- La fonction successeur $'$ est définie par $y = x'$,
- Les projections id_k^p sont définies par $y = x_k$.

Nous allons montrer que toutes les fonctions récursives sont arithmétiques.

Lemme 3.4. *Si $f_1, \ldots, f_q\colon \mathbb{N}^p \to \mathbb{N}$ et $g\colon \mathbb{N}^q \to \mathbb{N}$ sont arithmétiques, alors $g \circ (f_1, \ldots, f_q)$ est arithmétique. Si $f\colon \mathbb{N}^{p+1} \to \mathbb{N}$ est arithmétique, alors $\mu(f)$ est arithmétique.*

Démonstration. Soient $\phi_1(x_1, \ldots, x_p, y_1), \ldots, \phi_q(x_1, \ldots, x_p, y_q)$ des formules définissant respectivement $f_1, \ldots, f_q$ et $\psi(y_1, \ldots, y_q, z)$ une formule définissant g. Alors $\chi(x_1, \ldots, x_p, z)$ défini par

$$\exists y_1 \exists \ldots \exists y_q \left(\phi_1(x_1, \ldots, x_p, y_1) \wedge \cdots \wedge \phi_q(x_1, \ldots, x_p, y_q) \wedge \psi(y_1, \ldots, y_q, z)\right)$$

définit $g \circ (f_1, \ldots, f_q)$.

Si $\phi(y, x_1, \ldots, x_p, z)$ définit f, alors $\chi(x_1, \ldots, x_p, y)$ défini par

$$\phi(y, x_1, \ldots, x_p, \mathbf{0}) \wedge \forall u(u < y \to \exists v \left(\phi(u, x_1, \ldots, x_p, v) \wedge v \neq \mathbf{0}\right))$$

définit $\mu(f)$. □

Pour montrer qu'une fonction construite par récursion à partir de fonctions arithmétiques, nous aurons besoin du lemme suivant.

Lemme 3.5. *Pour tout $k \in \mathbb{N}$ et toute suite $(a_i)_{0 \leq i \leq k}$, il existe $s, t \in \mathbb{N}$ tels que pour tout $i \in \{0, \ldots, k\}$, a_i est le reste de la division euclidienne de s par $t(i+1)+1$.*

Démonstration. Montrons qu'il existe $t \in \mathbb{N}$ tel que $a_i < t(i+1)+1$ pour tout i et tel que les $t(i+1)+1$ soient premiers entre eux. Nous pourrons alors appliquer le théorème des restes chinois.

Soit $n \in \mathbb{N}$ tel que $n \geq k$ et $n \geq a_i$ pour tout i. Alors $t = n!$ convient. D'une part,

$$a_i \leq t < t(i+1)+1$$

pour tout i. D'autre part, si p est un diviseur commun de $t(i+1)+1$ et $t(j+1)+1$ avec $0 \leq i < j \leq k$, alors p ne divise pas t car sinon il ne diviserait pas $t(i+1)+1$. En particulier $p > n$. Or p divise

$$(t(j+1)+1) - (t(i+1)+1) = t(j-i)$$

donc p divise $j-i$. C'est absurde car $j - i \leq k \leq n$. □

Nous sommes maintenant en mesure de démontrer que toutes les fonctions récursives sont arithmétiques.

Théorème 3.6. *Toute fonction récursive est arithmétique.*

Démonstration. Il reste à montrer que si $f\colon \mathbb{N}^p \to \mathbb{N}$ et $g\colon \mathbb{N}^{p+2} \to \mathbb{N}$ sont arithmétique, alors $\rho(f,g)$ l'est aussi. Soient ϕ et ψ des formules définissant respectivement f et g.

Alors $\rho(f,g)(k, x_1, \ldots, x_p) = z$ si et seulement s'il existe une suite $(z_i)_{0 \leq i \leq k}$ telle que

$$\begin{aligned} z_0 &= f(x_1, \ldots, x_p), \\ z_{i+1} &= g(i, z_i, x_1, \ldots, x_p) \text{ pour tout } i < k, \\ z_k &= z. \end{aligned}$$

En effet, si une telle suite existe, alors $z_i = \rho(f,g)(i, x_1, \ldots, x_p)$ pour tout $i \leq k$ (par récurrence). Si $\rho(f,g)(k, x_1, \ldots, x_p) = z$, la suite $(\rho(f,g)(i, x_1, \ldots, x_p)_{0 \leq i \leq k}$ convient.

C'est équivalent à demander l'existence d'une suite $(z_i)_{0 \leq i \leq k}$ telle que

$$\begin{aligned} &\phi(x_1, \ldots, x_p, z_0), \\ &\psi(i, z_i, x_1, \ldots, x_p, z_{i+1}) \text{ pour tout } i < k, \\ &z_k = z. \end{aligned}$$

Vu le lemme précédent, c'est équivalent à demander l'existence de $s, t \in \mathbb{N}$ tels que

$$\exists q_0 \exists z_0 \, (z_0 < t+1 \ \wedge \ s = (t+1)q_0 + z_0 \ \wedge \ \phi(x_1, \ldots, x_p, z_0)),$$

$$\begin{aligned} \forall i (i < k \to \exists q_i \exists z_i \exists q_{i+1} \exists z_{i+1} \, &(z_i < t \cdot (i+1) + 1 \ \wedge \ s = (t \cdot (i+1) + 1) \cdot q_i + z_i \\ &\wedge \ z_{i+1} < t \cdot (i+2) + 1 \ \wedge \ s = (t \cdot (i+2) + 1) \cdot q_{i+1} + z_{i+1} \ \wedge \ \psi(i, z_i, x_1, \ldots, x_p, z_{i+1}))) \end{aligned}$$

et

$$\exists q_k \exists z_k \, (z_k < t(k+1) + 1 \ \wedge \ s = (t(k+1)+1)q_0 + z_0 \ \wedge \ z_k = z).$$

La conjonction de ces trois formules précédée de $\exists s \exists t$ définit $\rho(f,g)$. □

Corollaire 3.7. *Toute relation récursive est arithmétique.*

Démonstration. Soit $R \subset \mathbb{N}^p$ un ensemble récursif et $\phi(x_1, \ldots, x_p, y)$ une fonction définissant sa fonction caractéristique. Alors $\phi(x_1, \ldots, x_p, \mathbf{1})$ définit R. □

3.2 Formules rudimentaires et fonctions rudimentaires

Nous allons maintenant renforcer le résultat de la sous-section précédente en montrant que les formules représentant des fonctions récursives sont toujours équivalentes à des formules d'une certaine forme.

