Chapitre VIII

Théorèmes de limitation

L'objet de ce chapitre est d'établir plusieurs résultats apparus dans les années 1930 exprimant des limitations intrinsèques des logiques du premier ordre, au premier rang desquels les célèbres théorèmes d'incomplétude de Gödel affirmant que certains énoncés ne sont *pas* prouvables, plus précisément l'existence de formules vraies dans la structure $(\mathbb{N}, 0, S, +, \cdot)$ et non prouvables à partir du système de Peano du premier ordre PA_1 d'une part, et d'autre part l'impossibilité pour une théorie consistante incluant PA_1 de prouver sa propre consistance. Ces résultats sont importants pour la théorie des ensembles car ils rendent caduc tout espoir de définir un système qui serait complet et dont la consistance pourrait être établie sans hypothèses plus fortes.

> ▷ *Dans le chapitre* VII*, on a établi pour la logique du premier ordre un résultat positif important, le théorème de complétude, ainsi que quelques résultats négatifs exprimant des limitations à son pouvoir d'expression. Ces résultats pourraient suggérer que la logique du premier ordre est relativement rudimentaire, à la façon de la logique propositionnelle du chapitre* VI*. Il n'en est rien : la logique du premier ordre est extrêmement riche, en particulier il ne peut exister de moyen effectif exhaustif pour reconnaître les formules valides ou pour prouver la consistance d'une théorie. La signification des résultats négatifs de ce chapitre est avant tout l'affirmation — positive ! — de cette richesse.* ◁

Le chapitre contient des démonstrations essentiellement complètes des résultats annoncés, à l'exception du second théorème d'incomplétude qui n'est établi que pour les théories incluant Z et non PA_1, une restriction naturelle en vue des applications à la théorie des ensembles et rendant possible un développement rigoureux et néanmoins de longueur modérée. Les trois premières sections contiennent des résultats préparatoires, qui du reste ont un intérêt indépendant. La section 1 est consacrée aux fonctions récursives, qui sont une formalisation de la notion informelle de fonction effectivement calculable. Dans la section 2, on montre comment, par une numérotation soigneuse, on peut coder les notions de base de la logique à l'intérieur de la

structure $(\mathbb{N}, 0, S, +, \cdot)$ à l'aide de fonctions et de relations récursives. Dans la section 3, on montre qu'une version affaiblie de l'arithmétique de Peano dans laquelle l'induction est omise prouve suffisamment de formules pour représenter en un sens convenable toutes les fonctions récursives. Enfin, on établit dans la section 4 les théorèmes de limitation visés, à partir d'un argument diagonal commun assez simple et en tout cas rapide — et rien n'empêche le lecteur curieux de commencer par là.

Sources et compléments. *On a cherché ici à isoler les points techniques importants mais en allant au plus vite vers les résultats de la section* 4. *Un traitement plus détaillé et complet se trouve en particulier dans le livre* [109] *de P. Smith.*

1. Fonctions et relations récursives

Le point de départ est la recherche d'une notion précise d'effectivité pour les fonctions de $\mathbb{N}$ dans $\mathbb{N}$, ou plus généralement de $\mathbb{N}^p$ dans $\mathbb{N}^q$, fondée sur l'existence d'une *définition algorithmique*, c'est-à-dire d'une recette explicite faisant passer de l'argument à la valeur (aux antipodes de la représentation ensembliste des fonctions comme un magma indifférencié de couples). On étudie ici sous le nom de fonctions récursives une famille particulière de telles fonctions possédant une définition algorithmique simple, en l'occurrence une construction à partir de fonctions de base au moyen de règles de construction naturelles. Une fois les notions de base introduites, l'objet principal de la section est de vérifier que la plupart des fonctions et relations usuelles sur les entiers sont récursives.

> ▷ *Il s'agit de vérifications fastidieuses mais faciles : à partir des fonctions de base (qui jouent le rôle d'axiomes), on construit de proche en proche une liste de fonctions de plus en plus riche, et, à chaque étape, la question est de vérifier que la nouvelle fonction ou la nouvelle relation dont on veut établir le caractère récursif possède bien une définition du type autorisé à partir des éléments précédents de la liste. C'est typiquement le genre de vérification qu'il est indispensable d'effectuer une fois, mais que l'on peut ensuite oublier sans grand dommage.*
>
> *Avant toute description, on pourra noter que, quel que soit le type de définition considéré, pour peu qu'une définition soit un mot sur un alphabet fini, il existe au plus une infinité dénombrable de fonctions définissables, alors que l'ensemble de toutes les fonctions de $\mathbb{N}$ dans $\mathbb{N}$ est un ensemble non dénombrable, d'où il résulte qu'il existe nécessairement une infinité non dénombrable de fonctions non définissables — et même, en un sens informel, que la densité des fonctions possédant une définition est nulle.* ◁

Cette section est divisée en trois sous-sections. Les deux premières sont consacrées aux fonctions primitives récursives, qui forment une sous-famille de la famille des fonctions récursives. Dans la sous-section 1.1, on définit les fonctions primitives récursives à partir d'un schéma de récurrence et on montre qu'elles forment une famille close par de nombreuses opérations, en particulier la minimisation bornée. Dans la sous-section 1.2, ce résultat est appliqué à la construction d'un codage des suites finies d'entiers par les

entiers. Enfin, dans la sous-section 1.3, on définit les fonctions récursives générales, pour lesquelles on passe à un contexte de fonctions partielles, et on les relie aux fonctions calculables par machine de Turing.

1.1. Fonctions primitives récursives

▶ Résumé.— Définies par récurrence à partir de fonctions de base, les fonctions primitives récursives contiennent la plupart des fonctions usuelles et sont closes par minimisation bornée. ◀

1.1.1.— Le point de vue retenu ici est qu'une fonction de $\mathbb{N}^p$ dans $\mathbb{N}$ est effective si elle peut être construite à partir de fonctions de base simples telles que la fonction successeur par des opérations préservant le caractère effectif, typiquement des définitions récursives du type considéré dans la section III.3.

Définition (primitif récursif).— (i) Pour $f_1, \dots, f_q : \mathbb{N}^p \to \mathbb{N}$, et $g : \mathbb{N}^q \to \mathbb{N}$, on note $\mathsf{comp}(g, f_1, \dots, f_q)$ la fonction f de $\mathbb{N}^p$ dans $\mathbb{N}$ définie par

$$f(\vec{n}) := g(f_1(\vec{n}), \dots, f_q(\vec{n})),$$

dite définie par *composition* à partir de g, $f_1, \dots, f_q$.

(ii) Pour $g : \mathbb{N}^p \to \mathbb{N}$ et $h : \mathbb{N}^{p+2} \to \mathbb{N}$, on note $\mathsf{rec}(g, h)$ la fonction f de $\mathbb{N}^{p+1}$ dans $\mathbb{N}$ définie par

$$f(\vec{n}, k) := g(\vec{n}) \text{ pour } k = 0, \quad \text{et } f(\vec{n}, k) := h(\vec{n}, k, f(\vec{n}, k-1)) \text{ pour } k > 0,$$

dite définie par *récursion* de base g et de pas h.

(iii) Une fonction f de $\mathbb{N}^p$ dans $\mathbb{N}$ est dite *primitive récursive* si elle peut s'obtenir par un nombre fini de compositions et de récursions à partir des fonctions zero, succ et $\mathsf{proj}_{p,i}$ avec $1 \leqslant i \leqslant p$, où

- $\mathsf{zero} : \mathbb{N}^0 \to \mathbb{N}$ est définie [1] par $\mathsf{zero} := 0$,
- $\mathsf{succ} : \mathbb{N} \to \mathbb{N}$ est définie par $\mathsf{succ}(n) := S(n) = n + 1$,
- $\mathsf{proj}_{p,i} : \mathbb{N}^p \to \mathbb{N}$ est définie par $\mathsf{proj}_{p,i}(n_1, \dots, n_p) := n_i$.

(iv) Un sous-ensemble R de $\mathbb{N}^p$ — ou, de façon équivalente, une relation R sur $\mathbb{N}^p$ — est dit *primitif récursif* si sa fonction indicatrice $\mathbf{1}_R$, définie par $\mathbf{1}_R(\vec{n}) := 1$ pour $\vec{n}$ dans R et $\mathbf{1}_R(\vec{n}) := 0$ sinon, est primitive récursive.

Exemple. Vue comme une fonction de $\mathbb{N}^2$ dans $\mathbb{N}$, l'addition add *est primitive récursive, puisqu'elle admet la définition récursive* $\mathsf{add}(n, k) = n$ *pour* $k = 0$ *et*

$$\mathsf{add}(n, k) = n + k = S(n + (k-1)) = \mathsf{succ}(\mathsf{add}(n, k-1))$$

pour $k > 0$. *On a donc* $\mathsf{add} = \mathsf{rec}(g, h)$, *où* g *est la fonction identité de* $\mathbb{N}$, *c'est-à-dire* $\mathsf{proj}_{1,1}$, *et où* h *est la fonction de* $\mathbb{N}^3$ *dans* $\mathbb{N}$ *définie par* $h(n, k, m) := \mathsf{succ}(m)$. *On a donc* $h = \mathsf{comp}(\mathsf{succ}, \mathsf{proj}_{3,3})$, *d'où, finalement,*

$$\mathsf{add} = \mathsf{rec}(\mathsf{proj}_{1,1}, \mathsf{comp}(\mathsf{succ}, \mathsf{proj}_{3,3}))$$

qui témoigne de ce que add *est primitive récursive.*

1. C'est la constante 0 ; on pourrait éviter les fonctions à zéro argument en partant d'une fonction constante à un argument, mais définir par récursion les fonctions de $\mathbb{N}$ dans $\mathbb{N}$ poserait ensuite un problème (mineur) qui obligerait à traiter ce cas séparément.

1.1.2.— On commence la (longue) vérification que toutes les fonctions simples sont primitives récursives.

Lemme.— (i) *Supposons que* π *est une permutation de* $\{1, ..., \mathsf{p}\}$. *Si une fonction* f *de* $\mathbb{N}^{\mathsf{p}}$ *dans* $\mathbb{N}$ *est primitive récursive, il en est de même de la fonction* f^{π} *de* $\mathbb{N}^{\mathsf{p}}$ *dans* $\mathbb{N}$ *définie par* $\mathsf{f}^{\pi}(\mathsf{n}_1, ..., \mathsf{n}_{\mathsf{p}}) := \mathsf{f}(\mathsf{n}_{\pi(1)}, ..., \mathsf{n}_{\pi(\mathsf{p})})$.

(ii) *Sont primitifs récursifs, pour tous* p, m, *la fonction constante* $\mathsf{const}_{\mathsf{p},\mathsf{m}}$ *de* $\mathbb{N}^{\mathsf{p}}$ *dans* $\mathbb{N}$ *de valeur* m, *l'addition, la multiplication et l'exponentiation, les relations d'égalité et d'ordre, les singletons.*

Démonstration. (i) Par définition, on a

$$\mathsf{f}^{\pi} = \mathsf{comp}(\mathsf{f}, \mathsf{proj}_{\mathsf{p},\pi(1)}, ..., \mathsf{proj}_{\mathsf{p},\pi(\mathsf{p})}),$$

et, en substituant à f dans cette expression une définition de f en termes des fonctions de base, on obtient une définition de f^{π} en termes des mêmes fonctions de base, ce qui atteste du caractère primitif récursif de f^{π}.

(ii) On montre par récurrence sur p que la fonction $\mathsf{const}_{\mathsf{p},0}$ est primitive récursive. Pour $\mathsf{p} = 0$, la fonction est zero, et le résultat est vrai par définition. Pour $\mathsf{p} = 1$, la fonction $\mathsf{const}_{1,0}$ est définie par la récursion $\mathsf{const}_{1,0}(\mathsf{k}) := 0$ pour $\mathsf{k} = 0$ et

$$\mathsf{const}_{1,0}(\mathsf{k}) := \mathsf{const}_{1,0}(\mathsf{k} - 1)$$

pour $\mathsf{k} > 0$, donc $\mathsf{const}_{1,0}$ est définie par récursion de base zero et de pas h, où h est définie par $\mathsf{h}(\mathsf{n}, \mathsf{m}) := \mathsf{m}$, soit $\mathsf{h} = \mathsf{proj}_{2,2}$; on a donc

$$\mathsf{const}_{1,0} = \mathsf{rec}(\mathsf{zero}, \mathsf{proj}_{2,2}),$$

et $\mathsf{const}_{1,0}$ est primitive récursive. Enfin, pour $\mathsf{p} \geqslant 2$, on a

$$\mathsf{const}_{\mathsf{p},0}(\vec{\mathsf{n}}) = \mathsf{const}_{1,0}(\mathsf{n}_1) = \mathsf{const}_{1,0}(\mathsf{proj}_{\mathsf{p},1}(\vec{\mathsf{n}})),$$

donc $\mathsf{const}_{\mathsf{p},0} = \mathsf{comp}(\mathsf{const}_{1,0}, \mathsf{proj}_{\mathsf{p},1})$, à nouveau une fonction primitive récursive. Ensuite, on montre par récurrence sur m que $\mathsf{const}_{\mathsf{p},\mathsf{m}}$ est primitive récursive pour tout m. Pour $\mathsf{m} = 0$, le résultat vient d'être établi. Pour $\mathsf{m} > 0$, on trouve $\mathsf{const}_{\mathsf{p},\mathsf{m}}(\vec{\mathsf{n}}) = S(\mathsf{const}_{\mathsf{p},\mathsf{m}-1}(\vec{\mathsf{n}}))$ pour tout $\vec{\mathsf{n}}$, et on déduit

$$\mathsf{const}_{\mathsf{p},\mathsf{m}} = \mathsf{comp}(\mathsf{succ}, \mathsf{const}_{\mathsf{p},\mathsf{m}-1}).$$

Par hypothèse de récurrence, $\mathsf{const}_{\mathsf{p},\mathsf{m}-1}$ est primitive récursive, donc il en est de même de $\mathsf{const}_{\mathsf{p},\mathsf{m}}$.

On a vu dans l'exemple du §1.1.1 que l'addition est primitive récursive puisque définie par récursion à partir de la fonction successeur. La multiplication mult est définie à partir de l'addition par la récursion $\mathsf{mult}(\mathsf{n}, \mathsf{k}) := 0$ pour $\mathsf{k} = 0$ et

$$\mathsf{mult}(\mathsf{n}, \mathsf{k}) = \mathsf{n} \cdot \mathsf{k} = (\mathsf{n} \cdot (\mathsf{k} - 1)) + \mathsf{n}$$

pour $\mathsf{k} > 0$, d'où $\mathsf{mult} = \mathsf{rec}(\mathsf{const}_{1,0}, \mathsf{h})$, où h est la fonction de $\mathbb{N}^3$ dans $\mathbb{N}$ définie par $\mathsf{h}(\mathsf{n}, \mathsf{k}, \mathsf{m}) := \mathsf{m} + \mathsf{n}$, soit $\mathsf{comp}(\mathsf{add}, \mathsf{proj}_{3,3}, \mathsf{proj}_{3,1})$, d'où

$$\mathsf{mult} = \mathsf{rec}(\mathsf{const}_{1,0}, \mathsf{comp}(\mathsf{add}, \mathsf{proj}_{3,3}, \mathsf{proj}_{3,1})).$$

Comme $\mathsf{const}_{1,0}$ et add sont primitives récursives, mult l'est également. De même encore, l'exponentielle exp est primitive récursive puisque définie par la récursion $\mathsf{exp}(\mathsf{n}, \mathsf{k}) := 1$ pour $\mathsf{k} = 0$ et

$$\mathsf{exp}(\mathsf{n}, \mathsf{k}) = \mathsf{n}^{\mathsf{k}} = \mathsf{mult}(\mathsf{exp}(\mathsf{n}, \mathsf{k} - 1), \mathsf{n})$$

pour $\mathsf{k} > 0$. Ensuite, le singleton $\{0\}$ est primitif récursif, puisque sa fonction indicatrice $\mathbf{1}_{\{0\}}$ est définie par la récursion

$$\mathbf{1}_{\{0\}}(\mathsf{k}) := 1 \text{ pour } \mathsf{k} = 0, \text{ et } \mathbf{1}_{\{0\}}(\mathsf{k}) := 0 \text{ pour } \mathsf{k} > 0,$$

d'où $\mathbf{1}_{\{0\}} = \mathsf{rec}(\mathsf{const}_{0,1}, \mathsf{const}_{2,0})$. Puis l'ensemble $2\mathbb{N}$ des nombres pairs est primitif récursif puisque l'on a $\mathbf{1}_{2\mathbb{N}}(\mathsf{k}) = 1$ pour $\mathsf{k} = 0$ et

$$\mathbf{1}_{2\mathbb{N}}(\mathsf{k}) = \mathbf{1}_{\{0\}}(\mathbf{1}_{2\mathbb{N}}(\mathsf{k} - 1))$$

pour $\mathsf{k} > 0$, d'où $\mathbf{1}_{2\mathbb{N}} = \mathsf{rec}(\mathsf{const}_{0,1}, \mathsf{comp}(\mathbf{1}_{\{0\}}, \mathsf{proj}_{2,2}))$. Ensuite, la fonction moitie telle que $\mathsf{moitie}(\mathsf{n})$ est la partie entière de $\mathsf{n}/2$ est primitive récursive, car on peut écrire $\mathsf{moitie}(\mathsf{k}) = 0$ pour $\mathsf{k} = 0$ et

$$\mathsf{moitie}(\mathsf{k}) = \mathsf{moitie}(\mathsf{k}-1) + \mathbf{1}_{2\mathbb{N}}(\mathsf{k})$$

pour $\mathsf{k} > 0$, donc moitie est définie par récursion de base zero et de pas la fonction h telle que $\mathsf{h}(\mathsf{k}, \mathsf{m})$ est $\mathsf{m} + \mathbf{1}_{2\mathbb{N}}(\mathsf{k})$, d'où

$$\mathsf{moitie} = \mathsf{rec}(\mathsf{zero}, \mathsf{comp}(\mathsf{add}, \mathsf{proj}_{2,2}, \mathsf{comp}(\mathbf{1}_{2\mathbb{N}}, \mathsf{proj}_{2,1}))).$$

Considérons alors la décrémentation decr définie par $\mathsf{decr}(\mathsf{n}) := 0$ pour $\mathsf{n} = 0$ et par $\mathsf{decr}(\mathsf{n}) := \mathsf{n} - 1$ sinon. Pour $\mathsf{n} \geqslant 1$, on a

$$\mathsf{decr}(\mathsf{n}) = \mathsf{moitie}(\mathsf{n}) + \mathsf{moitie}(\mathsf{decr}(\mathsf{n}-1) + 1),$$

comme on le vérifie en séparant les cas de n pair et impair. Comme decr est définie par la récursion $\mathsf{decr}(\mathsf{k}) = 0$ pour $\mathsf{k} = 0$ et

$$\mathsf{decr}(\mathsf{k}) = \mathsf{moitie}(\mathsf{decr}(\mathsf{k}-1) + 1) + \mathsf{moitie}(\mathsf{k}) \text{ pour } \mathsf{k} > 0,$$

elle est primitive récursive, puisque définie par récursion de base zero et de pas la fonction h définie par $\mathsf{h}(\mathsf{k}, \mathsf{m}) = \mathsf{moitie}(\mathsf{m}+1) + \mathsf{moitie}(\mathsf{k})$, laquelle est primitive récursive puisque add et moitie le sont. Ensuite, la différence positive diff est primitive récursive puisque définie par la récursion $\mathsf{diff}(\mathsf{n}, \mathsf{k}) = \mathsf{n}$ pour $\mathsf{k} = 0$ et

$$\mathsf{diff}(\mathsf{n}, \mathsf{k}) = \mathsf{decr}(\mathsf{diff}(\mathsf{n}, \mathsf{k}-1)) \text{ pour } \mathsf{k} > 0.$$

Comme la fonction indicatrice $\mathbf{1}_{=}$ de la relation d'égalité est définie par

$$\mathbf{1}_{=}(\mathsf{n}_1, \mathsf{n}_2) = \mathbf{1}_{\{0\}}(\mathsf{diff}(\mathsf{n}_1, \mathsf{n}_2) + \mathsf{diff}(\mathsf{n}_2, \mathsf{n}_1)),$$

on en déduit qu'elle est primitive récursive comme composée de fonctions primitives récursives, le caractère primitif récursif de la fonction $(\mathsf{n}_1, \mathsf{n}_2) \mapsto \mathsf{diff}(\mathsf{n}_2, \mathsf{n}_1)$ résultant de celui de diff et du point (i). De même, les fonctions indicatrices des relations $\leqslant$ et $<$ sont primitives récursives puisque l'on a

$$\mathbf{1}_{\leqslant}(\mathsf{n}_1, \mathsf{n}_2) = \mathbf{1}_{\{0\}}(\mathsf{diff}(\mathsf{n}_1, \mathsf{n}_2)), \text{ et } \mathbf{1}_{<}(\mathsf{n}_1, \mathsf{n}_2) = \mathbf{1}_{\leqslant}(\mathsf{n}_1 + 1, \mathsf{n}_2).$$

Enfin, pour tous $\mathsf{m}_1, \ldots, \mathsf{m}_\mathsf{p}$, on a $\mathbf{1}_{\{\vec{\mathsf{m}}\}}(\vec{\mathsf{n}}) = \mathbf{1}_{=}(\mathsf{m}_1, \mathsf{n}_1) \cdot \cdots \cdot \mathbf{1}_{=}(\mathsf{m}_\mathsf{p}, \mathsf{n}_\mathsf{p})$. Pour $\mathsf{p} = 1$, on a $\mathbf{1}_{\{\mathsf{m}\}} = \mathsf{comp}(\mathbf{1}_{=}, \mathsf{proj}_{1,1}, \mathsf{const}_{1,\mathsf{m}})$, donc $\mathbf{1}_{\{\mathsf{m}\}}$ est primitive récursive. Ensuite, on utilise une récurrence sur p, et, comme la multiplication est primitive récursive, on conclut que $\mathbf{1}_{\{\vec{\mathsf{m}}\}}$ l'est aussi. □

▷ *On évite souvent l'argument utilisé ici pour montrer que la fonction décrémentation est primitive récursive en plaçant* decr *dans les fonctions de base. Noter aussi que, si la récursion est introduite de sorte que* $\mathsf{f}(\vec{\mathsf{n}}, \mathsf{k}{+}1)$ *est exprimé en fonction de* $\mathsf{f}(\vec{\mathsf{n}}, \mathsf{k})$ *— et non* $\mathsf{f}(\vec{\mathsf{n}}, \mathsf{k})$ *en fonction de* $\mathsf{f}(\vec{\mathsf{n}}, \mathsf{k}{-}1)$ *— alors* decr *est définie par la récursion évidente* $\mathsf{decr}(\mathsf{k}{+}1) = \mathsf{k}$ *pour* $\mathsf{k} > 0$. *D'une façon générale, il existe de nombreuses variantes dans la définition des fonctions (primitives) récursives, le point important étant que ces variantes mènent toutes à la même famille de fonctions, où on trouve toutes les fonctions arithmétiques de base.* ◁

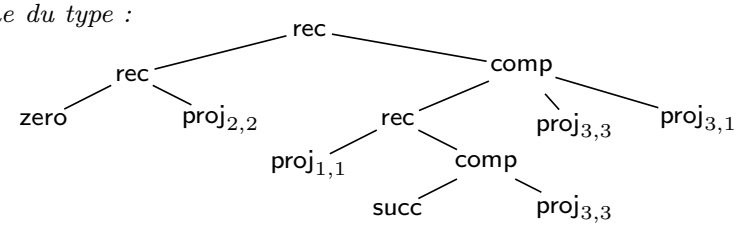
1.1.3.— Le caractère primitif récursif d'une fonction f est une propriété globale de f, mettant en jeu la construction de f comme un tout : une valeur $\mathsf{f}(\mathsf{n})$ particulière ne peut *rien* indiquer quant au caractère (primitif) récursif ou non de f.

▷ *On recommande de se familiariser avec les définitions récursives en explicitant quelques-unes des définitions mentionnées ci-dessus, par exemple en vérifiant que*

$$\mathsf{rec}(\mathsf{rec}(\mathsf{zero}, \mathsf{proj}_{2,2}), \mathsf{comp}(\mathsf{rec}(\mathsf{proj}_{1,1}, \mathsf{comp}(\mathsf{succ}, \mathsf{proj}_{3,3})), \mathsf{proj}_{3,3}, \mathsf{proj}_{3,1})),$$

est une définition primitive récursive de la multiplication.

Noter la structure arborescente sous-jacente dans une définition comme ci-dessus, qui, comme dans le cas des formules et des preuves, peut être visualisée dans un diagramme du type :



◁

1.1.4.— L'étape suivante en vue de montrer la richesse de la famille des fonctions primitives récursives consiste à établir des résultats de clôture. Ainsi, le résultat suivant correspond à une définition par cas et montre que l'on peut définir des fonctions primitives récursives en amalgamant des fragments de fonctions primitives récursives, pourvu que la relation discriminante soit elle-même primitive récursive.

Lemme.— *Si* f_1 *et* f_2 *sont des fonctions primitives récursives et si* R *est une relation primitive récursive, alors la fonction* f *définie par* $\mathsf{f}(\vec{n}) := \mathsf{f}_1(\vec{n})$ *si* $\mathsf{R}(\vec{n})$ *est vraie, et* $\mathsf{f}(\vec{n}) := \mathsf{f}_2(\vec{n})$ *sinon, est primitive récursive.*

Démonstration. On a $\mathsf{f}(\vec{n}) = \mathsf{f}_1(\vec{n}) \cdot \mathbf{1}_{\mathsf{R}}(\vec{n}) + \mathsf{f}_2(\vec{n}) \cdot \mathbf{1}_{\{0\}}(\mathbf{1}_{\mathsf{R}}(\vec{n}))$, d'où

$$\mathsf{f} = \mathsf{comp}(\mathsf{add}, \mathsf{comp}(\mathsf{mult}, \mathsf{f}_1, \mathbf{1}_{\mathsf{R}}), \mathsf{comp}(\mathsf{mult}, \mathsf{f}_2, \mathsf{comp}(\mathbf{1}_{\{0\}}, \mathbf{1}_{\mathsf{R}}))),$$

et f est primitive récursive puisque add, mult, $\mathbf{1}_{\{0\}}$ le sont, comme on l'a vu en 1.1.2, de même que f_1, f_2 et $\mathbf{1}_{\mathsf{R}}$ par hypothèse. □

1.1.5.— On vérifie ensuite que l'ensemble des fonctions primitives récursives est clos par sommations et produits finis.

Lemme.— *Si* f *est une fonction primitive récursive de* $\mathbb{N}^{\mathsf{p}+1}$ *dans* $\mathbb{N}$, *il en est de même des fonctions* f_1 *et* f_2 *définies par* $\mathsf{f}_1(\vec{n}, \mathsf{k}) := \sum_{\mathsf{i} \leqslant \mathsf{k}} \mathsf{f}(\vec{n}, \mathsf{i})$ *et par* $\mathsf{f}_2(\vec{n}, \mathsf{k}) := \prod_{\mathsf{i} \leqslant \mathsf{k}} \mathsf{f}(\vec{n}, \mathsf{i})$.

Démonstration. Les fonctions f_1 et f_2 sont définies par les récursions

$$\mathsf{f}_1(\vec{n}, \mathsf{k}) = \begin{cases} \mathsf{f}(\vec{n}, 0) \\ \mathsf{f}_1(\vec{n}, \mathsf{k}-1) + \mathsf{f}(\vec{n}, \mathsf{k}) \end{cases} \qquad \mathsf{f}_2(\vec{n}, \mathsf{k}) = \begin{cases} \mathsf{f}(\vec{n}, 0) & \text{pour } \mathsf{k} = 0, \\ \mathsf{f}_2(\vec{n}, \mathsf{k}-1) \cdot \mathsf{f}(\vec{n}, \mathsf{k}) & \text{pour } \mathsf{k} > 0, \end{cases}$$

donc sont primitives récursives. □

1.1.6.— De là, on déduit des résultats de clôture des ensembles récursifs.

Proposition (clôture).— *Pour tout* p, *la famille des sous-ensembles primitifs récursifs de* $\mathbb{N}^{\mathsf{p}}$ *est close par réunion, intersection, complémentaire, et elle contient tous les sous-ensembles finis et co-finis de* $\mathbb{N}^{\mathsf{p}}$.

Démonstration. Soient $\mathsf{R}_1, \mathsf{R}_2$ des sous-ensembles primitifs récursifs de $\mathbb{N}^{\mathsf{p}}$. L'ensemble $\mathsf{R}_1 \cap \mathsf{R}_2$ est primitif récursif, puisque l'on a $\mathbf{1}_{\mathsf{R}_1 \cap \mathsf{R}_2}(\vec{n}) = \mathbf{1}_{\mathsf{R}_1}(\vec{n}) \cdot \mathbf{1}_{\mathsf{R}_2}(\vec{n})$, soit

$$\mathbf{1}_{\mathsf{R}_1 \cap \mathsf{R}_2} = \mathsf{comp}(\mathsf{mult}, \mathbf{1}_{\mathsf{R}_1}, \mathbf{1}_{\mathsf{R}_2}).$$

Ensuite, $\mathbb{N}^{\mathsf{p}} \setminus \mathsf{R}_1$ est primitif récursif, puisque l'on a $\mathbf{1}_{\mathbb{N}^{\mathsf{p}} \setminus \mathsf{R}_1}(\vec{n}) = \mathbf{1}_{\{0\}}(\mathbf{1}_{\mathsf{R}_1}(\vec{n}))$, et, par conséquent, $\mathbf{1}_{\mathbb{N}^{\mathsf{p}} \setminus \mathsf{R}_1} = \mathsf{comp}(\mathbf{1}_{\{0\}}, \mathbf{1}_{\mathsf{R}_1})$. On en déduit que $\mathsf{R}_1 \cup \mathsf{R}_2$ est primitif récursif comme complémentaire de l'intersection des complémentaires de R_1 et R_2.

Enfin, on a vu en 1.1.2 que les singletons sont primitifs récursifs. Par réunion finie de tels singletons, on obtient tous les sous-ensembles finis de $\mathbb{N}^p$, et, par complémentation, tous les sous-ensembles co-finis. □

1.1.7.— On verra plus loin que, si R est une relation primitive récursive sur $\mathbb{N}^{p+1}$, il est faux en général que la projection de R sur $\mathbb{N}^p$, c'est-à-dire la relation $\exists i\,(R(\vec{n}, i))$, soit primitive récursive. En revanche, on a un résultat de clôture pour une quantification bornée, c'est-à-dire lorsque $\exists i$ est remplacé par $\exists i \leqslant k$.

Proposition (quantification bornée).— *Si* R *est une relation primitive récursive sur* $\mathbb{N}^{p+1}$ *et si* h *est une fonction primitive récursive sur* $\mathbb{N}^p$*, il en est de même des relations* R_1 *et* R_2 *sur* $\mathbb{N}^p$ *définies par*

$$R_1(\vec{n}) \Leftrightarrow \exists i \leqslant h(\vec{n})\,(R(\vec{n}, i)) \quad \text{et} \quad R_2(\vec{n}) \Leftrightarrow \forall i \leqslant h(\vec{n})\,(R(\vec{n}, i)).$$

Démonstration. Observons d'abord que, si R est une relation primitive récursive sur $\mathbb{N}^{p+1}$, il en est de même de $R_\exists$ et $R_\forall$ sur $\mathbb{N}^p$ définies par

$$(\vec{n}, k) \in R_\exists \Leftrightarrow \exists i \leqslant k\,(R(\vec{n}, i)) \text{ et } (\vec{n}, k) \in R_\forall \Leftrightarrow \forall i \leqslant k\,(R(\vec{n}, i)).$$

En effet, par définition, on a alors

$$\mathbf{1}_{R_\exists}(\vec{n}, k) = \mathbf{1}_{\geqslant 1}(\textstyle\sum_{i \leqslant k} \mathbf{1}_R(\vec{n}, i)) \quad \text{et} \quad \mathbf{1}_{R_\forall}(\vec{n}, k) = \prod_{i \leqslant k} \mathbf{1}_R(\vec{n}, i),$$

donc $R_\exists$ et $R_\forall$ sont primitives récursives par 1.1.5, et 1.1.6 pour $\mathbf{1}_{\geqslant 1}$.

Considérons alors la relation R_1 de l'énoncé. En termes de $R_\exists$, on trouve

$$\mathbf{1}_{R_1} = \mathsf{comp}(\mathbf{1}_{R_\exists}, \mathsf{proj}_{p,1}, \dots, \mathsf{proj}_{p,p}, \mathsf{comp}(h, \mathsf{proj}_{p,1}, \dots, \mathsf{proj}_{p,p})),$$

donc R_1 est primitive récursive. L'argument est le même pour R_2 avec $R_\forall$. □

1.1.8.— Pour énoncer une contrepartie en termes de fonctions, on introduit une nouvelle opération, la *minimisation bornée*.

