de toutes les itérations finies du saut Turing ? Nous allons voir que non : la théorie de $(\boldsymbol{\mathcal{D}}, \leqslant)$ est de complexité *maximale*. Qu'entendons-nous par là ? Considérons T_2, la théorie du second-ordre de $(\mathbb{N}, \times, +, 0, 1)$, c'est-à-dire l'ensemble des formules du second ordre qui sont vraies dans $\mathbb{N}$. On rappelle qu'une formule du second ordre est de la forme $\forall X\ \exists Y\ \ldots F(X, Y, \ldots)$ où les variables $X, Y, \ldots$ sont des ensembles d'entiers, et où F est une formule du premier ordre, paramétrée par ces ensembles.

La théorie T_2 est donc l'ensemble des formules du second ordre qui sont vraies dans $\mathbb{N}$. Si l'on a accès à T_2, on peut savoir si une formule de la théorie du premier ordre de $(\boldsymbol{\mathcal{D}}, \leqslant)$ est vraie : les quantifications $\exists \mathbf{a}$ et $\forall \mathbf{a}$ peuvent se remplacer par des quantifications sur les éléments de $2^{\mathbb{N}}$ et, à l'aide du théorème 9-3.4 relativisé aux paramètres du second ordre — lequel permet de transformer un prédicat $\Sigma^0_n(X, Y, \ldots)$ en une formule Σ_n de l'arithmétique —, on peut transformer une formule du premier ordre F de $(\boldsymbol{\mathcal{D}}, \leqslant)$ en une formule de second ordre équivalente F^* de $(\mathbb{N}, \times, +, 0, 1)$. Simpson a montré que l'inverse était vrai aussi : via un codage ingénieux, il est possible de transformer une formule de l'arithmétique du second ordre en une formule équivalente de $(\boldsymbol{\mathcal{D}}, \leqslant)$.

Insistons avant d'aller plus loin sur la complexité *extrême* de T_2. Nous étudierons dans la partie IV tous les détails de la complexité de T_2 restreinte aux formules Π^1_1, c'est-à-dire restreinte aux formules au sein desquelles les quantifications du second ordre sont toutes universelles. Nous verrons que cette théorie a déjà un degré Turing considérablement élevé comparé à $\emptyset'$ ou même à toutes les itérations finies de $\emptyset'$. Ce degré Turing est néanmoins bien défini, et il est *absolu* dans le sens où la valeur de vérité d'une formule Π^1_1 sera la même dans les modèles transitifs de la théorie des ensembles partageant les mêmes ordinaux calculables (voir la partie IV pour une définition formelle). À partir du niveau de complexité Π^1_2 des formules, ce sens devient plus flou. La valeur de vérité de ce genre de formule restera toutefois inchangée dans tous les modèles transitifs de la théorie des ensembles qui partagent cette fois non pas les même ordinaux calculables, mais les même ordinaux dénombrables. Sous réserve d'accepter l'absoluité des ordinaux dénombrables, la vérité des formules Π^1_2 est elle aussi absolue. Le degré Turing du niveau Π^1_2 de T_2 est quant à lui plus élevé que tous les degrés Turing abordés dans le présent livre (il s'agit en quelque sorte du supremum de tous les singletons Π^1_1, dont nous verrons la définition dans la section 30-4). La valeur de vérité d'une formule Π^1_3 pourra quant à elle différer entre deux modèles de ZFC qui partagent les mêmes ordinaux, et le sens qu'il y a à dire qu'une telle formule est *vraie* ou *fausse* s'évanouit ici encore un peu plus. Quant au degré Turing de la théorie Π^1_3 de T_2, de là où nous sommes, c'est-à-dire le monde des choses calculables, depuis lequel

nous observons la structure de l'univers, même les meilleurs télescopes ne permettent pas de le voir : il est tout simplement trop éloigné de nous.

Voilà donc la complexité de la théorie des degrés Turing ! Nous livrons ci-après la preuve moderne du théorème de Simpson, qui diffère de celle qui fut originellement produite, et qui présente son intérêt propre. Le résultat de Simpson découle du théorème suivant, où $n \in \mathbb{N}$..