Définition 3.8. Une formule *rudimentaire* est une formule construite uniquement à partir de négations, conjonctions, disjonctions et quantifications bornées (i.e. $\forall x < y$ et $\exists x < y$, où $\forall x < y \, \phi(x)$ est une abréviation pour $\forall x \, (x < y \to \phi(x))$ et $\exists x < y \, \phi(x)$ est une abréviation pour $\exists x \, (x < y \wedge \phi(x))$).

Une formule $\exists$-*rudimentaire* est une formule de la forme $\exists x \, \phi(x)$ où ϕ est rudimentaire.

Une formule $\exists$-*rudimentaire généralisée* est une formule obtenue par conjonctions, disjonctions, quantifications bornées et quantifications existentielles non bornées à partir de formules rudimentaires.

En inspectant la preuve du théorème 3.6, on se rend compte que les formules construites sont $\exists$-rudimentaires généralisées. Nous avons donc le résultat suivant.

Proposition 3.9. *Toute fonction récursive est définissable par une formule $\exists$-rudimentaire généralisée.*

Nous pouvons encore renforcer ce résultat en montrant que toute formule ∃-rudimentaire généralisée est arithmétiquement équivalente à une formule ∃-rudimentaire. Deux formules $\phi(x)$ et $\psi(x)$ sont arithmétiquement équivalentes si $\phi(\mathbf{a})$ est correcte si et seulement si $\psi(\mathbf{a})$ est correcte, pour tout nombre a. Pour cela, on a le résultat suivant.

Lemme 3.10 (Propriétés de fermeture)**.** *(i) Toute formule rudimentaire est équivalente à une formule ∃-rudimentaire,*

(ii) une conjonction de formules ∃-rudimentaires est arithmétiquement équivalente à une formule ∃-rudimentaire,

(iii) une disjonction de formules ∃-rudimentaires est équivalente à une formule ∃-rudimentaire,

(iv) la formule obtenue par quantification universelle bornée d'une formule ∃-rudimentaire est arithmétiquement équivalente à une formule ∃-rudimentaire,

(v) la formule obtenue par quantification existentielle bornée d'une formule ∃-rudimentaire est équivalente à une formule ∃-rudimentaire,

(vi) la formule obtenue par quantification existentielle d'une formule ∃-rudimentaire est arithmétique équivalente à une formule ∃-rudimentaire.

Démonstration. (i) La formule ϕ est logiquement équivalent à $\exists t\,(t = t \wedge \phi)$.

(ii) La formule $\exists u\,\phi(u) \wedge \exists v\ \psi(v)$ est arithmétiquement équivalent à $\exists w\,(\exists u < w\,\phi(u) \wedge \exists v < w\,\psi(v))$. En effet, on peut toujours trouver un nombre plus grand que deux nombres donnés.

(iii) La formule $\exists u\,\phi(u) \vee \exists v\,\psi(v)$ est logiquement équivalent à $\exists w\,(\phi(w) \vee \psi(w))$.

(iv) La formule $\forall u < v\,\exists t\,\phi(u,t)$ est arithmétiquement équivalent à $\exists w \forall u < v\,\exists t < w\,\phi(u,t)$. En effet, on peut toujours trouver un nombre majorant une famille finie de nombre.

(v) La formule $\exists u < v\,\exists t\,\phi(u,t)$ est logiquement équivalent à $\exists t \exists u < v\,\phi(u,t)$.

(vi) La formule $\exists u \exists v\,\phi(u,v)$ est arithmétiquement équivalent à $\exists w \exists u < w\,\exists v < w\,\phi(u,v)$.

□

En procédant par induction sur la complexité, nous pouvons montrer que toute formule ∃-rudimentaire généralisée est arithmétiquement équivalente à une formule ∃-rudimentaire. La proposition 3.9 peut donc être améliorée.

Proposition 3.11. *Toute fonction récursive est définissable par une formule ∃-rudimentaire.*

Finalement, cette proposition permet de démontrer un résultat qui sera utile par la suite. Commençons par la définition de fonction rudimentaire.

Définition 3.12. Une fonction $f\colon \mathbb{N}^p \to \mathbb{N}$ est *rudimentaire* si elle est définissable par une formule rudimentaire, i.e. s'il existe une formule rudimentaire $\phi(x_1, \ldots, x_p, y)$ telle que pour tout $n_1, \ldots, n_p, m \in \mathbb{N}$, on a $f(n_1, \ldots, n_p) = m$ si et seulement si $\phi(\mathbf{n_1}, \ldots, \mathbf{n_p}, \mathbf{m})$ est correct.

Proposition 3.13. *Toute fonction récursive peut s'obtenir comme composition de fonctions rudimentaires.*

Démonstration. Soit $f\colon \mathbb{N}^p \to \mathbb{N}$ une fonction récursive. Cette fonction est définissable par une formule ∃-rudimentaire $\exists z \phi(x_1, \ldots, x_p, y, z)$. Soit R la relation définie par ϕ, i.e.

$$(n_1, \ldots, n_p, m, q) \in R \quad \text{ssi} \quad \phi(\mathbf{n_1}, \ldots, \mathbf{n_p}, \mathbf{m}, \mathbf{q}) \text{ est correct.}$$

Définissons les fonctions auxiliaires

$$g(n_1, \ldots, n_p) = \begin{cases} \text{le plus petit } r \text{ tel que} \\ \qquad \exists y < r \exists z < r\, R n_1 \ldots n_p y z & \text{si un tel } r \text{ existe} \\ \text{indéfini} & \text{sinon} \end{cases}$$

et

$$h(n_1, \ldots, n_p, r) = \begin{cases} \text{le plus petit } m < r \text{ tel que} \\ \qquad \exists z < r\, R n_1 \ldots n_p m z & \text{si un tel } m \text{ existe} \\ \text{indéfini} & \text{sinon.} \end{cases}$$

On a $f(n_1, \ldots, n_p) = h(n_1, \ldots, n_p, g(n_1, \ldots, n_p))$ donc $f = h \circ (\mathrm{id}_1^p, \ldots, \mathrm{id}_p^p, g)$. En effet, si

$$g(n_1, \ldots, n_p) = r,$$

alors $h(n_1, \ldots, n_p, r)$ est défini. Si

$$h(n_1, \ldots, n_p, r) = m,$$

alors il existe $q \in \mathbb{N}$ tel que

$$(n_1, \ldots, n_p, m, q) \in R,$$

i.e.

$$\phi(\mathbf{n_1}, \ldots, \mathbf{n_p}, \mathbf{m}, \mathbf{p})$$

est correct. Mais alors $f(n_1, \ldots, n_p) = m$. Réciproquement si $f(n_1, \ldots, n_p)$ est défini alors $g(n_1, \ldots, n_p)$ est défini.