Définition (minimisation bornée).— Pour $h : \mathbb{N}^p \to \mathbb{N}$ et $R \subseteq \mathbb{N}^{p+1}$, on pose

$$\mu m{<}h(\vec{n})\,(R(\vec{n}, m)) := \begin{cases} \text{le plus petit } m < h(\vec{n}) \text{ vérifiant } R(\vec{n}, m) \text{ s'il existe,} \\ h(\vec{n}) \text{ sinon ;} \end{cases}$$

la fonction f de $\mathbb{N}^p$ dans $\mathbb{N}$ définie par $f(\vec{n}) := \mu m{<}h(\vec{n})\,(R(\vec{n}, m))$ est dite définie par *minimisation bornée* à partir de h et R, et notée $\mathsf{minim}_<(h, R)$.

Exemple. Un cas typique est celui où, partant de $g : \mathbb{N}^{p+1} \to \mathbb{N}$ *et de* $h : \mathbb{N}^p \to \mathbb{N}$*, on définit* f *par*

$$f(\vec{n}) =: \mu m{<}h(\vec{n})\,(g(\vec{n}, m) = 0),$$

qui associe à $\vec{n}$ *le plus petit* m *tel que* $g(\vec{n}, m)$ *s'annule s'il en existe un inférieur à* $h(\vec{n})$*, et* $h(\vec{n})$ *à défaut. Remarquer que, si* g *est une fonction primitive récursive, il en est de même de la relation* $g(\vec{n}, m) = 0$*.*

1.1.9.— La contrepartie de 1.1.7 pour les fonctions s'énonce comme suit.

Proposition (minimisation bornée).— *Si* h *et* R *sont primitives récursives, il en est de même de* $\mathsf{minim}_<(h, R)$*.*

Démonstration. Soit f la fonction $\mathsf{minim}_<(\mathsf{h},\mathsf{R})$. Dire que m est le premier entier pour lequel $\mathsf{R}(\vec{\mathsf{n}},\mathsf{m})$ est vrai signifie qu'il existe exactement m entiers i avec la propriété que R est faux pour tous les $\mathsf{j} \leqslant \mathsf{i}$. Par ailleurs, $\mathsf{h}(\vec{\mathsf{n}})$ est le nombre d'entiers inférieurs à $\mathsf{h}(\vec{\mathsf{n}})$ (!). On a donc dans tous les cas $\mathsf{f}(\vec{\mathsf{n}}) = \sum_{\mathsf{i}\leqslant\mathsf{h}(\vec{\mathsf{n}})} \mathsf{g}(\vec{\mathsf{n}},\mathsf{i})$, où g est définie par $\mathsf{g}(\vec{\mathsf{n}},\mathsf{i}) := 1$ si on a $\forall \mathsf{j}\leqslant\mathsf{i}(\neg\mathsf{R}(\vec{\mathsf{n}},\mathsf{j}))$ et par $\mathsf{g}(\vec{\mathsf{n}},\mathsf{i}) := 0$ si on a $\exists \mathsf{j}\leqslant\mathsf{i}(\mathsf{R}(\vec{\mathsf{n}},\mathsf{j}))$. Comme vu dans la démonstration de 1.1.7, la relation « $\exists \mathsf{j}\leqslant\mathsf{i}(\mathsf{R}(\vec{\mathsf{n}},\mathsf{j}))$ » est primitive récursive, et, par 1.1.4, il en est de même de g qui est définie par cas à partir de fonctions et relations primitives récursives. Enfin f est obtenue par sommation à partir de g et h, donc elle est primitive récursive par 1.1.5. □

On notera que l'introduction de la borne h est essentielle dans le résultat précédent, d'abord pour garantir que la fonction f est partout définie, puis pour montrer qu'elle est primitive récursive. De fait, on verra plus loin que le résultat est en défaut si la borne est omise.

1.2. Représentation des suites finies

▶ Résumé.— Il existe des codages des suites finies d'entiers par des entiers tels que toutes les opérations associées soient primitives récursives. ◀

1.2.1.— L'ensemble des suites finies d'entiers est un ensemble dénombrable, et on peut donc numéroter ses éléments. Il existe plusieurs façons de le faire ; pour les besoins de ce chapitre, on utilisera deux numérotations distinctes. Ces numérotations sont non bijectives : toute suite d'entiers reçoit un numéro, mais il existe des numéros ne correspondant à aucune suite.

1.2.2.— La première numérotation utilise la fonction traditionnellement appelée fonction β de Gödel ; elle est très simple, mais non univoque : une même suite peut recevoir plusieurs numéros distincts.

▷ *Le but cherché est de pouvoir remplacer au moindre coût une quantification du type « il existe une suite finie d'entiers telle que… » par une quantification « il existe un entier tel que… », qui sera le point essentiel dans la démonstration de* 3.3.2 *: or, pour cela, nulle injectivité n'est requise.* ◁

Lemme.— *Il existe une fonction primitive récursive* $\mathsf{beta} : \mathbb{N}^3 \to \mathbb{N}$ *telle que, pour toute suite finie d'entiers naturels* $(\mathsf{n}_0,\ldots,\mathsf{n}_\mathsf{k})$*, il existe* (*au moins*) *deux entiers* s,t *vérifiant* $\mathsf{beta}(\mathsf{s},\mathsf{t},\mathsf{i}) = \mathsf{n}_\mathsf{i}$ *pour* $0 \leqslant \mathsf{i} \leqslant \mathsf{k}$.

Démonstration. Soient $\mathsf{quot}(\mathsf{n},\mathsf{k})$ et $\mathsf{reste}(\mathsf{n},\mathsf{k})$ le quotient et le reste de la division euclidienne de n par k pour $\mathsf{k} \neq 0$, complétés par $\mathsf{quot}(\mathsf{n},0) := 0$ et $\mathsf{reste}(\mathsf{n},0) := \mathsf{n}$. Alors, quot est définie par la minimisation bornée

$$\mathsf{quot}(\mathsf{n},\mathsf{k}) = \mu\mathsf{q}\leqslant\mathsf{n}\,((\mathsf{k}=0\wedge\mathsf{q}=0)\vee((\mathsf{k}>0\wedge\mathsf{k}\cdot\mathsf{q}\leqslant\mathsf{n}<\mathsf{k}\cdot(\mathsf{q}+1)))).$$

La condition ci-dessus est une combinaison booléenne de conditions mettant en jeu la multiplication et l'ordre, donc elle est primitive récursive, et la fonction quot est primitive récursive par 1.1.9. La fonction reste est alors définie par

$$\mathsf{reste}(\mathsf{n},\mathsf{k}) := \mathsf{diff}(\mathsf{n},\mathsf{k}\cdot\mathsf{quot}(\mathsf{n},\mathsf{k})),$$

donc elle est primitive récursive puisque les fonction diff, mult, et quot le sont.

Soit alors beta la fonction de $\mathbb{N}^3$ dans $\mathbb{N}$ définie par

$$\mathsf{beta}(\mathsf{s},\mathsf{t},\mathsf{i}) := \mathsf{reste}(\mathsf{s},(\mathsf{i}+1)\mathsf{t}+1).$$

Que beta soit primitive récursive résulte de ce que l'addition, la multiplication, et la fonction reste le sont.

Soit alors $(n_0, ..., n_k)$ une suite finie d'entiers. Posons $m := \max(n_0, ..., n_k, k)$ et $t = m!$. Supposons qu'un entier r divise à la fois $(i+1)t+1$ et $(j+1)t+1$ avec $0 \leqslant i < j \leqslant k$. Alors, r divise $(j-i)t$. Or, t et $(i+1)t+1$ sont premiers entre eux, donc r doit diviser $j-i$. Mais on a alors $r \leqslant k$, donc $r \leqslant m$, et, par construction, r divise t. Comme r divise $(i+1)t+1$ par hypothèse, r ne peut être que 1. Autrement dit les nombres $t+1$, $2t+1$, ...,$(k+1)t+1$ sont deux à deux premiers entre eux. Par le lemme des restes chinois, il existe donc un entier s tel que le reste de la division de s par $(i+1)t+1$ soit n_i, c'est-à-dire un entier s satisfaisant l'égalité $\mathsf{beta}(s,t,i) = n_i$ pour $0 \leqslant i \leqslant k$. □

▷ *Le codage de Gödel n'est pas très naturel, et il n'est pas immédiat de déterminer les couples d'entiers codant une suite donnée. D'un autre côté, sa définition ne requiert que des opérations arithmétiques très simples, ce qui sera un élément important pour son application ultérieure dans la sous-section* 3.3. *À titre d'exemple, considérons le cas de la suite* $(2,1)$. *On a ici* $m = t = 2$, *et les couples d'entiers* (s,t) *vérifiant* $\mathsf{beta}(s,t,0) = 2$ *et* $\mathsf{beta}(s,t,1) = 1$, *c'est-à-dire codant, au sens de la fonction* beta, *la suite* $(2,1)$ *sont tous les couples* $(s,2)$ *vérifiant* $s \equiv 2 \pmod 3$ *et* $s \equiv 1 \pmod 5$, *c'est-à-dire les couples* $(11+15r, 2)$ *avec* $r \geqslant 0$. ◁

1.2.3.— La seconde numérotation, cette fois injective, repose sur la décomposition des entiers en produit de facteurs premiers : le principe est de représenter $(n_0, ..., n_k)$ par $p_0^{n_0+1} \cdots p_k^{n_k+1}$, où p_i est le $i^{\text{ième}}$ nombre premier.

Définition (codage des suites).— On définit le *code* $\langle n_0, ..., n_k \rangle$ de la suite $(n_0, ..., n_k)$ comme $p_0^{n_0+1} \cdots p_k^{n_k+1}$; on pose $\langle\rangle := 1$. On note Suite l'ensemble des codes de suite. Si n est un code, on note $\mathsf{lg}(n)$ la longueur de la suite de code n, et, pour $i \leqslant \mathsf{lg}(n)$, on note $\mathsf{coord}(n,i)$ le $i^{\text{ième}}$ facteur de la suite de code n. Si n, m sont des codes, on note $\mathsf{concat}(n,m)$ le code de la concaténation des suites de codes n et m. On prolonge lg, coord et concat par 0 là où les clauses précédentes n'attribuent pas de valeur.

▷ *Par exemple, l'unique numéro associé à la suite* $(2,1)$ via *ce second codage est* $2^3 \cdot 3^2$, *soit* 72. *Le codage ainsi défini est injectif* (*un seul entier par suite*), *mais non surjectif* (*tous les entiers ne sont pas des numéros de suite*) *: seuls les entiers dont les facteurs premiers forment un segment initial de la suite des nombres premiers codent une suite. De nombreux codages alternatifs sont possibles, voir par exemple l'exercice* 51. ◁

1.2.4.— Le point important pour la suite est que chacune des relations et fonctions définies ci-dessus est primitive récursive.

Proposition (codage des suites).— *L'ensemble* Suite *et chacune des fonctions* lg, coord, *et* concat *sont primitifs récursifs.*

Démonstration. La relation « k divise n », notée $k \mid n$, est primitive récursive puisque définie par la quantification bornée $\exists q \leqslant n\, (n = k \cdot q)$. La relation « p est premier » est primitive récursive puisqu'à son tour définie à partir d'une quantification bornée

$$p \neq 0 \ \wedge\ p \neq 1 \ \wedge\ \forall k \leqslant p\, (k \mid p \Rightarrow (k = 1 \vee k = p)).$$

Ensuite, la fonction « factorielle » est primitive récursive puisque définie par récurrence à partir de la multiplication. De là, la fonction prem associant à chaque entier k le $\mathsf{k}^{\text{ième}}$ nombre premier est primitive récursive puisque définie par récurrence et minimisation bornée par $\mathsf{prem}(\mathsf{k}) := 2$ pour $\mathsf{k} = 0$ et

$$\mathsf{prem}(\mathsf{k}) := \mu\mathsf{p}{\leqslant}\mathsf{prem}(\mathsf{k}{-}1)!{+}1\ (\text{« } \mathsf{p} \text{ est premier » et } \mathsf{p} > \mathsf{prem}(\mathsf{k}{-}1))$$

pour $\mathsf{k} \geqslant 1$, définition légitime puisque tous les facteurs premiers de $\mathsf{prem}(\mathsf{k}{-}1)!{+}1$ sont plus grands que $\mathsf{prem}(\mathsf{k}{-}1)$, et donc $\mathsf{prem}(\mathsf{k})$ est borné supérieurement par l'entier $\mathsf{prem}(\mathsf{k}{-}1)!{+}1$. Alors, n code une suite, c'est-à-dire appartient à l'ensemble Suite, si, et seulement si, les diviseurs premiers de n forment un segment initial de la suite des nombres premiers, donc si, et seulement si, on a

$$\mathsf{n} = 1 \vee (\mathsf{n} > 1 \wedge \forall \mathsf{k}{\leqslant}\mathsf{n}\,(\mathsf{prem}(\mathsf{k}) \mid \mathsf{n} \Rightarrow \mathsf{prem}(\mathsf{k}{-}1) \mid \mathsf{n})),$$

et donc Suite est un ensemble primitif récursif.

Alors, $\mathsf{lg}(\mathsf{n})$ est l'indice du plus petit nombre premier ne divisant pas n, donc encore le nombre de facteurs premiers de n, donc la fonction longueur lg est à son tour définie par minimisation bornée par $\mathsf{lg}(\mathsf{n}) := 0$ pour $\mathsf{n} \notin \mathsf{Suite}$ ou $\mathsf{n} = 1$, et

$$\mathsf{lg}(\mathsf{n}) := \mu\mathsf{k}{\leqslant}\mathsf{n}\,(\mathsf{prem}(\mathsf{k}+1) \not\mid \mathsf{n}) \text{ sinon.}$$

Ensuite, $\mathsf{coord}(\mathsf{n},\mathsf{i})$ est le plus grand k tel que $\mathsf{p}_\mathsf{i}^{\mathsf{k}+1}$ divise n, donc la fonction coordonnée coord admet la définition par minimisation bornée

$$\mathsf{coord}(\mathsf{n},\mathsf{i}) := \mu\mathsf{k}{\leqslant}\mathsf{n}{+}1\,(\mathsf{exp}(\mathsf{prem}(\mathsf{i}),\mathsf{k}+2) \not\mid \mathsf{n}).$$

Enfin, $\mathsf{concat}(\mathsf{n}_1,\mathsf{n}_2)$ s'obtient en multipliant n_1 par l'entier obtenu en translatant de $\mathsf{lg}(\mathsf{n}_1)$ les indices des facteurs premiers de n_2, donc la fonction de concaténation concat est définie par $\mathsf{concat}(\mathsf{n}_1,\mathsf{n}_2) := 0$ pour $\mathsf{n}_1 \notin \mathsf{Suite}$ ou $\mathsf{n}_2 \notin \mathsf{Suite}$, par $\mathsf{concat}(\mathsf{n}_1,\mathsf{n}_2) := \mathsf{n}_1$ pour $\mathsf{n}_1 \in \mathsf{Suite}$ et $\mathsf{n}_2 = 1$, et par

$$\mathsf{concat}(\mathsf{n}_1,\mathsf{n}_2) := \mathsf{n}_1 \cdot \textstyle\prod_{\mathsf{i}<\mathsf{lg}(\mathsf{n}_2)} \mathsf{exp}(\mathsf{prem}(\mathsf{lg}(\mathsf{n}_1){+}\mathsf{i}{+}1),\mathsf{coord}(\mathsf{n}_2,\mathsf{i}){+}1)$$

sinon. Toutes les fonctions précédentes sont donc primitives récursives. □

1.2.5.— Une application importante de l'existence d'un codage primitif récursif des suites finies quelconques est la possibilité d'utiliser des définitions par récursion complète, où la valeur de la fonction en k dépend de l'ensemble des valeurs en $0,\dots,\mathsf{k}-1$, et pas seulement en $\mathsf{k}-1$.

Proposition (récursion complète).— *Si* $\mathsf{g} : \mathbb{N}^\mathsf{p} \to \mathbb{N}$ *et* $\mathsf{h} : \mathbb{N}^{\mathsf{p}+2} \to \mathbb{N}$ *sont primitives récursives, alors la fonction* f *de* $\mathbb{N}^{\mathsf{p}+1}$ *dans* $\mathbb{N}$ *définie par*

$$\mathsf{f}(\vec{\mathsf{n}},\mathsf{k}) := \begin{cases} \mathsf{g}(\vec{\mathsf{n}}) & \text{pour } \mathsf{k} = 0, \\ \mathsf{f}(\vec{\mathsf{n}},\mathsf{k}) := \mathsf{h}(\vec{\mathsf{n}},\mathsf{k},\langle \mathsf{f}(\vec{\mathsf{n}},0),\dots,\mathsf{f}(\vec{\mathsf{n}},\mathsf{k}-1)\rangle) & \text{pour } \mathsf{k} > 0 \end{cases}$$

est primitive récursive.

Démonstration. Définissons f^* par $\mathsf{f}^*(\vec{\mathsf{n}},\mathsf{k}) := \langle \mathsf{f}(\vec{\mathsf{n}},0),\dots,\mathsf{f}(\vec{\mathsf{n}},\mathsf{k})\rangle$. Alors, f^* est primitive récursive, puisque définie par $\mathsf{f}^*(\vec{\mathsf{n}},\mathsf{k}) := \langle \mathsf{g}(\vec{\mathsf{n}})\rangle$ pour $\mathsf{k} = 0$ et

$$\mathsf{f}^*(\vec{\mathsf{n}},\mathsf{k}) := \mathsf{concat}(\mathsf{f}^*(\vec{\mathsf{n}},\mathsf{k}-1),\mathsf{h}(\vec{\mathsf{n}},\mathsf{k},\mathsf{f}^*(\vec{\mathsf{n}},\mathsf{k}-1)))$$

pour $\mathsf{k} > 0$, et que, par construction, $\langle \mathsf{m}\rangle$ est $\mathsf{p}_0^{\mathsf{m}+1}$, c'est-à-dire $\mathsf{exp}(2,\mathsf{m}+1)$, une fonction primitive récursive de m. Or, on a $\mathsf{f}(\vec{\mathsf{n}},\mathsf{k}) = \mathsf{coord}(\mathsf{f}^*(\vec{\mathsf{n}},\mathsf{k}),\mathsf{k})$, et on déduit

$$\mathsf{f} = \mathsf{comp}(\mathsf{coord},\mathsf{f}^*,\mathsf{proj}_{\mathsf{p}+1,\mathsf{p}+1}),$$

donc f est primitive récursive puisque f^* et coord le sont. □

1.3. Fonctions et relations récursives

► Résumé.— Fonctions et relations récursives s'obtiennent en passant aux fonctions partielles et ajoutant la minimisation non bornée. ◄

1.3.1.— On considère maintenant les fonctions récursives générales. Celles-ci sont définies à partir des mêmes fonctions de base que les fonctions primitives récursives des sections 1.1 et 1.2, mais en autorisant une opération supplémentaire, à savoir la définition par minimisation (non bornée) qui fait passer d'une fonction g à la fonction f telle que $f(\vec{n})$ est le plus petit m vérifiant $g(\vec{n}, m) = 0$ *quand il existe*. Le changement de point de vue par rapport à la minimisation bornée de 1.1.9 est important car, rien ne garantissant l'existence d'un tel entier m, il se peut que $f(\vec{n})$ ne soit pas définie, et on passe donc des fonctions totales aux fonctions partielles, dont le domaine est une partie éventuellement propre de $\mathbb{N}^p$.

1.3.2.— Dans toute la suite, $f(\vec{n}) = m$ signifie « $f(\vec{n})$ est définie et vaut m », et, de même, $f(\vec{n}) \geqslant m$ signifie « $f(\vec{n})$ est définie et vaut m ou plus ». On étend aux fonctions partielles les opérations de composition et de récursion : si g, $h_1, ..., h_q$ sont des fonctions partielles de $\mathbb{N}^q$ et $\mathbb{N}^p$ dans $\mathbb{N}$, on note $\mathsf{comp}(g, h_1, ..., h_q)$ la fonction partielle f telle que $f(\vec{n}) = m$ est vraie si, et seulement si, il existe $m_1, ..., m_q$ vérifiant $h_i(\vec{n}) = m_i$ pour $1 \leqslant i \leqslant q$, et $g(m_1, ..., m_q) = m$. De même, on note $\mathsf{rec}(g, h)$ la fonction f telle que l'égalité $f(\vec{n}, k) = m$ est vraie si, et seulement si, il existe une suite finie d'entiers m_0, $m_1, ..., m_k = m$ vérifiant $g(\vec{n}) = m_0$ et $h(\vec{n}, i, m_{i-1}) = m_i$ pour $i = 1, ..., k$. Noter que, dans ce cas, si $f(\vec{n}, k)$ n'est pas définie, il en est de même de $f(\vec{n}, j)$ pour $j \geqslant k$.

1.3.3.— Avec ces conventions, on peut introduire l'opération générale de minimisation et la famille des fonctions récursives.

Définition (minimisation, récursif).— (i) Pour $g : \mathbb{N}^{p+1} \to \mathbb{N}$ partielle, on note $\mathsf{minim}(g)$ la fonction partielle f de $\mathbb{N}^p$ dans $\mathbb{N}$, dite obtenue par *minimisation* à partir de g, définie par

$f(\vec{n}) := m$ si on a $g(\vec{n}, k) > 0$ pour $k = 0, 1, ..., m-1$ et $g(\vec{n}, m) = 0$,

et non définie si on a $g(\vec{n}, k) \neq 0$ pour tout k.

(ii) Une fonction partielle $f : \mathbb{N}^p \to \mathbb{N}$ est dite *récursive* si elle peut s'obtenir à partir des fonctions zero, succ et $\mathsf{proj}_{p,i}$ par un nombre fini de compositions, de récursions, et de minimisations. Une relation sur $\mathbb{N}^p$, ou un sous-ensemble de $\mathbb{N}^p$, est dite *récursive* si sa fonction indicatrice l'est.

Exemple. Par définition, toute fonction primitive récursive est récursive, et toute relation primitive récursive est récursive. Si R est un sous-ensemble récursif de $\mathbb{N}^p$, la fonction $g : \mathbb{N}^{p+1} \to \mathbb{N}$ définie par $g(\vec{n}, k) := \mathbf{1}_{\mathbb{N}^p \setminus R}(\vec{n})$ vaut 0 pour $\vec{n}$ dans R, et 1 sinon, indépendamment de k, et elle est récursive puisque R l'est. Considérons alors $f := \mathsf{minim}(g)$. Par construction, $f(\vec{n})$ est définie et vaut 0 pour $\vec{n} \in R$, et n'est pas définie pour $\vec{n} \notin R$: ainsi, pour tout ensemble récursif (donc en particulier tout ensemble primitif récursif) il existe une fonction récursive dont le domaine

est exactement R. *Si* R *est une partie propre de* $\mathbb{N}^{\mathsf{p}}$, *la fonction ainsi définie ne peut être primitive récursive puisqu'elle n'est pas partout définie*[2].

▷ *Il est important de se rappeler qu'une fonction récursive n'est, en général, pas partout définie : il peut être utile de la souligner en parlant alors de fonction* récursive partielle, *et d'appeler* récursive totale *une fonction récursive partielle partout définie* (*comme par exemple une fonction indicatrice*).

On notera une similarité entre les notions de fonction (*primitive*) *récursive et de formule prouvable d'une logique : dans les deux cas, la condition est l'existence d'un témoin qui est une suite finie* (*c'est-à-dire un mot*) *obéissant à des règles syntaxiques, définition par règles de formation à partir de fonctions de base dans un cas, preuve par règles de déduction à partir d'axiomes dans l'autre.* ◁

1.3.4.— On va maintenant mentionner brièvement le lien entre la notion de fonction récursive et celle de fonction calculable par machine de Turing. Le point de départ est la notion de fonction effectivement calculable.

« Définition » (calculable, décidable).— Une fonction (partielle) f (sur $\mathbb{N}^{\mathsf{p}}$) est dite *calculable* s'il existe un algorithme uniforme[3] qui, pour chaque $\vec{\mathsf{n}}$ dans $\mathbb{N}^{\mathsf{p}}$, s'arrête et donne la valeur de $\mathsf{f}(\vec{\mathsf{n}})$ si $\vec{\mathsf{n}}$ est dans le domaine de f, et ne s'arrête pas sinon. De même, une relation R (sur $\mathbb{N}^{\mathsf{p}}$) est dite *décidable* s'il existe un algorithme uniforme qui, pour chaque $\vec{\mathsf{n}}$ dans $\mathbb{N}^{\mathsf{p}}$, détermine si $\mathsf{R}(\vec{\mathsf{n}})$ est vraie ou non.

1.3.5.— La définition précédente ne devient formelle que lorsque l'on précise le type d'algorithme considéré, c'est-à-dire lorsque l'on fixe un modèle de calcul. Celui des *machines de Turing* est usuel, voir par exemple [17].

▷ *Informellement, une machine de Turing* M *est un programme fini spécifiant une famille de transitions légales sur un ensemble* (*infini*) *de configurations, et un calcul de* M *est une suite* (*finie ou infinie*) *de configurations* $(\mathsf{c}_0, \mathsf{c}_1, ...)$ *telle que* $(\mathsf{c}_{\mathsf{n}}, \mathsf{c}_{\mathsf{n}+1})$ *soit une transition légale de* M *pour tout* n. *Dans la version la plus simple, une configuration est une suite de cases indexées par des entiers et contenant chacune au plus une lettre d'un alphabet* Σ *fixé, plus la spécification d'une case particulière dite accessible, plus celle d'un état pris dans un ensemble fini* Q *correspondant intuitivement au contenu de la mémoire vive de* M *: une telle configuration est donc codée par un triplet* $(\mathsf{R}, \mathsf{n}, \mathsf{q})$ *dans* $(\Sigma \cup \{\square\})^{\mathbb{Z}} \times \mathbb{Z} \times \mathsf{Q}$, *où on utilise* $\square$ *pour représenter le blanc* (*absence de caractère*). *On considère une seule sorte de transition, consistant à lire le caractère* s *de la case accessible* n, *remplacer celui-ci par un nouveau caractère* s', *rendre accessible une nouvelle case* n' *avec* $\mathsf{n}' = \mathsf{n} \pm 1$, *et passer dans un nouvel état* q', *les nouvelles valeurs ne dépendant que de* s *et de* q. *Une transition consiste donc à passer de* $(\mathsf{R}, \mathsf{n}, \mathsf{q})$ *à* $(\mathsf{R}', \mathsf{n}', \mathsf{q}')$ *avec* $\mathsf{R}'(\mathsf{i}) = \mathsf{R}(\mathsf{i})$ *pour* $\mathsf{i} \neq \mathsf{n}$, *et* $\mathsf{n}' = \mathsf{n} + \mathsf{d}$ *avec* $\mathsf{d} = \pm 1$, *et* $\mathsf{R}'(\mathsf{n})$, d, *et* q' *dépendant seulement de* $\mathsf{R}(\mathsf{n})$ *et de* q. *La machine de Turing* M *est le programme spécifiant les transitions légales, par exemple sous la forme d'une liste finie de quintuplets* $(\mathsf{s}, \mathsf{q}, \mathsf{s}', \mathsf{q}', \mathsf{d})$ *correspondant à « ancien caractère, ancien état, nouveau caractère, nouvel état, déplacement ». Si, pour chaque valeur de* (s, q), *il existe au plus une transition légale à partir de* (s, q), *la machine* M *est dite* déterministe, *et alors un seul calcul part d'une configuration donnée.* ◁

2. Cet exemple laisse ouverte la question de l'existence d'une fonction récursive partout définie mais non primitive récursive : on verra plus loin qu'il en existe.

3. L'algorithme doit être le même pour *toutes* les valeurs de l'argument.

1.3.6.— Notant $[\mathsf{n}]$ le nombre (suite de chiffres 0 à 9) représentant n en base dix, on peut introduire une notion de calculabilité (presque) précise.

Définition (MT-calculable, MT-décidable).— Une fonction partielle f de $\mathbb{N}^{\mathsf{p}}$ dans $\mathbb{N}$ est dite *calculable par machine de Turing*, ou *MT-calculable*, s'il existe une machine de Turing déterministe M d'alphabet $\{0, 1, ..., 9\}$ calculant f au sens où, quels que soient les entiers $\mathsf{n}_1, ..., \mathsf{n}_{\mathsf{p}}$, le calcul de M à partir de la configuration initiale associée au mot $[\mathsf{n}_1]\square[\mathsf{n}_2]\square\cdots\square[\mathsf{n}_{\mathsf{p}}]$ se termine avec la configuration finale associée au mot $[f(\vec{\mathsf{n}})]$ si la valeur $f(\vec{\mathsf{n}})$ est définie, et ne se termine pas si $f(\vec{\mathsf{n}})$ n'est pas définie.

Une relation R sur $\mathbb{N}^{\mathsf{p}}$ est dite *décidable par machine de Turing*, ou *MT-décidable*, s'il existe une machine de Turing déterministe M décidant R au sens où, pour tous $\mathsf{n}_1, ..., \mathsf{n}_{\mathsf{p}}$, le calcul de M à partir de la configuration initiale associée au mot $[\mathsf{n}_1]\square[\mathsf{n}_2]\square\cdots\square[\mathsf{n}_{\mathsf{p}}]$ se termine dans un état acceptant si $\mathsf{R}(\vec{\mathsf{n}})$ est vraie, et dans un état refusant si $\mathsf{R}(\vec{\mathsf{n}})$ est fausse.

1.3.7.— Une fonction f est donc MT-calculable s'il existe un algorithme calculant f et implantable sur machine de Turing. La définition explicite des machines de Turing rend clair que les calculs de machine de Turing correspondent à l'exécution d'un algorithme, et, de là, toute fonction MT-calculable doit être considérée comme calculable. La *thèse de Church–Turing* TC est l'affirmation que, réciproquement, tout algorithme peut être implanté (moyennant une traduction convenable) sur une machine de Turing, et donc en particulier toute fonction calculable [4] est MT-calculable et tout ensemble décidable est MT-décidable. À ce jour, aucun modèle de calcul ne contredit la thèse de Church–Turing, ce qui justifie les approximations « calculable = MT-calculable » et « décidable = MT-décidable ».

> ▷ *Noter que l'acceptation ou le rejet de la thèse de Church–Turing ne met pas en cause la justesse des résultats portant sur les machines de Turing ; en revanche, si la thèse de Church–Turing venait à être rejetée, c'est la portée de ces résultats qui serait limitée, puisqu'ils référeraient à un modèle jugé non exhaustif.* ◁

1.3.8.— Pour ce qui nous concerne, le point essentiel est que la MT-calculabilité est exactement équivalente à la propriété d'être récursive.

Proposition (équivalence).— *Une fonction de $\mathbb{N}^{\mathsf{p}}$ dans $\mathbb{N}$ est récursive si, et seulement si, elle est calculable par machine de Turing ; une relation sur $\mathbb{N}^{\mathsf{p}}$ est récursive si, et seulement si, elle est décidable par machine de Turing.*

> *Principe de la démonstration.* Dans un sens, il suffit de vérifier que les fonctions de base zero, succ, $\mathsf{proj}_{\mathsf{p,i}}$ sont MT-calculables, puis que la famille des fonctions MT-calculables est close par composition, récursion, et minimisation, ce qui est facile une fois les machines de Turing précisément définies.

4. La version considérée ici est une version forte ; une variante plus faible concernerait seulement les fonctions partout définies.

Dans l'autre sens, *a priori* plus difficile, il s'agit de coder les configurations d'une machine de Turing M par des entiers. Soit r_M la fonction associant à chaque choix de valeurs initiales $\vec{\mathsf{n}}$ et chaque entier t, le code de la configuration obtenue après t étapes de calcul de M à partir des données $\vec{\mathsf{n}}$. Par définition d'une machine de Turing, la configuration à l'instant t s'obtient de façon simple à partir de la configuration à l'instant $\mathsf{t}-1$ et de la machine M, qui est un objet fini. De là, si le codage des configurations est fait de façon raisonnable, la fonction r_M est définie par une récursion sur t et, de ce fait, elle est primitive récursive. Ensuite, f peut être extrait de r_M à l'aide d'une unique minimisation correspondant à la recherche du plus petit t pour lequel la $\mathsf{t}^{\text{ième}}$ étape du calcul est une configuration terminale de M. On conclut que f est récursive. □

▷ *Dans la mesure où le modèle de calcul des machines de Turing relève des langages de programmation impératifs tandis que celui des fonctions récursives relève des langages de programmation fonctionnels, le résultat ci-dessus fait écho au fait que langages impératifs et langages fonctionnels ont la même capacité de calcul. La différence principale entre le point de vue de la calculabilité (par machine de Turing) et celui de la récursivité est que le premier est local en ce qu'il met en jeu la valeur de la fonction pour chaque choix de valeurs pour les arguments, alors que le second est global en ce qu'il considère la fonction par le biais d'une définition indépendante de toute évaluation. Les deux approches ne sont néanmoins pas si opposées puisque, dans l'approche « calculabilité », le point important est l'uniformité du programme, une propriété globale.* ◁

1.3.9.— Comme la thèse de Church–Turing TC est l'affirmation que toute fonction calculable est MT-calculable, et que toute relation décidable est MT-décidable, on peut réénoncer l'équivalence de 1.3.8 comme suit.