Théorème 4.4 (Slaman et Woodin [206])
Tout sous-ensemble dénombrable $R \subseteq \boldsymbol{\mathcal{D}}^n$ de n-uplets de degrés Turing est uniformément définissable dans $(\boldsymbol{\mathcal{D}}, \leqslant)$ avec un nombre fini de paramètres. Formellement, il existe une formule $F(x_1, \ldots, x_n, y_1, \ldots, y_m)$ telle que, pour tout $R \subseteq \boldsymbol{\mathcal{D}}^n$, il existe des paramètres $\mathbf{p}_1, \ldots, \mathbf{p}_m \in \boldsymbol{\mathcal{D}}$ pour lesquels $(\mathbf{a}_1, \ldots, \mathbf{a}_n) \in \boldsymbol{\mathcal{D}}$ si, et seulement si, $F(\mathbf{a}_1, \ldots, \mathbf{a}_n, \mathbf{p}_1, \ldots, \mathbf{p}_m)$ est vrai dans $(\boldsymbol{\mathcal{D}}, \leqslant)$.

Voyons tout de suite comment utiliser le théorème 4.4 pour montrer le théorème de Simpson : il suffit de coder un modèle standard de l'arithmétique dans les degrés Turing.

Théorème 4.5 (Simpson [201])
La théorie du premier ordre des degrés Turing est many-one équivalente à celle de l'arithmétique du second ordre.

Preuve. Nous avons déjà vu dans les paragraphes précédents comment transformer un énoncé de $(\boldsymbol{\mathcal{D}}, \leqslant)$ en un énoncé équivalent de l'arithmétique du second ordre. Voyons à présent comment faire l'inverse.

L'idée est de coder un modèle standard de $(\mathbb{N}, +, \times, 0, 1)$ dans les degrés Turing. Un tel modèle sera codé par un ensemble fini de paramètres codant pour un ensemble N de degrés Turing qui représentent $\mathbb{N}$, avec un degré spécifique représentant 0 et un autre représentant 1. Les relations $+$ et $\times$ sont elles aussi codées par un ensemble fini de paramètres.

Il est possible de créer une formule de $\boldsymbol{\mathcal{D}}$ qui vérifie si un ensemble fini de paramètres code bien pour le modèle standard de l'arithmétique : il s'agit simplement de vérifier les axiomes de l'arithmétique de Robinson (voir la section 9-2.3), lesquels sont en nombre fini, et de vérifier ensuite que tout sous-ensemble du modèle possède un plus petit élément. On peut se reporter au théorème 9-3.13 pour voir que ces conditions sont nécessaires et suffisantes pour vérifier que l'on a bien affaire au modèle standard des entiers. La quantification universelle « tout sous-ensemble du modèle possède un plus petit élément » peut être remplacée par une quantification universelle sur les degrés Turing utilisés comme paramètres pour coder des sous-ensembles de N (notre ensemble de degrés qui représente $\mathbb{N}$).

Étant donné une formule F de l'arithmétique du second ordre, on peut finalement la transformer en une formule équivalente F^* dans $\boldsymbol{\mathcal{D}}$, en remplaçant les quantifications sur les ensembles par des quantifications sur les paramètres codant pour ces ensembles. La formule du second ordre de l'arithmétique sera donc interprétée dans $\boldsymbol{\mathcal{D}}$ par la formule : il existe des paramètres codant pour un modèle standard de l'arithmétique, tel que F^* est vérifiée dans ce modèle. ∎

Passons à présent au codage de Slaman et Woodin. Le lemme qui suit constitue la partie difficile de la preuve. Il repose sur un forcing qui peut sembler relativement simple dans son principe, mais dont l'exécution s'avère délicate et demande pas mal d'astuce pour être menée à bien.