De plus g et h sont rudimentaires. La fonction g est définie par la formule $\psi(x_1, \ldots, x_p, t)$ définie par

$$\exists y < t \exists z < t \, \phi(x_1, \ldots, x_p, y, z) \wedge \forall u < t \forall y < u \forall z < u \, \neg\phi(x_1, \ldots, x_p, y, z)$$

et la fonction h est définie par la formule $\chi(x_1, \ldots, x_p, t, y)$ définie par

$$y < t \wedge \exists z < t \, \phi(x_1, \ldots, x_p, y, z) \wedge \forall u < y \forall z < u \, \neg\phi(x_1, \ldots, x_p, y, z).$$

Les fonctions id_k^p sont également rudimentaires dont la conclusion est immédiate. □

3.3 Arithmétique minimale et représentabilité

Nous allons encore renforcer les résultats établis précédemment. Pour cela, nous allons adapter le concept de fonction arithmétique à une théorie plus faible que la théorie standard de l'arithmétique.

Définition 3.14. La théorie *minimale* de l'arithmétique, notée **Q**, est l'ensemble des formules impliquées par les axiomes suivants (par souci de lisibilité, nous omettons les quantificateurs universelles au début de chaque formule).

(Q1) $\mathbf{0} \neq x'$,
(Q2) $x' = y' \to x = y$,
(Q3) $x + \mathbf{0} = x$,
(Q4) $x + y' = (x + y)'$,
(Q5) $x \cdot \mathbf{0} = \mathbf{0}$,
(Q6) $x \cdot y' = (x \cdot y) + x$,
(Q7) $x \not< 0$,
(Q8) $x < y' \leftrightarrow (x < y \vee x = y)$,
(Q9) $\mathbf{0} < y \leftrightarrow y \neq \mathbf{0}$,
(Q10) $x' < y \leftrightarrow (x < y \wedge x' \neq y)$.

Cette théorie est trop faible que pour démontrer la plupart des résultats de l'arithmétique. Cependant, ses axiomes sont élémentaires et toute théorie de l'arithmétique convenable doit contenir **Q**. De plus, nous verrons que toute formule $\exists$-rudimentaire est vraie dans **Q** si elle l'est dans **N**.

La notion de relation arithmétique s'adapte aux théories plus faible.

Définition 3.15. Soit T une théorie dans le langage de l'arithmétique. Une relation $R \subset \mathbb{N}^p$ est *définissable* dans la théorie T s'il existe une formule $\phi(x_1, \ldots, x_p)$ telle que $(n_1, \ldots, n_p) \in R$ implique que $\phi(\mathbf{n_1}, \ldots, \mathbf{n_p})$ est un théorème de T et $(n_1, \ldots, n_p) \notin R$ implique que $\neg\phi(\mathbf{n_1}, \ldots, \mathbf{n_p})$ est un théorème. On dit que la formule ϕ définit R dans T.

Remarque. Une relation est arithmétique si et seulement si elle est définissable dans la théorie standard de l'arithmétique. C'est le cas car la théorie standard de l'arithmétique est complète et consistante.

Nous définissons également une notion plus forte pour les fonctions totales.

Définition 3.16. Soit T une théorie dans le langage de l'arithmétique. Une fonction $f \colon \mathbf{N}^p \to \mathbf{N}$ est *représentable* dans la théorie T s'il existe une formule $\phi(x_1, \ldots, x_p, y)$ telle que $f(n_1, \ldots, n_p) = m$ implique que

$$\forall y \, (\phi(\mathbf{n_1}, \ldots, \mathbf{n_p}, y) \leftrightarrow y = \mathbf{m})$$

est un théorème de T. On dit que la formule ϕ représente f dans T.

Remarque. La phrase $\forall y\,(\phi(\mathbf{n_1},\ldots,\mathbf{n_p},y)\leftrightarrow y=\mathbf{m})$ est logiquement équivalente à la conjonction de

$$\phi(\mathbf{n_1},\ldots,\mathbf{n_p},\mathbf{m})$$

avec

$$\forall y\,(y\neq\mathbf{m}\rightarrow\neg\phi(\mathbf{n_1},\ldots,\mathbf{n_p},y)).$$

Cette dernière phrase est plus forte que la conjonction des phrases $\neg\phi(\mathbf{n_1},\ldots,\mathbf{n_p},\mathbf{q})$ pour $q\neq m$. En effet, nous verrons dans la dernière section de ce travail qu'il existe des modèles de $\mathbf{Q}$ pour lesquels des formules $\phi(x)$ aussi simples que $x+1=1+x$ telles que les phrases $\phi(\mathbf{0}),\phi(\mathbf{1}),\ldots$ sont vraies mais que la phrase $\forall x\,\phi(x)$ est fausse.

Nous allons montrer que les fonctions récursives sont représentables dans $\mathbf{Q}$. La première étape de la preuve consiste à démontrer, comme nous l'avons suggéré précédemment, que toutes les formules $\exists$-rudimentaires correctes sont des théorèmes de l'arithmétique minimale.

Théorème 3.17. *Une formule $\exists$-rudimentaire est correcte si et seulement si c'est un théorème de $\mathbf{Q}$.*

Démonstration. Comme tout axiome de $\mathbf{Q}$ est correct, tout théorème de $\mathbf{Q}$ est correct donc la condition est suffisante. Nous postposons la preuve de la nécessité à la sous-section suivante. □

Ce théorème peut être démontré directement. Cependant, nous présentons dans la sous-section suivante une manière plus générale de le démontrer.

En utilisant le résultat précédent, nous pouvons passer à l'étape suivante, qui consiste à montrer que les fonctions rudimentaires sont représentables.

Lemme 3.18. *Toute fonction rudimentaire est représentable dans $\mathbf{Q}$ par une formule rudimentaire.*

Démonstration. Les axiomes (Q7) et (Q8) permettent de montrer, par une récurrence directe sur m, que

$$x<\mathbf{m}\leftrightarrow(x=\mathbf{0}\vee\cdots\vee x=\mathbf{m}-\mathbf{1}).$$

Les axiomes (Q9) et (Q10), quant à eux, permettent de montrer que

$$\mathbf{m}<x\leftrightarrow(x\neq\mathbf{0}\wedge\cdots\wedge x\neq\mathbf{m}).$$

Ainsi

$$\forall x\,(x<\mathbf{m}\vee x=\mathbf{m}\vee\mathbf{m}<x) \tag{1}$$

est un théorème de $\mathbf{Q}$.