Corollaire (équivalence).— (TC) *Une fonction de $\mathbb{N}^\mathsf{p}$ dans $\mathbb{N}$ est calculable si, et seulement si, elle est récursive ; une relation sur $\mathbb{N}^\mathsf{p}$ est décidable si, et seulement si, elle est récursive.*

1.3.10.— Par ailleurs, on tire de la démonstration de 1.3.8 l'application technique suivante, qui sera utilisée plus loin.

Corollaire (unique minimisation).— *Toute fonction récursive peut s'obtenir par une unique minimisation à partir d'une fonction primitive récursive.*

Démonstration. Si f est une fonction récursive, elle est MT-calculable ; or on a vu dans la démonstration de 1.3.8 que, si M est une machine de Turing calculant f, il existe une fonction primitive récursive r_M codant les calculs de M à partir de laquelle la fonction calculée par M, à savoir ici f, s'obtient par une unique minimisation. □

▷ *Une démonstration directe ne passant pas par les machines de Turing serait bien sûr possible, mais elle requerrait des vérifications fastidieuses.* ◁

1.3.11.— Signalons pour terminer cet aperçu schématique qu'il existe toute une théorie de la récursivité dont les résultats mentionnés ici ne sont que les toutes premières étapes. En particulier, il existe plusieurs hiérarchies de complexité dont les ensembles récursifs constituent le premier niveau.

2. Arithmétisation de la syntaxe

Notre but est maintenant d'*arithmétiser* la syntaxe des logiques du premier ordre, à savoir fixer une numérotation des formules et, plus généralement, de toutes les notions mises en jeu dans cette logique. Le point ici est que l'on cherche à définir une telle arithmétisation de façon suffisamment régulière pour que tous les ensembles et fonctions apparaissant soient primitifs récursifs. Comme dans la section précédente, il s'agit de vérifications fastidieuses mais dépourvues de difficulté.

> ▷ *Tous les composants syntaxiques d'une logique du premier ordre, termes, formules, preuves, ont été définis comme des mots sur un alphabet ad hoc. Dans le cas d'une signature finie ou dénombrable, les ensembles de mots concernés sont dénombrables, et il est facile d'en fixer une numérotation. Le seul point requérant ici du soin sera le résultat que la fonction qui, aux numéros d'une formule* Φ *et d'un terme* t *et à un indice* i *associe celui de la formule* $\Phi(\mathsf{x_i}\leftarrow\mathsf{t})$ *est récursive.* ◁

Cette section comporte deux sous-sections. Dans la sous-section 2.1, on numérote les formules des logiques du premier ordre à signature au plus dénombrable de sorte que les paramètres syntaxiques associés à celles-ci puissent se lire comme fonctions primitives récursives des numéros associés. Dans la sous-section 2.2, on numérote de même les preuves en logiques du premier ordre de façon à ce que la prouvabilité corresponde à une relation semi-récursive (*alias* récursivement énumérable).

2.1. Numérotation des formules

> ► Résumé.— On numérote les formules du premier ordre de sorte que les fonctions syntaxiques associées soient primitives récursives. ◄

2.1.1.— Dans toute la suite, on considère des logiques du premier ordre associées à des signatures au plus dénombrables. Il existe alors une signature maximale dans laquelle toute signature dénombrable est incluse.

Notation (signature $\Sigma_{\max}$).— On note $\Sigma_{\max}$ la signature contenant, pour chaque arité k, une suite dénombrable $(\mathsf{s_{k,i}})_{\mathsf{i}\geqslant 1}$ de symboles d'opération k-aire (avec $\mathsf{k}\geqslant 0$), et une suite dénombrable$(\mathsf{r_{k,i}})_{\mathsf{i}\geqslant 1}$ de symboles de relation k-aire, (avec $\mathsf{k}\geqslant 1$), et $\mathcal{L}_{\max}$ la logique du premier ordre associée.

Toutes les logiques du premier ordre à signature au plus dénombrable, telles $\mathcal{L}_{\mathrm{arith}}$ ou $\mathcal{L}_{\mathrm{ens}}$, pourront être considérées comme incluses dans $\mathcal{L}_{\max}$. Dans le cas de l'arithmétique, on conviendra que les symboles $\mathsf{0}$, S, $+$, et $\cdot$ correspondent respectivement à $\mathsf{s}_{0,1}$, $\mathsf{s}_{1,1}$, $\mathsf{s}_{2,1}$, et $\mathsf{s}_{2,2}$.

2.1.2.— Le principe pour une numérotation des termes et des formules de $\mathcal{L}_{\max}$ est naturel : les formules sont des suites finies de symboles, donc, une fois ceux-ci numérotés, par exemple suivant le principe de VII.1.1.10, on obtient une numérotation des formules à partir de toute numérotation des suites finies d'entiers, par exemple celle de 1.2.3.

Définition (numérotation).— (i) Pour t terme de $\mathcal{L}_{\max}$, on définit inductivement $\ulcorner\mathsf{t}\urcorner$ par $\ulcorner\mathsf{x_i}\urcorner := \langle 0,\mathsf{i}\rangle$, $\ulcorner\mathsf{s_{0,i}}\urcorner := \langle 1,0,\mathsf{i}\rangle$, et

$\ulcorner\mathsf{t}\urcorner := \langle\langle 1,\mathsf{k},\mathsf{i}\rangle,\ulcorner\mathsf{t_1}\urcorner,\dots,\ulcorner\mathsf{t_k}\urcorner\rangle$ si t est $\mathsf{s_{k,i}}(\mathsf{t_1},\dots,\mathsf{t_k})$.

(ii) Pour Φ formule atomique de $\mathcal{L}_{\max}$, on définit $\ulcorner\Phi\urcorner$ par

$\ulcorner\Phi\urcorner := \langle\langle 2,\mathsf{k},\mathsf{i}\rangle,\ulcorner\mathsf{t_1}\urcorner,\dots,\ulcorner\mathsf{t_k}\urcorner\rangle$ si Φ est $\mathsf{r_{k,i}}(\mathsf{t_1},\dots,\mathsf{t_k})$, en convenant que le symbole d'égalité $=$ reçoit le numéro $\langle 2,2,0\rangle$.

(iii) Pour Φ formule de $\mathcal{L}_{\max}$, on définit inductivement $\ulcorner\Phi\urcorner$ par

$\ulcorner\Phi\urcorner := \langle 1,\ulcorner\Psi\urcorner\rangle$ pour $\Phi = \neg(\Psi)$, $\ulcorner\Phi\urcorner := \langle\ulcorner\mathsf{c}\urcorner,\ulcorner\Psi\urcorner,\ulcorner\Theta\urcorner\rangle$ pour $\Phi = (\Psi)\,\mathsf{c}\,(\Theta)$ avec $\ulcorner\Rightarrow\urcorner := 2$, $\ulcorner\vee\urcorner := 3$, $\ulcorner\wedge\urcorner := 4$, et $\ulcorner\Phi\urcorner := \langle\ulcorner Q\mathsf{x_i}\urcorner,\ulcorner\Psi\urcorner\rangle$ pour $\Phi = Q\mathsf{x_i}(\Psi)$ avec $\ulcorner\exists\mathsf{x_i}\urcorner := \langle 1,\mathsf{i}\rangle$ et $\ulcorner\forall\mathsf{x_i}\urcorner := \langle 2,\mathsf{i}\rangle$.

Exemple. On trouve donc $\ulcorner\mathsf{x_2}\urcorner = \langle 0,2\rangle = 2^{1+0}.3^{1+2} = 54$. *De même, il vient*

$$\ulcorner\mathsf{0}\urcorner = \ulcorner\mathsf{s_{0,1}}\urcorner = \ulcorner 1,0,1\urcorner = \langle 1,0,1\rangle = 2^{1+1}.3^{1+0}.5^{1+1} = 300,$$

puis $\ulcorner=\urcorner = \langle 2,2,0\rangle = 2^{1+2}.3^{1+2}.5^{1+0} = 1080$. *De là, on déduit*

$$\ulcorner\mathsf{x_2}=\mathsf{0}\urcorner = \langle\ulcorner=\urcorner,\ulcorner\mathsf{x_2}\urcorner,\ulcorner\mathsf{0}\urcorner\rangle = \langle 1080,54,300\rangle = 2^{1081}.3^{55}.5^{301},$$

qui est un (très) grand entier.

▷ *On voit sur l'exemple le caractère contingent de la représentation : le choix des numéros des symboles de base est sans importance, de même que celui de l'ordre de description des objets. Le point important est que le codage ainsi obtenu est non ambigu, et suffisamment simple au sens où, partant d'un entier, on peut effectivement retrouver le terme ou la formule dont il est le numéro.* ◁

2.1.3.— Dans notre contexte, la notion de simplicité visée est celle d'ensemble primitif récursif, et le premier résultat indispensable est le suivant.

Lemme.— *L'ensemble* Var *des numéros des variables de* $\mathcal{L}_{\max}$ *est un ensemble primitif récursif. Il en est de même de l'ensemble* Term *des numéros des termes, et de l'ensemble* Form *des numéros des formules.*

Démonstration. Par construction, on a $\mathsf{n_i} < \mathsf{n}$ si n est le code de la suite $(\mathsf{n_0},\dots,\mathsf{n_k})$. Il en résulte que l'on peut inverser les définitions en utilisant des quantifications bornées et déduire de 1.1.7 que l'on ne sort pas du cadre primitif récursif. Ainsi, on peut caractériser l'ensemble Var des numéros de variables en définissant $\mathsf{Var}(\mathsf{n})$ comme étant

$$\exists \mathsf{i}{<}\mathsf{n}\,(\mathsf{i}\geqslant 1\wedge \mathsf{n}=\langle 0,\mathsf{i}\rangle),$$

et l'ensemble Var est donc primitif récursif. Le même argument montre que l'ensemble Const des numéros des symboles de constantes de $\mathcal{L}_{\max}$, c'est-à-dire les entiers de la forme $\langle 1,0,\mathsf{i}\rangle$ avec $\mathsf{i}\geqslant 1$, est primitif récursif, puis qu'il en est de même pour l'ensemble des numéros de symboles de fonctions d'arité k, de la forme $\langle 1,\mathsf{k},\mathsf{i}\rangle$, et de ceux des symboles de relations d'arité k, de la forme $\langle 2,\mathsf{k},\mathsf{i}\rangle$.

Pour les termes, comme, par construction, le numéro de $\mathsf{s}(\mathsf{t_1},\dots,\mathsf{t_k})$ est plus grand que celui de s et de chaque $\mathsf{t_i}$, on peut à nouveau utiliser des quantifications bornées. Mais, comme la définition est récursive et non directe, il faut en outre utiliser une récursion, et même une récursion complète, puisque le numéro de $\mathsf{s}(\mathsf{t_1},\dots,\mathsf{t_k})$ s'obtient à partir des numéros de s et des $\mathsf{t_i}$, qui sont des prédécesseurs du numéro à définir, mais pas le prédécesseur immédiat en général. On peut caractériser

l'ensemble Term des numéros de termes en définissant $\mathsf{Term}(n)$ comme

$$\begin{aligned}&\mathsf{Var}(n) \vee \mathsf{Const}(n)\\ &\quad\vee (\mathsf{Suite}(n) \wedge \exists k, i{<}n\, (\mathsf{lg}(n) = k+1 \wedge \mathsf{coord}(n,0) = \langle 1,k,i\rangle\\ &\qquad\wedge \forall j{\leqslant}k\, (j \geqslant 1 \Rightarrow \mathsf{Term}(\mathsf{coord}(n,j))))),\end{aligned}$$

une définition par récursion complète, où la valeur (0 ou 1) de $\mathsf{Term}(n)$ est définie à partir de la suite des valeurs $\mathsf{Term}(k)$ pour $0 \leqslant k < n$: avec les notations de 1.2.5, on a $\mathbf{1}_{\mathsf{Term}}(n) = 0$ pour $n = 0$ et $\mathbf{1}_{\mathsf{Term}}(n) = f(n, \langle \mathbf{1}_{\mathsf{Term}}(0), \dots, \mathbf{1}_{\mathsf{Term}}(n-1)\rangle)$ pour $n > 0$, où f est définie par

$$\begin{aligned}f(n,m) := {}& \mathbf{1}_{\mathsf{Var}}(n) + \mathbf{1}_{\mathsf{Const}}(n) + \mathbf{1}_{\mathsf{Suite}}(n) \cdot \textstyle\sum_{k<n, i<n}\\ &(\mathbf{1}_{\ll \mathsf{coord}(n,0)=\langle 1,k,i\rangle \gg}(n) \cdot \mathbf{1}_{\ll \mathsf{lg}(n)=k+1\gg}(n) \cdot \prod_{1\leqslant j\leqslant k} \mathsf{coord}(m, \mathsf{coord}(n,j))).\end{aligned}$$

On conclut que Term est un ensemble primitif récursif.

Ensuite, on peut caractériser l'ensemble Atom des numéros de formules atomiques en définissant $\mathsf{Atom}(n)$ comme

$$\begin{aligned}&\mathsf{Suite}(n) \wedge\\ &\quad((\mathsf{lg}(n) = 3 \wedge \mathsf{coord}(n,0) = \ulcorner{=}\urcorner\\ &\qquad\wedge \mathsf{Term}(\mathsf{coord}(n,1)) \wedge \mathsf{Term}(\mathsf{coord}(n,2)))\\ &\quad\vee \exists k, i{<}n\, (\mathsf{lg}(n) = k+1 \wedge \mathsf{coord}(n,0) = \langle 2,k,i\rangle\\ &\qquad\wedge \forall j{\leqslant}k\, (j \geqslant 1 \Rightarrow \mathsf{Term}(\mathsf{coord}(n,j))))),\end{aligned}$$

et il est donc primitif récursif.

Enfin l'argument pour l'ensemble Form des numéros de formules est du même type que pour les termes, en définissant $\mathsf{Form}(n)$ comme

$$\begin{aligned}&\mathsf{Atom}(n) \vee (\mathsf{Suite}(n) \wedge\\ &\quad((\mathsf{coord}(n,0) = \ulcorner\neg\urcorner \wedge \mathsf{lg}(n) = 2 \wedge \mathsf{Form}(\mathsf{coord}(n,1)))\\ &\qquad\vee ((\mathsf{coord}(n,0) = \ulcorner\Rightarrow\urcorner \vee \mathsf{coord}(n,0) = \ulcorner\wedge\urcorner \vee \mathsf{coord}(n,0) = \ulcorner\vee\urcorner)\\ &\qquad\quad\wedge \mathsf{lg}(n) = 3 \wedge \mathsf{Form}(\mathsf{coord}(n,1)) \wedge \mathsf{Form}(\mathsf{coord}(n,2)))\\ &\qquad\vee \exists i{<}n\, ((\mathsf{coord}(n,0) = \ulcorner\exists x_i\urcorner \vee \mathsf{coord}(n,0) = \ulcorner\forall x_i\urcorner)\\ &\qquad\quad\wedge \mathsf{lg}(n) = 2 \wedge \mathsf{Form}(\mathsf{coord}(n,1))))).\end{aligned}$$

C'est une définition par récursion complète, et donc Form est primitif récursif. □

2.1.4.— Si Φ est une formule, t un terme et i un entier naturel, la substitution de t aux occurrences libres de x_i dans Φ fournit une nouvelle formule $\Phi(x_i{\leftarrow}t)$. Passant aux numéros, on obtient une fonction de $\mathbb{N}^3$ dans $\mathbb{N}$. Il est important pour la suite que celle-ci soit suffisamment simple.

Proposition (substitution).— *La fonction sur* $\mathbb{N}^3$ *définie par*

$$\mathsf{subst}_{\mathsf{Form}}(\ulcorner\Phi\urcorner, \ulcorner t\urcorner, i) := \ulcorner\Phi(x_i{\leftarrow}t)\urcorner, \qquad (\#1)$$

et prolongée par 0 *hors de* $\mathsf{Form} \times \mathsf{Term} \times \mathbb{N}$*, est primitive récursive.*

Démonstration. On définit d'abord une fonction de substitution $\mathsf{subst}_{\mathsf{Term}}$ pour les termes par

$\mathsf{subst}_{\mathsf{Term}}(n,m,i) := 0$ pour $\neg\mathsf{Term}(n)$ ou $\neg\mathsf{Term}(m)$,
$\mathsf{subst}_{\mathsf{Term}}(n,m,i) := n$ pour $\mathsf{Const}(n)$ ou $\mathsf{Var}(n)$ avec $\mathsf{coord}(n,1) \neq i$,
$\mathsf{subst}_{\mathsf{Term}}(n,m,i) := m$ pour $n = \langle 0, i\rangle$,
$\mathsf{subst}_{\mathsf{Term}}(n,m,i) := \langle \mathsf{coord}(n,0), \mathsf{subst}_{\mathsf{Term}}(\mathsf{coord}(n,1),m,i), \dots, \mathsf{subst}_{\mathsf{Term}}(\mathsf{coord}(n,k),m,i)\rangle$
pour $\mathsf{Term}(n)$ et $\neg\mathsf{Var}(n)$ et $\neg\mathsf{Const}(n)$ et $\mathsf{lg}(n) = k+1$.

Comme les clauses ci-dessus correspondent à une récursion complète, la fonction $\mathsf{subst}_{\mathsf{Term}}$ ainsi définie est primitive récursive, et, par construction, on a

$$\mathsf{subst}_{\mathsf{Term}}(\ulcorner t\urcorner, \ulcorner u\urcorner, i) = \ulcorner t(x_i{\leftarrow}u)\urcorner,$$

pour tous termes t, u et tout indice i.

Ensuite, on définit de même une fonction de substitution pour les formules par

$\mathsf{subst}_{\mathsf{Form}}(\mathsf{n},\mathsf{m},\mathsf{i}) := 0$ pour $\neg\mathsf{Form}(\mathsf{n})$ ou $\neg\mathsf{Term}(\mathsf{m})$,

$\mathsf{subst}_{\mathsf{Form}}(\mathsf{n},\mathsf{m},\mathsf{i}) := \langle \mathsf{coord}(\mathsf{n},0), \mathsf{subst}_{\mathsf{Term}}(\mathsf{coord}(\mathsf{n},1),\mathsf{m},\mathsf{i}), \ldots$
$\ldots, \mathsf{subst}_{\mathsf{Term}}(\mathsf{coord}(\mathsf{n},\mathsf{k}),\mathsf{m},\mathsf{i})\rangle$
pour $\mathsf{Atom}(\mathsf{n})$ et $\mathsf{lg}(\mathsf{n}) = \mathsf{k}+1$,

$\mathsf{subst}_{\mathsf{Form}}(\mathsf{n},\mathsf{m},\mathsf{i}) := \langle \mathsf{coord}(\mathsf{n},0), \mathsf{subst}_{\mathsf{Form}}(\mathsf{coord}(\mathsf{n},1),\mathsf{m},\mathsf{i})\rangle$
pour $\mathsf{Form}(\mathsf{n})$ et $\neg\mathsf{Atom}(\mathsf{n})$ et $\mathsf{coord}(\mathsf{n},0) = \ulcorner\neg\urcorner$,

$\mathsf{subst}_{\mathsf{Form}}(\mathsf{n},\mathsf{m},\mathsf{i}) := \langle \mathsf{coord}(\mathsf{n},0), \mathsf{subst}_{\mathsf{Form}}(\mathsf{coord}(\mathsf{n},1),\mathsf{m},\mathsf{i}),$
$\ldots, \mathsf{subst}_{\mathsf{Form}}(\mathsf{coord}(\mathsf{n},2),\mathsf{m},\mathsf{i})\rangle$
pour $\mathsf{Form}(\mathsf{n})$ et $\neg\mathsf{Atom}(\mathsf{n})$ et $\mathsf{coord}(\mathsf{n},0) = \ulcorner\Rightarrow\urcorner, \ulcorner\vee\urcorner$, ou $\ulcorner\wedge\urcorner$,

$\mathsf{subst}_{\mathsf{Form}}(\mathsf{n},\mathsf{m},\mathsf{i}) := \langle \mathsf{coord}(\mathsf{n},0), \mathsf{subst}_{\mathsf{Form}}(\mathsf{coord}(\mathsf{n},2),\mathsf{m},\mathsf{i})\rangle$
pour $\mathsf{Form}(\mathsf{n})$ et $\neg\mathsf{Atom}(\mathsf{n})$ et $\mathsf{Suite}(\mathsf{coord}(\mathsf{n},0))$
et $\mathsf{coord}(\mathsf{coord}(\mathsf{n},0),0) = 1$ ou 2 et $\mathsf{coord}(\mathsf{coord}(\mathsf{n},0),1) \neq \mathsf{i}$,

$\mathsf{subst}_{\mathsf{Form}}(\mathsf{n},\mathsf{m},\mathsf{i}) := \mathsf{n}$ pour $\mathsf{Form}(\mathsf{n})$ et $\neg\mathsf{Atom}(\mathsf{n})$ et $\mathsf{Suite}(\mathsf{coord}(\mathsf{n},0))$
et $\mathsf{coord}(\mathsf{coord}(\mathsf{n},0),0) = 1$ ou 2 et $\mathsf{coord}(\mathsf{coord}(\mathsf{n},0),1) = \mathsf{i}$.

La fonction $\mathsf{subst}_{\mathsf{Form}}$ est définie par une récursion complète par rapport à la variable n, et elle est donc primitive récursive. D'autre part, par construction, la relation $\mathsf{subst}_{\mathsf{Form}}(\ulcorner\Phi\urcorner, \ulcorner\mathsf{t}\urcorner, \mathsf{i}) = \ulcorner\Phi(\mathsf{x}_\mathsf{i}\leftarrow\mathsf{t})\urcorner$ est vérifiée. □

2.1.5.— Comme application directe on déduit :

Corollaire (substitution).— *La fonction* subst *associant à tout couple de la forme* $(\ulcorner\Phi\urcorner, \mathsf{n})$*, avec* $\Phi(\mathsf{x}_1)$ *formule à une variable libre de* $\mathcal{L}_{\max}$ *et* n *entier, le numéro de la formule* $\Phi(\mathsf{S}^\mathsf{n}0)$ *est primitive récursive.*

Démonstration. La fonction f associant à un entier k le numéro du terme $\mathsf{S}^\mathsf{k}0$ est primitive récursive, puisque définie par la récursion $\mathsf{f}(\mathsf{k}) = \langle 1,0,1\rangle$ pour $\mathsf{k} = 0$ et

$$\mathsf{f}(\mathsf{k}) = \langle\langle 1,1,1\rangle, \mathsf{f}(\mathsf{k}-1)\rangle$$

pour $\mathsf{k} > 0$. En la composant avec la fonction $\mathsf{subst}_{\mathsf{Form}}$ de 2.1.4, et avec la fonction constante de valeur 1, on obtient la fonction cherchée. □

2.2. Numérotation des preuves

▶ Résumé.— On numérote les preuves de sorte que les formules prouvables à partir d'une théorie récursive forment un ensemble semi-récursif. ◀

2.2.1.— Ayant numéroté les formules, on numérote maintenant les preuves, qui sont des suites de formules obéissant à des règles. À nouveau, le point est de définir la numérotation de façon à ce que les notions utiles correspondent à des relations simples, en l'occurrence primitives récursives.

Convention (récursif).— Un ensemble de formules T de $\mathcal{L}_{\max}$ est dit primitif récursif (*resp.,* récursif) si l'ensemble $\ulcorner\mathsf{T}\urcorner$ des numéros des formules appartenant à T l'est.

2.2.2.— On commence par les numéros des axiomes de $\mathcal{L}_{\max}$.

Lemme.— *L'ensemble des axiomes de $\mathcal{L}_{\max}$ est primitif récursif.*

Démonstration. Les axiomes se répartissent en plusieurs familles. La première est celle des instances d'axiomes de la logique propositionnelle. Considérons le cas de l'axiome $\mathsf{X}_1 \Rightarrow (\mathsf{X}_2 \Rightarrow \mathsf{X}_1)$. Une formule Φ de $\mathcal{L}_{\max}$ est une instance de cet axiome si, et seulement si, il existe des formules Φ_1, Φ_2 telles que Φ est $\Phi_1 \Rightarrow (\Phi_2 \Rightarrow \Phi_1)$. Un entier n est le numéro d'une telle formule si, et seulement si, il existe des numéros de formule n_1, n_2, nécessairement inférieurs à n, tels que n est le code de $\Phi_1 \Rightarrow (\Phi_2 \Rightarrow \Phi_1)$, en appelant Φ_i la formule de numéro n_i, donc si, et seulement si, n satisfait

$$\exists \mathsf{n}_1, \mathsf{n}_2 < \mathsf{n}\, (\mathsf{Form}(\mathsf{n}_1) \wedge \mathsf{Form}(\mathsf{n}_2) \wedge \mathsf{n} = \langle \ulcorner \Rightarrow \urcorner, \mathsf{n}_1, \langle \ulcorner \Rightarrow \urcorner, \mathsf{n}_2, \mathsf{n}_1 \rangle \rangle).$$

Cette condition ne met en jeu que des quantifications bornées et des fonctions primitives récursives, et elle définit donc un ensemble primitif récursif. Il en est de même des instances de chacun des axiomes de la logique propositionnelle, et, comme il existe un nombre fini de ceux-ci, on déduit que l'ensemble des instances d'axiomes de $\mathcal{L}_\bullet$ est un sous-ensemble primitif récursif de Form.

La seconde famille d'axiomes comprend les formules $\forall \mathsf{x}(\Phi \Rightarrow \Psi) \Rightarrow (\Phi \Rightarrow \forall \mathsf{x}(\Psi))$. Là encore, il s'agit d'une condition purement syntaxique et un entier n est le numéro d'un tel axiome si, et seulement si, il existe des entiers i, n_1, n_2 strictement inférieurs à n correspondant respectivement à l'indice de la variable x et aux numéros des formules Φ et Ψ et donc à partir desquels n s'exprime de façon primitive récursive. L'argument est similaire pour les axiomes du type $\neg \forall \mathsf{x}(\Phi) \Leftrightarrow \exists \mathsf{x}(\neg \Phi)$.

Le cas des axiomes $\forall \mathsf{x}(\Phi) \Rightarrow \Phi(\mathsf{x} \leftarrow \mathsf{t})$ est *a priori* plus délicat puisqu'y figure la substitution et que, de surcroît, on requiert la condition supplémentaire que le terme t soit libre pour la variable x dans Φ. Or, la question de la substitution a été réglée en 2.1.4. Ensuite, il est facile de définir récursivement une relation primitive récursive $\mathsf{Occure}(\mathsf{m}, \mathsf{i})$ telle que, si m est le numéro d'un terme t ou d'une formule Φ, alors $\mathsf{Occure}(\mathsf{m}, \mathsf{i})$ est vrai si, et seulement si, la variable x_i a au moins une occurrence dans t ou dans Φ. Par construction, $\mathsf{Occure}(\mathsf{m}, \mathsf{i})$ ne peut être vraie que pour $\mathsf{i} < \mathsf{m}$. Une récursion parallèle à celle utilisée pour définir la fonction $\mathsf{subst}_{\mathsf{Form}}$ permet alors de définir une relation $\mathsf{Libre}(\mathsf{n}, \mathsf{m}, \mathsf{i})$ de sorte que, si n est le numéro d'une formule Φ et m le numéro d'un terme t, alors $\mathsf{Libre}(\mathsf{n}, \mathsf{m}, \mathsf{i})$ est vraie si, et seulement si, t est libre pour x_i dans Φ. La récursion se fait sur n, et on observe que t est libre pour x_i dans $Q\mathsf{x}(\Psi)$ si ou bien x_i n'apparaît pas dans $Q\mathsf{x}(\Psi)$, ou bien x n'apparaît pas dans t et t est libre pour x_i dans Ψ. On peut donc prendre comme clauses de définition

$\mathsf{Libre}(\mathsf{n}, \mathsf{m}, \mathsf{i}) = 0$ pour $\neg \mathsf{Form}(\mathsf{n})$ ou $\neg \mathsf{Term}(\mathsf{m})$,
$\mathsf{Libre}(\mathsf{n}, \mathsf{m}, \mathsf{i}) = 1$ pour $\mathsf{Atom}(\mathsf{n})$,
$\mathsf{Libre}(\mathsf{n}, \mathsf{m}, \mathsf{i}) = \mathsf{Libre}(\mathsf{coord}(\mathsf{n}, 1), \mathsf{m}, \mathsf{i})$
pour $\mathsf{Form}(\mathsf{n})$ et $\neg \mathsf{Atom}(\mathsf{n})$ et $\mathsf{coord}(\mathsf{n}, 0) = \ulcorner \neg \urcorner$,
$\mathsf{Libre}(\mathsf{n}, \mathsf{m}, \mathsf{i}) = \min(\mathsf{Libre}(\mathsf{coord}(\mathsf{n}, 1), \mathsf{m}, \mathsf{i}), \mathsf{Libre}(\mathsf{coord}(\mathsf{n}, 2), \mathsf{m}, \mathsf{i}))$
pour $\mathsf{Form}(\mathsf{n})$ et $\neg \mathsf{Atom}(\mathsf{n})$ et $\mathsf{coord}(\mathsf{n}, 0) = \ulcorner \wedge \urcorner$, $\ulcorner \vee \urcorner$ ou $\ulcorner \Rightarrow \urcorner$,
$\mathsf{Libre}(\mathsf{n}, \mathsf{m}, \mathsf{i}) = \max(\neg \mathsf{Occure}(\mathsf{n}, \mathsf{i}), \inf(\mathsf{Occure}(\mathsf{n}, \mathsf{i}), \mathsf{Libre}(\mathsf{coord}(\mathsf{n}, 3), \mathsf{m}, \mathsf{i})))$
pour $\mathsf{Form}(\mathsf{n})$ et $\neg \mathsf{Atom}(\mathsf{n})$ et $\mathsf{coord}(\mathsf{n}, 0) = \ulcorner \exists \urcorner$ ou $\ulcorner \forall \urcorner$.

La relation Libre est donc primitive récursive. A partir de là, il est facile d'établir que les numéros des axiomes du type $\forall \mathsf{x}(\Phi) \Rightarrow \Phi(\mathsf{x} \leftarrow \mathsf{t})$ avec t libre pour x dans Φ est un ensemble primitif récursif.

Enfin les axiomes pour l'égalité ne posent pas de problème, puisque l'on peut écrire directement une formule explicite. □

2.2.3.— On arrive maintenant au résultat escompté : l'arithmétisation patiemment définie garantit que les preuves en logique du premier ordre sont des notions récursives.