Lemme 4.6 (Slaman et Woodin [206]). Toute anti-chaîne dénombrable dans les degrés Turing est uniformément définissable avec trois paramètres. ⋆

Preuve. Soit $(\mathbf{a}_n)_{n\in\mathbb{N}}$ une anti-chaîne dans les degrés Turing, et soit $\mathbf{b}$ un majorant de cette anti-chaîne. Nous allons définir deux degrés $\mathbf{g}_0, \mathbf{g}_1$ tels que pour tout degré $\mathbf{y} \leqslant \mathbf{b}$ ne majorant aucun $\mathbf{a}_i$, alors $\mathbf{g}_0 \cup \mathbf{y}$ et $\mathbf{g}_1 \cup \mathbf{y}$ ont une borne inférieure, et cette borne inférieure est $\mathbf{y}$. En d'autre termes, tout degré à la fois sous $\mathbf{g}_0 \cup \mathbf{y}$ et sous $\mathbf{g}_1 \cup \mathbf{y}$ doit aussi être sous $\mathbf{y}$. À l'inverse, il existera pour tout i un degré à la fois sous $\mathbf{g}_0 \cup \mathbf{a}_i$ et sous $\mathbf{g}_1 \cup \mathbf{a}_i$ qui ne sera pas sous $\mathbf{a}_i$. Il s'ensuit que chaque $\mathbf{a}_i$ sera un élément minimal satisfaisant la formule

$$F(\mathbf{x}) = \mathbf{x} \leqslant \mathbf{b} \ \wedge \exists \mathbf{c}\ (\mathbf{c} \not\leqslant \mathbf{x} \ \wedge \ \mathbf{c} \leqslant \mathbf{g}_0 \cup \mathbf{x} \ \wedge \ \mathbf{c} \leqslant \mathbf{g}_1 \cup \mathbf{x}).$$

En particulier, les degrés $\mathbf{a}_i$ seront exactement les degrés $\mathbf{x}$ satisfaisant la formule $F(\mathbf{x}) \wedge \forall \mathbf{y} \leqslant \mathbf{x}\ \neg F(\mathbf{y})$. Le fait que les $\mathbf{a}_i$ forment une anti-chaîne est utilisé uniquement pour les définir comme solutions minimales de F, mais n'intervient plus par la suite.

Nous utiliserons pour la construction des degrés $\mathbf{g}_0$ et $\mathbf{g}_1$ le fait suivant : tout degré Turing contient un ensemble X calculable en n'importe quel sous-ensemble infini de X. On peut le voir de la manière suivante : étant donné un ensemble Y quelconque, on définit X comme étant l'ensemble des préfixes $\sigma \prec Y$, via un codage des chaînes finies par des entiers.

Soit B un représentant de $\mathbf{b}$ et, pour tout n, soit A_n un représentant de $\mathbf{a}_n$ calculable en n'importe lequel de ses sous-ensembles infinis. Nous allons définir deux ensembles G_0, G_1 tels que pour tout i il existe $C \not\leqslant_T A_i$ tel que $C \leqslant_T G_0 \oplus A_i$ et $C \leqslant_T G_1 \oplus A_i$, et tel que pour tout $Y \leqslant_T B$ et tout D tel que $D \leqslant_T G_0 \oplus Y$ et $D \leqslant_T G_1 \oplus Y$, alors $Y \geqslant_T D$ ou bien $Y \geqslant_T A_j$ pour un certain j.

On procède à cet effet via un forcing qui présente des similarités avec celui du théorème 2.7.

Soit $\mathbb{P}$ l'ensemble de conditions de la forme (σ_0, σ_1, n) pour $\sigma_0, \sigma_1 \in 2^{<\mathbb{N}}$, avec $|\sigma_0| = |\sigma_1|$ et $n \in \mathbb{N}$. L'entier n sert à restreindre les extensions possibles, la chaîne σ_0 est utilisée pour le premier générique G_0 et la chaîne σ_1 pour le deuxième générique G_1. Tout comme dans la preuve du théorème 2.7, on peut voir G_0 et G_1 comme étant construits par colonne. L'entier n indique que la construction sera dorénavant restreinte sur les n premières colonnes : pour une colonne $k \leqslant n$, si $a \notin A_k$, il n'y a alors aucune restriction pour le bit $\langle k, a\rangle$ des deux génériques. Si en revanche $a \in A_k$, alors le bit $\langle k, a\rangle$ des deux génériques doit être identique (sans nécessairement être égal à $A_k(a)$).
Formellement, $(\sigma_0, \sigma_1, n) \succeq (\tau_0, \tau_1, m)$ si $\sigma_0 \preceq \tau_0$, si $\sigma_1 \preceq \tau_1$, si $n \leqslant m$, et si de plus la condition suivante est vérifiée : pour tout $k \leqslant n$, alors pour tout $a \in A_k$ tel que $|\sigma_i| \leqslant \langle k, a\rangle < |\tau_i|$, les valeurs $\tau_0(\langle k, a\rangle)$ et $\tau_1(\langle k, a\rangle)$ doivent être les mêmes.