Soit $f\colon\mathbb{N}^p\rightarrow\mathbb{N}$ une fonction rudimentaire et $\phi(x_1,\ldots,x_p,y)$ la formule la définissant. Définissons $\psi(x_1,\ldots,x_p,y)$ par

$$\phi(x_1,\ldots,x_p,y)\wedge\forall z<y\,\neg\phi(x_1,\ldots,x_p,z)$$

et montrons que ψ représente f.

Supposons que $f(n_1,\ldots,n_p)=m$. Comme ϕ définit f dans $\mathbf{N}$, les phrases $\phi(\mathbf{n_1},\ldots,\mathbf{n_p},\mathbf{m})$ et $\neg\phi(\mathbf{n_1},\ldots,\mathbf{n_p},\mathbf{q})$ sont correctes pour tout $q\neq m$, et en particulier pour tout $q<m$. Ainsi, la phrase

$$\forall z<\mathbf{m}\,\neg\phi(\mathbf{n_1},\ldots,\mathbf{n_p},z) \tag{2}$$

est correcte donc $\psi(\mathbf{n_1},\ldots,\mathbf{n_p},\mathbf{m})$ est correcte. Comme il s'agit d'une phrase rudimentaire, c'est un théorème de $\mathbf{Q}$.

Il faut ensuite montrer que

$$y\neq\mathbf{m}\rightarrow\neg(\phi(\mathbf{n_1},\ldots,\mathbf{n_p},y)\wedge\forall z<y\,\neg\phi(\mathbf{n_1},\ldots,\mathbf{n_p},z))$$

(le quantificateur $\forall y$ est omis). C'est équivalent à

$$\phi(\mathbf{n_1},\ldots,\mathbf{n_p},y)\rightarrow(y=\mathbf{m}\vee\exists z<y\,\phi(\mathbf{n_1},\ldots,\mathbf{n_p},z)).$$

Pour montrer cela, il est suffisant de montrer

$$\phi(\mathbf{n_1},\ldots,\mathbf{n_p},y)\rightarrow(y=\mathbf{m}\vee\mathbf{m}<y)$$

puisque $\phi(\mathbf{n_1},\ldots,\mathbf{n_p},\mathbf{m})$ est un théorème. C'est une conséquence directe des équations 1 et 2. □

La troisième étape consiste à montrer que les compositions de fonctions rudimentaires sont représentables dans **Q**.

Proposition 3.19. *Toute composition de fonctions rudimentaires est représentable dans* **Q** *par une formule* $\exists$*-rudimentaire.*

Démonstration. Soient $f_1, \ldots, f_q \colon \mathbb{N}^p \to \mathbb{N}$ et $g \colon \mathbb{N}^q \to \mathbb{N}$ des fonctions rudimentaires et $\phi_1(x_1, \ldots, x_p, y_1)$, $\ldots$, $\phi_q(x_1, \ldots, x_p, y_q)$ les formules les représentant. Considérons la formule $\chi(x_1, \ldots, x_p, z)$, définie par

$$\exists y_1 \exists \ldots \exists y_q \, (\phi_1(x_1, \ldots, x_p, y_1) \wedge \cdots \wedge \phi_q(x_1, \ldots, x_p, y_q) \wedge \psi(y_1, \ldots, y_q, z)).$$

Montrons que χ représente h. Si $f_1(n_1, \ldots, n_p) = m_1, \ldots, f_q(n_1, \ldots, n_p) = m_q$ et $g(m_1, \ldots, m_q) = r = h(n_1, \ldots, n_p)$, alors

$$\begin{gathered} \phi_1(\mathbf{n_1}, \ldots, \mathbf{n_p}, y) \leftrightarrow y = \mathbf{m_1} \\ \vdots \\ \phi_q(\mathbf{n_1}, \ldots, \mathbf{n_p}, y) \leftrightarrow y = \mathbf{m_q} \end{gathered}$$

et

$$\psi(\mathbf{m_1}, \ldots, \mathbf{m_q}, z) \leftrightarrow z = \mathbf{r}$$

sont des théorèmes de **Q** (à nouveau, les quantificateurs sont omis). Mais les q premières équations impliquent directement que

$$\begin{aligned} \exists y_1 \ldots \exists y_q \, (\phi_1(\mathbf{n_1}, \ldots, \mathbf{n_p}, y_1) \wedge \cdots \wedge \phi_q(\mathbf{n_1}, \ldots, \mathbf{n_p}, y_q) \wedge \psi(y_1, \ldots, y_q, z)) \\ \leftrightarrow \exists y_1 \ldots \exists y_q \, (y_1 = \mathbf{n_1} \wedge \cdots \wedge y_q = \mathbf{n_q} \wedge \psi(y_1, \ldots, y_q, z)) \end{aligned}$$

est un théorème. Or, il est évident que

$$\exists y_1 \ldots \exists y_q \, (y_1 = \mathbf{n_1} \wedge \cdots \wedge y_q = \mathbf{n_q} \wedge \psi(y_1, \ldots, y_q, z)) \leftrightarrow \psi(\mathbf{m_1}, \ldots, \mathbf{m_q}, z)$$

est un théorème. Ainsi, par définition de ψ, on sait que

$$\exists y_1 \ldots \exists y_q \, (\phi_1(\mathbf{n_1}, \ldots, \mathbf{n_p}, y_1) \wedge \cdots \wedge \phi_q(\mathbf{n_1}, \ldots, \mathbf{n_p}, y_q) \wedge \psi(y_1, \ldots, y_q, z)) \leftrightarrow z = \mathbf{r}$$

est un théorème. □

Théorème 3.20. *Toute fonction récursive est représentable dans* **Q** *par une formule* $\exists$*-rudimentaire.*

Démonstration. C'est une conséquence directe de la proposition 3.13 et de la proposition précédente. □

Théorème 3.21. *Toute relation récursive est définissable dans* **Q** *par une formule* $\exists$*-rudimentaire.*

Démonstration. Soit $R \subset \mathbb{N}^p$ et $\phi(x_1, \ldots, x_p, y)$ une formule représentant sa fonction caractéristique. Si $(n_1, \ldots, n_p) \in R$, alors

$$\phi(\mathbf{n_1}, \ldots, \mathbf{n_p}, y) \leftrightarrow y = \mathbf{1}$$

est un théorème de **Q** donc en particulier, $\phi(\mathbf{n_1}, \ldots, \mathbf{n_p}, \mathbf{1})$ est un théorème. Si $(n_1, \ldots, n_p) \notin R$, alors

$$\phi(\mathbf{n_1}, \ldots, \mathbf{n_p}, y) \leftrightarrow y = \mathbf{0}$$

est un théorème de **Q** donc en particulier, $\neg\phi(\mathbf{n_1}, \ldots, \mathbf{n_p}, \mathbf{1})$ est un théorème. Ainsi $\phi(x_1, \ldots, x_p, \mathbf{1})$ définit R. □

3.4 Modèles et plongements

Comme dit précédemment, nous allons présenter un cadre plus général pour démontrer le théorème 3.17. Nous commençons par présenter la notion de morphisme de modèle.