Proposition (complexité des preuves).— *Si* T *est un ensemble primitif récursif (resp., récursif) de formules de* $\mathcal{L}_{\max}$*, la relation* $\mathsf{Preuve}_{\mathsf{T}}(\mathsf{n},\mathsf{m})$ *exprimant que* m *est le numéro d'une preuve à partir de* T *pour la formule de numéro* n *est primitive récursive (resp., récursive).*

Démonstration. Soit n le numéro de Φ. Un entier m est le numéro d'une preuve de Φ à partir de T si, et seulement si, c'est le numéro d'une suite finie de formules finissant avec Φ et telle que chaque formule est un axiome, ou un élément de T, ou est obtenue par généralisation à partir d'une formule antérieure de la suite, ou est obtenue par coupure à partir de deux formules antérieures de la suite. En notant $\mathsf{Axiom}(\mathsf{n})$ la relation primitive récursive caractérisant les numéros d'axiomes construite en 2.2.2, on peut définir $\mathsf{Preuve}_{\mathsf{T}}(\mathsf{n},\mathsf{m})$ comme

$$\begin{aligned}&\mathsf{Form}(\mathsf{n}) \wedge \mathsf{Suite}(\mathsf{m}) \wedge \mathsf{coord}(\mathsf{m},\mathsf{lg}(\mathsf{m})-1) = \mathsf{n}\\ &\quad\wedge \forall \mathsf{k}{<}\mathsf{lg}(\mathsf{m})\,(\mathsf{Form}(\mathsf{coord}(\mathsf{m},\mathsf{k}))\\ &\quad\wedge (\mathsf{Axiom}(\mathsf{coord}(\mathsf{m},\mathsf{k})) \vee \mathbf{1}^{\ulcorner\mathsf{T}\urcorner}(\mathsf{coord}(\mathsf{m},\mathsf{k}))\\ &\qquad\vee \exists \mathsf{i}{<}\mathsf{k}\,(\mathsf{coord}(\mathsf{m},\mathsf{k}) = \mathsf{gen}(\mathsf{coord}(\mathsf{m},\mathsf{i}),\mathsf{coord}(\mathsf{coord}(\mathsf{m},\mathsf{k}),1)))\\ &\qquad\vee \exists \mathsf{i},\mathsf{j}{<}\mathsf{k}\,(\mathsf{coord}(\mathsf{m},\mathsf{k}) = \mathsf{coupure}(\mathsf{coord}(\mathsf{m},\mathsf{i}),\mathsf{coord}(\mathsf{m},\mathsf{j})))),\end{aligned}$$

où gen et $\mathsf{coupure}$ sont les deux fonctions primitives récursives définies par

$$\mathsf{gen}(\mathsf{n},\mathsf{i}) := \langle {}^{\ulcorner}\forall \mathsf{x}_{\mathsf{i}}{}^{\urcorner},\mathsf{n}\rangle,$$

$$\mathsf{coupure}(\mathsf{n},\mathsf{p}) := \mathsf{coord}(\mathsf{p},2) \quad \text{si l'on a } \mathsf{Form}(\mathsf{p}) \wedge \mathsf{coord}(\mathsf{p},0) = {}^{\ulcorner}\Rightarrow{}^{\urcorner} \wedge \mathsf{coord}(\mathsf{p},1) = \mathsf{n},$$

$$\mathsf{coupure}(\mathsf{n},\mathsf{p}) := 0 \text{ sinon.}$$

La complexité de T dicte alors la complexité de $\mathsf{Preuve}_{\mathsf{T}}$: si T est primitif récursif, c'est-à-dire si la fonction $\mathbf{1}^{\ulcorner\mathsf{T}\urcorner}$ est primitive récursive, alors $\mathsf{Preuve}_{\mathsf{T}}$ est une relation primitive récursive ; si $\mathbf{1}^{\ulcorner\mathsf{T}\urcorner}$ est récursive, alors $\mathsf{Preuve}_{\mathsf{T}}$ est récursive. □

Exemple. Les axiomes du système de Peano du premier ordre PA_1 *forment une famille infinie de formules closes de* $\mathcal{L}_{\mathrm{arith}}$*, donc* a fortiori *de* $\mathcal{L}_{\max}$*. Cette famille est primitive récursive. En effet, d'abord, la famille des numéros de formules de* $\mathcal{L}_{\mathrm{arith}}$ *est primitive récursive puisqu'obtenue à partir de celle de* $\mathcal{L}_{\max}$ *en restreignant les symboles. Ensuite, les axiomes d'induction sont obtenus en énumérant les formules* Ind_{Φ} *:*

$$(\Phi(\mathsf{x}_1{\leftarrow}0) \wedge \forall \mathsf{x}_1(\Phi{\Rightarrow}\Phi(\mathsf{x}_1{\leftarrow}\mathsf{S}(\mathsf{x}_1)))) \Rightarrow \forall \mathsf{x}_1(\Phi)$$

pour Φ *formule de* $\mathcal{L}_{\mathrm{arith}}$*, et il existe une fonction primitive récursive associant au numéro de toute formule* Φ *le numéro de la formule* Ind_{Φ} *correspondant. La proposition implique donc que la relation* $\mathsf{Preuve}_{\mathsf{PA}_1}(\mathsf{n},\mathsf{m})$ *exprimant que* m *code une preuve à partir de* PA_1 *pour la formule de numéro* n *est primitive récursive.*

2.2.4.— Le résultat de 2.2.3 entraîne que l'ensemble des formules prouvables à partir d'un ensemble récursif de formules est la *projection* d'un ensemble récursif. On a vu dans la section 1 que la famille des relations primitives récursives est close par quantification bornée, et le même argument montre qu'il en est de même de la famille des relations récursives. En revanche, on n'a rien affirmé quant aux quantifications quelconques, ou, ce qui revient au même, quant aux projections d'ensembles (primitifs) récursifs. On introduit donc une nouvelle notion.

Définition (semi-récursif).— Un sous-ensemble S de $\mathbb{N}^{\mathsf{p}}$ est dit *semi-récursif* [5] s'il existe $\mathsf{q} \geqslant \mathsf{p}$ et R récursif sur $\mathbb{N}^{\mathsf{q}}$ tels que S est la projection de R, c'est-à-dire que $\vec{\mathsf{n}} \in \mathsf{S}$ équivaut à $\exists \vec{\mathsf{m}}\, ((\vec{\mathsf{n}}, \vec{\mathsf{m}}) \in \mathsf{R})$.

Par définition, tout ensemble récursif est semi-récursif, mais on verra dans la section 4 qu'il existe des ensembles semi-récursifs non récursifs. Il est facile de vérifier que tout ensemble semi-récursif peut s'exprimer comme l'image d'une fonction récursive partielle (donc être « récursivement énuméré »), et que tout ensemble semi-récursif non vide peut s'exprimer comme l'image d'une fonction récursive totale, voir exercice 52.

2.2.5.— On peut alors conclure avec l'important résultat suivant, récompense (bien méritée !) des efforts des deux premières parties de ce chapitre.

Corollaire (complexité de la prouvabilité).— *Si* T *est une théorie récursive de* $\mathcal{L}_{\max}$*, l'ensemble des formules closes prouvables à partir de* T *est semi-récursif.*

Démonstration. Une formule close Φ de numéro n est prouvable à partir de T si, et seulement si, il existe un entier m tel que m est le numéro d'une preuve de Φ à partir de T, si, et seulement si, la relation $\exists \mathsf{x}\, (\mathsf{Preuve}_{\mathsf{T}}(\mathsf{n}, \mathsf{x}))$ est satisfaite. Par 2.2.3, la relation $\mathsf{Preuve}_{\mathsf{T}}$ est récursive, et sa projection est semi-récursive. □

3. L'arithmétique de Robinson

Dans cette dernière section préparatoire, on établit des résultats de prouvabilité à partir du sous-système du système de Peano PA obtenu en omettant l'axiome d'induction mais en ajoutant une définition de l'ordre, appelé *arithmétique de Robinson* et noté $\mathsf{PA}_{\text{faible}}$. On introduit alors une notion de *représentabilité* d'une fonction f dans $\mathsf{PA}_{\text{faible}}$, signifiant essentiellement que $\mathsf{PA}_{\text{faible}}$ prouve que $\mathsf{f}(\vec{\mathsf{n}})$ est la valeur de f en $\vec{\mathsf{n}}$. Le principal résultat technique, établi en 3.3.2, affirme que, en dépit du pouvoir de preuve très limité de $\mathsf{PA}_{\text{faible}}$, toute fonction récursive y est représentable.

▷ *On a vu dans ce qui précède que les fonctions récursives sont simples du point de vue de la complexité puisque, par exemple, elles sont calculables par machine de Turing, c'est-à-dire au moyen d'un programme rudimentaire. Le résultat de représentabilité que l'on va établir ici montre que ces fonctions sont également simples du point de vue de la prouvabilité. Ce résultat est essentiel pour l'obtention des résultats d'impossibilité de la section* 4*, et, en un sens, il constitue le noyau dur de leur démonstration. Outre les prévisibles vérifications plus ou moins automatiques, on a ici à établir un résultat non trivial, à savoir la clôture de la famille des fonctions représentables par définition récursive, où la difficulté vient de la nécessité de pouvoir parler de suites d'entiers à l'intérieur de la structure* $(\mathbb{N}, 0, S, +, \cdot)$*. Comme on a préparé le terrain dans la section* 1*, les choses s'enchaîneront bien, mais il y a là un point fondamentalement délicat.* ◁

5. Le terme « récursivement énumérable » est aussi employé, mais vieilli.

Trois sous-sections dans cette section. On commence dans la sous-section 3.1 avec des résultats généraux sur les modèles de l'arithmétique de Robinson. Ensuite, on montre dans la sous-section 3.2 que toute formule dont toutes les quantifications universelles de la forme prénexe sont bornées (voir 3.2.3) satisfaite dans $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ est prouvable dans $\mathsf{PA}_{\text{faible}}$, et on en déduit dans la sous-section 3.3 que toute fonction récursive totale sur $\mathbb{N}^{\mathsf{p}}$ est représentable dans $\mathsf{PA}_{\text{faible}}$, ce qui signifie essentiellement que $\mathsf{PA}_{\text{faible}}$ prouve la valeur correcte de la fonction aux entiers standards.

3.1. Modèles du système $\mathsf{PA}_{\text{faible}}$

▶ Résumé.— Tout modèle de l'arithmétique de Robinson $\mathsf{PA}_{\text{faible}}$ est extension finale de la structure $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$. ◀

3.1.1.— On étudie dans toute cette section un sous-système faible de l'arithmétique de Peano dans lequel l'axiome d'induction est absent.

Définition (système $\mathsf{PA}_{\text{faible}}$).— On appelle *système de Robinson* le système $\mathsf{PA}_{\text{faible}}$ de $\mathcal{L}_{\text{arith}+}$ formé par les six premiers axiomes du système de Peano (III.1.2.1), auquel on ajoute

$$\forall \mathsf{x}, \mathsf{y}\,(\mathsf{x} \leqslant \mathsf{y} \Leftrightarrow \exists \mathsf{z}\,(\mathsf{z} + \mathsf{x} = \mathsf{y})). \tag{$\mathsf{Def}_{\leqslant}$}$$

Outre la définition $\mathsf{Def}_{\leqslant}$, le système $\mathsf{PA}_{\text{faible}}$ contient donc les axiomes

$$\forall \mathsf{x}\,(\mathsf{x} \neq 0 \Leftrightarrow \exists \mathsf{y}\,(\mathsf{x} = S(\mathsf{y})), \tag{Succ_1}$$
$$\forall \mathsf{x}, \mathsf{y}\,(x \neq y \Rightarrow S(x) \neq S(y)), \tag{Succ_2}$$
$$\forall \mathsf{x}\,(\mathsf{x} + 0 = \mathsf{x}), \tag{Add_1}$$
$$\forall \mathsf{x}, \mathsf{y}\,(\mathsf{x} + S(\mathsf{y}) = S(\mathsf{x} + \mathsf{y})), \tag{Add_2}$$
$$\forall \mathsf{x}\,(\mathsf{x} \cdot 0 = 0), \tag{Mult_1}$$
$$\forall \mathsf{x}, \mathsf{y}\,(\mathsf{x} \cdot S(\mathsf{y}) = \mathsf{x} \cdot \mathsf{y} + \mathsf{x}). \tag{Mult_2}$$

Famille finie de sept formules de la logique $\mathcal{L}_{\text{arith}+}$, le système $\mathsf{PA}_{\text{faible}}$ est équivalent à l'unique formule qui en est la conjonction. Par hypothèse, la structure $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ est un modèle de $\mathsf{PA}_{\text{faible}}$, puisque $\mathsf{Def}_{\leqslant}$ correspond à la construction de l'ordre usuel des entiers.

▷ *Noter que* $\mathsf{PA}_{\text{faible}}$ *ne diffère du système* PA^- *de* VII.3.1.3 *que par l'ajout de l'ordre. La définition de ce dernier utilise l'addition de* z *à gauche : dans le cas de* $(\mathbb{N}, 0, S, +, \cdot)$*, l'addition est commutative, et il est donc indifférent de l'utiliser à gauche ou à droite. Mais le cadre axiomatique de* $\mathsf{PA}_{\text{faible}}$ *est si faible qu'il ne garantit pas la commutativité de l'addition (cf. exercice* 49*), et il n'est donc pas indifférent de référer à une addition à gauche ou à droite. On notera aussi que* $\mathsf{PA}_{\text{faible}}$ *n'est pas formellement un sous-système de* PA_1 *puisqu'il contient un axiome supplémentaire. En fait,* $\mathsf{PA}_{\text{faible}}$ *est un sous-système du système obtenu en ajoutant à* PA_1 *la définition* $\mathsf{Def}_{\leqslant}$*, lequel, comme on a vu en* VII.3.1.5*, est une extension conservative de* PA_1*.* ◁

3.1.2.— Comme la plupart des propriétés d'arithmétique prouvables à partir des axiomes de Peano ont des preuves qui utilisent l'induction, on peut s'attendre à ce que très peu puissent être prouvées à partir de $\mathsf{PA}_{\text{faible}}$. Le

travail va consister à voir qu'un assez grand nombre restent néanmoins accessibles. Comme on l'a vu au chapitre VII, le plus commode pour établir que $\mathsf{PA}_{\text{faible}}$ prouve Φ est la méthode sémantique consistant à établir que Φ est vraie dans tous les modèles de $\mathsf{PA}_{\text{faible}}$. De là, la première tâche est d'étudier (un peu) les modèles de $\mathsf{PA}_{\text{faible}}$. Le principal résultat dans cette direction sera que tout modèle de $\mathsf{PA}_{\text{faible}}$ admet un segment initial qui est une copie de $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$.

3.1.3.— On rappelle que l'on écrit $\mathsf{S}^{\mathsf{n}}\mathsf{0}$ pour $\mathsf{S}(\cdots(\mathsf{S}(\mathsf{0}))\cdots)$, n symboles S. Le but est d'établir que, dans tout modèle de $\mathsf{PA}_{\text{faible}}$, les interprétations des termes $\mathsf{S}^{\mathsf{n}}\mathsf{0}$ forment une copie de $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$. Pour cela, on commence par établir que $\mathsf{PA}_{\text{faible}}$ prouve diverses propriétés des termes $\mathsf{S}^{\mathsf{n}}\mathsf{0}$ mettant en jeu le successeur, l'addition, et la multiplication.

Lemme.— *Le système* $\mathsf{PA}_{\text{faible}}$ *prouve les formules suivantes :*

- $\mathsf{S}^{\mathsf{p}}\mathsf{0} \neq \mathsf{S}^{\mathsf{q}}\mathsf{0}$ *pour* $\mathsf{p} \neq \mathsf{q}$ *;* (#2)
- $\mathsf{S}^{\mathsf{p}}\mathsf{0} + \mathsf{S}^{\mathsf{q}}\mathsf{0} = \mathsf{S}^{\mathsf{r}}\mathsf{0}$ *pour* $\mathsf{p} + \mathsf{q} = \mathsf{r}$ *;* (#3)
- $\mathsf{S}^{\mathsf{p}}\mathsf{0} \cdot \mathsf{S}^{\mathsf{q}}\mathsf{0} = \mathsf{S}^{\mathsf{r}}\mathsf{0}$ *pour* $\mathsf{p} \cdot \mathsf{q} = \mathsf{r}$ *;* (#4)
- $\forall \mathsf{x}, \mathsf{y}\,(\mathsf{x} + \mathsf{y} = \mathsf{0} \Rightarrow (\mathsf{x} = \mathsf{0} \wedge \mathsf{y} = \mathsf{0}))$. (#5)

Démonstration. Soit $\mathcal{M}$ un modèle de $\mathsf{PA}_{\text{faible}}$. Pour (#2), on montre que $\mathsf{p} < \mathsf{q}$ entraîne $(\mathsf{S}^{\mathsf{p}}\mathsf{0})^{\mathcal{M}} \neq (\mathsf{S}^{\mathsf{q}}\mathsf{0})^{\mathcal{M}}$ pour tout q en utilisant une récurrence sur p. On suppose $\mathsf{p} < \mathsf{q}$. Alors, q n'est pas nul, et on pose $\mathsf{q}' = \mathsf{q} - 1$. Pour $\mathsf{p} = 0$, on trouve $(\mathsf{S}^{\mathsf{q}}\mathsf{0})^{\mathcal{M}} = \mathsf{S}^{\mathcal{M}}((\mathsf{S}^{\mathsf{q}'}\mathsf{0})^{\mathcal{M}})$, donc $\mathsf{0}^{\mathcal{M}} \neq (\mathsf{S}^{\mathsf{q}}\mathsf{0})^{\mathcal{M}}$ puisque $\mathcal{M}$ satisfait l'axiome Succ_1. Supposons $\mathsf{p} > 0$. Soit $\mathsf{p}' = \mathsf{p} - 1$. On a alors $(\mathsf{S}^{\mathsf{p}}\mathsf{0})^{\mathcal{M}} = \mathsf{S}^{\mathcal{M}}((\mathsf{S}^{\mathsf{p}'}\mathsf{0})^{\mathcal{M}})$. L'hypothèse de récurrence entraîne $(\mathsf{S}^{\mathsf{p}'}\mathsf{0})^{\mathcal{M}} \neq (\mathsf{S}^{\mathsf{q}'}\mathsf{0})^{\mathcal{M}}$, d'où, en utilisant l'axiome Succ_2, $\mathsf{S}((\mathsf{S}^{\mathsf{p}'}\mathsf{0})^{\mathcal{M}}) \neq \mathsf{S}((\mathsf{S}^{\mathsf{q}'}\mathsf{0})^{\mathcal{M}})$, qui est $(\mathsf{S}^{\mathsf{p}}\mathsf{0})^{\mathcal{M}} \neq (\mathsf{S}^{\mathsf{q}}\mathsf{0})^{\mathcal{M}}$.

Ensuite, on a établi en VII.3.1.4 comme illustration de la méthode sémantique la prouvabilité des formules (#3) à partir de PA^-. Ces formules sont donc *a fortiori* prouvables à partir de $\mathsf{PA}_{\text{faible}}$. Pour (#4), l'argument est similaire, en utilisant les axiomes Mult_1 et Mult_2.

Pour (#5), soient a, b des éléments quelconques du domaine de $\mathcal{M}$. Il s'agit de montrer que $a +^{\mathcal{M}} b = \mathsf{0}^{\mathcal{M}}$ entraîne $a = \mathsf{0}^{\mathcal{M}}$ et $b = \mathsf{0}^{\mathcal{M}}$. Or, par Succ_1, $b \neq \mathsf{0}^{\mathcal{M}}$ entraîne l'existence de b' dans le domaine de $\mathcal{M}$ vérifiant $b = \mathsf{S}^{\mathcal{M}}(b')$. Par Add_2, on a alors $a +^{\mathcal{M}} b = \mathsf{S}^{\mathcal{M}}(a +^{\mathcal{M}} b')$, et, par Succ_1 à nouveau, ce dernier élément n'est pas $\mathsf{0}^{\mathcal{M}}$. Par conséquent, $a +^{\mathcal{M}} b = \mathsf{0}^{\mathcal{M}}$ entraîne $b = \mathsf{0}^{\mathcal{M}}$, et, de là, $a = \mathsf{0}^{\mathcal{M}}$ par l'axiome Add_1. □

3.1.4.— On continue avec des formules mettant l'ordre en jeu.

Lemme.— *Le système* $\mathsf{PA}_{\text{faible}}$ *prouve les formules suivantes :*

- $\mathsf{S}^{\mathsf{p}}\mathsf{0} \leqslant \mathsf{S}^{\mathsf{q}}\mathsf{0}$ *pour* $\mathsf{p} \leqslant \mathsf{q}$ *;* (#6)
- $\forall \mathsf{x}, \mathsf{y}\,(\mathsf{x} \leqslant \mathsf{y} \Rightarrow \mathsf{S}(\mathsf{x}) \leqslant \mathsf{S}(\mathsf{y}))$ *;* (#7)
- $\forall \mathsf{x}\,(\mathsf{x} \leqslant \mathsf{S}^{\mathsf{p}}\mathsf{0} \Leftrightarrow (\mathsf{x} = \mathsf{0} \vee \mathsf{x} = \mathsf{S}^{\mathsf{1}}\mathsf{0} \vee \cdots \vee \mathsf{x} = \mathsf{S}^{\mathsf{p}}\mathsf{0}))$ *;* (#8)
- $\forall \mathsf{x}\,(\mathsf{x} \leqslant \mathsf{S}^{\mathsf{p}}\mathsf{0} \vee \mathsf{S}^{\mathsf{p}}\mathsf{0} \leqslant \mathsf{x})$. (#9)

Démonstration. Soit $\mathcal{M}$ un modèle de $\mathsf{PA}_{\text{faible}}$. Supposons $\mathsf{p} \leqslant \mathsf{q}$. Soit $\mathsf{r} = \mathsf{q} - \mathsf{p}$. On a alors $\mathsf{r} + \mathsf{p} = \mathsf{q}$, donc, par (#3), $\mathsf{PA}_{\text{faible}}$ prouve $\mathsf{S}^{\mathsf{r}}\mathsf{0} + \mathsf{S}^{\mathsf{p}}\mathsf{0} = \mathsf{S}^{\mathsf{q}}\mathsf{0}$, donc *a fortiori* $\exists \mathsf{z}\,(\mathsf{z} + \mathsf{S}^{\mathsf{p}}\mathsf{0} = \mathsf{S}^{\mathsf{q}}\mathsf{0})$, soit $\mathsf{S}^{\mathsf{p}}\mathsf{0} \leqslant \mathsf{S}^{\mathsf{q}}\mathsf{0}$.

Supposons maintenant $\mathcal{M} \models a \leqslant b$, soit $c +^{\mathcal{M}} a = b$. Comme $\mathcal{M}$ satisfait Add_2, on déduit $c +^{\mathcal{M}} \mathsf{S}^{\mathcal{M}}(a) = \mathsf{S}^{\mathcal{M}}(b)$, donc $\mathsf{S}^{\mathcal{M}}(a) \leqslant^{\mathcal{M}} \mathsf{S}^{\mathcal{M}}(b)$.

Pour (#8), un sens, à savoir $\mathsf{x} = \mathsf{S}^{\mathsf{q}}\mathsf{0} \Rightarrow \mathsf{x} \leqslant \mathsf{S}^{\mathsf{p}}\mathsf{0}$ pour $\mathsf{q} \leqslant \mathsf{p}$, résulte directement de (#6). Pour l'autre sens, on utilise une récurrence sur p. Si $\mathcal{M} \models a \leqslant \mathsf{S}^{\mathsf{p}}\mathsf{0}$ est satisfait, il existe c vérifiant $c +^{\mathcal{M}} a = (\mathsf{S}^{\mathsf{p}}\mathsf{0})^{\mathcal{M}}$. Supposons d'abord $\mathsf{p} = 0$. Alors, (#5) entraîne $a = \mathsf{0}^{\mathcal{M}}$, ce qui est (#8). Supposons ensuite $\mathsf{p} > 0$. Soit $\mathsf{p}' := \mathsf{p} - 1$. Si a est $\mathsf{0}^{\mathcal{M}}$, on a fini. Sinon, il existe a' tel que a est $\mathsf{S}^{\mathcal{M}}(a')$. Par Add_2, $c +^{\mathcal{M}} a$ est $\mathsf{S}^{\mathcal{M}}(c +^{\mathcal{M}} a')$, et, par Succ_2, on déduit $c +^{\mathcal{M}} a' = (\mathsf{S}^{\mathsf{p}'}\mathsf{0})^{\mathcal{M}}$, d'où $a' \leqslant (\mathsf{S}^{\mathsf{p}'}\mathsf{0})^{\mathcal{M}}$. L'hypothèse de récurrence implique que a' est $(\mathsf{S}^{\mathsf{i}}\mathsf{0})^{\mathcal{M}}$ pour l'un (au moins) des entiers $\mathsf{0}, \ldots, \mathsf{p}'$, et, de là, que a est $(\mathsf{S}^{\mathsf{i}}\mathsf{0})^{\mathcal{M}}$ pour l'un (au moins) des entiers $1, \ldots, \mathsf{p}$.

Enfin, on montre (#9) par récurrence sur p. Soit a un élément quelconque du domaine de $\mathcal{M}$. Supposons $\mathsf{p} = 0$. Puisque $\mathcal{M}$ satisfait Add_1, on a $a +^{\mathcal{M}} \mathsf{0} = a$, donc $\mathsf{0}^{\mathcal{M}} \leqslant^{\mathcal{M}} a$. Supposons $\mathsf{p} > 0$, et soit $\mathsf{p}' := \mathsf{p} - 1$. Si a est $\mathsf{0}^{\mathcal{M}}$, on obtient $a \leqslant^{\mathcal{M}} (\mathsf{S}^{\mathsf{p}}\mathsf{0})^{\mathcal{M}}$ par (#6). Sinon, par Succ_1, il existe a' tel que a soit $\mathsf{S}^{\mathcal{M}}(a')$. Par hypothèse de récurrence, on a $a' \leqslant^{\mathcal{M}} (\mathsf{S}^{\mathsf{p}'}\mathsf{0})^{\mathcal{M}}$ ou $\mathsf{S}^{\mathsf{p}'}\mathsf{0}^{\mathcal{M}} \leqslant^{\mathcal{M}} a'$. Dans le premier cas, on déduit l'inégalité $a \leqslant^{\mathcal{M}} \mathsf{S}^{\mathsf{p}}\mathsf{0}$ de (#7). Dans le second cas, on déduit de même l'inégalité $(\mathsf{S}^{\mathsf{p}}\mathsf{0})^{\mathcal{M}} \leqslant^{\mathcal{M}} a$. □

▷ *On notera qu'à ce point rien ne permet d'affirmer que* $\mathsf{PA}_{\text{faible}}$ *prouve que* $\leqslant$ *soit une relation d'ordre — et, de fait, il ne le prouve pas : il existe des modèles de* $\mathsf{PA}_{\text{faible}}$ *où l'interprétation de* $\leqslant$ *n'est pas un ordre.* ◁

3.1.5.— Pour décrire la situation, on introduit maintenant les notions de *sous-structure* et d'*extension finale* d'une structure.

Définition (sous-structure, extension, extension finale).—

(i) Pour $\mathcal{S}, \mathcal{S}_\bullet$ structures de type Σ, on dit que $\mathcal{S}_\bullet$ est *sous-structure* de $\mathcal{S}$, ou que $\mathcal{S}$ est *extension* de $\mathcal{S}_\bullet$, si $\mathrm{Dom}(\mathcal{S}_\bullet)$ est inclus dans $\mathrm{Dom}(\mathcal{S})$ et que les opérations et relations de $\mathcal{S}_\bullet$ sont induites par celles de $\mathcal{S}$.

(ii) Si r est un symbole de relation binaire dans Σ, on dit que $\mathcal{S}$ est extension r-*finale* de $\mathcal{S}_\bullet$ si $\mathcal{S}$ est extension de $\mathcal{S}_\bullet$ et si, pour tout a dans $\mathrm{Dom}(\mathcal{S})$ et tout b dans $\mathrm{Dom}(\mathcal{S}_\bullet) \setminus \mathrm{Dom}(\mathcal{S})$, on a $a\, \mathsf{r}^{\mathcal{S}}\, b$.

▷ *Une sous-structure d'un groupe est un sous-groupe, une sous-structure d'un corps est un sous-corps, etc. Noter que la définition d'une extension* r*-finale ne suppose pas que l'interprétation de* r *soit un ordre.* ◁

3.1.6.— Les sous-structures peuvent être caractérisées en termes de clôture.

Lemme.— *Si* $\mathcal{S}$ *est une structure de type* Σ *et si* A *est un sous-ensemble du domaine de* $\mathcal{S}$*, les opérations et relations de* $\mathcal{S}$ *induisent une sous-structure de domaine* A *si, et seulement si,* A *est clos par les opérations de* $\mathcal{S}$*. Dans ce cas,* A *contient* $\mathsf{c}^{\mathcal{S}}$ *pour chaque symbole de constante* c *de* Σ*.*

Démonstration. Supposons que $\mathcal{S}_\bullet$ est une structure de type Σ de domaine A. Pour chaque symbole d'opération s de Σ, l'interprétation de s dans $\mathcal{S}_\bullet$ doit être une opération partout définie. Par définition, cela n'est vérifié par la restriction de $\mathsf{s}^{\mathcal{S}}$ que si, et seulement si, l'ensemble A est clos par $\mathsf{s}^{\mathcal{S}}$. □

3.1.7.— On peut maintenant établir notre résultat principal sur les modèles de $\mathsf{PA}_{\text{faible}}$, à savoir que tout tel modèle est extension $\leqslant$-finale d'une

(copie de) $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$, c'est-à-dire inclut une copie de $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$, éventuellement complétée par des éléments tous postérieurs au sens de $\leqslant^{\mathcal{M}}$.

Proposition (extension finale).— *Si $\mathcal{M}$ est modèle de $\mathsf{PA}_{\text{faible}}$, il existe une sous-structure $\mathcal{M}_\bullet$ de $\mathcal{M}$ dont le domaine est $\{(\mathsf{S}^{\mathsf{n}}\mathsf{0})^{\mathcal{M}} \mid \mathsf{n} \in \mathbb{N}\}$. La structure $\mathcal{M}_\bullet$ est isomorphe à $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$, et $\mathcal{M}$ est extension $\leqslant$-finale de $\mathcal{M}_\bullet$ (voir figure 1).*

Démonstration. Posons $\mathbb{N}_\bullet := \{(\mathsf{S}^{\mathsf{n}}\mathsf{0})^{\mathcal{M}} \mid \mathsf{n} \in \mathbb{N}\}$. D'après 3.1.6, pour montrer que les opérations et relations de $\mathcal{M}$ induisent une structure bien définie $\mathcal{M}_\bullet$ de domaine $\mathbb{N}_\bullet$, il suffit de vérifier que $\mathbb{N}_\bullet$ est clos par toutes les opérations de $\Sigma_{\text{arith}+}$, donc de Σ_{arith}. C'est le cas pour le symbole de constante $\mathsf{0}$, puisque $\mathsf{0}^{\mathcal{M}}$ est dans $\mathbb{N}_\bullet$ par hypothèse ; c'est le cas pour S, puisque, par construction, on a $\mathsf{S}(\mathsf{S}^{\mathsf{p}}\mathsf{0}) = \mathsf{S}^{\mathsf{p}+1}\mathsf{0}$, donc $\mathsf{S}^{\mathcal{M}}((\mathsf{S}^{\mathsf{p}}\mathsf{0})^{\mathcal{M}}) = (\mathsf{S}^{\mathsf{p}+1}\mathsf{0})^{\mathcal{M}}$; c'est le cas pour $+$ et $\cdot$, puisque, par (#3) et (#4), on a $(\mathsf{S}^{\mathsf{p}}\mathsf{0})^{\mathcal{M}} +^{\mathcal{M}} (\mathsf{S}^{\mathsf{q}}\mathsf{0})^{\mathcal{M}} = (\mathsf{S}^{\mathsf{p}+\mathsf{q}}\mathsf{0})^{\mathcal{M}}$ et $(\mathsf{S}^{\mathsf{p}}\mathsf{0})^{\mathcal{M}} \cdot^{\mathcal{M}} (\mathsf{S}^{\mathsf{q}}\mathsf{0})^{\mathcal{M}} = (\mathsf{S}^{\mathsf{p}\mathsf{q}}\mathsf{0})^{\mathcal{M}}$.

Ensuite, l'application $f : \mathsf{n} \mapsto (\mathsf{S}^{\mathsf{n}}\mathsf{0})^{\mathcal{M}}$ est une surjection de $\mathbb{N}$ sur $\mathbb{N}_\bullet$, et elle est injective en vertu de (#2). Par conséquent, f est une bijection. Par construction, on a $f(0) = \mathsf{0}^{\mathcal{M}}$, et $f(S(\mathsf{n})) = \mathsf{S}^{\mathcal{M}}(f(\mathsf{n}))$. Puis, par (#3) et (#4) à nouveau, on a $f(\mathsf{p} + \mathsf{q}) = f(\mathsf{p}) +^{\mathcal{M}} f(\mathsf{q})$ et $f(\mathsf{p} \cdot \mathsf{q}) = f(\mathsf{p}) \cdot^{\mathcal{M}} f(\mathsf{q})$. Enfin, par (#6), $\mathsf{p} \leqslant \mathsf{q}$ équivaut à $f(\mathsf{p}) \leqslant^{\mathcal{M}} f(\mathsf{q})$. La bijection f établit donc un isomorphisme entre les structures $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ et $\mathcal{M}_\bullet$.

Enfin soit a un élément de $\mathbb{N}_\bullet$, disons $a = (\mathsf{S}^{\mathsf{n}}\mathsf{0})^{\mathcal{M}}$, et b un élément quelconque du domaine de $\mathcal{M}$. Par (#9), on a $a \leqslant^{\mathcal{M}} b$ ou $b \leqslant^{\mathcal{M}} a$. Alors, par (#8), $b \leqslant^{\mathcal{M}} a$ entraîne qu'il existe un entier $\mathsf{p} \leqslant \mathsf{n}$ tel que b soit $(\mathsf{S}^{\mathsf{p}}\mathsf{0})^{\mathcal{M}}$, donc appartienne à $\mathbb{N}_\bullet$: c'est dire que $\mathcal{M}$ est extension $\leqslant$-finale de $\mathcal{M}_\bullet$. □

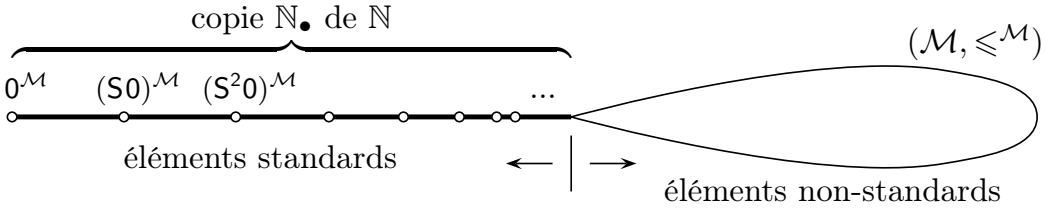


FIGURE 1.— Modèle de $\mathsf{PA}_{\text{faible}}$: une copie de $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ suivie d'éventuels éléments non-standards ; à la différence du cas particulier des modèles de l'arithmétique de la figure VII.2, la relation $\leqslant^{\mathcal{M}}$ n'a aucune raison en général d'être un ordre sur les éléments non-standards.