Considérons G_0, G_1 deux ensembles suffisamment génériques pour ce forcing. Pour chaque A_n, et pour $a \in A_n$ suffisamment grand, on aura

$$G_0(\langle n, a\rangle) = G_1(\langle n, a\rangle),$$

par définition de ce qu'est une extension valide dans ce forcing. En particulier ; les ensembles $X_0^n, X_1^n \subseteq A_n$ définis par $X_i^n(a) = 0$ si $a \notin A_n$, et $X_i^n(a) = G_i(\langle n, a\rangle)$ sinon, sont les mêmes sauf pour un nombre fini de bits, et sont donc tous les deux calculés par $G_0 \oplus A_n$ et $G_1 \oplus A_n$.

Montrons que si G_0, G_1 sont suffisamment génériques, alors aucun A_n ne peut calculer les ensembles X_0^n, X_1^n ainsi définis. Étant donné une condition $\langle \sigma_0, \sigma_1, n\rangle$, on peut prendre n'importe quelle extension pour le côté σ_0, ce qui force alors certains bits de l'extension pour l'autre côté. En considérant le fait qu'il y a nécessairement des sous-ensembles infinis de A_n non calculables en A_n, on peut nécessairement trouver une extension $\tau_0 \succeq \sigma_0$ telle que pour une fonctionnelle Φ_e donnée, $\Phi_e(A_n)$ ne produise jamais la restriction de τ_0 qui sera faite pour en faire un préfixe de X_0^n — soit parce que $\Phi_e(A_n)$ sera partielle, soit parce qu'elle produira une chaîne incompatible avec le préfixe de X_0^n ainsi forcé. Comme X_1^n coïncide avec X_0^n sauf sur un nombre fini de bits, alors A_n ne calculera pas non plus X_1^n. Cela nous donne la première partie de ce que l'on cherche à montrer : pour tout A_n, il existe un ensemble calculable en $G_0 \oplus A_n$ et en $G_1 \oplus A_n$, mais pas en A_n.

Il reste à montrer que pour tout $Y \leqslant_T B$ tel que Y ne calcule aucun A_n, si $G_0 \oplus Y$ et $G_1 \oplus Y$ calculent un même ensemble C, alors $Y \geqslant_T C$. Soit alors $p = (\sigma_0, \sigma_1, n)$ une condition et soient Φ_{e_0}, Φ_{e_1} une paire de fonctionnelles. On sépare dans un premier temps la chaîne σ_0 de notre condition p. S'il existe x et $\tau_0 \succeq \sigma_0$ tels que pour tout $\rho_0 \succeq \tau_0$ on a $\Phi_{e_0}(Y \oplus \rho_0, x)\uparrow$, on considère alors une chaîne τ_1 telle que la condition (τ_0, τ_1, n) forme une extension valide, pour laquelle on aura forcé la partialité de $\Phi_{e_0}(Y \oplus G_0)$.

Supposons à présent que pour tout x et pour tout $\tau_0 \succeq \sigma_0$ il existe $\rho_0 \succeq \tau_0$ tel que $\Phi_{e_0}(Y \oplus \rho_0, x) \downarrow$. Supposons dans un premier temps qu'il existe une extension $\tau \succeq \sigma_0$ telle que, pour toutes extensions $\rho_0, \rho_1 \succeq \tau$ et pour tout x, on ait $\Phi_{e_0}(Y \oplus \rho_0, x) \downarrow = a$ et $\Phi_{e_0}(Y \oplus \rho_1, x) \downarrow = b$ impliquent $a = b$. Alors, on ne peut produire qu'un unique ensemble via $\Phi_{e_0}(Y \oplus G_0)$ pour un générique G_0 quelconque, et cet ensemble est calculable alors en Y. On force donc $\Phi_{e_0}(Y \oplus G_0)$ à calculer quelque chose qui est déjà calculable en Y. Supposons finalement que, pour tout $\tau \succeq \sigma_0$, il existe $\rho_0, \rho_1 \succeq \tau$ et x tels que $\Phi_{e_0}(Y \oplus \rho_0, x) \downarrow \neq \Phi_{e_0}(Y \oplus \rho_1, x) \downarrow$. Le lemme suivant s'avère utile.