Définition 3.22. Soient L un langage et $\mathcal{N}, \mathcal{M}$ des modèles de ce langage. Un *morphisme* est une application $\mu \colon \mathcal{N} \to \mathcal{M}$ telle que

— pour toute constante c de L, on a
$$\mu\left(c^{\mathcal{N}}\right) = c^{\mathcal{M}},$$

— pour toute fonction k-aire f et $a_1, \ldots, a_k \in |\mathcal{N}|$, on a
$$\mu\left(f^{\mathcal{N}}(a_1, \ldots, a_k)\right) = f^{\mathcal{M}}(\mu(a_1), \ldots, \mu(a_k)),$$

— pour tout prédicat k-aire P et $a_1, \ldots, a_k \in |\mathcal{N}|$, on a
$$P^{\mathcal{N}}(a_1, \ldots, a_k) \text{ si et seulement si } P^{\mathcal{M}}(\mu(a_1), \ldots, \mu(a_k)).$$

Un *plongement* est un morphisme injectif.

Il est immédiat que les morphismes préservent les termes fermés.

Proposition 3.23. *Si $\mu \colon \mathcal{N} \to \mathcal{M}$ est un morphisme, alors pour tout terme fermé t, on a*
$$t^{\mathcal{M}} = \mu\left(t^{\mathcal{N}}\right).$$

Démonstration. Il suffit de procéder par récurrence sur la complexité de t. □

Les morphismes préservent également les formules atomiques n'utilisant pas l'identité.

Lemme 3.24. *Si $\mu \colon \mathcal{N} \to \mathcal{M}$ est un morphisme, alors pour toute phrase ϕ de la forme $P(t_1, \ldots, t_k)$, on a*
$$\mathcal{M} \models \phi \text{ si et seulement si } \mathcal{N} \models \phi.$$

Démonstration. C'est évident vu la définition d'un morphisme. □

Les plongements, quant à eux, préservent les formules utilisant l'identité.

Lemme 3.25. *Si $\iota \colon \mathcal{N} \to \mathcal{M}$ est un plongement, alors pour toute phrase ϕ de la forme $t_1 = t_2$, on a*
$$\mathcal{M} \models \phi \text{ si et seulement si } \mathcal{N} \models \phi.$$

Démonstration. En effet, vu l'injectivité de ι, on a
$$\mathcal{M} \models \phi \quad \text{ssi} \quad t_1^{\mathcal{M}} = t_2^{\mathcal{M}} \quad \text{ssi} \quad \iota(t_1^{\mathcal{N}}) = \iota(t_2^{\mathcal{N}}) \quad \text{ssi} \quad t_1^{\mathcal{N}} = t_2^{\mathcal{N}} \quad \text{ssi} \quad \mathcal{N} \models \phi.$$

□

Ainsi, les plongements conservent toutes les phrases sans quantificateurs.

Proposition 3.26. *Si $\iota \colon \mathcal{N} \to \mathcal{M}$ est un plongement, alors pour toute phrase ϕ ne contenant pas de quantificateur, on a*
$$\mathcal{M} \models \phi \text{ si et seulement si } \mathcal{N} \models \phi.$$

Démonstration. Il suffit de procéder par récurrence sur la complexité de ϕ, le cas de base découlant des deux lemmes précédents. □

De plus, comme on pourrait le deviner, les plongements conservent les propriétés existentielles.

Proposition 3.27. *Si $\iota \colon \mathcal{N} \to \mathcal{M}$ est un plongement, alors pour toute formule $\phi(x_1, \ldots, x_n)$ libre en les variables $x_1, \ldots, x_n$ et ne contenant pas de quantificateur, on a*
$$\mathcal{N} \models \exists x_1 \ldots \exists x_n \, \phi(x_1, \ldots, x_n) \text{ implique } \mathcal{M} \models \exists x_1 \ldots \exists x_n \, \phi(x_1, \ldots, x_n).$$

Démonstration. On procède par récurrence sur n. Le cas $n = 0$ est traité par la proposition précédente.

Supposons que la propriété est vraie pour n et montrons la pour $n + 1$. On a

$$\mathcal{N} \models \exists x_0 \exists x_1 \dots \exists x_n \, \phi(x_0, x_1, \dots, x_n)$$

si et seulement s'il existe $n \in |\mathcal{N}|$ tel que

$$\mathcal{N}_n^{x_0} \models \exists x_1 \dots \exists x_n \, \phi(c, x_1, \dots, x_n)$$

où $\mathcal{N}_n^{x_0}$ est le modèle obtenu en remplaçant la variable x_0 par une constante c dont l'interprétation est n.

Or, $\phi(c, x_1, \dots, x_n)$ est libre en les variables $x_1, \dots, x_n$ et ι est un plongement de $\mathcal{N}_n^{x_0}$ dans $\mathcal{M}_{\iota(n)}^{x_0}$ donc

$$\mathcal{M}_{\iota(n)}^{x_0} \models \exists x_1 \dots \exists x_n \, \phi(c, x_1, \dots, x_n),$$

ce qui implique

$$\mathcal{M} \models \exists x_0 \exists x_1 \dots \exists x_n \, \phi(x_0, x_1, \dots, x_n).$$

□

Nous allons maintenant considérer ces résultats dans le cas particulier de l'arithmétique.

Proposition 3.28. *Soit $\mathcal{M}$ un modèle de $\mathbf{Q}$. Alors l'application*

$$\iota \colon \mathbb{N} \to \mathcal{M} \quad n \mapsto \mathbf{n}^{\mathcal{M}}$$

est un plongement.