▷ *Le résultat ci-dessus étend celui du* §VII.3.3.6. *Il s'applique à tous les modèles de l'arithmétique puisque ceux-ci sont modèles de* $\mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot, \leqslant)$, *donc, par hypothèse, de* $\mathsf{PA}_{\text{faible}}$. *En revanche, comme* $\mathsf{PA}_{\text{faible}}$ *est beaucoup plus faible que* PA_1 *et,* a fortiori, *que* $\mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ *— au moins de façon conjecturale pour le moment — il doit exister des modèles de* $\mathsf{PA}_{\text{faible}}$ *non modèles de* PA_1, *et des modèles de* PA_1 *non modèles de l'arithmétique, c'est-à-dire qui ne sont pas élémentairement équivalents à* $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$. *Le résultat ci-dessus montre que, aussi exotiques soient les modèles de* $\mathsf{PA}_{\text{faible}}$, *néanmoins ils commencent tous par une copie des entiers.* ◁

3.2. Absoluité des formules Σ_1

▶ Résumé.— Toute formule Σ_1 satisfaite dans $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ est prouvable à partir de $\mathsf{PA}_{\text{faible}}$. ◀

3.2.1.— Notre but dans cette section est d'établir un résultat général de prouvabilité à partir de $\mathsf{PA}_{\text{faible}}$, à savoir que toutes les formules suffisamment simples, en l'occurrence celles dont les seules quantifications non bornées de la forme prénexe sont existentielles, sont automatiquement prouvables dans $\mathsf{PA}_{\text{faible}}$ dès qu'elles sont satisfaites dans $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$.

▷ *Deux modèles quelconques de* $\mathsf{PA}_{\text{faible}}$ *n'ont aucune raison de satisfaire les mêmes formules du premier ordre : par exemple, on a mentionné qu'il existe des modèles de* $\mathsf{PA}_{\text{faible}}$ *où l'addition n'est pas commutative, ce qui signifie qu'il existe des modèles de* $\mathsf{PA}_{\text{faible}}$ *où la formule* $\forall \mathsf{x}, \mathsf{y}\,(\mathsf{x} + \mathsf{y} = \mathsf{y} + \mathsf{x})$ *est satisfaite, comme* $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$*, et d'autres où elle est fausse (cf. exercice* 49*). On va montrer ici que ce type de situation ne peut se produire pour certaines formules syntaxiquement simples, dites* de complexité Σ_1*. La raison est que les formules de complexité* Σ_1 *vraies dans une structure* $\mathcal{M}_\bullet$ *restent automatiquement vraies dans toute extension finale de* $\mathcal{M}_\bullet$ *: comme tout modèle de* $\mathsf{PA}_{\text{faible}}$ *est extension finale de* $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$*, les formules de complexité* Σ_1 *vraies dans* $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ *doivent être vraies dans tout modèle de* $\mathsf{PA}_{\text{faible}}$ *et donc, par complétude, prouvables à partir de* $\mathsf{PA}_{\text{faible}}$*.* ◁

3.2.2.— On commence avec le cas, trivial, d'une extension quelconque et des formules sans quantificateur.

Lemme.— *Si* $\mathcal{S}$ *est une extension de* $\mathcal{S}_\bullet$*, alors, pour toute formule* $\Phi(\vec{\mathsf{x}})$ *sans quantificateur et tous* $\vec{a}$ *dans le domaine de* $\mathcal{S}_\bullet$*, la relation* $\mathcal{S}_\bullet \models \Phi(\vec{a})$ *équivaut à* $\mathcal{S} \models \Phi(\vec{a})$*. En particulier,* $\mathcal{S}_\bullet$ *et* $\mathcal{S}$ *satisfont les mêmes formules closes sans quantificateur.*

Démonstration. Une induction montre d'abord $\mathsf{t}(\vec{a})^{\mathcal{S}_\bullet} = \mathsf{t}(\vec{a})^{\mathcal{S}}$ pour tout terme $\mathsf{t}(\vec{\mathsf{x}})$ et tous $\vec{a}$ dans le domaine de $\mathcal{S}_\bullet$. On en déduit l'équivalence souhaitée pour les formules atomiques, puis, inductivement, pour toutes les formules sans quantificateur, c'est-à-dire pour les combinaisons booléennes de formules atomiques. □

3.2.3.— On étend le résultat précédent au cas, plus intéressant, des extensions finales et des formules dites à quantifications bornées. On rappelle que, suivant la convention VII.1.1.9, $\exists \mathsf{x} \leqslant \mathsf{y}\,(\ldots)$ signifie $\exists \mathsf{x}\,(\mathsf{x} \leqslant \mathsf{y} \wedge \ldots)$, et $\forall \mathsf{x} \leqslant \mathsf{y}\,(\ldots)$ signifie $\forall \mathsf{x}\,(\mathsf{x} \leqslant \mathsf{y} \Rightarrow \ldots)$.

Définition (formules Δ_0 et Σ_1).— On dit qu'une formule Φ de $\mathcal{L}_{\text{arith}+}$ est *de complexité* Δ_0 — ou simplement est Δ_0 — si les seules quantifications figurant dans Φ sont des quantifications bornées $\exists \mathsf{x} \leqslant \mathsf{t}$ ou $\forall \mathsf{x} \leqslant \mathsf{t}$ avec t terme sans occurrence de x. On dit que Φ est *de complexité* Σ_1 si les seules quantifications universelles figurant dans Φ sont des quantifications bornées $\forall \mathsf{x} \leqslant \mathsf{t}$ et si, dans l'arbre associé à Φ, il n'y a aucun symbole $\neg$, $\Rightarrow$, $\Leftrightarrow$ au-dessus d'une quantification existentielle.

▷ *De façon équivalente, la famille des formules de complexité Σ_1 est la plus petite famille contenant les formules sans quantificateur et close par conjonction, disjonction, quantification universelle bornée, et quantification existentielle. En particulier, toute formule du type $\exists \mathsf{x}_1, \ldots, \mathsf{x}_k\, (\Psi)$ avec Ψ de complexité Δ_0 est de complexité Σ_1.* ◁

3.2.4.— La notion de formule Σ_1 se simplifie pour les formules prénexes.

Définition (prénexe).— Une formule du premier ordre est dite *prénexe* si tous les quantificateurs précèdent tous les autres symboles.

Il est facile de vérifier que les axiomes de $\mathcal{L}_\Sigma$ impliquent que toute formule de $\mathcal{L}_\Sigma$ est prouvablement équivalente à une formule prénexe. Par définition, une formule prénexe Φ est alors Σ_1 si, et seulement si, toutes les quantifications universelles de Φ sont des quantifications bornées $\forall \mathsf{x} \leqslant \mathsf{t}$.

▷ *L'équivalence précédente n'est pas valable pour des formules générales puisque, par exemple, la suite de symboles $\neg\exists \mathsf{x}$ (interdite dans une formule prénexe) cache une quantification universelle.* ◁

3.2.5.— On parle d'*absoluité* pour exprimer que la valeur « vrai/faux » d'une formule ne change pas entre une structure et une autre. Le résultat suivant exprime que les formules closes Δ_0 sont absolues et les formules closes Σ_1 semi-absolues vers le haut vis-à-vis des extensions finales.

Proposition (absoluité).— *Supposons $\mathcal{S}$ extension $\leqslant$-finale de $\mathcal{S}_\bullet$.*

(i) *Pour toute formule $\Phi(\vec{\mathsf{x}})$ de complexité Δ_0 et tous $\vec{a}$ dans le domaine de $\mathcal{S}_\bullet$, la relation $\mathcal{S}_\bullet \models \Phi(\vec{a})$ équivaut à $\mathcal{S} \models \Phi(\vec{a})$. En particulier, $\mathcal{S}_\bullet$ et $\mathcal{S}$ satisfont les mêmes formules closes de complexité Δ_0.*

(ii) *Pour toute formule $\Phi(\vec{\mathsf{x}})$ de complexité Σ_1 et tous $\vec{a}$ dans le domaine de $\mathcal{S}_\bullet$, la relation $\mathcal{S}_\bullet \models \Phi(\vec{a})$ entraîne $\mathcal{S} \models \Phi(\vec{a})$. En particulier, toute formule close de complexité Σ_1 vraie dans $\mathcal{S}_\bullet$ est vraie dans $\mathcal{S}$.*

Démonstration. (i) Soit $S := \mathrm{Dom}(\mathcal{S})$ et $S_\bullet := \mathrm{Dom}(\mathcal{S}_\bullet)$. On montre l'équivalence par induction sur Φ. Par 3.2.2, l'équivalence est vraie pour les formules atomiques puisque $\mathcal{S}$ est extension de $\mathcal{S}_\bullet$, et elle est préservée par négation, conjonction, disjonction, implication et équivalence. Il reste à voir que l'équivalence est aussi préservée par quantification bornée. On considère le cas d'une quantification universelle. Supposons que $\Phi(\vec{\mathsf{x}})$ est $\forall \mathsf{y} \leqslant \mathsf{t}(\vec{\mathsf{x}})\, (\Psi(\vec{\mathsf{x}}, \mathsf{y}))$ et que $\vec{a}$ est une suite d'éléments de $S_\bullet$. On cherche le lien entre la satisfaction de $\forall \mathsf{y} \leqslant \mathsf{t}\,(\vec{a})\, (\Psi(\vec{a}, \mathsf{y}))$ dans $\mathcal{S}_\bullet$ et dans $\mathcal{S}$. Or, soit b un élément de S vérifiant $b \leqslant^{\mathcal{S}} \mathsf{t}(\vec{a})^{\mathcal{S}}$. Comme $\mathcal{S}$ est extension de $\mathcal{S}_\bullet$, on a $\mathsf{t}(\vec{a})^{\mathcal{S}} = \mathsf{t}(\vec{a})^{\mathcal{S}_\bullet}$, et, comme $\mathcal{S}$ est extension $\leqslant$-finale de $\mathcal{S}_\bullet$ et que $\mathsf{t}(\vec{a})^{\mathcal{S}_\bullet}$ est dans $S_\bullet$, la relation $b \leqslant^{\mathcal{S}} \mathsf{t}(\vec{a})^{\mathcal{S}}$ entraîne $b \in S_\bullet$. Par conséquent, les éléments vérifiant $b \leqslant \mathsf{t}(\vec{a})$ dans $\mathcal{S}_\bullet$ et $\mathcal{S}$ coïncident. Par hypothèse d'induction, pour tout tel élément b, les relations $\mathcal{S}_\bullet \models \Psi(\vec{a}, b)$ et $\mathcal{S} \models \Psi(\vec{a}, b)$ sont équivalentes, et, par conséquent, il en est de même de $\mathcal{S}_\bullet \models \Phi(\vec{a})$ et $\mathcal{S} \models \Phi(\vec{a})$. Le cas du quantificateur $\exists$ est similaire.

(ii) On procède de même. Sans perte de généralité, on suppose Φ prénexe. Par (i), l'implication est vraie pour toutes les formules sans quantificateur. Le passage à

la conjonction et à la disjonction est facile. L'argument pour les quantifications universelles bornées est le même que ci-dessus. Enfin, pour une quantification existentielle, supposons que $\Phi(\vec{x})$ est $\exists y\,(\Psi(\vec{x},y))$ avec $\Psi(\vec{x},y)$ de complexité Σ_1. Supposons $\mathcal{S}_\bullet \models \Phi(\vec{a})$. On a donc $\mathcal{S}_\bullet \models \Psi(\vec{a},b)$ pour un certain b dans $S_\bullet$. Par hypothèse d'induction, on déduit $\mathcal{S} \models \Psi(\vec{a},b)$, d'où $\mathcal{S} \models \exists y\,(\Psi(\vec{a},y))$, et par conséquent $\mathcal{S} \models \Phi(\vec{a})$. □

▷ *Noter que, dans* (ii)*, il n'y a en général aucune raison que l'implication réciproque soit vraie : si $\mathcal{S}$ satisfait $\Phi(\vec{a})$, il existe un élément b dans S vérifiant $\Psi(\vec{a},b)$, mais, faute de borne sur la quantification, rien ne permet d'affirmer que cet élément appartient à $S_\bullet$.* ◁

3.2.6.— On déduit de ce qui précède un résultat de complétude faible affirmant que toute formule suffisamment simple vraie dans les entiers est prouvable dans le système de Robinson. On verra plus loin avec le premier théorème d'incomplétude de Gödel que ce résultat ne s'étend pas à des formules plus compliquées.

Proposition (prouvabilité).— *Toute formule arithmétique close de complexité Σ_1 satisfaite dans $(\mathbb{N},0,S,+,\cdot,\leqslant)$ est prouvable à partir de $\mathsf{PA}_{\text{faible}}$.*

Démonstration. Supposons que Φ est une formule close de complexité Σ_1 de $\mathcal{L}_{\text{arith}+}$ satisfaite dans la structure $(\mathbb{N},0,S,+,\cdot,\leqslant)$. Par 3.2.5, Φ est satisfaite dans toute extension finale de $(\mathbb{N},0,S,+,\cdot,\leqslant)$, donc, par 3.1.7, dans tout modèle de $\mathsf{PA}_{\text{faible}}$. Par le théorème de complétude (VII.3.1.2), il en résulte que Φ est prouvable à partir de $\mathsf{PA}_{\text{faible}}$. □

3.2.7.— On en déduit un résultat de complétude du système $\mathsf{PA}_{\text{faible}}$ vis-à-vis des formules Δ_0, c'est-à-dire des formules à quantifications bornées.

Corollaire (Δ_0-complétude).— *Si Φ est une formule arithmétique close de complexité Δ_0, alors l'une au moins des formules Φ ou $\neg\Phi$ est prouvable à partir de $\mathsf{PA}_{\text{faible}}$.*

Démonstration. Si Φ est Δ_0, alors à la fois Φ et $\neg\Phi$ sont Δ_0, donc *a fortiori* Σ_1, et l'une des deux est satisfaite dans la structure $(\mathbb{N},0,S,+,\cdot,\leqslant)$. On applique alors à cette dernière le résultat de 3.2.6. □

3.3. Représentabilité

▶ Résumé.— Toute fonction récursive totale et toute relation récursive sur $\mathbb{N}^p$ est Σ_1-représentable dans $\mathsf{PA}_{\text{faible}}$. ◀

3.3.1.— On peut maintenant revenir aux fonctions récursives et étudier leurs relations avec le système $\mathsf{PA}_{\text{faible}}$. Il s'agit de montrer que, si f est une fonction récursive, alors $\mathsf{PA}_{\text{faible}}$ calcule en un certain sens la valeur de f.

▷ *On a déjà rencontré la notion de fonction définissable : si $\mathcal{S}$ est une structure de domaine S, une fonction f de S^p dans S est dite définissable dans $\mathcal{S}$ s'il*

existe une formule $\Phi(\vec{x}, y)$ *telle que* $b = f(\vec{a})$ *est vrai dans* $\mathcal{S}$ *si et seulement* $\mathcal{S}$ *satisfait* $\Phi(\vec{a}, b)$. *De fait, on peut montrer que toute fonction récursive sur* $\mathbb{N}^p$ *est définissable dans la structure* $(\mathbb{N}, 0, S, +, \cdot)$ (*exercice* 54). *Mais ce n'est pas cette notion qui est adaptée ici, puisque ce ne sont pas des résultats relatifs à la satisfaction dans une structure particulière qui sont visés, mais des résultats de prouvabilité à partir de* $\mathsf{PA}_{\text{faible}}$. *Le problème est qu'alors parler des entiers ne fait pas sens, puisque la théorie* $\mathsf{PA}_{\text{faible}}$ *ne connaît que leur contrepartie formelle, à savoir les termes* $S^n 0$. *De ce fait, ce à quoi on s'intéresse pour une fonction* f *de* $\mathbb{N}^p$ *dans* $\mathbb{N}$ *est l'existence d'une formule* $\Phi(\vec{x}, y)$ *telle que, quand on a* $f(n_1, \dots, n_p) = n$, *alors* $\mathsf{PA}_{\text{faible}}$ *prouve* $\Phi(S^{n_1} 0, \dots, S^{n_p} 0, S^n 0)$. *De plus, comme un modèle de* $\mathsf{PA}_{\text{faible}}$ *peut contenir bien d'autres éléments que les interprétations des termes* $S^n 0$, *on demande que* $\mathsf{PA}_{\text{faible}}$ *prouve la formule* $\forall y{\neq}S^n 0\,(\neg\Phi(S^{n_1} 0, \dots, S^{n_p} 0, y))$. *Noter que cette condition, qui affirme le caractère fonctionnel de* Φ *aux valeurs standards des arguments, est plus faible que l'affirmation que* Φ *est fonctionnelle partout.* ◁

Définition (représentable).— Soient T une théorie de $\mathcal{L}_\Sigma$ avec $\Sigma \supseteq \Sigma_{\text{arith}}$ et f une fonction partout définie de $\mathbb{N}^p$ dans $\mathbb{N}$. Une formule $\Phi(x_1, \dots, x_p, y)$ de $\mathcal{L}_\Sigma$ est dite *représenter* f dans T si, pour tous $n_1, \dots, n_p, n$ dans $\mathbb{N}$ vérifiant $f(n_1, \dots, n_p) = n$,

- T prouve $\Phi(S^{n_1} 0, \dots, S^{n_p} 0, S^n 0)$, et
- T prouve $\forall y{\neq}S^n 0\,(\neg\Phi(S^{n_1} 0, \dots, S^{n_p} 0, y))$.

3.3.2.— Notre but est maintenant d'établir le résultat suivant.

Proposition (représentabilité).— *Toute fonction récursive totale peut être représentée dans* $\mathsf{PA}_{\text{faible}}$ *par une formule de complexité* Σ_1.

▷ *À un point spécifique près, la démonstration de ce résultat n'est pas difficile : elle consiste naturellement à vérifier que les fonctions récursives de base sont représentables, puis que l'ensemble des fonctions représentables est clos par les opérations de composition, récursion, et minimisation.* ◁

3.3.3.— Dans toute la suite, on écrit « Σ_1*-représentable* » pour « représentable par une formule de complexité Σ_1 ». Par ailleurs, dans tout modèle $\mathcal{M}$ de $\mathsf{PA}_{\text{faible}}$, les éléments de la forme $(S^n 0)^{\mathcal{M}}$ seront dits *standards* : ce sont eux qui correspondent aux entiers naturels, par opposition aux éventuels éléments dits non-standards qui ne correspondent à aucun entier naturel. Le premier résultat n'utilise même pas les axiomes de $\mathsf{PA}_{\text{faible}}$.

Lemme.— *Les fonctions* zero, succ *et* $\mathsf{proj}_{p,i}$ *pour chaque* p, i *sont* Σ_1*-représentables dans* $\mathsf{PA}_{\text{faible}}$.

Démonstration. Soit $\Phi(y)$ la formule $y = 0$. La formule $\Phi(0)$, c'est-à-dire $0 = 0$, est alors valide, donc prouvable à partir des seuls axiomes de $\mathcal{L}_{\text{arith}+}$. Il en est de même de la formule $\forall y{\neq}0\,(y \neq 0)$. Par définition, la formule $\Phi(y)$ représente donc dans $\mathsf{PA}_{\text{faible}}$ la fonction (à zéro argument) zero. De même, $y = S(x)$ représente dans $\mathsf{PA}_{\text{faible}}$ la fonction succ, et $y = x_i$ représente $\mathsf{proj}_{p,i}$. Les formules précédentes sont sans quantificateur, donc certainement de complexité Σ_1. □

3.3.4.— Il s'agit maintenant de montrer que la famille des fonctions Σ_1-représentables est close par les opérations définissant les fonctions récursives. On commence par la composition.

Lemme.— *Les fonctions Σ_1-représentables dans $\mathsf{PA}_{\text{faible}}$ sont closes par composition.*

Démonstration. Supposons $\mathsf{f} = \mathsf{comp}(\mathsf{g}, \mathsf{f}_1, \dots, \mathsf{f}_q)$. Supposons que Φ_i représente f_i pour $1 \leqslant i \leqslant q$, et que Ψ représente g. Soit $\Phi(x_1, \dots, x_p, y)$ la formule

$$\exists y_1, \dots, y_q \, (\Phi_1(x_1, \dots, x_p, y_1) \wedge \dots \wedge \Phi_q(x_1, \dots, x_p, y_q) \wedge \Psi(y_1, \dots, y_q, y)), \qquad (\#10)$$

et soient $n_1, \dots, n_p$ des entiers. Posons $m_i := f_i(n_1, \dots, n_p)$ et $n := g(m_1, \dots, m_q)$. Soit $\mathcal{M}$ un modèle de $\mathsf{PA}_{\text{faible}}$. Pour chaque i, puisque Φ_i représente g_i, il existe exactement un élément b_i du domaine de $\mathcal{M}$ vérifiant $\Phi_i(S^{n_1}0, \dots, S^{n_p}0, b_i)$, à savoir $(S^{m_i}0)^{\mathcal{M}}$. Ensuite, puisque Ψ représente g, il existe exactement un élément a du domaine de $\mathcal{M}$ vérifiant $\Phi(S^{m_1}0, \dots, S^{m_q}0, a)$, à savoir $(S^n0)^{\mathcal{M}}$. De là, $\mathcal{M}$ satisfait les formules $\Phi(S^{m_1}0, \dots, S^{m_q}0, S^n0)$ et $\forall y{\neq}S^n0 \, (\neg\Phi(S^{m_1}0, \dots, S^{m_q}0, y))$. Par le théorème de complétude, on déduit que $\mathsf{PA}_{\text{faible}}$ prouve $\Phi(S^{m_1}0, \dots, S^{m_q}0, S^n0)$ et $\forall y{\neq}S^n0 \, (\neg\Phi(S^{m_1}0, \dots, S^{m_q}0, y))$, c'est-à-dire que Φ représente f dans $\mathsf{PA}_{\text{faible}}$. Par ailleurs, si chacune des formules $\Phi_1, \dots, \Phi_q$, Ψ est de complexité Σ_1, il en est de même de Φ. □

3.3.5.— Avant de considérer les définitions par récursion, partie délicate de la construction, on considère les définitions par minimisation. On sait que, même si g est une fonction totale, la fonction $\mathsf{minim}(\mathsf{g})$ n'est pas nécessairement totale, puisque, pour un choix $\vec{n}$ donné, il n'existe pas nécessairement d'entier k pour lequel on ait $\mathsf{g}(\vec{n}, \mathsf{k}) = 0$. On dira ici que f est définie par *minimisation totale* à partir de g si f est définie par minimisation à partir de g, et que f et g sont totales (c'est-à-dire partout définies).

Lemme.— *La famille des fonctions récursives totales est la clôture des fonctions de base par composition, récursion, et minimisation totale.*

Démonstration. Par définition, toute fonction obtenue à partir des fonctions de base par composition, récursion, et minimisation totale est une fonction récursive totale. Le problème est qu'inversement, il se pourrait que, dans la définition d'une fonction récursive totale, on utilise à des étapes intermédiaires des fonctions partielles. La question est donc de montrer que l'on peut l'éviter, c'est-à-dire que toute fonction récursive totale peut être construite en n'utilisant que des minimisations totales : cela résulte de 1.3.10, où est affirmé précisément que l'on peut toujours se contenter d'une unique minimisation finale. □

3.3.6.— On peut alors facilement conclure pour la minimisation.

Lemme.— *Les fonctions Σ_1-représentables dans $\mathsf{PA}_{\text{faible}}$ sont closes par minimalisation totale.*

Démonstration. Supposons que f est obtenue par minimisation totale à partir de g, et que $\Psi(\vec{x}, y, z)$ est une formule de complexité Σ_1 représentant g dans $\mathsf{PA}_{\text{faible}}$. Soit $\Phi(\vec{x}, y)$ la formule

$$\Psi(\vec{x}, y, 0) \wedge \forall y'{<}y \, \exists z{\neq}0 \, (\Psi(\vec{x}, y', z)). \qquad (\#11)$$

La formule Φ est elle aussi Σ_1 puisqu'obtenue à partir d'instances de Ψ par conjonction, quantification universelle bornée et quantification existentielle. On va mon-

trer que Φ représente f dans $\mathsf{PA}_{\text{faible}}$. Soit $\mathcal{M}$ un modèle de $\mathsf{PA}_{\text{faible}}$, et soient $\vec{n}$ des entiers naturels. Soit $\mathsf{m} := \mathsf{f}(\vec{n})$, qui existe puisque, par hypothèse, f est totale, et, pour chaque $\mathsf{k} < \mathsf{m}$, soit $\mathsf{m_k} := \mathsf{g}(\vec{n}, \mathsf{k})$. Par hypothèse, on a $\mathsf{g}(\vec{n}, \mathsf{m}) = 0$ et $\mathsf{g}(\vec{n}, \mathsf{k}) = \mathsf{m_k} \neq 0$ pour $\mathsf{k} < \mathsf{m}$. Puisque Ψ représente g, on a $\mathcal{M} \models \Psi(\overrightarrow{\mathsf{S^n0}}, \mathsf{S^m0}, 0)$. Par ailleurs, par (#8), $\mathcal{M} \models b < \mathsf{S^m0}$ entraîne $b = 0$ ou $b = \mathsf{S0}$ ou ... ou $b = \mathsf{S^{m-1}0}$, d'où, comme $\mathcal{M}$ satisfait $\Psi(\overrightarrow{\mathsf{S^n0}}, \mathsf{S^k0}, \mathsf{S^{m_k}0})$,

$\mathcal{M} \models \forall \mathsf{y'}{<}\mathsf{S^m0}\, \exists \mathsf{z}{\neq}0\, (\Psi(\overrightarrow{\mathsf{S^n0}}, \mathsf{y'}, \mathsf{z}))$, et donc $\mathcal{M} \models \Phi(\overrightarrow{\mathsf{S^n0}}, \mathsf{S^m0})$.

Par ailleurs, supposons $\mathcal{M} \models \Phi(\overrightarrow{\mathsf{S^n0}}, b)$. *A priori*, on ne sait pas que b doit être standard. Mais, par (#9), on a $\mathcal{M} \models b \leqslant \mathsf{S^m0} \vee \mathsf{S^m0} < b$. Or, $\mathcal{M} \models \mathsf{S^m0} < b$ est impossible : en effet, par définition, on aurait alors $\mathcal{M} \models \exists \mathsf{z}{\neq}0\, (\Psi(\overrightarrow{\mathsf{S^n0}}, \mathsf{S^m0}, \mathsf{z}))$, alors que l'hypothèse que Ψ représente g et que l'on a $\mathsf{g}(\vec{n}, \mathsf{m}) = 0$ implique

$\mathcal{M} \models \forall \mathsf{z}{\neq}0\, (\neg\Psi(\overrightarrow{\mathsf{S^n0}}, \mathsf{S^m0}, 0))$.

On a donc $\mathcal{M} \models b \leqslant \mathsf{S^m0}$. De là, b est standard, et, alors, il est clair que la seule valeur possible est $b = (\mathsf{S^m0})^{\mathcal{M}}$. Par conséquent, Φ représente f dans $\mathsf{PA}_{\text{faible}}$. □

3.3.7.— On passe à la clôture de la famille des fonctions représentables dans $\mathsf{PA}_{\text{faible}}$ vis-à-vis des définitions par récursion. La construction est délicate, parce qu'il faut être capable de parler d'une suite de valeurs de longueur quelconque. C'est ici que l'on utilise la fonction beta de 1.2.2, qui permet de passer d'une quantification du type « il existe une suite d'entiers telle que ... » à une unique quantification « il existe un entier tel que ... ».

Lemme.— *La fonction* beta *est* Σ_1*-représentable dans* $\mathsf{PA}_{\text{faible}}$ *par une formule* $\mathsf{B}(\mathsf{x_1}, \mathsf{x_2}, \mathsf{x_3}, \mathsf{y})$ *pour laquelle* $\mathsf{PA}_{\text{faible}}$ *prouve*

$$\mathsf{B}(\mathsf{x_1}, \mathsf{x_2}, \mathsf{x_3}, \mathsf{S^m0}) \Rightarrow \forall \mathsf{y}{\neq}\mathsf{S^m0}\, (\neg\mathsf{B}(\mathsf{x_1}, \mathsf{x_2}, \mathsf{x_3}, \mathsf{y})). \qquad (\#12)$$

Démonstration. On utilise $\mathsf{x} \neq \mathsf{y}$ comme raccourci pour $\neg(\mathsf{x} = \mathsf{y})$, et $\mathsf{x} < \mathsf{y}$ comme un raccourci pour $\mathsf{x} \leqslant \mathsf{y} \wedge \mathsf{x} \neq \mathsf{y}$. Soit $\Phi(\mathsf{x_1}, \mathsf{x_2}, \mathsf{x_3})$ la formule Δ_0

$(\mathsf{x_2} = 0 \wedge \mathsf{x_3} = 0) \vee (\mathsf{x_2} \neq 0 \wedge \exists \mathsf{x_4}{\leqslant}\mathsf{x_1}\, (\mathsf{x_1} = \mathsf{x_2} \cdot \mathsf{x_4} + \mathsf{x_3} \wedge \mathsf{x_3} < \mathsf{x_2}))$.

Alors, Φ représente la fonction « reste de la division euclidienne ». En effet, si n et p sont des entiers et si r est le reste de la division euclidienne de n par p, alors, en notant q le quotient, on a $\mathsf{n} = \mathsf{p} \cdot \mathsf{q} + \mathsf{r}$ et $\mathsf{r} \leqslant \mathsf{p}$ et $\mathsf{r} \neq \mathsf{p}$, donc $\mathsf{PA}_{\text{faible}}$ prouve les formules $\mathsf{S^n0} = \mathsf{S^p0} \cdot \mathsf{S^q0} + \mathsf{S^r0}$, et $\mathsf{S^r0} \leqslant \mathsf{S^p0}$ et $\mathsf{S^r0} \neq \mathsf{S^p0}$, donc $\Phi(\mathsf{S^n0}, \mathsf{S^p0}, \mathsf{S^r0})$. Par ailleurs, soit $\mathcal{M}$ un modèle quelconque de $\mathsf{PA}_{\text{faible}}$. Supposons $\mathcal{M} \models \Phi(\mathsf{S^n0}, \mathsf{S^p0}, a)$ avec $\mathsf{p} \geqslant 1$. Il existe alors b dans le domaine de $\mathcal{M}$ tel que $\mathcal{M}$ satisfait à la fois

$b \leqslant \mathsf{S^n0}$, $\mathsf{S^n0} = \mathsf{S^p0} \cdot b + a$, $a \leqslant \mathsf{S^p0}$, et $a \neq \mathsf{S^p0}$.

Si a et b sont standards, les seules valeurs possibles sont $a = (\mathsf{S^r0})^{\mathcal{M}}$ et $b = (\mathsf{S^q0})^{\mathcal{M}}$. Or, par (#8), $\mathcal{M} \models a \leqslant \mathsf{S^p0}$ entraîne que a est standard, et $\mathcal{M} \models b \leqslant \mathsf{S^n0}$ entraîne que b est standard. On a donc $a = (\mathsf{S^r0})^{\mathcal{M}}$ et $b = (\mathsf{S^q0})^{\mathcal{M}}$.

Soit $\Psi(\mathsf{x_1}, \mathsf{x_2}, \mathsf{x_3}, \mathsf{y})$ la formule $\Phi(\mathsf{x_1}, \mathsf{S}(\mathsf{S}(\mathsf{x_3}) \cdot \mathsf{x_2}), \mathsf{y})$. Alors, Ψ est une formule Δ_0, tout comme Φ, et le fait que Φ représente la fonction « reste » implique que Ψ représente beta.

Soit alors $\mathsf{B}(\mathsf{x_1}, \mathsf{x_2}, \mathsf{x_3}, \mathsf{y})$ la formule

$\Psi(\mathsf{x_1}, \mathsf{x_2}, \mathsf{x_3}, \mathsf{y}) \wedge \forall \mathsf{z}{<}\mathsf{y}\, (\neg\Psi(\mathsf{x_1}, \mathsf{x_2}, \mathsf{x_3}, \mathsf{z}))$.

Alors, B, qui est encore une formule Δ_0, et donc *a fortiori* Σ_1, représente aussi beta. En effet, soient $\mathsf{n}, \mathsf{p}, \mathsf{i}$ des entiers, et soit $\mathsf{m} = \mathsf{beta}(\mathsf{n}, \mathsf{p}, \mathsf{i})$. Puisque Ψ représente beta, la théorie $\mathsf{PA}_{\text{faible}}$ prouve $\Psi(\mathsf{S^n0}, \mathsf{S^p0}, \mathsf{S^i0}, \mathsf{S^m0})$. Par (#8), $\mathsf{PA}_{\text{faible}}$ prouve $\mathsf{z} < \mathsf{S^m0} \Rightarrow (\mathsf{z} = 0 \vee \cdots \vee \mathsf{z} = \mathsf{S^{m-1}0})$ et, pour $\mathsf{j} = 0, \ldots, \mathsf{m} - 1$, elle prouve aussi $\neg\Psi(\mathsf{S^n0}, \mathsf{S^p0}, \mathsf{S^i0}, \mathsf{S^j0})$. Par conséquent, $\mathsf{PA}_{\text{faible}}$ prouve $\mathsf{B}(\mathsf{S^n0}, \mathsf{S^p0}, \mathsf{S^i0}, \mathsf{S^m0})$. Par ailleurs, puisque Ψ représente beta, on a

$$\mathsf{PA}_{\text{faible}} \vdash \Psi(\mathsf{S}^n 0, \mathsf{S}^p 0, \mathsf{S}^i 0, y) \Rightarrow y = \mathsf{S}^m 0,$$

donc, *a fortiori*, $\mathsf{PA}_{\text{faible}} \vdash \mathsf{B}(\mathsf{S}^n 0, \mathsf{S}^p 0, \mathsf{S}^i 0, y) \Rightarrow y = \mathsf{S}^m 0$, et on conclut que B représente beta.