Lemme 4.7. Pour tout $\tau \succeq \sigma$, il existe x et deux extensions $\rho_0, \rho_1 \succeq \tau$ qui diffèrent sur seulement un bit et tels que

$$\Phi_{e_0}(Y \oplus \rho_0, x) \downarrow = a \neq \Phi_{e_0}(Y \oplus \rho_1, x) \downarrow = b. \qquad \star$$

Preuve. Il suffit de trouver deux extensions de τ de même taille et incompatibles sur un certain x. Soient $i_0, \ldots, i_k$ les bits sur lesquelles ces extensions diffèrent. On inverse le bit i_0 dans la première extension, et on l'étend pour obtenir une valeur a_0 pour x. Si $a_0 \neq a$, on a terminé. Sinon, on inverse à son tour i_1 dans cette nouvelle extension, que l'on étend encore pour obtenir une valeur a_1, et ainsi de suite. Si chaque valeur $a_1 = a_2 = \cdots = a_{k-1}$, alors $a_{k-1} \neq b$, et notre chaîne diffère maintenant d'un seul bit de celle qui a produit b. ∎

On se sert du lemme précédent pour calculer à l'aide de Y une suite de quadruplets $(\tau_{0,m}, \tau_{1,m}, i_m, x_m)_{m \in \mathbb{N}}$ avec $i_m < i_{m+1}$ telle que, pour tout m, les chaînes $\tau_{0,m}, \tau_{1,m}$ diffèrent sur exactement le bit i_m et soient incompatibles sur x_n. La suite de bits $(i_m)_{m \in \mathbb{N}}$ est une suite infinie et Y-calculable. Rappelons que notre condition de forcing est de la forme (σ_0, σ_1, n). S'il existe i_m tel que $i_m = \langle k, a \rangle$ pour $k > m$, cela entraîne que pour toute extension τ' de σ_1 telle que $\langle \tau_{0,m}, \tau', n \rangle$ est une extension valide, alors $\langle \tau_{1,m}, \tau', n \rangle$ en est aussi une, car il n'y a aucune contrainte sur le bit i_m. On peut donc trouver une extension de τ' de σ_1 qui force une valeur pour x_n (à supposer que l'on ne puisse pas forcer la partialité de ce côté là), et l'on prend l'extension $\tau_{0,m}$ ou $\tau_{1,m}$ de σ_0 qui force une valeur différente. On force donc $\Phi_{e_0}(Y \oplus G_0)$ et $\Phi_{e_1}(Y \oplus G_1)$ à être différents.

Si à présent il n'existe pas $i_m = \langle k, a \rangle$ pour $k > m$, alors il doit exister par le principe des tiroirs un certain $k \leqslant m$ pour lequel une infinité de i_m est de la forme $\langle k, a \rangle$. Aussi n'est-il pas possible d'avoir $A_k(a) = 1$ pour chacun de ces i_m, car on aurait alors un sous-ensemble infini de A_k et Y-calculable, or par hypothèse tout sous-ensemble infini de A_k calcule A_k, et Y ne calcule pas A_k. Il doit donc exister $\tau_{0,m}, \tau_{1,m}$ et $i_m = \langle k, a \rangle$ tel que $A_k(a) = 0$. Là encore, toute extension τ' de σ_1 qui est compatible avec $\tau_{0,m}$ le sera aussi avec $\tau_{1,m}$, car il n'y a aucune contrainte sur le bit i_m. On peut alors trouver

une extension τ' de σ_1 qui force une valeur sur x_m —à moins que l'on ne puisse forcer la partialité de ce côté là— et choisir une extension parmi $\tau_{0,m}$ et $\tau_{1,m}$ qui force une autre valeur sur x_m. Cela conclut la preuve. ∎

Et voilà. La preuve du lemme ne fut pas sans difficulté, mais nous sommes à présent presque au bout de nos peines. Montrons finalement que tout sous-ensemble dénombrable de $\boldsymbol{\mathcal{D}}^n$ peut être codé dans les degrés Turing.