Démonstration. L'axiome (Q1) permet de montrer chacune des phrases $\mathbf{0} \neq \mathbf{1}$, $\mathbf{0} \neq \mathbf{2}$, $\mathbf{0} \neq \mathbf{3}$, etc, car $\mathbf{1}, \mathbf{2}, \mathbf{3}, \dots$ se terminent par $'$. Ensuite, l'axiome (Q2) permet de prouver que $\mathbf{m} \neq \mathbf{n}$ si $m \neq n$. Ainsi ι est injectif.

Clairement $\iota(0) = \mathbf{0}$ et $\iota(n') = \iota(n)'$. Les axiomes (Q3) et (Q4) permettent de montrer, par une récurrence directe sur m, que $\iota(n + m) = \iota(n) + \iota(m)$. En procédant de manière similaire, les axiomes (Q5) et (Q6) impliquent que $\iota(n \cdot m) = \iota(n) \cdot \iota(m)$.

Finalement, montrons que $n < m$ si et seulement si $\iota(n) < \iota(m)$, par récurrence sur m. Le cas de base est acquis vu (Q7). Supposons le résultat acquis pour m, alors l'axiome (Q8) implique

$$\begin{aligned} n < m' \quad \text{ssi} \quad n < m \text{ ou } n = m \quad \text{ssi} \quad \iota(n) < \iota(m) \text{ ou } \iota(n) = \iota(m) \\ \text{ssi} \quad \iota(n) < \iota(m)' \quad \text{ssi} \quad \iota(n) < \iota(m'). \end{aligned}$$

Ainsi ι est bien un plongement. □

Nous pouvons généraliser le résultat 3.26 aux phrases rudimentaires.

Proposition 3.29. *Si $\mathcal{M}$ est un modèle de $\mathbf{Q}$, alors pour toute phrase rudimentaire, on a*

$$\mathcal{M} \models \phi \text{ si et seulement si } \mathbb{N} \models \phi.$$

Démonstration. Nous procédons comme dans la preuve de la proposition 3.26. Il reste à traiter les cas où ϕ est de la forme $\exists u < t \, \psi(u)$ ou $\forall u < \mathbf{n} \, \psi(v)$ où t est un terme fermé. Il existe alors un $m \in \mathbb{N}$ tel que $\mathbb{N} \models t = \mathbf{m}$. En procédant par récurrence sur m, on a $u < \mathbf{n}$ si et seulement si $u = \mathbf{0} \vee u = \mathbf{1} \vee \dots \vee u = \mathbf{m} - \mathbf{1}$. Ainsi,

$$\begin{aligned} \mathcal{M} \models \exists u < t \, \psi(u) \quad \text{ssi} \quad \mathcal{M} \models \exists u < \mathbf{m} \, \psi(u) \quad \text{ssi} \quad \mathcal{M} \models \psi(\mathbf{0}) \vee \psi(\mathbf{1}) \vee \dots \vee \psi(\mathbf{m} - \mathbf{1}) \\ \text{ssi} \quad \mathbb{N} \models \psi(\mathbf{0}) \vee \psi(\mathbf{1}) \vee \dots \vee \psi(\mathbf{m} - \mathbf{1}) \quad \text{ssi} \quad \mathbb{N} \models \exists u < \mathbf{m} \, \psi(u) \quad \text{ssi} \quad \mathbb{N} \models \exists u < t \, \psi(u) \end{aligned}$$

et

$$\begin{aligned} \mathcal{M} \models \forall u < t \, \psi(u) \quad \text{ssi} \quad \mathcal{M} \models \forall u < \mathbf{m} \, \psi(u) \quad \text{ssi} \quad \mathcal{M} \models \psi(\mathbf{0}) \vee \psi(\mathbf{1}) \vee \dots \vee \psi(\mathbf{m} - \mathbf{1}) \\ \text{ssi} \quad \mathbb{N} \models \psi(\mathbf{0}) \vee \psi(\mathbf{1}) \vee \dots \vee \psi(\mathbf{m} - \mathbf{1}) \quad \text{ssi} \quad \mathbb{N} \models \forall u < \mathbf{m} \, \psi(u) \quad \text{ssi} \quad \mathbb{N} \models \forall u < t \, \psi(u). \end{aligned}$$

□

Nous pouvons maintenant démontrer le théorème 3.17.

Preuve du théorème 3.17. Si $\mathcal{M}$ est un modèle de $\mathbf{Q}$, alors en procédant comme dans la preuve de la proposition 3.27, on peut montrer que pour toute formule $\exists$-rudimentaire ϕ, on a

$$\mathbb{N} \models \phi \text{ implique } \mathcal{M} \models \phi.$$

Ainsi, pour toute formule $\exists$-rudimentaire $\phi \in \mathbf{N}$, on a $\phi \in \mathbf{Q}$. □

4 Indécidabilité et incomplétude

Nous sommes maintenant en mesure de démontrer les résultats mentionnés dans l'introduction.

4.1 Diagonalisation et théorème du point fixe

Nous définissons la diagonalisation d'une formule et démontrons un théorème concernant cette diagonalisation. Tous les résultats qui nous intéressent vont découler de ce théorème.

Définition 4.1. Étant donné une formule ϕ, on définit sa *diagonalisation* par $\exists x\,(x = \ulcorner\phi\urcorner \wedge \phi)$. Cette notion est particulièrement intéressante dans le cas d'une formule $\phi(x)$ dont la variable x est libre. En effet, dans ce cas, la diagonalisation de ϕ est équivalente à $\phi(\ulcorner\phi\urcorner)$.

Théorème 4.2 (Théorème du point fixe)**.** *Soit T une théorie contenant $\mathbf{Q}$. Alors pour toute formule $\beta(y)$ il existe une phrase γ telle que $\gamma \leftrightarrow \beta(\ulcorner\gamma\urcorner)$ est un théorème de T.*

Démonstration. La fonction $\mathrm{diag}\colon \mathbb{N} \to \mathbb{N} \quad \ulcorner\phi\urcorner \mapsto \ulcorner\exists x\,(x = \ulcorner\phi\urcorner \wedge \phi)\urcorner$ est récursive. Ainsi, il existe une formule $\delta(x, y)$ la représentant dans T.