Or, soit $\mathcal{M}$ un modèle quelconque de $\mathsf{PA}_{\text{faible}}$, et supposons

$$\mathcal{M} \models \mathsf{B}(a, b, c, \mathsf{S}^m 0). \qquad (\#13)$$

Soit d un élément du domaine de $\mathcal{M}$ distinct de $(\mathsf{S}^m 0)^{\mathcal{M}}$. Par (#8), on a nécessairement $\mathcal{M} \models d < \mathsf{S}^m 0$ ou $\mathcal{M} \models \mathsf{S}^m 0 < d$. Dans le premier cas, l'hypothèse (#13) entraîne $\mathcal{M} \not\models \mathsf{B}(a, b, c, d)$. Dans le second cas, si l'on avait $\mathcal{M} \models \mathsf{B}(a, b, c, d)$, on déduirait $\mathcal{M} \not\models \mathsf{B}(a, b, c, \mathsf{S}^m 0)$, contrairement à (#13). Par conséquent, on a nécessairement $\mathcal{M} \not\models \mathsf{B}(a, b, c, d)$ et donc $\mathsf{PA}_{\text{faible}}$ prouve

$$\mathsf{B}(x_1, x_2, x_3, \mathsf{S}^m 0) \Rightarrow \forall y{\neq}\mathsf{S}^m 0\, (\neg \mathsf{B}(x_1, x_2, x_3, y)). \qquad \square$$

▷ *Noter que la fin de la démonstration ci-dessus n'est pas spécifique à la fonction* beta *: dès qu'une fonction* f *est représentable, il existe une formule représentant* f *vérifiant la contrepartie de* (#12). ◁

3.3.8.— On est maintenant prêt pour traiter les définitions récursives.

Lemme.— *Les fonctions Σ_1-représentables dans $\mathsf{PA}_{\text{faible}}$ sont closes par définition par récursion.*

Démonstration. Soient g et h des fonctions respectivement de $\mathbb{N}^p$ et de $\mathbb{N}^{p+2}$ dans $\mathbb{N}$ représentées dans $\mathsf{PA}_{\text{faible}}$ par les formules $\Psi(\vec{x}, y)$ et $\Theta(\vec{x}, x_{p+1}, x_{p+2}, y)$ de complexité Σ_1, et soit f la fonction rec(g, h). Affirmer $f(\vec{n}, k) = m$ dans $\mathbb{N}$, c'est affirmer qu'il existe des entiers $m_0, m_1, \ldots, m_k$ vérifiant

$$m_0 = g(\vec{n}) \wedge m_1 = h(\vec{n}, 1, m_0) \wedge \cdots \wedge m_k = h(\vec{n}, k, m_{k-1}) \wedge m = m_k. \qquad (\#14)$$

En vertu de 1.2.2, cette condition est vérifiée si, et seulement si, il existe deux entiers s, t vérifiant

$$\begin{aligned} &\mathsf{beta}(s, t, 0) = g(\vec{n}) \wedge \mathsf{beta}(s, t, 1) = h(\vec{n}, 1, \mathsf{beta}(s, 0)) \wedge \cdots \\ &\qquad \wedge \mathsf{beta}(s, t, k) = h(\vec{n}, k, \mathsf{beta}(s, k-1)) \wedge m = \mathsf{beta}(s, t, k), \end{aligned}$$

donc

$$\begin{aligned} &\mathsf{beta}(s, t, 0) = g(\vec{n}) \\ &\qquad \wedge \forall i{<}k\, (\mathsf{beta}(s, t, i+1) = h(\vec{n}, i+1, \mathsf{beta}(s, t, i))) \wedge m = \mathsf{beta}(s, t, k), \end{aligned}$$

donc si, et seulement si, il existe deux entiers s, t vérifiant

$$\begin{aligned} &\exists x_1\, (\mathsf{beta}(s, t, 0) = x_1 \wedge g(\vec{n}) = x_1) \\ &\qquad \wedge \forall x_2{<}k\, \exists x_3\, \exists x_4\, (\mathsf{beta}(s, t, x_2) = x_3 \wedge \mathsf{beta}(s, t, S(x_2)) = x_4 \\ &\qquad\qquad \wedge h(\vec{n}, S(x_2), x_3) = x_4) \wedge \mathsf{beta}(s, t, k) = m. \end{aligned}$$

Soit alors $\Phi(s, t, \vec{x}, y, z)$ la formule

$$\begin{aligned} &\exists x_1 (\mathsf{B}(s, t, 0, x_1) \wedge \Psi(\vec{x}, x_1)) \\ &\qquad \wedge \forall x_2{<}y\, \exists x_3\, \exists x_4\, (\mathsf{B}(s, t, x_2, x_3) \wedge \mathsf{B}(s, t, S(x_2), x_4) \\ &\qquad\qquad \wedge \Theta(\vec{x}, S(x_2), x_3, x_4)) \wedge \mathsf{B}(s, t, y, z) \qquad (\#15) \end{aligned}$$

obtenue à partir de la précédente en substituant des variables $\vec{x}$ aux entiers de la suite $\vec{n}$ et en remplaçant les fonctions g, h et beta par les formules Φ, Θ et B qui, par hypothèse, les représentent dans $\mathsf{PA}_{\text{faible}}$. On va montrer que $\exists s, t\, (\Phi)$, qui, par construction, est une formule Σ_1, représente f dans $\mathsf{PA}_{\text{faible}}$. Pour cela, soient $\vec{n}$ et k des entiers naturels, et $\mathcal{M}$ un modèle de $\mathsf{PA}_{\text{faible}}$, de domaine M.

Il s'agit de montrer que

- $\mathcal{M}$ satisfait $\exists s, t(\Phi(s, t, \overrightarrow{S^n 0}, S^k 0, a))$ pour au plus un a dans M, et
- $\mathcal{M}$ satisfait $\exists s, t(\Phi(s, t, \overrightarrow{S^n 0}, S^k 0, S^m 0))$ pour $m = f(\vec{n}, k)$.

Supposons $\mathcal{M} \models \exists s, t\, (\Phi(s, t, \overrightarrow{S^n 0}, S^k 0, a))$. Il existe donc s, t et a_0 dans M tels que le modèle $\mathcal{M}$ satisfait

$$\mathsf{B}(s,t,0,a_0) \wedge \Psi(\overrightarrow{\mathsf{S}^n 0}, a_0)$$
$$\wedge\, \forall \mathsf{x}_2 {<} \mathsf{S}^k 0\, \exists \mathsf{x}_3, \mathsf{x}_4\, (\mathsf{B}(s,t,\mathsf{x}_2,\mathsf{x}_3) \wedge \mathsf{B}(s,t,\mathsf{S}(\mathsf{x}_2),\mathsf{x}_4)$$
$$\wedge\, \Theta(\overrightarrow{\mathsf{S}^n 0}, \mathsf{S}(\mathsf{x}_2), \mathsf{x}_3, \mathsf{x}_4)) \wedge \mathsf{B}(s,t,\mathsf{S}^k 0, a_0).$$

On montre par induction sur i que $\mathcal{M}$ satisfait $\mathsf{B}(s,t,\mathsf{S}^i 0, \mathsf{S}^{m_i} 0)$, où $\mathsf{m_i}$ est l'entier défini par (#14). Pour $\mathsf{i} = 0$, puisque $\mathcal{M}$ satisfait $\Psi(\overrightarrow{\mathsf{S}^n 0}, a_0)$ et que Ψ représente g, on doit avoir $a_0 = (\mathsf{S}^{m_0} 0)^{\mathcal{M}}$ avec $\mathsf{m}_0 = \mathsf{g}(\vec{\mathsf{n}})$. Ensuite, pour chaque entier i entre 0 et $\mathsf{k}-1$, le système $\mathsf{PA}_{\text{faible}}$ prouve, donc le modèle $\mathcal{M}$ satisfait, $\mathsf{S}^i 0 < \mathsf{S}^k 0$, et l'hypothèse que $\mathcal{M}$ satisfait le fragment central de la formule implique que, pour tout tel i, le modèle $\mathcal{M}$ satisfait

$$\exists \mathsf{x}\, \exists \mathsf{x}'\, (\mathsf{B}(s,t,\mathsf{S}^i 0, \mathsf{x}) \wedge \mathsf{B}(s,t,\mathsf{S}^{i+1} 0, \mathsf{x}') \wedge \Theta(\overrightarrow{\mathsf{S}^n 0}, \mathsf{S}^{i+1} 0, \mathsf{x}, \mathsf{x}')).$$

Soient a et a' tels que $\mathcal{M}$ satisfait à la fois

$$\mathsf{B}(s,t,\mathsf{S}^i 0, a),\ \ \mathsf{B}(s,t,\mathsf{S}^{i+1} 0, a'),\ \text{et}\ \ \Theta(\overrightarrow{\mathsf{S}^n 0}, \mathsf{S}^{i+1} 0, a, a').$$

Par hypothèse d'induction, on a $\mathcal{M} \models \mathsf{B}(s,t,\mathsf{S}^i 0, \mathsf{S}^{m_i} 0)$, et l'hypothèse additionnelle d'unicité incluse dans la formule B établie en 1.2.2 garantit [6] $a = (\mathsf{S}^{m_i} 0)^{\mathcal{M}}$. La formule $\Theta(\overrightarrow{\mathsf{S}^n 0}, \mathsf{S}^{i+1} 0, a, a')$, c'est-à-dire $\Theta(\overrightarrow{\mathsf{S}^n 0}, \mathsf{S}^{i+1} 0, \mathsf{S}^{m_i} 0, a')$, entraîne alors

$$a' = (\mathsf{S}^{m_{i+1}} 0)^{\mathcal{M}}, \text{ donc } \mathcal{M} \models \mathsf{B}(s,t,\mathsf{S}^{i+1} 0, \mathsf{S}^{m_{i+1}} 0),$$

et la récurrence continue. À la fin, on obtient que a est nécessairement $(\mathsf{S}^m 0)^{\mathcal{M}}$, avec $\mathsf{m} = \mathsf{f}(\vec{\mathsf{n}}, \mathsf{k})$.

Le second point est facile. Avec les mêmes notations, en supposant $\mathsf{beta}(s,t,\mathsf{i}) = \mathsf{m_i}$ pour $0 \leqslant \mathsf{i} \leqslant \mathsf{k}$, on montre que $\mathcal{M}$ satisfait $\Phi(s,t,\overrightarrow{\mathsf{S}^n 0}, \mathsf{S}^k 0, \mathsf{S}^m 0)$. Le seul point à noter est l'utilisation de (#8) pour passer

de « $(\mathsf{x}_2 = 0 \vee \cdots \vee \mathsf{x}_2 = \mathsf{S}^{k-1} 0) \Rightarrow \ldots$ » à « $(\mathsf{x}_2 < \mathsf{S}^k 0) \Rightarrow \ldots$ ». □

La démonstration de la proposition 3.3.2 est donc complète.

3.3.9.— On passe à la représentabilité pour les relations récursives.

Définition (représentable).— Une formule $\Phi(\mathsf{x}_1, \ldots, \mathsf{x}_p)$ est dite *représenter* une relation R de $\mathbb{N}^p$ dans T si, pour tous $\mathsf{n}_1, \ldots, \mathsf{n}_p$ dans $\mathbb{N}$,

- T prouve $\Phi(\mathsf{S}^{n_1} 0, \ldots, \mathsf{S}^{n_p} 0)$ lorsque $\mathsf{R}(\mathsf{n}_1, \ldots, \mathsf{n}_p)$ est vraie, et
- T prouve $\neg\Phi(\mathsf{S}^{n_1} 0, \ldots, \mathsf{S}^{n_p} 0)$ lorsque $\mathsf{R}(\mathsf{n}_1, \ldots, \mathsf{n}_p)$ est fausse.

3.3.10.— Le résultat est alors analogue à celui des fonctions.

Corollaire (représentabilité).— *Toute relation récursive sur $\mathbb{N}^p$ peut être représentée dans le système $\mathsf{PA}_{\text{faible}}$ par une formule de complexité Σ_1.*

Démonstration. Supposons que R est une relation récursive sur $\mathbb{N}^p$. La fonction indicatrice de R est alors une fonction récursive totale. Il existe donc une formule $\Phi(\vec{\mathsf{x}}, \mathsf{y})$ de complexité Σ_1 représentant $\mathbf{1}_{\mathsf{R}}$ dans $\mathsf{PA}_{\text{faible}}$. Soit $\Phi'(\vec{\mathsf{x}})$ la formule $\Phi(\vec{\mathsf{x}}, \mathsf{S}^1 0)$, et soit $\vec{\mathsf{n}}$ une suite d'entiers quelconque. Si $\mathsf{R}(\vec{\mathsf{n}})$ est vraie, alors $\mathsf{PA}_{\text{faible}}$ prouve $\Phi(\overrightarrow{\mathsf{S}^n 0}, \mathsf{S}^1 0)$, donc $\Phi'(\overrightarrow{\mathsf{S}^n 0})$. Si $\mathsf{R}(\vec{\mathsf{n}})$ est fausse, alors $\mathsf{PA}_{\text{faible}}$ prouve $\Phi(\overrightarrow{\mathsf{S}^n 0}, 0)$, et donc $\neg\Phi(\overrightarrow{\mathsf{S}^n 0}, \mathsf{S}^1 0)$, soit $\neg\Phi'(\overrightarrow{\mathsf{S}^n 0})$, par définition de la représentabilité d'une fonction. □

6. Rien ne permet d'affirmer que s et t sont des éléments standards de $\mathcal{M}$: on ne peut donc pas affirmer que $\mathsf{B}(s,t,\mathsf{S}^i 0, a)$ est satisfait pour un a au plus en général. Comme on ne peut pas introduire une suite de variables de longueur k puisque k est variable, et que l'on doit se contenter d'une sous-formule locale reliant les valeurs en i et en $\mathsf{i}+1$, l'unicité de la valeur en i est essentielle.

3.3.11.— Il ne peut y avoir unicité des formules Σ_1 représentant une fonction ou une relation récursive. Pour la plupart des développements de la section 4, il est loisible d'utiliser n'importe quelle formule mais, pour 4.5.9, il est utile que la formule soit précisément celle qui résulte des démonstrations des lemmes 3.3.4, 3.3.6, et 3.3.8, c'est-à-dire suive pas à pas la définition de la fonction représentée. On pose une définition pour ce cas.

Définition (représentation fidèle).— On dit qu'une formule Φ représente *fidèlement* une fonction f s'il existe une définition récursive def_f de f telle que Φ est la formule Σ_1 déduite de def_f en partant des formules de 3.3.3 et en utilisant précisément (#10), (#11), et (#15) pour mimer chaque étape de def_f. De même, *mutatis mutandis*, pour une relation.

4. Indécidabilité et incomplétude

Grâce aux résultats préparatoires accumulés dans les sections précédentes, on va maintenant établir les théorèmes de limitation annoncés, qui ont en commun d'énoncer des résultats d'impossibilité traduisant des limitations intrinsèques de la logique du premier ordre.

> ▷ *On constatera que les arguments techniques pour tous les « grands » théorèmes sont assez brefs, une fois le lemme diagonal établi. On pourra aussi noter que, de tous les résultats préliminaires obtenus dans les sections précédentes, un seul est crucial, à savoir l'existence d'une numérotation des formules telle que la fonction de substitution associée soit Σ_1-représentable dans le système* $\mathsf{PA}_{\text{faible}}$. ◁

Cette (longue) section comporte cinq sous-sections. Dans la sous-section 4.1, on établit une forme générique de l'argument diagonal, appelée lemme diagonal, mettant en jeu une formule d'arithmétique quelconque. Ensuite, en appliquant le lemme diagonal à des formules convenablement choisies, on déduit successivement le théorème d'indécidabilité de Church dans la sous-section 4.2, le théorème de Tarski sur la non-définissabilité de la vérité arithmétique dans la sous-section 4.3, puis le premier théorème d'incomplétude de Gödel dans la sous-section 4.4, et enfin le second théorème d'incomplétude de Gödel dans la sous-section 4.5, ici sous une forme non optimale puisqu'établi seulement pour les théories au moins aussi fortes que la théorie de Zermelo, alors que l'arithmétique de Peano serait suffisante.

4.1. Le lemme diagonal

> ▶ Résumé.— Le lemme diagonal est l'argument général d'autoréférence à la base de tous les résultats ultérieurs de ce chapitre. ◀

4.1.1.— Le lemme diagonal est une variation sur l'argument éponyme, et, comme toujours, il combine autoréférence et négation. L'autoréférence vient ici de la possibilité d'appliquer une formule arithmétique à une (autre) formule *via* l'arithmétisation de la syntaxe : *a priori*, les formules d'arith-

métique s'appliquent aux entiers, mais, une fois les formules numérotées par des entiers, rien n'empêche d'appliquer une formule au numéro d'une autre formule, et même — c'est là que l'autoréférence arrive — au sien ou à celui de sa négation.

> ▷ *Le principe est le suivant. Grâce à l'arithmétisation, on peut faire comme si les formules s'appliquaient aux formules, ou encore les entiers aux entiers. Soit* $\Phi(\mathsf{x})$ *une formule d'arithmétique quelconque. Si l'on pose* $\Phi'(\mathsf{x}) = \Phi(\mathsf{x}(\mathsf{x}))$*, et* $\Delta = \neg\Phi'(\neg\Phi')$*, on a alors l'« égalité »*
>
> $$\Phi(\Delta) = \Phi(\neg\Phi'(\neg\Phi')) = \Phi'(\neg\Phi') = \neg\Delta.$$
>
> *Vis-à-vis de* Φ*, la formule* Δ *est alors (équivalente à) sa propre négation : si* $\Phi(\mathsf{x})$ *signifie « x est satisfait », alors* Δ*, affirmant sa propre fausseté, ne peut être ni vraie ni fausse, sur le modèle du paradoxe du menteur qui constate qu'aucune valeur de vérité ne peut être attribuée à la phrase « Je mens ». De même, si* $\Phi(\mathsf{x})$ *signifie « x est prouvable »,* Δ *affirme sa propre non-prouvabilité.* ◁

4.1.2.— Dans toute la suite, l'expression « formule d'arithmétique » réfère à une formule de $\mathcal{L}_{\mathrm{arith}+}$, c'est-à-dire une formule du premier ordre vis-à-vis de la signature $(0, \mathsf{S}, +, \cdot, \leqslant)$. On rappelle que $\mathcal{L}_{\max}$ est une logique du premier ordre avec des familles dénombrables de symboles d'opération et de relation de toutes les arités (2.1.1). On rappelle aussi l'existence, établie en 2.1.5, d'une numérotation des formules et d'une fonction récursive subst vérifiant $\mathsf{subst}(\ulcorner\Phi\urcorner, \mathsf{k}) = \ulcorner\Phi(\mathsf{S}^{\mathsf{k}}0)\urcorner$ pour tous Φ et k.

4.1.3.— On fixe une fois pour toutes une formule $\Upsilon(\mathsf{x}, \mathsf{y}, \mathsf{z})$ de complexité Σ_1 de $\mathcal{L}_{\mathrm{arith}+}$ représentant fidèlement [7] la fonction subst dans $\mathsf{PA}_{\mathrm{faible}}$, laquelle existe par 2.1.5 et 3.3.2. Le résultat central est alors le suivant.

Proposition (lemme diagonal).— *Soit* $\Phi(\mathsf{z})$ *une formule de* $\mathcal{L}_{\max}$ *à une variable libre. Notant* $\Theta(\mathsf{x})$ *pour* $\exists\mathsf{z}\,(\Upsilon(\mathsf{x}, \mathsf{x}, \mathsf{z}) \wedge \Phi(\mathsf{z}))$ *et* n *pour* $\ulcorner\neg\Theta(\mathsf{x})\urcorner$*, on appelle* $\mathsf{Diag}(\Phi)$ *la formule close* $\neg\Theta(\mathsf{S}^{\mathsf{n}}0)$*. Alors, si* Δ *est* $\mathsf{Diag}(\Phi)$*, on a*

$$\mathsf{PA}_{\mathrm{faible}} \vdash \Phi(\mathsf{S}^{\ulcorner\Delta\urcorner}0) \Leftrightarrow \neg\Delta, \qquad (\#16)$$

De plus, si $\Phi(\mathsf{x})$ *est* Σ_1*, alors* $\mathsf{Diag}(\Phi)$ *est la négation d'une formule* Σ_1*.*

> *Démonstration.* Écrivant Δ pour $\mathsf{Diag}(\Phi)$, on a par définition
>
> $$\Delta = \neg\exists\mathsf{z}\,(\Upsilon(\mathsf{S}^{\mathsf{n}}0, \mathsf{S}^{\mathsf{n}}0, \mathsf{z}) \wedge \Phi(\mathsf{z})),$$
>
> donc, si Φ est de complexité Σ_1, alors Δ est la négation d'une formule Σ_1. Par le théorème de complétude, il suffit pour établir l'équivalence (#16) de montrer que tout modèle de $\mathsf{PA}_{\mathrm{faible}}$ satisfaisant $\Phi(\mathsf{S}^{\ulcorner\Delta\urcorner}0)$ satisfait $\neg\Delta$, et que tout modèle de $\mathsf{PA}_{\mathrm{faible}}$ satisfaisant $\neg\Delta$ satisfait $\Phi(\mathsf{S}^{\ulcorner\Delta\urcorner}0)$.
>
> Par construction, on a $\ulcorner\Delta\urcorner = \ulcorner\neg\Theta(\mathsf{S}^{\mathsf{n}}0)\urcorner = \mathsf{subst}(\mathsf{n}, \mathsf{n})$, et, par conséquent, puisque la formule Υ représente la fonction subst dans $\mathsf{PA}_{\mathrm{faible}}$, on a
>
> $$\mathsf{PA}_{\mathrm{faible}} \vdash \Upsilon(\mathsf{S}^{\mathsf{n}}0, \mathsf{S}^{\mathsf{n}}0, \mathsf{S}^{\ulcorner\Delta\urcorner}0), \qquad (\#17)$$
>
> $$\mathsf{PA}_{\mathrm{faible}} \vdash \forall\mathsf{z}(\Upsilon(\mathsf{S}^{\mathsf{n}}0, \mathsf{S}^{\mathsf{n}}0, \mathsf{z}) \Rightarrow \mathsf{z} = \mathsf{S}^{\ulcorner\Delta\urcorner}0). \qquad (\#18)$$
>
> Supposons que $\mathcal{M}$ est modèle de $\mathsf{PA}_{\mathrm{faible}}$ et satisfait $\Phi(\mathsf{S}^{\ulcorner\Delta\urcorner}0)$. Par (#17), $\mathcal{M}$ satisfait aussi $\Upsilon(\mathsf{S}^{\mathsf{n}}0, \mathsf{S}^{\mathsf{n}}0, \mathsf{S}^{\ulcorner\Delta\urcorner}0)$, donc il existe c, à savoir $(\mathsf{S}^{\ulcorner\Delta\urcorner}0)^{\mathcal{M}}$, tel que $\mathcal{M}$ satisfait $\Upsilon(\mathsf{S}^{\mathsf{n}}0, \mathsf{S}^{\mathsf{n}}0, c) \wedge \Phi(c)$. Par conséquent, $\mathcal{M}$ satisfait $\Theta(\mathsf{S}^{\mathsf{n}}0)$, c'est-à-dire $\neg\Delta$.

7. L'hypothèse que la représentation est fidèle ne sera pas utilisée avant 4.5.9.

Inversement, supposons que $\mathcal{M}$ est modèle de $\mathsf{PA}_{\text{faible}}$ et satisfait $\neg\Delta$. Il existe donc dans le domaine de $\mathcal{M}$ un élément c satisfaisant $\Upsilon(\mathsf{S}^n0, \mathsf{S}^n0, c)$ et $\Phi(c)$. D'après (#18), l'élément c est forcément $(\mathsf{S}^{\ulcorner\Delta\urcorner}0)^{\mathcal{M}}$, et l'hypothèse que $\mathcal{M}$ satisfait $\Phi(c)$ signifie que $\mathcal{M}$ satisfait $\Phi(\mathsf{S}^{\ulcorner\Delta\urcorner}0)$. □

4.2. Le théorème d'indécidabilité de Church

▶ Résumé.— Ni l'ensemble des formules d'arithmétique prouvables à partir de $\mathsf{PA}_{\text{faible}}$ ni l'ensemble de celles qui sont valides ne sont récursifs. ◀

4.2.1.— Une première application du lemme diagonal est le théorème d'indécidabilité de Church affirmant que l'ensemble des formules d'arithmétique qui sont prouvables à partir de $\mathsf{PA}_{\text{faible}}$ et l'ensemble des formules d'arithmétique qui sont valides sont des ensembles compliqués, à savoir non récursifs. Moyennant la thèse de Church–Turing, cela signifie qu'il n'existe aucun moyen effectif permettant de décider si une formule d'arithmétique est ou non prouvable à partir de $\mathsf{PA}_{\text{faible}}$, ou encore est valide, c'est-à-dire prouvable à partir des seuls axiomes de la logique $\mathcal{L}_{\text{arith}+}$.

▷ *On notera que le théorème ne dit* rien *quant à une formule particulière : ce qu'il affirme, c'est qu'il n'existe pas de méthode uniforme permettant de décider, pour* toute *formule d'arithmétique* Φ*, si* Φ *est ou non valide, autrement dit, il n'existe pas de recette miraculeuse valable pour toutes les formules à la fois.* ◁

4.2.2.— On rappelle qu'un ensemble de formules T est dit récursif si l'ensemble des numéros des formules appartenant à T est récursif.

Proposition (théorème de Church).— *Si* T *est un ensemble consistant de formules d'arithmétique incluant* $\mathsf{PA}_{\text{faible}}$*, l'ensemble des formules prouvables à partir de* T *n'est pas récursif.*

Démonstration. Soit $\overline{\mathsf{T}}$ l'ensemble des formules de $\mathcal{L}_{\text{arith}+}$ prouvables à partir de T. Supposons $\overline{\mathsf{T}}$ récursif, et soit $\ulcorner\overline{\mathsf{T}}\urcorner$ l'ensemble $\{\ulcorner\Phi\urcorner \mid \Phi \in \overline{\mathsf{T}}\}$ des numéros des formules de $\overline{\mathsf{T}}$. Alors, $\ulcorner\overline{\mathsf{T}}\urcorner$ est aussi récursif. Par 3.3.10, il existe donc une formule $\Phi(x)$ représentant $\ulcorner\overline{\mathsf{T}}\urcorner$ (vue comme relation unaire) dans $\mathsf{PA}_{\text{faible}}$. Soit $\Delta := \mathsf{Diag}(\Phi)$ associée suivant 4.1.3. Par construction, on a

$$\mathsf{PA}_{\text{faible}} \vdash \Phi(\mathsf{S}^{\ulcorner\Delta\urcorner}0) \Leftrightarrow \neg\Delta. \qquad (\#19)$$

On se demande si Δ est ou non prouvable à partir de T.

Supposons $\mathsf{T} \vdash \Delta$. Cela signifie que $\ulcorner\Delta\urcorner$ est dans $\ulcorner\overline{\mathsf{T}}\urcorner$, et l'hypothèse que Φ représente $\ulcorner\overline{\mathsf{T}}\urcorner$ dans $\mathsf{PA}_{\text{faible}}$ implique $\mathsf{PA}_{\text{faible}} \vdash \Phi(\mathsf{S}^{\ulcorner\Delta\urcorner}0)$. En utilisant (#19) et une coupure, on déduit $\mathsf{PA}_{\text{faible}} \vdash \neg\Delta$, et donc *a fortiori* $\mathsf{T} \vdash \neg\Delta$, ce qui contredit la consistance de T.

Supposons maintenant $\mathsf{T} \nvdash \Delta$. Alors, $\ulcorner\Delta\urcorner$ n'est pas dans $\ulcorner\overline{\mathsf{T}}\urcorner$, et l'hypothèse que Φ représente $\ulcorner\overline{\mathsf{T}}\urcorner$ dans $\mathsf{PA}_{\text{faible}}$ implique $\mathsf{PA}_{\text{faible}} \vdash \neg\Phi(\mathsf{S}^{\ulcorner\Delta\urcorner}0)$. Par (#19), on déduit $\mathsf{PA}_{\text{faible}} \vdash \Delta$, donc $\mathsf{T} \vdash \Delta$, ce qui contredit l'hypothèse $\mathsf{T} \nvdash \Delta$. L'hypo-

thèse que $\overline{\mathsf{T}}$ est récursif — de même que l'hypothèse[8] que $\ulcorner\overline{\mathsf{T}}\urcorner$ est représentable dans $\mathsf{PA}_{\text{faible}}$ — est donc contradictoire. □

4.2.3.— Une application directe du théorème de Church est l'existence d'un ensemble semi-récursif non récursif.

Corollaire (complexité).— *Si* T *est un ensemble récursif et consistant de formules d'arithmétique incluant* $\mathsf{PA}_{\text{faible}}$*, alors l'ensemble des formules prouvables à partir de* T *est un ensemble semi-récursif non récursif.*

Démonstration. Comme ci-dessus, soit $\overline{\mathsf{T}}$ l'ensemble des formules prouvables à partir de T. Par 2.2.5, $\overline{\mathsf{T}}$ est semi-récursif, et, par le théorème de Church, il est non récursif. □

4.2.4.— Une seconde application du théorème de Church est l'impossibilité d'un algorithme permettant de reconnaître les formules de $\mathcal{L}_{\text{arith}+}$ qui sont valides (c'est-à-dire satisfaites dans toute réalisation).

Corollaire (validité).— *L'ensemble des formules de* $\mathcal{L}_{\text{arith}+}$ *qui sont valides est semi-récursif non récursif.*

Démonstration. D'après le théorème de complétude, une formule de $\mathcal{L}_{\text{arith}+}$ est valide si, et seulement si, elle est prouvable à partir d'un ensemble vide d'hypothèses. Avec les notations précédentes, l'ensemble des formules valides est donc $\overline{\emptyset}$ et, par 2.2.5, cet ensemble est semi-récursif puisque l'ensemble vide l'est.

Notons alors Θ la formule de $\mathcal{L}_{\text{arith}+}$ qui est la conjonction des axiomes constituant $\mathsf{PA}_{\text{faible}}$. Par le théorème de la déduction (VII.2.2.2), il y a équivalence, pour toute formule Φ de $\mathcal{L}_{\text{arith}+}$, entre $\mathsf{PA}_{\text{faible}} \vdash \Phi$, c'est-à-dire $\{\Theta\} \vdash \Phi$, et $\vdash \Theta \Rightarrow \Phi$, c'est-à-dire l'appartenance de $\Theta \Rightarrow \Phi$ à $\overline{\emptyset}$. Soit f la fonction primitive récursive définie par $\mathsf{f}(\mathsf{n}) := \langle \ulcorner\Rightarrow\urcorner, \ulcorner\Theta\urcorner, \mathsf{n}\rangle$. Pour toute formule Φ, on a $\mathsf{f}(\ulcorner\Phi\urcorner) = \ulcorner\Theta\Rightarrow\Phi\urcorner$, et donc $\mathsf{n} \in \ulcorner\overline{\mathsf{PA}_{\text{faible}}}\urcorner$ équivaut à $\mathsf{f}(\mathsf{n}) \in \ulcorner\overline{\emptyset}\urcorner$. Si $\overline{\emptyset}$ était récursif, il en serait de même de l'ensemble $\overline{\mathsf{PA}_{\text{faible}}}$, ce qui contredirait le théorème de Church. □

Le résultat précédent formalise le fait que la logique du premier ordre n'est pas triviale. Si les formules valides se limitaient à des évidences telles que $\mathsf{x}=\mathsf{y} \Rightarrow \mathsf{y}=\mathsf{x}$, on s'attendrait à ce qu'elles puissent être décrites simplement, et donc à ce que l'ensemble des numéros des formules valides soit simple, en particulier récursif.

▷ *Noter que le caractère récursif ou non de l'ensemble des numéros des formules valides d'une logique* $\mathcal{L}_\Sigma$ *dépend du pouvoir d'expression de* $\mathcal{L}_\Sigma$*, et donc de la signature* Σ*. On vient de voir que, si* Σ *est la signature* $\Sigma_{\text{arith}+}$ *de l'arithmétique, alors l'ensemble des numéros des formules valides est non récursif. Avec un peu de travail supplémentaire, on peut éliminer les symboles d'opération et obtenir le même résultat dès que la signature contient au moins un symbole de relation binaire. En revanche, si une signature* Σ *ne contient que des symboles de relation unaire et pas de symbole d'opération, alors le pouvoir d'expression de* $\mathcal{L}_\Sigma$ *est faible et, dans ce cas, l'ensemble des numéros des formules valides de* $\mathcal{L}_\Sigma$ *est récursif.* ◁

8. Les deux conditions sont équivalentes : si $\ulcorner\overline{\mathsf{T}}\urcorner$ est représentable dans $\mathsf{PA}_{\text{faible}}$, on peut décider $\mathsf{n} \in \overline{\mathsf{T}}$ en énumérant toutes les formules prouvables à partir de T jusqu'à voir apparaître $\Phi(\mathsf{S}^\mathsf{n}0)$ ou $\neg\Phi(\mathsf{S}^\mathsf{n}0)$.