Preuve du théorème 4.4. Dans ce qui suit, les variables en majuscule dénotent des ensembles ou suites de degrés Turing. Soit un ensemble dénombrable de n-uplets $R \subseteq \boldsymbol{\mathcal{D}}^n$. Soit $\mathbf{b}$ un majorant sur l'ensemble des degrés concernés par R (c'est-à-dire sur la réunion des projections de R sur chaque coordonnée), et soit $(\mathbf{x}_i)_{i\in\mathbb{N}}$ une liste de tous les degrés sous $\mathbf{b}$. Notons que $\mathbf{b}$ n'est qu'un majorant, et que certains $\mathbf{x}_i$ peuvent donc ne pas être des degrés du n-uplet R.

On trouve alors une anti-chaîne $(\mathbf{c}_i^k)_{k\leqslant n,\, i\in\mathbb{N}}$ telle que $B = (\mathbf{c}_i^k \cup \mathbf{x}_i)_{k\leqslant n,\, i\in\mathbb{N}}$ forme un ensemble de degrés calculatoirement indépendants (on montre sans peine qu'une telle anti-chaîne existe, par extensions finies). Pour $k \leqslant n$ fixé, soit $C_k = (\mathbf{c}_i^k)_{i\in\mathbb{N}}$. On définit finalement

$$S = \{\mathbf{c}_{i_1}^1 \cup \mathbf{x}_{i_1} \cup \cdots \cup \mathbf{c}_{i_n}^n \cup \mathbf{x}_{i_n} \ : \ (\mathbf{x}_{i_1}, \ldots, \mathbf{x}_{i_n}) \in R\}.$$

Comme B est calculatoirement indépendant, il existe pour tout $\mathbf{a} \in S$ un n-uplet de degrés $\mathbf{b}_1, \ldots, \mathbf{b}_n \in B$, unique à l'ordre près, tel que

$$\mathbf{a} = \mathbf{b}_1 \cup \cdots \cup \mathbf{b}_n.$$

Par ailleurs, l'indépendance calculatoire de B garantit également que pour le degré $\mathbf{b}_1$ il existe un unique $i \leqslant n$ tel que $\mathbf{b}_1 = \mathbf{c}_m^i \cup \mathbf{x}_m$ pour un certain m, et ce m est lui aussi unique. Il en va de même pour $\mathbf{b}_2$, $\mathbf{b}_3$, …, ce qui permet de définir R de la manière suivante : $(\mathbf{x}_1, \ldots, \mathbf{x}_n) \in R$ si

$$\mathbf{x}_1 \leqslant \mathbf{b} \wedge \cdots \wedge \mathbf{x}_n \leqslant \mathbf{b} \wedge \exists \mathbf{y}_1 \in C_1 \ \ldots \exists \mathbf{y}_n \in C_n \ (\mathbf{x}_1 \cup \mathbf{y}_1) \cup \cdots \cup (\mathbf{x}_n \cup \mathbf{y}_n) \in S.$$

Comme chaque C_i et comme S sont des anti-chaînes, elles sont définissables d'après le lemme 4.6. Une telle formule est donc définissable dans les degrés Turing. ∎

Slaman et Woodin ont utilisé leur technique de codage comme point de départ à une étude complexe de la *rigidité* des degrés Turing. Une structure est dite *rigide* si elle n'admet pas d'automorphisme autre que l'identité, c'est-à-dire dans le cas des degrés, de bijection on triviale $f : \boldsymbol{\mathcal{D}} \to \boldsymbol{\mathcal{D}}$ telle que $\mathbf{a} \leqslant \mathbf{b} \leftrightarrow f(\mathbf{a}) \leqslant f(\mathbf{b})$. Par exemple, la structure $(\mathbb{R}, +, \times, \leqslant)$ des réels est une structure rigide. Étant donné un automorphisme $f : \mathbb{R} \to \mathbb{R}$, on doit avoir $f(0) + f(1) = f(1)$ et donc $f(0) = 0$, puis $f(1) = f(1) \times f(1)$