Soit $\alpha(x)$ la formule

$$\exists y\,(\delta(x, y) \wedge \beta(y)).$$

Soit $a = \ulcorner\alpha\urcorner$ et γ la phrase

$$\exists x\,(x = \mathbf{a} \wedge \alpha(x)).$$

La phrase γ est logiquement équivalente à $\alpha(\mathbf{a})$, i.e.

$$\exists y\,(\delta(\mathbf{a}, y) \wedge \beta(y)).$$

Soit $g = \ulcorner\gamma\urcorner$. Comme γ est la diagonalisation de α, on a $\mathrm{diag}(a) = g$ donc

$$\forall y(\delta(\mathbf{a}, y) \leftrightarrow y = \mathbf{g}).$$

Ainsi $\alpha(\mathbf{a})$ est logiquement équivalent à

$$\exists y\,(y = \mathbf{g} \wedge \beta(y)),$$

qui est équivalent à $\beta(\mathbf{g})$. La phrase $\gamma \leftrightarrow \beta(\ulcorner\gamma\urcorner)$ est donc un théorème. □

4.2 Théorèmes limitatifs

Maintenant que le théorème du point fixe est acquis, nous pouvons résoudre les problèmes présentés dans l'introduction.

Théorème 4.3 (Tarski)**.** *Soit T une théorie consistante contenant $\mathbf{Q}$. L'ensemble des codes des théorèmes de T n'est pas définissable dans T.*

Démonstration. Supposons qu'il existe une formule $\nu(x)$ définissant l'ensemble des théorèmes. Par le théorème du point fixe, il existe une formule γ telle que

$$\gamma \leftrightarrow \neg\nu(\ulcorner\gamma\urcorner) \tag{3}$$

est un théorème de T.

Si γ n'est pas un théorème, alors $\neg\nu(\ulcorner\gamma\urcorner)$ en est un puisque ν défini l'ensemble des codes de phrases correctes. Mais alors γ est un théorème vu (3).

On sait donc que γ est un théorème, donc $\nu(\ulcorner\gamma\urcorner)$ aussi. Mais par (3), $\neg\nu(\ulcorner\gamma\urcorner)$ est un théorème. C'est impossible si T est consistant. □

Remarque. La phrase γ de la preuve précédente affirme d'elle même qu'elle est fausse. Elle est comparable à la phrase "Cette phrase est fausse." du paradoxe du menteur.

Théorème 4.4 (Indécidabité)**.** *Soit T une théorie consistante contenant $\mathbf{Q}$. L'ensemble des théorème de T n'est pas décidable.*

Démonstration. C'est évident au vu du théorème précédent et du théorème 3.21. □

Théorème 4.5 (Church)**.** *L'ensemble des formules valides (i.e. vraies dans tous les modèles) d'un langage contenant le langage de l'arithmétique n'est pas décidable.*

Démonstration. Soit C la conjonction des axiomes de **Q**. Alors $C \to \phi$ est valide si et seulement ϕ est dans **Q**. S'il existait une procédure permettant de déterminer si une formule est valide ou non, il existerait aussi une procédure permettant de déterminer si une formule est dans **Q** ou non. Or ce n'est pas le cas vu le théorème précédent. □

Théorème 4.6 (Premier théorème d'incomplétude de Gödel)**.** *Il n'existe pas d'extension axiomatisable, consistante et complète de* **Q**.

Démonstration. Par le théorème 2.10, toute extension axiomatisable et complète est décidable. Par le théorème d'indécidabilité, aucune extension consistante n'est décidable. □

5 Arithmétique de Peano et modèles non standards

Dans cette section, nous commençons par donner un modèle non standard de l'arithmétique minimale. Nous introduisons ensuite l'arithmétique de Peano, qui permet de faire des récurrences.

5.1 Ordinaux

Le modèle non standard que nous allons présenter est celui des ordinaux. Nous présentons ici une construction naïve des ordinaux, qui est néanmoins suffisante pour montrer les propriétés qui nous intéressent.

Commençons par rappeler la notion d'ensemble bien ordonné.

Définition 5.1. Un *bon ordre* sur un ensemble S est un ordre total $\leq$ tel que pour tout $T \subset S$ non vide, T admet un minimum.

Un ensemble S est *bien ordonné* s'il est muni d'un bon ordre $\leq$.

Nous donnons maintenant une notion d'isomorphie pour ces ensembles.

Définition 5.2. Deux ensembles bien ordonnés A et B sont *isomorphes* s'il existe une bijection $f\colon A \to B$ préservant l'ordre, i.e. telle que pour tout $x \leq y$, on a $f(x) \leq f(y)$.

Les ordinaux sont les classes d'isomorphies des ensembles bien ordonnés.

Définition 5.3. Un *ordinal* est une classe d'isomorphie d'un ensemble bien ordonné.

Exemple. La classe d'équivalence de l'ensemble vide, notée $\mathbf{0}$, est un ordinal. Les classes d'équivalence $\mathbf{1} = [\{0\}]$, $\mathbf{2} = [\{0, 1\}]$, etc. sont des ordinaux. La classe d'équivalence $\omega = [\mathbb{N}]$ est le premier ordinal infini.

Nous définissons maintenant le successeur.

Définition 5.4. Si S est un ensemble bien ordonné, on définit son *successeur* par $S' = S \cup \{\infty\}$ où ∞ est un symbole n'appartenant pas à S et tel que $x \leq \infty$ pour tout $x \in S$. Cette construction revient à ajouter une borne supérieure à S.

On vérifie sans mal que si les ensembles bien ordonnés A et B sont isomorphes, alors les ensembles bien ordonnés A' et B' le sont aussi. La définition suivante est donc licite.

Définition 5.5. Si $\alpha = [A]$ est un ordinal, on définit son *successeur* par $\alpha' = [A']$.

Nous pouvons maintenant définir l'addition d'ordinaux.

Définition 5.6. Si A et B sont deux ensembles bien ordonnés, on définit leur *somme*, $A \oplus B$, en munissant l'ensemble $A \cup B$ (A et B sont disjoints) d'un ordre $\leq$ en posant $x \leq y$ pour tout $x \in A$ et $y \in B$.

À nouveau, l'opération $\oplus$ permet de définir une addition sur les ordinaux.

Définition 5.7. Si $\alpha = [A]$ et $\beta = [B]$ sont des ordinaux, on définit leur *somme* par $\alpha + \beta = [A \oplus B]$.

Finalement, nous donnons une définition pour la multiplication d'ordinaux.

Définition 5.8. Si A et B sont des ensembles bien ordonnés, leur *produit*, noté $A \odot B$, est l'ensemble $A \times B$ muni de l'ordre lexicographique.

Si $\alpha = [A]$ et $\beta = [B]$ sont des ordinaux, on définit leur *produit* par $\alpha \cdot \beta = [A \odot B]$.

Définir un ordre sur les ordinaux est un peu plus compliqué. Nous avons besoin de la notion de morphisme.