4.3. Le théorème de non-définissabilité de la vérité de Tarski

▶ Résumé.— Il est impossible de définir la vérité dans un modèle de $\mathsf{PA}_{\text{faible}}$. La logique du second ordre n'a pas de théorème de complétude. ◀

4.3.1.— Une deuxième application, très simple, du lemme diagonal est le théorème de Tarski qui affirme l'impossibilité de définir de l'intérieur la vérité (c'est-à-dire la satisfaction) dans toute structure qui est modèle de $\mathsf{PA}_{\text{faible}}$, donc en particulier dans la structure $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$.

▷ *On notera que le théorème de Tarski réfute (ou conforte) le paradoxe de Berry, en montrant que l'on ne saurait, de l'intérieur de $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$, parler de ce qui y est vrai, donc en particulier de ce qui y est définissable.* ◁

Proposition (théorème de Tarski I).— *Si $\mathcal{M}$ est modèle de $\mathsf{PA}_{\text{faible}}$ de signature Σ finie ou dénombrable incluant $\Sigma_{\text{arith}+}$, il n'existe pas de formule $\mathsf{Sat}_{\mathcal{M}}(\mathsf{x})$ de $\mathcal{L}_\Sigma$ vérifiant, pour toute formule close Φ de $\mathcal{L}_\Sigma$,*

$$\mathcal{M} \models \Phi \quad \Leftrightarrow \quad \mathcal{M} \models \mathsf{Sat}_{\mathcal{M}}(\mathsf{S}^{\ulcorner\Phi\urcorner}0). \qquad (\#20)$$

En d'autres termes : l'ensemble des numéros des formules de $\mathcal{L}_\Sigma$ vraies dans $\mathcal{M}$ n'est pas définissable dans $\mathcal{M}$.

Démonstration. Sans perte de généralité, on peut supposer $\Sigma \subseteq \Sigma_{\max}$. Supposons que $\mathsf{Sat}_{\mathcal{M}}(\mathsf{x})$ satisfait (#20). Soit $\Delta := \mathsf{Diag}(\mathsf{Sat}_{\mathcal{M}})$. Par le lemme diagonal, on a

$$\mathsf{PA}_{\text{faible}} \vdash \mathsf{Sat}_{\mathcal{M}}(\mathsf{S}^{\ulcorner\Delta\urcorner}0) \Leftrightarrow \neg\Delta,$$

donc, puisque $\mathcal{M}$ est modèle de $\mathsf{PA}_{\text{faible}}$,

$$\mathcal{M} \models \mathsf{Sat}_{\mathcal{M}}(\mathsf{S}^{\ulcorner\Delta\urcorner}0) \Leftrightarrow \neg\Delta. \qquad (\#21)$$

Par construction, Δ est soit vraie, soit fausse dans $\mathcal{M}$. Or, (#20) et (#21) se contredisent. En effet, Si Δ est vraie, (#21) implique $\mathcal{M} \not\models \mathsf{Sat}_{\mathcal{M}}(\mathsf{S}^{\ulcorner\Delta\urcorner}0)$, d'où $\mathcal{M} \not\models \Delta$ par (#20), ce qui contredit l'hypothèse. Si Δ est fausse, (#21) implique $\mathcal{M} \models \mathsf{Sat}_{\mathcal{M}}(\mathsf{S}^{\ulcorner\Delta\urcorner}0)$, d'où $\mathcal{M} \models \Delta$ par (#20), à nouveau une contradiction. L'hypothèse que $\mathsf{Sat}_{\mathcal{M}}$ existe est donc intenable. □

4.3.2.— Le résultat de 4.3.1 s'applique en particulier à $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$.

Corollaire (théorème de Tarski II).— *Il n'existe pas de formule $\mathsf{Sat}_{\mathbb{N}}(\mathsf{x})$ de $\mathcal{L}_{\text{arith}+}$ vérifiant, pour toute formule close Φ de $\mathcal{L}_{\text{arith}+}$,*

$$(\mathbb{N}, 0, S, +, \cdot, \leqslant) \models \Phi \quad \Leftrightarrow \quad (\mathbb{N}, 0, S, +, \cdot, \leqslant) \models \mathsf{Sat}_{\mathbb{N}}(\ulcorner\Phi\urcorner).$$

Démonstration. Dans le cas particulier de la structure $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$, l'interprétation du terme $\mathsf{S}^{\mathsf{n}}0$ est, par définition, l'entier n lui-même. □

4.3.3.— On déduit du théorème de Tarski plusieurs applications. La première est que la théorie du premier ordre de $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ est un ensemble compliqué, au sens où il ne saurait être récursif, ni même semi-récursif. On commence par un résultat préparatoire.

Lemme.— *Tout sous-ensemble semi-récursif de $\mathbb{N}^{\mathsf{p}}$ dont le complémentaire est semi-récursif est récursif.*

Démonstration. Supposons que S est la projection sur $\mathbb{N}^{\mathsf{p}}$ d'un ensemble récursif R_+ de $\mathbb{N}^{\mathsf{q}}$, et que $\mathbb{N}^{\mathsf{p}} \setminus \mathsf{S}$ est la projection d'un ensemble récursif R_- de $\mathbb{N}^{\mathsf{r}}$. Quitte à remplacer R_+ ou R_- par son produit avec une puissance convenable de $\mathbb{N}$, on peut supposer $\mathsf{r} = \mathsf{q}$. La fonction $\mathbf{1}_{\mathsf{R}_+} + \mathbf{1}_{\mathsf{R}_-}$ est récursive totale, et il en est de même de la fonction f qui à $\vec{\mathsf{n}}$ associe le plus petit m tel que $\mathbf{1}_{\mathsf{R}_+}(\vec{\mathsf{n}}, \mathsf{m}) + \mathbf{1}_{\mathsf{R}_-}(\vec{\mathsf{n}}, \mathsf{m})$ vaut 1 : en effet, f est récursive puisqu'elle se définit comme

$$\mathsf{minim}(\mathsf{comp}(\mathbf{1}_{\{1\}}, \mathsf{comp}(\mathsf{add}, \mathbf{1}_{\mathsf{R}_+}, \mathbf{1}_{\mathsf{R}_-}))),$$

et elle est totale puisque la réunion des projections de R_+ et R_- est $\mathbb{N}^{\mathsf{p}}$. On trouve alors $\mathbf{1}_{\mathsf{S}}(\vec{\mathsf{n}}) = \mathbf{1}_{\mathsf{R}_+}(\vec{\mathsf{n}}, \mathsf{f}(\vec{\mathsf{n}}))$, donc $\mathbf{1}_{\mathsf{S}}$ est une fonction récursive, et S est récursif. □

4.3.4.— On rappelle que $\mathsf{Th}_1\mathcal{S}$ est la famille des formules closes du premier ordre satisfaites dans la structure $\mathcal{S}$.

Proposition (indécidabilité).— *L'ensemble $\mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ n'est pas semi-récursif.*

Démonstration. Supposons que $\mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ est semi-récursif. Par construction, $\Phi \notin \mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ équivaut à $\neg\Phi \in \mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot, \leqslant)$, donc le complémentaire dans $\mathbb{N}$ de $\ulcorner\mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot, \leqslant)\urcorner$ est également semi-récursif. Par 4.3.3, l'ensemble $\ulcorner\mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot, \leqslant)\urcorner$ est donc récursif, puis représentable dans $\mathsf{PA}_{\text{faible}}$ par une formule $\Psi(\mathsf{x})$ de complexité Σ_1. Pour toute formule Φ close, la relation $(\mathbb{N}, 0, S, +, \cdot, \leqslant) \models \Phi$, c'est-à-dire $\ulcorner\Phi\urcorner \in \ulcorner\mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot, \leqslant)\urcorner$, équivaut alors à la relation $(\mathbb{N}, 0, S, +, \cdot, \leqslant) \models \Psi(\ulcorner\Phi\urcorner)$. C'est dire que Ψ satisfait à l'équivalence (#20), ce qui contredit le théorème de Tarski (4.3.1). □

▷ *Un résultat semblable vaut pour la théorie de tout modèle $\mathcal{M}$ de PA_1 : reprenant la démonstration ci-dessus et supposant $\mathsf{Th}_1(\mathcal{M})$ semi-récursif donc récursif, on peut représenter $\ulcorner\mathsf{Th}_1(\mathcal{M})\urcorner$ par une formule Ψ de complexité Σ_1, et (en considérant son complémentaire) par une formule Ψ' de complexité Π_1, de sorte que PA_1 prouve $\Psi \Leftrightarrow \Psi'$. Alors, $\mathcal{M} \models \Phi$ équivaut à $(\mathbb{N}, 0, S, +, \cdot) \models \Psi(\ulcorner\Phi\urcorner)$, qui implique $\mathcal{M} \models \Psi(\ulcorner\Phi\urcorner)$ puisque Ψ est Σ_1, tandis que $\mathcal{M} \models \Psi'(\ulcorner\Phi\urcorner)$ implique la relation $(\mathbb{N}, 0, S, +, \cdot) \models \Psi'(\ulcorner\Phi\urcorner)$ puisque Ψ' est Π_1, d'où $\mathcal{M} \models \Phi$. Donc, $\mathcal{M} \models \Phi$ équivaut à $\mathcal{M} \models \Psi(\ulcorner\Phi\urcorner)$, et on conclut comme plus haut.* ◁

4.3.5.— Moyennant la thèse de Church–Turing, le résultat précédent implique qu'aucun procédé algorithmique uniforme ne peut décider si un énoncé d'arithmétique est satisfait dans $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$. Il est naturel de se demander si l'indécidabilité persiste pour des énoncés de forme particulière, typiquement l'existence d'un zéro pour une équation diophantienne. On signale sans démonstration le raffinement suivant.

Proposition (théorème MRDP [9]).— *Il existe un polynôme à coefficients entiers $P(\mathsf{x}, \vec{\mathsf{y}})$ en dix variables tel que $\{\mathsf{n} \mid \exists \vec{\mathsf{y}}\,(P(\mathsf{n}, \vec{\mathsf{y}}) = 0)\}$ est semi-récursif non récursif.*

▷ *La démonstration consiste à montrer par des codages extrêmement délicats que, pour* tout *sous-ensemble semi-récursif S de $\mathbb{N}$, il existe un polynôme entier $Q(\mathsf{x}, \vec{\mathsf{y}})$ tel que S est $\{\mathsf{n} \mid \exists \vec{\mathsf{y}}\,(Q(\mathsf{n}, \vec{\mathsf{y}}) = 0)\}$. Le théorème montre que la résolubilité d'une équation diophantienne à 9 inconnues est indécidable, et résout négativement le dixième problème de Hilbert.* ◁

4.3.6.— La seconde application du théorème de Tarski est une forme faible du premier théorème d'incomplétude de Gödel dans laquelle on n'exhibe pas de formule non prouvable explicite.

Proposition (premier théorème d'incomplétude de Gödel, forme faible).— *Pour toute théorie récursive T incluse dans $\mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot, \leqslant)$, il existe une formule vraie dans $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ non prouvable à partir de T.*

Démonstration. Soit $\overline{\mathsf{T}}$ l'ensemble des formules closes prouvables à partir de T. Toute formule close de $\overline{\mathsf{T}}$ est satisfaite dans $(\mathbb{N}, 0, S, +, \cdot)$, donc on a

$$\overline{\mathsf{T}} \subseteq \mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot).$$

Par 2.2.5, $\overline{\mathsf{T}}$ est semi-récursif. Comme $\mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot)$ n'est pas semi-récursif, l'inclusion de $\overline{\mathsf{T}}$ dans $\mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot)$ doit être stricte, ce qui signifie qu'il existe une formule close Φ vraie dans $(\mathbb{N}, 0, S, +, \cdot)$ et non prouvable à partir de T. □

Noter que le résultat s'applique en particulier au système de Peano PA_1, puisque ce dernier est un ensemble (primitif) récursif, ainsi que vu dans l'exemple de 2.2.3 (voir 4.4.5 ci-dessous).

4.3.7.— Une troisième application du théorème de Tarski est l'impossibilité d'une notion satisfaisante de prouvabilité pour la logique du second ordre.

Proposition (second ordre).— *Il n'existe pas de notion de preuve pour la logique du second ordre telle que toute formule valide soit prouvable et que l'ensemble des numéros des formules prouvables soit semi-récursif.*

Démonstration. Le système PA complété de $\mathsf{Def}_{\leqslant}$ est équivalent à l'unique formule du second ordre Θ qui est la conjonction des sept formules de III.1.2.1 et de $\mathsf{Def}_{\leqslant}$. Par le résultat du §VII.4.3.4, le seul modèle de Θ est la structure $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$, et donc dire qu'une formule close Φ est vraie dans la structure $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ équivaut à dire que la formule $\Theta \Rightarrow \Phi$ est valide. S'il existait une notion de preuve vérifiant les conditions de l'énoncé, cela équivaudrait à dire que $\Theta \Rightarrow \Phi$ est prouvable, et il en résulterait que $\mathsf{Th}_2(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ serait semi-récursif, puis *a fortiori* que $\mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ le serait, contredisant 4.3.4. □

4.3.8.— Le résultat précédent illustre l'intérêt spécifique du premier ordre comparé au second. En logique du premier ordre, il existe une notion de

9. « Matiyasevich–Robinson–Davis–Putnam ».

preuve suffisamment simple pour que les formules prouvables forment une famille explicitement énumérable, et en même temps suffisamment riche pour que toutes les formules valides soient prouvables. En logique du second ordre, il ne peut exister de notion de preuve réunissant ces deux vertus antagonistes : si une notion de preuve était suffisamment riche pour qu'il y ait complétude, alors elle devrait être trop compliquée pour que les formules prouvables puissent être énumérées effectivement.

4.4. Le premier théorème d'incomplétude de Gödel

▶ Résumé.— Pour toute axiomatisation de l'arithmétique T, il existe une formule canonique explicite vraie dans $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ et non prouvable à partir de T. ◀

4.4.1.— On va maintenant appliquer le lemme diagonal non plus à la notion sémantique de satisfaction, mais à la notion syntaxique de prouvabilité et en déduire le premier théorème d'incomplétude de Gödel qui affirme que toute axiomatisation raisonnable de l'arithmétique est incomplète : il existe une formule explicite qui est vraie mais non prouvable.

4.4.2.— Si T est une théorie récursive, alors, par 2.2.3, la relation $\mathsf{Preuve}_{\mathsf{T}}$ est récursive, et donc, par 3.3.10, elle est représentable dans $\mathsf{PA}_{\text{faible}}$ par une formule de complexité Σ_1. Dans la suite, on fixe une telle formule, et on considère la formule diagonale associée.

Définition (formule de Gödel).— Soit T une théorie récursive de $\mathcal{L}_{\max}$.

(i) On fixe une formule $\underline{\mathsf{Preuve}}_{\mathsf{T}}(\mathsf{x}, \mathsf{y})$ de complexité Σ_1 représentant fidèlement la relation $\mathsf{Preuve}_{\mathsf{T}}(\mathsf{n}, \mathsf{m})$ dans $\mathsf{PA}_{\text{faible}}$, et on écrit $\underline{\mathsf{Prouvable}}_{\mathsf{T}}(\mathsf{x})$ pour $\exists \mathsf{y}\,(\underline{\mathsf{Preuve}}_{\mathsf{T}}(\mathsf{x}, \mathsf{y}))$.

(ii) On pose alors $\Delta_{\mathsf{T}} := \mathsf{Diag}(\underline{\mathsf{Prouvable}}_{\mathsf{T}})$; on appelle Δ_{T} la *formule de Gödel* pour T.

▷ *La formule* $\underline{\mathsf{Prouvable}}_{\mathsf{T}}(\mathsf{x})$ *code la prouvabilité à partir de* T *: pour toute formule close* Φ*, il y a équivalence entre* $\mathsf{T} \vdash \Phi$ *et* $(\mathbb{N}, 0, S, +, \cdot, \leqslant) \models \underline{\mathsf{Prouvable}}_{\mathsf{T}}(\ulcorner \Phi \urcorner)$*. En revanche, dans un modèle non-standard de* $\mathsf{PA}_{\text{faible}}$*, les éléments non-standards peuvent coder des preuves ne correspondant à aucune preuve standard, et donc il se peut que* $\underline{\mathsf{Prouvable}}_{\mathsf{T}}(\ulcorner \Phi \urcorner)$ *y soit satisfaite sans que* Φ *soit prouvable.* ◁

4.4.3.— On va établir la forme forte du premier théorème d'incomplétude de Gödel, qui se distingue du résultat de 4.3.6 en ce que, cette fois, on exhibe une formule vraie dans $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ mais non prouvable à partir de T. Pour énoncer le résultat complet, on introduit un renforcement de la notion usuelle de consistance d'une théorie.

Définition (ω-consistance faible).— Pour Σ incluant Σ_{arith}, une théorie consistante T de $\mathcal{L}_\Sigma$ est dite *faiblement ω-consistante* si, quand T prouve $\neg\Phi(\mathsf{S}^{\mathsf{n}}\mathsf{0})$ pour tout n, alors T ne prouve pas $\exists \mathsf{x}\,(\Phi(\mathsf{x}))$.

▷ *Appelons* ω-règle *l'opération (infinitaire) consistant à déduire* $\forall x\,(\Phi(x))$ *de la conjonction des formules* $\Phi(0)$, $\Phi(S0)$, $\Phi(S^2 0)$ *... et notons* $\vdash_\omega$ *la relation de prouvabilité obtenue en ajoutant la* ω*-règle aux règles de la logique du premier ordre. La* ω*-consistance faible d'une théorie* T *est un cas particulier de la consistance vis-à-vis de la* ω*-logique (« *ω*-consistance »), à savoir la non-existence de* Φ *telle que l'on ait à la fois* $\mathsf{T} \vdash_\omega \Phi$ *et* $\mathsf{T} \vdash_\omega \neg\Phi$. ◁

4.4.4.— Le premier théorème d'incomplétude affirme que, pour T convenable, la formule Δ_T de 4.4.2 est vraie mais non prouvable à partir de T.

▷ *L'appellation « incomplétude » s'oppose à la « complétude » du théorème du chapitre* VII. *Ce sont des notions distinctes : la complétude affirme qu'une formule satisfaite dans tout modèle d'une théorie* T *est prouvable à partir de* T *; l'incomplétude affirme que* Δ_T *est satisfaite dans le modèle* $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ *de* T *et non prouvable à partir de* T. *Il n'y a pas de contradiction :* Δ_T *est satisfaite dans un modèle de* T*, mais pas dans tous les modèles.* ◁

Proposition (premier théorème d'incomplétude).— *Supposons que* T *est une théorie récursive consistante incluant* $\mathsf{PA}_{\text{faible}}$.

(i) *Si* $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ *est modèle de* T*, alors* Δ_T *est satisfaite dans* $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ *;*

(ii) *La théorie* T *ne prouve pas* Δ_T *;*

(iii) *Si, de plus,* T *est faiblement* ω*-consistante, alors* T *ne prouve pas* $\neg\Delta_\mathsf{T}$.

Démonstration. Dans toute la suite, on écrit Δ pour Δ_T, et on pose $\mathsf{n} := \ulcorner\Delta\urcorner$. Par le lemme diagonal (4.1.3) on a alors

$$\mathsf{PA}_{\text{faible}} \vdash \underline{\mathsf{Prouvable}}_\mathsf{T}(S^\mathsf{n}0) \Leftrightarrow \neg\Delta. \qquad (\#22)$$

(i) Supposons $(\mathbb{N}, 0, S, +, \cdot, \leqslant) \not\models \Delta$. Par $(\#22)$, $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ satisfait alors $\underline{\mathsf{Prouvable}}_\mathsf{T}(S^\mathsf{n}0)$, et il existe un entier p vérifiant $(\mathbb{N}, 0, S, +, \cdot, \leqslant) \models \underline{\mathsf{Preuve}}_\mathsf{T}(S^\mathsf{n}0, \mathsf{p})$. Par construction, cet entier p est le numéro d'une preuve de Δ à partir de T, et on a donc $\mathsf{T} \vdash \Delta$, d'où $(\mathbb{N}, 0, S, +, \cdot, \leqslant) \models \Delta$ puisque l'on suppose $(\mathbb{N}, 0, S, +, \cdot, \leqslant) \models \mathsf{T}$, contredisant l'hypothèse. Donc, $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ doit satisfaire Δ.

(ii) Supposons $\mathsf{T} \vdash \Delta$. Soit p le numéro d'une preuve de Δ à partir de T. La relation $\mathsf{Preuve}_\mathsf{T}(\mathsf{n}, \mathsf{p})$ est alors vraie dans $\mathbb{N}$, donc, puisque la formule $\underline{\mathsf{Preuve}}_\mathsf{T}$ représente la relation $\mathsf{Preuve}_\mathsf{T}$ dans $\mathsf{PA}_{\text{faible}}$, le système $\mathsf{PA}_{\text{faible}}$ prouve $\underline{\mathsf{Preuve}}_\mathsf{T}(S^\mathsf{n}0, S^\mathsf{p}0)$, donc, *a fortiori*, $\exists y(\underline{\mathsf{Preuve}}_\mathsf{T}(S^\mathsf{n}0, y))$, c'est-à-dire $\underline{\mathsf{Prouvable}}_\mathsf{T}(S^\mathsf{n}0)$. Par $(\#22)$, on en déduit que $\mathsf{PA}_{\text{faible}}$, donc *a fortiori* T, prouvent $\neg\Delta$, et, de là, que T prouve à la fois Δ et $\neg\Delta$, contredisant l'hypothèse que T est consistante. Donc,T ne prouve pas Δ.

(iii) Supposons $\mathsf{T} \vdash \neg\Delta$. Toujours par $(\#22)$, on déduit $\mathsf{T} \vdash \underline{\mathsf{Prouvable}}_\mathsf{T}(S^\mathsf{n}0)$, c'est-à-dire $\exists y(\underline{\mathsf{Preuve}}_\mathsf{T}(S^\mathsf{n}0, y))$. D'après (ii), T ne prouve pas Δ, et, par conséquent, il ne peut exister aucun entier naturel p tel que p soit le numéro d'une preuve de Δ à partir de T. Pour chaque entier naturel p, la relation $\mathsf{Preuve}_\mathsf{T}(\mathsf{n}, \mathsf{p})$ est donc fausse, et, de là, par définition de la représentabilité, $\mathsf{PA}_{\text{faible}}$, donc *a fortiori* T, prouvent $\neg\underline{\mathsf{Preuve}}_\mathsf{T}(S^\mathsf{n}0, S^\mathsf{p}0)$ pour tout p. Puisque T prouve $\exists y\,(\underline{\mathsf{Preuve}}_\mathsf{T}(S^\mathsf{n}0, y))$, l'hypothèse de ω-consistance faible est contredite. Donc,T ne prouve pas $\neg\Delta$. □

▷ *Dans la démonstration ci-dessus, la dissymétrie entre* Δ *et* $\neg\Delta$ *provient de l'impossibilité de passer directement de* $\mathsf{T} \vdash \underline{\mathsf{Prouvable}}_\mathsf{T}(\ulcorner\Phi\urcorner)$ *à* $\mathsf{T} \vdash \Phi$*, forçant à utiliser l'hypothèse,* a priori *plus forte, que* T *est* ω*-consistante. En effet, la relation* $\mathsf{T} \vdash \underline{\mathsf{Preuve}}_\mathsf{T}(S^{\ulcorner\Phi\urcorner}0, S^\mathsf{p}0)$ *pour un entier (standard)* p *implique* $\mathsf{T} \vdash \Phi$*, mais*

la relation $\mathsf{T} \vdash \exists \mathsf{y}\,(\underline{\mathsf{Preuve}}_{\mathsf{T}}(\mathsf{S}^{\ulcorner\Phi\urcorner}\mathsf{0}, \mathsf{y}))$*, elle, ne garantit pas l'existence d'un tel entier : si* $\mathcal{M}$ *est un modèle de* T*, il existe* b *dans le domaine de* $\mathcal{M}$ *tel que* $\mathcal{M}$ *vérifie* $\underline{\mathsf{Preuve}}_{\mathsf{T}}(\mathsf{S}^{\ulcorner\Phi\urcorner}\mathsf{0}, b)$*, mais, si* b *est non-standard, c'est-à-dire distinct de* $(\mathsf{S}^{\mathsf{p}}\mathsf{0})^{\mathcal{M}}$ *pour tout* p*, on ne peut conclure. Signalons ici que, par un procédé voisin de celui utilisé pour passer de* Ψ *à* B *dans la démonstration du* §3.3.7*, on peut définir une formule* $\underline{\mathsf{Prouvable}}'_{\mathsf{T}}$ *telle que, si* Δ'_{T} *est la formule déduite par le lemme diagonal, alors* T *ne prouve ni* Δ'_{T}*, ni* $\neg\Delta'_{\mathsf{T}}$ *sous l'hypothèse que* T *est consistante, et non nécessairement faiblement* ω*-consistante.*

Par ailleurs, lorsque (i) *s'applique, il est clair que* T *ne prouve pas* $\neg\Delta_{\mathsf{T}}$*, puisque* Δ_{T} *est satisfaite dans* $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$*. Mais cela ne vaut que si* $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ *est modèle de* T*, autrement dit si* T *a un modèle standard, ce que ne réclame pas la démonstration de* (iii). ◁

4.4.5.— Par le premier théorème d'incomplétude, aucune théorie récursive T incluse dans $\mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot, \leqslant)$, autrement dit aucune axiomatisation de $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ récursive et du premier ordre [10], ne prouve la formule de Gödel Δ_{T} associée et donc ne peut être complète. On obtient ainsi en particulier une réponse négative aux questions de VII.3.3.8.

Corollaire (incomplétude I).— *Si* PA_1 *est consistant, la formule de Gödel* Δ_{PA_1} *est une formule d'arithmétique vraie dans* $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ *mais non prouvable à partir de* PA_1*.*

▷ *Ce résultat contredit l'intuition superficielle que le système de Peano* PA_1 *(ou tout autre système récursif du premier ordre) puisse décrire l'arithmétique des nombres entiers de façon exhaustive. Les limites qu'il trace sont infranchissables, et toute tentative de contournement serait vaine.* ◁

4.4.6.— On montrera plus loin que le passage de l'arithmétique de Peano à la théorie des ensembles de Zermelo–Fraenkel renforce strictement le cadre axiomatique : le système ZFC prouve davantage d'énoncés d'arithmétique que le système PA_1. Dès lors, on pourrait penser que le système ZFC donne une description complète des nombres entiers, et que sa propre incomplétude ne se manifeste qu'à un niveau supérieur. Le premier théorème d'incomplétude appliqué à ZFC montre directement que ce n'est pas le cas.

Corollaire (incomplétude II).— *Si le système* ZFC *est consistant, la formule de Gödel* Δ_{ZFC} *est une formule d'arithmétique qui est vraie dans la structure* $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ *mais non prouvable à partir de* ZFC*.*

▷ *Ainsi, même au niveau des nombres entiers, la description du monde mathématique fournie par le système* ZFC*, et, de même, par tout autre système axiomatique, est incomplète.* ◁

4.4.7.— Pour toute théorie récursive T, la formule de Gödel Δ_{T} est effective : avec suffisamment de courage, on pourrait l'écrire explicitement. Pour

10. Noter l'importance de chaque hypothèse : que T soit récursive écarte $\mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot, \leqslant)$, qui est complète par définition ; que T soit du premier ordre écarte PA, qui est récursive et caractérise $(\mathbb{N}, 0, S, +, \cdot)$, donc est complète.

autant, Δ_{T} est absconse et éloignée des énoncés usuellement considérés en théorie des nombres, et on peut se demander si l'incomplétude se manifeste pour des formules plus simples. On mentionnera deux résultats dans cette direction. L'un est un corollaire du théorème MRDP (4.3.5).

Proposition (incomplétude III).— *Pour toute théorie récursive* T *incluse dans* $\mathsf{Th}_1(\mathbb{N}, 0, S, +, \cdot, \leqslant)$*, il existe une équation diophantienne explicite qui n'a pas de zéro dans* $\mathbb{N}$ *mais telle que* T *ne prouve pas ce fait.*

4.4.8.— L'autre résultat concerne le théorème de Goodstein (II.4.2.4).

▷ *A priori, l'énoncé de* II.4.2.4 *n'est pas exprimé dans l'arithmétique* $\mathcal{L}_{\text{arith}+}$ *puisqu'il requiert des exponentiations itérées. Mais, exprimé sous la forme*

$$\forall \mathsf{k}\, \exists \mathsf{n}\, (\text{« la suite partant de } \mathsf{k} \text{ atteint la valeur } 0 \text{ après } \mathsf{n} \text{ étapes »}),$$

il peut être traduit en une formule de $\mathcal{L}_{\text{arith}}$ *car l'exponentielle, puis la fonction « écrire en base* p *itérée et remplacer* p *par* q *», sont récursives, donc définissables dans* $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ *(voir exercice* 54*).* ◁

Proposition (non-prouvabilité).— *Le théorème de Goodstein ne peut pas être établi à partir des axiomes de* PA_1*.*

▷ *Initialement établi par L. Kirby et J. Paris en 1983 par la méthode des indicatrices développée par J. Paris et L. Harrington dans* [4]*, le résultat peut aussi être démontré simplement en établissant que, si* $T(d)$ *est le nombre d'étapes pour que la suite de Goodstein à partir de* d *atteigne* 0*, alors la fonction* T *croît plus vite que toutes les fonctions pouvant être prouvées totales dans* PA_1*, la hiérarchie de ces dernières étant explicitement connue* [11]*. Cela ne démontre* pas *que la démonstration de la section* II.4 *fondée sur les ordinaux est la seule possible, mais montre qu'aucune démonstration fondée sur l'induction sur* $\mathbb{N}$ *ne peut suffire. De fait, pour chaque entier* d*, il peut exister dans* PA_1 *une preuve de convergence pour la suite de Goodstein partant de* d*, mais ce qui est affirmé est qu'il n'existe pas de preuve uniforme valable pour tous les entiers* d *à la fois : concaténer des preuves pour* $d := 0$*,* $d := 1$*, etc. ne donne pas une suite finie d'énoncés.* ◁

4.5. Le second théorème d'incomplétude de Gödel

▶ Résumé.— Une théorie incluant le système de Zermelo ne peut prouver sa propre consistance. Aucune axiomatisation effective ne peut établir la cohérence des mathématiques. ◀

4.5.1.— On termine avec le second théorème d'incomplétude, qui fournit un nouvel énoncé non démontrable dans une théorie T convenable, à savoir l'énoncé (qui n'est unique qu'au choix des représentations près) qui code la consistance de la théorie T. Ce résultat est important pour la théorie des ensembles, car il anéantit l'espoir d'une théorie fondationnelle prouvablement consistante et, comme on verra dans les chapitres XIII et XIV, place des bornes infranchissables sur le statut des axiomes de grands cardinaux.

4.5.2.— Pour énoncer le résultat, on a d'abord besoin d'arithmétiser la notion de consistance d'une théorie récursive T.

Définition (consistance).— (i) On fixe une formule $\underline{\mathsf{Opposé}}(\mathsf{x},\mathsf{y})$ de complexité Σ_1 représentant dans $\mathsf{PA}_{\text{faible}}$ la relation $\mathsf{Opposé}$ sur $\mathbb{N}^2$ telle que $\mathsf{Opposé}(\mathsf{p},\mathsf{q})$ est vraie si l'on a $\mathsf{p} = \ulcorner\Phi\urcorner$ et $\mathsf{q} = \ulcorner\neg\Phi\urcorner$ avec Φ formule close.

(ii) Pour T théorie de $\mathcal{L}_{\max}$, on note Cons_T la formule close

$$\neg\exists \mathsf{x},\mathsf{y}\,(\underline{\mathsf{Prouvable}}_\mathsf{T}(\mathsf{x}) \wedge \underline{\mathsf{Prouvable}}_\mathsf{T}(\mathsf{y}) \wedge \underline{\mathsf{Opposé}}(\mathsf{x},\mathsf{y})). \qquad (\#23)$$

▷ *La définition ci-dessus est légitime : par les résultats de la section 2, la relation* $\mathsf{Opposé}$ *est récursive, donc* Σ_1*-représentable dans* $\mathsf{PA}_{\text{faible}}$*. La formule* Cons_T *exprime que l'on ne peut pas prouver à partir de* T *à la fois une chose et son contraire, et elle est donc une contrepartie formelle de la notion de consistance. La formule* Cons_T *est satisfaite dans* $(\mathbb{N},0,S,+,\cdot,\leqslant)$ *si, et seulement si,* T *est consistante donc, en* ω*-logique, établir que* T *est consistante équivaut à démontrer que* Cons_T *est prouvable. En revanche,* Cons_T *peut être fausse dans un modèle non-standard sans que les éléments dont l'existence est affirmée codent une contradiction de* T*.* ◁

4.5.3.— La formule Cons_T traduit la consistance de la théorie T au niveau des entiers. Le résultat suivant est donc naturel.