Définition 5.9. Si S est un ensemble bien ordonné, un *segment initial* est une partie $T \subset S$ telle que si $x \leq y$ et $y \in T$, alors $x \in T$.

Un *morphisme* entre deux ensembles ordonnés A et B est une application $f\colon A \to B$ telle que $f(A)$ est un segment initial de B et telle que $x \leq y$ si et seulement si $f(x) \leq f(y)$. En particulier, un morphisme est toujours injectif.

Exemple. L'ensemble $\{0, 1, 2\}$ est un segment initial de $\mathbb{N}$. L'application inclusion est un morphisme.

Définition 5.10. On peut définir un *ordre* sur les ordinaux par $\alpha \leq \beta$ s'il existe un morphisme d'un représentant de α dans un représentant de β. L'ordre ainsi défini est bon.

On définit également un *ordre stricte* par $\alpha < \beta$ si et seulement si $\beta \not\leq \alpha$.

Les ordinaux, muni de la constante $\mathbf{0}$, des opérations $'$, $+$ et $\cdot$ et du prédicat $<$, forment un modèle de l'arithmétique minimale.

Remarque. Les phrases $\mathbf{0} + \mathbf{1} = \mathbf{1} + \mathbf{0}$, $\mathbf{2} + \mathbf{1} = \mathbf{1} + \mathbf{2}$, etc. sont vraies dans le modèle des ordinaux car elles appartiennent à $\mathbf{Q}$. Par contre, la phrase $\forall x\, x + \mathbf{1} = \mathbf{1} + x$ est fausse car $\mathbf{1} + \omega$ et $\omega + \mathbf{1}$ ne sont pas isomorphes. En effet, $\mathbf{1} + \omega$ est isomorphe à ω (rajouter une borne inférieure à $\mathbb{N}$ ne change pas la structure d'ordre), alors que $\omega + \mathbf{1}$ n'y est pas isomorphe (ajouter une borne supérieure, par contre, change la structure).

5.2 Arithmétique de Peano

Comme nous venons de le voir, des phrases aussi simples que $\forall x\, x + \mathbf{1} = \mathbf{1} + x$ ne sont pas des théorèmes de $\mathbf{Q}$. Pourtant, cette phrase peut être démontrée par récurrence. Nous introduisons donc la théorie de l'arithmétique de Peano, dans laquelle on ajoute des axiomes permettant de démontrer certaines propriétés par récurrence.

Définition 5.11. La théorie de l'arithmétique *de Peano*, notée $\mathbf{P}$, est la théorie engendrée par $\mathbf{Q}$ uni à l'ensemble des axiomes de la forme

$$(\phi(\mathbf{0}) \wedge \forall x\, (\phi(x) \to \phi(x'))) \to \forall x\, x,$$

où $\phi(x)$ est une formule libre en x. La formule $\phi(x)$ peut également être libre en d'autres variables $y_1, \ldots, y_p$ et dans ce cas, l'axiome est précédé par $\forall y_1 \ldots \forall y_p$.

Remarque. La théorie de Peano est engendrée par une infinité (dénombrable) d'axiomes. Cependant, ces axiomes suivent tous un certain schéma. Ainsi, la théorie $\mathbf{P}$ est axiomatisable.

Remarque. Il est important de remarquer que la théorie de Peano est plus faible que la théorie utilisée pour la construction des naturels. Pour construire les naturels, on utilise un axiome de la forme

$$(0 \in S \wedge \forall x\, (x \in S \to x' \in S)) \to \forall x\, x \in S$$

valable pour tout ensemble $S \subset \mathbb{N}$. La théorie $\mathbf{P}$, quant à elle, est une théorie du premier ordre et la phrase suivante n'est un axiome que si l'ensemble S est définissable. Or nous savons qu'il existe une infinité non dénombrable de sous-ensembles de $\mathbb{N}$ mais seulement une infinité dénombrable de sous-ensembles définissables.

Finalement, nous donnons un exemple de théorème qui est vrai dans la théorie des ensembles mais qui n'est pas un théorème de $\mathbf{P}$. Pour cela, rappelons l'énoncé du théorème de Ramsay.

Théorème 5.12 (Ramsey)**.** *Soient r, s, n des naturels non nuls tel que $n \geq r$. Il existe un entier $m \geq n$ tel que peu importe comment l'ensemble des parties de taille r de $\{1, \ldots, m\}$ est partitionné en s classes, il existe un sous-ensemble $X \subset \{1, \ldots, m\}$ de taille n tel que toutes les parties de taille r de X appartiennent à la même classe.*

Ce théorème peut être démontré dans $\mathbf{P}$. Cependant, il est possible de le renforcer en un théorème qui n'est pas démontrable dans $\mathbf{P}$.

Définition 5.13. Un ensemble $X \subset \mathbb{N}$ est *glorieux* si $\#X \geq \min X$.

Théorème 5.14 (Parris-Harrington)**.** *Soient r, s, n des naturels non nuls tel que $n \geq r$. Il existe un entier $m \geq n$ tel que peu importe comment l'ensemble des parties de tailles r de $\{1, \ldots, m\}$ est partitionné en s classes, il existe un sous-ensemble* glorieux *$X \subset \{1, \ldots, m\}$ de taille n tel que toutes les parties de taille r de X appartiennent à la même classe.*

Ce dernier théorème peut être démontré en démontrant d'abord une version infinie du théorème, puis en utilisant le théorème de compacité pour se ramener au cas fini. Cependant, il ne peut pas être démontré dans $\mathbf{P}$.

D'autres exemples de théorèmes indécidables dans $\mathbf{P}$ sont donnés par le théorème de Goodstein et le problème de l'hydre de Kirby et Paris (voir par exemple [3]).

Index

Références

[1] Stefan Bilaniuk. *A Problem Course in Mathematical Logic. Version 1.6.* URL : `http://euclid.trentu.ca/math/sb/pcml/pcml.html`.

[2] George S. Boolos, John P. Burgess et Richard C. Jeffrey. *Computability and Logic.* 5e éd. New York : Cambridge University Press, 2007. ISBN : 978-0-521-87752-7.

[3] El Jj. *Deux (deux ?) minutes pour l'hydre de Kirby & Paris.* URL : `http://eljjdx.canalblog.com/archives/2016/02/12/33360210.html`.

[4] Terence Tao. *245B, Notes 7 : Well-ordered sets, ordinals, and Zorns lemma (optional).* URL : `https://terrytao.wordpress.com/2009/01/28/245b-notes-7-well-ordered-sets-ordinals-and-zorns-lemma-optional/`.