Lemme.— *Si* Σ *est incluse dans* $\Sigma_{\max}$ *et si* T *est une théorie récursive consistante de* $\mathcal{L}_\Sigma$*, alors* $(\mathbb{N},0,S,+,\cdot)$ *satisfait* Cons_T*.*

Démonstration. Supposons $(\mathbb{N},0,S,+,\cdot) \not\models \mathsf{Cons}_\mathsf{T}$. Il existe donc deux entiers p,q tels que $(\mathbb{N},0,S,+,\cdot)$ satisfait $\underline{\mathsf{Prouvable}}_\mathsf{T}(\mathsf{p})$, $\underline{\mathsf{Prouvable}}_\mathsf{T}(\mathsf{q})$, et $\underline{\mathsf{Opposé}}(\mathsf{p},\mathsf{q})$. Puisque la formule $\underline{\mathsf{Opposé}}(\mathsf{x},\mathsf{y})$ représente la relation $\mathsf{Opposé}$, et que, dans $(\mathbb{N},0,S,+,\cdot)$, l'entier p est l'interprétation du terme $\mathsf{S}^\mathsf{p}0$, et de même pour q, il existe deux formules Φ, Ψ de $\mathcal{L}_\Sigma$ telles que p et q sont les numéros de Φ et Ψ, et on a soit $\Psi = \neg\Phi$, soit $\Phi = \neg\Psi$. D'autre part, par définition, il existe deux entiers n, m tels que $(\mathbb{N},0,S,+,\cdot)$ satisfait $\underline{\mathsf{Preuve}}_\mathsf{T}(\mathsf{p},\mathsf{n})$ et $\underline{\mathsf{Preuve}}_\mathsf{T}(\mathsf{q},\mathsf{m})$. Alors, n et m sont les numéros de deux suites finies de formules de $\mathcal{L}_\Sigma$ et, puisque la formule $\underline{\mathsf{Preuve}}(\mathsf{x},\mathsf{y})$ représente la relation $\mathsf{Preuve}_\mathsf{T}$, et que n et m sont les interprétations de $\mathsf{S}^\mathsf{n}0$ et $\mathsf{S}^\mathsf{m}0$, les relations $\mathsf{Preuve}_\mathsf{T}(\ulcorner\Phi\urcorner,\mathsf{n})$ et $\mathsf{Preuve}_\mathsf{T}(\ulcorner\Psi\urcorner,\mathsf{m})$ sont vérifiées, et on a donc $\mathsf{T}\vdash\Phi$ et $\mathsf{T}\vdash\Psi$, ce qui contredit la consistance de T puisque Φ et Ψ sont opposés. □

4.5.4.— Pour énoncer le second théorème d'incomplétude, on a besoin d'une dernière notion. Par 3.2.6, on sait que, si une formule close Φ est Σ_1 et vraie dans $(\mathbb{N},0,S,+,\cdot,\leqslant)$, alors Φ est prouvable dans $\mathsf{PA}_{\text{faible}}$, donc dans toute théorie T incluant $\mathsf{PA}_{\text{faible}}$. Il en résulte que $\underline{\mathsf{Prouvable}}_\mathsf{T}(\ulcorner\Phi\urcorner)$ est vraie dans $(\mathbb{N},0,S,+,\cdot,\leqslant)$, et, par conséquent, $(\mathbb{N},0,S,+,\cdot,\leqslant)$ satisfait

$$\Phi \Rightarrow \underline{\mathsf{Prouvable}}_\mathsf{T}(\mathsf{S}^{\ulcorner\Phi\urcorner}0) \qquad (\#24)$$

dès que Φ est une formule Σ_1.

Définition (absoluité).— Une théorie T de $\mathcal{L}_{\max}$ est dite *prouver l'absoluité des formules arithmétiques* Σ_1 si T prouve (#24) pour chaque formule close Φ de $\mathcal{L}_{\text{arith}}$ de complexité Σ_1 .

4.5.5.— Les définitions étant en place, on peut énoncer le résultat.

Proposition (second théorème d'incomplétude).— *Si* T *est une théorie récursive consistante incluant* $\mathsf{PA}_{\text{faible}}$ *et prouvant l'absoluité des formules arithmétiques* Σ_1*, alors* T *ne prouve pas* $\mathsf{Cons}_{\mathsf{T}}$*.*

Démonstration. Comme dans la démonstration du premier théorème d'incomplétude (4.4.4), soit Δ la formule de Gödel associée à T, c'est-à-dire la formule $\mathsf{Diag}(\underline{\mathsf{Prouvable}}_{\mathsf{T}})$ associée à $\underline{\mathsf{Prouvable}}_{\mathsf{T}}$ par le lemme diagonal. On sait que, par définition, la formule Δ est la négation d'une certaine formule Δ^o de complexité Σ_1.

D'après le premier théorème d'incomplétude, T ne prouve pas Δ, donc, pour montrer $\mathsf{T} \nvdash \Phi$, il suffit de montrer $\mathsf{T} \vdash \Phi \Rightarrow \Delta$, ou encore de montrer $\mathsf{T} \vdash \Delta^o \Rightarrow \neg\Phi$. On va appliquer cela à $\Phi := \mathsf{Cons}_{\mathsf{T}}$, en montrant que $\mathsf{T} \vdash \Delta^o$ implique la T-prouvabilité de deux numéros de formule opposés, à savoir $\ulcorner\Delta^o\urcorner$ et $\ulcorner\Delta\urcorner$.

D'un côté, puisque Δ^o est une formule arithmétique Σ_1, l'hypothèse que T prouve l'absoluité de ces formules entraîne, par définition,

$$\mathsf{T} \vdash \Delta^o \Rightarrow \underline{\mathsf{Prouvable}}_{\mathsf{T}}(\mathsf{S}^{\ulcorner\Delta^o\urcorner}0). \qquad (\#25)$$

D'un autre côté, par le lemme diagonal, on a la relation (#22), donc $\mathsf{PA}_{\text{faible}}$, et *a fortiori* T, prouvent $\neg\Delta \Rightarrow \underline{\mathsf{Prouvable}}_{\mathsf{T}}(\mathsf{S}^{\ulcorner\Delta\urcorner}0)$, d'où, puisque Δ est $\neg\Delta^o$,

$$\mathsf{T} \vdash \Delta^o \Rightarrow \underline{\mathsf{Prouvable}}_{\mathsf{T}}(\mathsf{S}^{\ulcorner\Delta\urcorner}0). \qquad (\#26)$$

Enfin, Δ^o et Δ sont opposées, donc $\mathsf{Opposé}(\ulcorner\Delta^o\urcorner, \ulcorner\Delta\urcorner)$ est vraie dans $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$. Puisque $\underline{\mathsf{Opposé}}(x, y)$ représente $\mathsf{Opposé}$ dans $\mathsf{PA}_{\text{faible}}$ donc dans T, on déduit

$$\mathsf{T} \vdash \underline{\mathsf{Opposé}}(\mathsf{S}^{\ulcorner\Delta^o\urcorner}0, \mathsf{S}^{\ulcorner\Delta\urcorner}0). \qquad (\#27)$$

En rapprochant (#25), (#26), et (#27), on obtient

$$\mathsf{T} \vdash \Delta^o \Rightarrow (\underline{\mathsf{Prouvable}}_{\mathsf{T}}(\mathsf{S}^{\ulcorner\Delta^o\urcorner}0) \wedge \underline{\mathsf{Prouvable}}_{\mathsf{T}}(\mathsf{S}^{\ulcorner\Delta\urcorner}0) \wedge \underline{\mathsf{Opposé}}(\mathsf{S}^{\ulcorner\Delta^o\urcorner}0, \mathsf{S}^{\ulcorner\Delta\urcorner}0)),$$

d'où $\mathsf{T} \vdash \Delta^o \Rightarrow \neg\mathsf{Cons}_{\mathsf{T}}$, qui, on l'a vu, entraîne $\mathsf{T} \nvdash \mathsf{Cons}_{\mathsf{T}}$. □

▷ *Noter la simplicité de l'argument précédent une fois la non-prouvabilité de la formule de Gödel* Δ_{T} *acquise : le seul élément supplémentaire est ici fourni directement par l'hypothèse sur l'absoluité des formules* Σ_1*, et on peut soupçonner que la véritable difficulté consiste à établir que telle ou telle théorie* T *prouve l'absoluité des formules* Σ_1*.*

Noter aussi que le second théorème d'incomplétude n'empêche pas toute démonstration de non-contradiction : rien n'interdit de montrer $\mathsf{Cons}_{\mathsf{T}}$ *dans une théorie* T' *plus forte que* T *— à commencer par* $\mathsf{T}' = \mathsf{T} + \mathsf{Cons}_{\mathsf{T}}$*. Ce qu'interdit le théorème de Gödel, c'est de trouver le point fixe d'une théorie prouvant sa propre non-contradiction.* ◁

4.5.6.— On va considérer ici le cas où T est une théorie des ensembles, typiquement Z ou ZF, et la fin de la démonstration est alors facile.

Proposition (absoluité).— *Le système de Zermelo* Z *et, a fortiori, le système de Zermelo–Fraenkel* ZFC*, prouvent l'absoluité des formules* Σ_1*.*

Démonstration. On sait par 3.2.6 que toute formule close Σ_1 vraie dans la structure $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ est prouvable à partir de $\mathsf{PA}_{\text{faible}}$. Dire qu'une théorie T prouve l'absoluité Σ_1, c'est dire que l'on peut construire une démonstration de

ce résultat à partir des axiomes de T. On a montré au chapitre III que les entiers et les ensembles d'entiers pouvaient être construits dans tout modèle d'une théorie des ensembles comme Z, et la question est de vérifier que la démonstration de 3.2.6 peut être entièrement formalisée dans le cadre de Z. Or, les ingrédients de cette démonstration sont, d'une part, le fait que les formules closes Σ_1 vraies dans $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ sont vraies dans tous les modèles de $\mathsf{PA}_{\text{faible}}$, et, d'autre part, le théorème de complétude qui permet de déduire du fait qu'une formule close est vraie dans tous les modèles de $\mathsf{PA}_{\text{faible}}$ le fait que cette formule est prouvable à partir de $\mathsf{PA}_{\text{faible}}$. Chacune des deux démonstrations se formalise sans problème dans le cadre de Z (et en particulier sans faire appel à la ω-règle) : pour l'établir, il suffit de relire la démonstration du théorème de complétude au chapitre VII en vérifiant que l'on n'introduit aucun ensemble dont l'existence ne soit garantie par Z. On notera que, la signature $\mathsf{\Sigma}_{\max}$ et les formules associées ayant été numérotées, il n'y a pas de problème de choix pour construire le modèle requis pour le théorème de complétude. □

4.5.7.— On déduit immédiatement de 4.5.5 et 4.5.6 le corollaire suivant.

Proposition (consistance de la théorie des ensembles).— *S'il est consistant, le système* Z *ne prouve pas la formule* $\mathsf{Cons_Z}$*, et, s'il est consistant, le système* ZFC *ne prouve pas la formule* $\mathsf{Cons_{ZFC}}$*. Plus généralement, si* T *est récursive, consistante, et inclut* Z*, elle ne prouve pas* $\mathsf{Cons_T}$*.*

4.5.8.— Le théorème de Gödel établit la non-prouvabilité de la formule particulière $\mathsf{Cons_T}$ traduisant la consistance de T, mais pas celle de *toute* formule traduisant cette consistance.

▷ *Si* $\mathsf{Cons_T}$ *est remplacée par une formule exprimant « il existe une preuve de* $\mathsf{\Phi}$ *et pas de preuve de plus petit numéro de* $\neg\mathsf{\Phi}$ *», alors on peut établir que la formule de consistance obtenue est, elle, prouvable.* ◁

4.5.9.— En revanche, la consistance d'une théorie peut être exprimée de plusieurs façons, et on peut facilement passer de la traduction d'une forme à celle d'une autre, pourvu qu'il s'agisse de traductions obéissant au même canevas. Typiquement, dès lors que T inclut $\mathsf{PA}_{\text{faible}}$, la consistance de T équivaut à la non-prouvabilité de $0 = 1$, c'est-à-dire de $\mathsf{0{=}S0}$. Or, on montre le résultat technique suivant.

Lemme.— *Dans le contexte de* 4.5.5*, écrivons* $\bot$ *pour* $\mathsf{0{=}S0}$*, et soit* $\mathsf{Cons'_T}$ *la formule* $\neg\underline{\mathsf{Prouvable}}_{\mathsf{T}}(\mathsf{S}^{\ulcorner\bot\urcorner}\mathsf{0})$*. Alors,* T *prouve* $\mathsf{Cons_T} \Leftrightarrow \mathsf{Cons'_T}$*.*

Démonstration. Si T prouve à la fois $\mathsf{\Phi}$ et $\neg\mathsf{\Phi}$, alors T prouve toute formule, donc en particulier $\mathsf{0{=}S0}$. En arithmétisant, on déduit que

si l'on a $\mathsf{Preuve_T}(\ulcorner\mathsf{\Phi}\urcorner, \mathsf{p})$ et $\mathsf{Preuve_T}(\ulcorner\mathsf{\Psi}\urcorner, \mathsf{q})$ et $\mathsf{Opposé}(\ulcorner\mathsf{\Phi}\urcorner, \ulcorner\mathsf{\Psi}\urcorner)$,
alors il existe r vérifiant $\mathsf{Preuve_T}(\ulcorner\bot\urcorner, \mathsf{r})$. (#28)

Mais on a plus : l'argument faisant passer de preuves de $\mathsf{\Phi}$ et $\mathsf{\Psi}$ à celle de $\mathsf{0{=}S0}$ est uniforme et ne dépend pas de ces preuves. En effet, soit $\mathsf{\Phi}_1, \ldots, \mathsf{\Phi}_n$ une preuve de

la formule $X_1 \Rightarrow (\neg X_1 \Rightarrow X_2)$ en logique booléenne. Supposant que l'on est dans le cas $\Psi = \neg\Phi$, on obtient alors une preuve de $0{=}S0$ en concaténant une preuve de Φ, une preuve de Ψ, puis $\Phi_1(X_1{\leftarrow}\Phi, X_2{\leftarrow}0{=}S0)$, ..., $\Phi_n(X_1{\leftarrow}\Phi, X_2{\leftarrow}0{=}S0)$, suivies de $\Psi \Rightarrow 0{=}S0$, et finalement $0{=}S0$. En particulier parce que la fonction de substitution est primitive récursive par la contrepartie de 2.1.5, cela implique l'existence d'une fonction f primitive récursive telle que

si l'on a $\mathsf{Preuve}_T(\ulcorner\Phi\urcorner, p)$, $\mathsf{Preuve}_T(\ulcorner\Psi\urcorner, q)$, $\mathsf{Opposé}(\ulcorner\Phi\urcorner, \ulcorner\Psi\urcorner)$, et $f(p,q) = r$, on a alors $\mathsf{Preuve}_T(\ulcorner\bot\urcorner, r)$. (#29)

Soit F une formule Σ_1 représentant fidèlement la fonction f. Parce que $\underline{\mathsf{Preuve}}_T$ représente fidèlement Preuve_T, la construction de $\underline{\mathsf{Preuve}}_T$ suit pas à pas celle de Preuve_T (ce que ne fait pas nécessairement toute formule représentant Preuve_T) et, de là, la démonstration de l'implication (#29) donnée plus haut induit pas à pas une démonstration parallèle de la formule

$$(\underline{\mathsf{Preuve}}_T(x,u) \wedge \underline{\mathsf{Preuve}}_T(y,v) \wedge \underline{\mathsf{Opposé}}(x,y) \wedge F(u,v,w)) \Rightarrow \underline{\mathsf{Preuve}}_T(S^{\ulcorner\bot\urcorner}0, w), \quad (\#30)$$

d'où, en quantifiant sur les variables x, y, u, v, w,

$$\exists x,y\,(\underline{\mathsf{Prouvable}}_T(x) \wedge \underline{\mathsf{Prouvable}}_T(y) \wedge \underline{\mathsf{Opposé}}(x,y)) \Rightarrow \underline{\mathsf{Prouvable}}_T(S^{\ulcorner\bot\urcorner}0),$$

qui, par contraposition, donne $\mathsf{Cons}'_T \Rightarrow \mathsf{Cons}_T$. On démontrerait $\mathsf{Cons}_T \Rightarrow \mathsf{Cons}'_T$ par un argument semblable. □

4.5.10.— Il est donc désormais loisible d'utiliser Cons'_T à la place de Cons_T dans 4.5.5, et on peut réénoncer 4.5.7 sous la forme suivante.

Corollaire (consistance de la théorie des ensembles).— *Si* T *est une théorie récursive, consistante, et incluant* Z*, elle ne prouve pas* Cons'_T.

▷ *Cette forme du second théorème est plus aisément compréhensible que celle de* 4.5.5. *Si* T *est consistante, alors il n'existe pas de preuve de* $0{=}1$ *dans* T *et donc, pour tout entier* n*, la relation* $\mathsf{Preuve}_T(\ulcorner\bot\urcorner, n)$ *est fausse. Arithmétisant une démonstration, on déduit que* T *prouve* $\neg\underline{\mathsf{Preuve}}_T(S^{\ulcorner\bot\urcorner}0, S^n0)$ *pour tout entier* n*. Ce que dit le second théorème d'incomplétude, c'est que, bien que* T *prouve chacune des formules* $\neg\underline{\mathsf{Preuve}}_T(S^{\ulcorner\bot\urcorner}0, S^n0)$*, elle ne prouve pas la formule* $\forall y\,(\neg\underline{\mathsf{Preuve}}_T(S^{\ulcorner\bot\urcorner}0, y))$ *: c'est un résultat analogue à celui de* 4.4.8*, où* PA_1 *peut prouver la convergence de chacune des suites de Goodstein séparément sans prouver l'énoncé* $\forall y\,(...)$*. Encore une fois, cela tient à ce qu'il n'existe pas de preuve uniforme en* n *et que la concaténation d'une suite de preuves n'est pas une preuve. Techniquement, la « faute » en revient aux possibles entiers non-standards, c'est-à-dire non de la forme* S^n0*, impossibles à exclure en logique du premier ordre. Noter que, par définition* (§4.4.3)*, si* T *est faiblement* ω*-consistante, alors, sous les hypothèses du corollaire,* T *ne prouve pas* $\neg\mathsf{Cons}'_T$. ◁

4.5.11.— Le second théorème d'incomplétude a des conséquences importantes en termes de fondations des mathématiques puisqu'il ruine les espoirs d'établir formellement la cohérence de l'édifice global.

« Proposition » (fondement des mathématiques).— *Quel que soit le cadre axiomatique $\mathcal{T}$ supposé non contradictoire, dès lors que $\mathcal{T}$ est formalisable dans une logique du premier ordre par un système récursif consistant incluant le système de Zermelo* Z*, il est impossible d'établir dans le cadre de $\mathcal{T}$ que les mathématiques sont non contradictoires.*

« Démonstration ». On a vu en III.4.2.5 que l'existence d'une représentation de tous les objets mathématiques comme ensembles purs ramène la non-contradiction de l'édifice mathématique entier à la seule non-contradiction du système ZF ou, plus généralement, d'un système T l'incluant. Moyennant l'adéquation de la prouvabilité à la notion informelle de démontrabilité (objet de consensus), et l'existence d'une arithmétisation fidèle de celle-ci (objet de démonstration), établir la non-contradiction d'un tel système T équivaut à montrer que la formule $\mathsf{Cons}_{\mathsf{T}}$ (ou la formule Cons'_T, ou tout autre formule de consistance prouvablement équivalente) est satisfaite dans $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$, et la seule façon envisageable de l'établir est de démontrer que T prouve $\mathsf{Cons}_{\mathsf{T}}$. Or, c'est précisément ce que le second théorème d'incomplétude affirme être impossible. □

On notera que l'impossibilité de montrer la cohérence globale de l'édifice, redonne de l'intérêt à considérer séparément des fragments plus modestes dont la cohérence, au moins relative, peut être établie directement.

▷ *Sans remettre en cause l'unité profonde des mathématiques, on peut souligner dès maintenant que l'approche du traité de Bourbaki* [6] *fondée sur la théorie des ensembles comme base axiomatique universelle procède d'une vision pré-Gödelienne aujourd'hui dépassée, voir section* XVI.1.2. ◁

4.5.12.— On mentionne encore, ici sans démonstration, le dernier ingrédient d'une forme forte du second théorème d'incomplétude.

Proposition (absoluité).— *Le système de Peano* PA_1 *prouve l'absoluité des formules Σ_1.*

▷ *La démonstration de ce résultat est délicate. Il s'agit de montrer à l'intérieur de tout modèle de* PA_1 *une version du théorème de complétude ; il est ici essentiel de disposer des axiomes d'induction, donc du système* PA_1 *et pas seulement de* $\mathsf{PA}_{\text{faible}}$*, mais, même avec cet outil, il reste à développer des arguments de codage plus délicats que dans le contexte de* Z*, où il s'agit d'une simple transcription de la démonstration du théorème de complétude. Notons que l'arithmétique de Robinson n'est pas suffisante ici ; on peut même douter de la signification d'un énoncé de consistance dans un système si faible qu'il ne prouve pas la commutativité de l'addition* (*exercice* 49). ◁

4.5.13.— En rapprochant de 4.5.5, on dédui(rai)t le résultat optimal.

Proposition (consistance de l'arithmétique).— *S'il est consistant, le système* PA_1 *ne prouve pas la formule* $\mathsf{Cons}_{\mathsf{PA}_1}$.

▷ *L'incertitude frustrante induite par le second théorème d'incomplétude au niveau de la théorie des ensembles se rencontre donc sous la même forme dès le niveau de l'arithmétique et du système* PA_1. ◁

4.5.14.— Pour terminer, en complément de ce qui précède, on mentionne une approche que nous n'exploiterons pas ici, mais qui dégage bien les propriétés requises de la formule $\underline{\mathsf{Prouvable}}_{\mathsf{T}}(\mathsf{x})$ et fournit un formalisme élégant. Pour toute formule Φ, notons $\Box_{\mathsf{T}}\Phi$ pour $\underline{\mathsf{Prouvable}}_{\mathsf{T}}(\mathsf{S}^{\ulcorner\Phi\urcorner}0)$. On appelle *conditions de Hilbert–Bernays–Löb* les trois assertions

(C_1) $\mathsf{T} \vdash \Phi$ entraîne $\mathsf{T} \vdash \Box_{\mathsf{T}}\Phi$,
(C_2) $\mathsf{T} \vdash \Box_{\mathsf{T}}(\Phi \Rightarrow \Psi) \Rightarrow (\Box_{\mathsf{T}}\Phi \Rightarrow \Box_{\mathsf{T}}\Psi)$,
(C_3) $\mathsf{T} \vdash \Box_{\mathsf{T}}\Phi \Rightarrow \Box_{\mathsf{T}}\Box_{\mathsf{T}}\Phi$.

Ce que nous avons fait ci-dessus est essentiellement d'établir que, $\underline{\mathsf{Prouvable}}_{\mathsf{T}}$ étant construit comme indiqué, les conditions (C_1) et (C_2) sont vérifiées quand T inclut $\mathsf{PA}_{\text{faible}}$, et ($C_3$) l'est quand T inclut Z.

Démonstration (esquisse). Supposons $\mathsf{T} \supseteq \mathsf{PA}_{\text{faible}}$. Pour ($C_1$), si T prouve Φ et si n est le numéro d'une preuve de Φ à partir de T, alors T prouve $\underline{\mathsf{Preuve}}_{\mathsf{T}}(\ulcorner\Phi\urcorner, \mathsf{S}^{\mathsf{n}}0)$, donc $\underline{\mathsf{Prouvable}}_{\mathsf{T}}(\ulcorner\Phi\urcorner)$, qui est $\Box_{\mathsf{T}}\Phi$. Pour (C_2), l'argument est celui de 4.5.9, et c'est là que l'hypothèse que les formules représentent fidèlement les fonctions et relations est nécessaire. Enfin, supposons que T prouve l'absoluité des formules Σ_1 d'arithmétique. Formaliser cette hypothèse montre que T prouve $\Psi \Rightarrow \Box_{\mathsf{T}}\Psi$ pour toute formule Σ_1 d'arithmétique Ψ. Comme $\Box_{\mathsf{T}}\Phi$ est, pour tout Φ, une formule Σ_1 d'arithmétique, on obtient (C_3). □

De ce point, il est facile d'établir le second théorème d'incomplétude — dans ce contexte, $\mathsf{Cons}'_{\mathsf{T}}$ s'énonce élégamment $\neg\Box_{\mathsf{T}}\bot$ — et divers raffinements ultérieurs, voir [109].

—==000==—

Exercices

Exercice 50 (fonctions récursives).— Soit F l'ensemble des fonctions pouvant s'obtenir par un nombre fini de compositions et de récursions à partir des fonctions constantes et des projections. Montrer que toute fonction dans F est primitive récursive. Montrer que, pour toute fonction f de $\mathbb{N}^{\mathsf{p}}$ dans $\mathbb{N}$ qui est dans F, il existe une constante C_{f} telle que, pour tous $\mathsf{n}_1, \ldots, \mathsf{n}_{\mathsf{p}}$ dans $\mathbb{N}$, on a $f(\mathsf{n}_1, \ldots, \mathsf{n}_{\mathsf{p}}) \leqslant \max(C_{\mathsf{f}}, \mathsf{n}_1, \ldots, \mathsf{n}_{\mathsf{p}})$. En déduire que F est un sous-ensemble strict de l'ensemble des fonctions primitives récursives.

Exercice 51 (codage des suites).— Partant de $\langle \mathsf{i}, \mathsf{j}\rangle := 2^{\mathsf{i}}(2\mathsf{j}+1)-1$, on code la suite vide par 0, la suite (n) par $1+\langle 0, \mathsf{n}\rangle$, et $(\mathsf{n}_1, \ldots, \mathsf{n}_{\mathsf{k}})$ par $1+\langle \mathsf{k}-1, \langle \mathsf{n}_1, \langle \mathsf{n}_2, \ldots, \langle \mathsf{n}_{\mathsf{k}-1}, \mathsf{n}_{\mathsf{k}}\rangle \cdots\rangle\rangle\rangle$. (i) Montrer que ce codage établit une bijection entre $\mathbb{N}$ et les suites finies d'entiers, et que la longueur d'une suite est la première projection de son code. (ii) Vérifier que les fonctions coord et concat correspondantes sont récursives, puis définissables dans PA^-.

Exercice 52 (semi-récursif).— Montrer que, pour $\mathsf{S} \subseteq \mathbb{N}^{\mathsf{p}}$, il y a équivalence entre : (i) S est semi-récursive ; (ii) S est MT-semi-décidable, au sens où il existe une machine de

Turing déterministe M dont le calcul à partir de l'entrée $\vec{n}$ est acceptant si $S(\vec{n})$ est vrai et ne se termine pas quand $S(\vec{n})$ est faux ; (iii) S est la projection d'une relation primitive récursive sur $\mathbb{N}^{p+1}$; (iv) Il existe une fonction récursive f dont le domaine est S ; (v) Il existe une fonction récursive totale f dont l'image est S ou bien S est vide.

Exercice 53 (système de Peano).— (i) Montrer que PA_1 prouve $\forall x, y(x+y = y+x)$. (ii) En déduire en utilisant une induction sur x que PA_1 prouve l'existence de la division euclidienne sous la forme $\forall x, y\, \exists q, r, s\, (x = S(y) \cdot q + r \wedge s + r = y)$. (iii) Montrer que la structure $\mathcal{M}$ formée par les polynômes de $\mathbb{Z}[X]$ à coefficient dominant positif munis des opérations usuelles est un modèle de $\mathsf{PA}_{\text{faible}}$. Peut-on diviser X par 2 dans $\mathcal{M}$? Qu'en déduit-on pour la formule ci-dessus ?

Exercice 54 (définissabilité).— Montrer que toute fonction récursive est définissable (VII.3.1.5) dans la structure $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$. [Suivre le même schéma que pour montrer que toute fonction récursive totale est représentable.]

Exercice 55 (couplage).— (i) Montrer que la bijection coupl de $\mathbb{N}^2$ dans $\mathbb{N}$ définie par $\mathsf{coupl}(n_1, n_2) = (n_1 + n_2)(n_1 + n_2 + 1)/2 + n_1$ est primitive récursive, de même que chacune des deux composantes coord_1, coord_2 de la bijection réciproque. (ii) En déduire l'existence d'une fonction primitive récursive frag de $\mathbb{N}^2$ dans $\mathbb{N}$ telle que, pour toute suite finie d'entiers $(n_0, \ldots, n_k)$, il existe un entier n tel que l'on ait $\mathsf{frag}(n, i) = n_i$ pour $i = 0, \ldots, k$. [Utiliser la fonction beta, et la fonction coupl pour contracter les deux premiers arguments de beta en un seul.]

Exercice 56 (récursion simultanée).— Soient g_1, g_2, h_1, h_2 des fonctions primitives récursives respectivement de $\mathbb{N}^p$, $\mathbb{N}^p$, $\mathbb{N}^{p+3}$, et $\mathbb{N}^{p+3}$ dans $\mathbb{N}$. Montrer que les fonctions f_1, f_2 de $\mathbb{N}^{p+1}$ dans $\mathbb{N}$ définies pour $i = 1, 2$, pour $k = 0$, par $f_i(\vec{n}, k) := g_i(\vec{n})$ et, pour $k > 0$, par $f_i(\vec{n}, k) := h_i(\vec{n}, k, f_1(\vec{n}, k-1), f_2(\vec{n}, k-1))$ sont primitives récursives.

Exercice 57 (consistance).— (i) Montrer que, si ZFC est consistant, alors il en est de même de la théorie $\mathsf{ZFC} + \neg\mathsf{Cons}_{\mathsf{ZFC}}$ [Utiliser le second théorème d'incomplétude]. (ii) Montrer que la théorie $\mathsf{ZFC} + \neg\mathsf{Cons}_{\mathsf{ZFC}}$ ne possède pas de modèle standard, c'est-à-dire tel que ω soit le sup des ordinaux $\underline{n}$ pour n entier naturel [Remarquer que, dans un modèle standard, la satisfaction de $\underline{\mathsf{Prouvable}}_T(S^{\ulcorner\Phi\urcorner}0)$ entraîne la prouvabilité de Φ]. (iii) Peut-on raffiner le théorème de complétude en un énoncé affirmant l'existence d'un modèle standard pour chaque théorie consistante ?

Repères chronologiques

▸ L'usage de définitions récursives remonte à R. Dedekind (1831–1916), vers 1888. La théorie des fonctions récursives a été développée par A. Church (1903–1995) dans les années 1930, en même temps que la notion de fonction MT-calculable par A. Turing (1912–1954). La description actuelle est due à S. Kleene (1909–1994).

▸ Les théorèmes d'incomplétude ont été établis par K. Gödel (1906–1978) en 1931.

▸ La non-définissabilité de la vérité a été observée par A. Tarski (1901–1983) en 1933.

▸ L'existence d'un ensemble semi-récursif non récursif a été établie par A. Turing en 1936. L'indécidabilité de la validité pour les logiques du premier ordre par A. Church.

▸ La non-prouvabilité du théorème de Goodstein à partir de l'arithmétique de Peano a été établie par L. Kirby (né en 1952) et J. Paris (né en 1944) en 1982.

Résumé du chapitre VIII

▸ Définies par récurrence à partir de fonctions de base, les fonctions primitives récursives contiennent la plupart des fonctions usuelles et sont closes par minimisation bornée.

▸ Il existe des codages des suites finies d'entiers par des entiers tels que toutes les opérations associées soient primitives récursives.

▸ Fonctions et relations récursives s'obtiennent en passant aux fonctions partielles et ajoutant la minimisation non bornée.

▸ On numérote les formules du premier ordre de sorte que les fonctions syntaxiques associées soient primitives récursives.

▸ On numérote les preuves de sorte que les formules prouvables à partir d'une théorie récursive forment un ensemble semi-récursif.

▸ Tout modèle de l'arithmétique de Robinson $\mathsf{PA}_{\text{faible}}$ est extension finale de la structure $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$.

▸ Toute formule Σ_1 satisfaite dans $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ est prouvable dans le système $\mathsf{PA}_{\text{faible}}$.

▸ Toute fonction récursive totale et toute relation récursive sur $\mathbb{N}^p$ est Σ_1-représentable dans $\mathsf{PA}_{\text{faible}}$.

▸ Le lemme diagonal est l'argument général d'autoréférence à la base des résultats négatifs de ce chapitre.

▸ Ni l'ensemble des formules d'arithmétique prouvables dans $\mathsf{PA}_{\text{faible}}$ ni l'ensemble de celles qui sont valides ne sont récursifs.

▸ Il est impossible de définir la vérité dans un modèle de $\mathsf{PA}_{\text{faible}}$. La logique du second ordre n'a pas de théorème de complétude.

▸ Pour toute axiomatisation de l'arithmétique T, il existe une formule canonique explicite vraie dans $(\mathbb{N}, 0, S, +, \cdot, \leqslant)$ et non prouvable dans T.

▸ Une théorie incluant le système de Zermelo ne peut prouver sa propre consistance. Aucune axiomatisation effective ne peut établir la cohérence des mathématiques.

—==000==—