Résultats autour de la conjecture d'Hartmanis et Stearns

Auteur : Georis, Nicolas
Promoteur(s) : Charlier, Emilie
Faculté : Faculté des Sciences
Diplôme : Master en sciences mathématiques, à finalité didactique
Année académique : 2025-2026
URI/URL : http://hdl.handle.net/2268.2/25528

Faculté des Sciences
Département de Mathématique

Résultats autour de la conjecture de Hartmanis et Stearns

Mémoire de fin d'études présenté en vue de l'obtention du titre de
Master en Sciences Mathématiques, à finalité didactique

Année académique 2025-2026

Auteur :
Nicolas Georis

Promotrice :
Émilie Charlier

Co-Promoteur :
Pierre Popoli

Remerciements

Je tiens tout d'abord à remercier chaleureusement mes promoteurs, Émilie Charlier et Pierre Popoli, pour leur accompagnement tout au long de ce travail, leurs relectures attentives, leurs suggestions toujours pertinentes, ainsi que toutes les discussions lors de nos réunions, qui m'apportaient toujours de nouvelles pistes de réflexion. Merci en particulier à Émilie Charlier de m'avoir proposé ce sujet qui m'a permis, en plus d'explorer des thématiques de calculabilité qui m'intéressaient, de découvrir des aspects de théorie des nombres auxquels je ne me serais sans doute pas intéressé autrement. Merci aussi à Pierre Popoli de s'être intéressé à mon sujet de mémoire et d'avoir rejoint son encadrement.

Je remercie également les enseignantes et enseignants rencontrés au cours de mes années de master pour leurs cours intéressants, pas uniquement de mathématiques, ainsi que les étudiantes et étudiants avec qui j'ai eu de nombreuses discussions, pas toujours mathématiques non plus. Un merci tout particulier à Morgane, pour tous les repas et cafés partagés.

Une pensée va également à mes amies et amis qui, malgré la distance, me soutiennent depuis de nombreuses années et qui m'ont accompagné dans l'accomplissement de ce master, et ce même sans nécessairement comprendre ce que je pouvais bien faire de mes journées, ni même ce dont ce mémoire peut parler : Ana, Huba, Léo, Leyla, Noëlie, Noémie et Shems, ainsi que ceux que j'aurais oublié de citer (oups). Merci à vous d'être là et de prendre le temps de lire ces lignes, si jamais vous passez par ici. Merci évidemment à Élisa pour tout ton soutien, tous les rires, toutes les discussions sur les sujets les plus incongrus, les élans créatifs et philosophiques, ainsi que pour supporter toutes mes angoisses et interrogations, et ce depuis bientôt treize ans au moment de l'écriture de ces lignes. Merci d'avoir été jusqu'ici une constante dans ma vie ; j'espère que nous partagerons encore quelques crises existentielles ensemble.

Enfin, merci infiniment à Michel et Élodie, sans qui, sûrement, la rédaction de ce mémoire, ni même l'accomplissement de mon master, n'auraient été possibles.

Table des matières

Chapitre 1

Introduction : La conjecture de Hartmanis–Stearns

Si des constantes usuelles telles que

$$\sqrt{2} = 1,41421356237309504880168872420969807856 9\cdots$$

ou

$$\pi = 3,14159265358979323846264338327950288419 7\cdots$$

admettent des descriptions géométriques très simples, leur représentation en base 10 semble être particulièrement complexe. À l'opposé de ces deux exemples, il est connu que les rationnels ont une représentation en base b relativement simple, celle-ci étant ultimement périodique, et ce pour tout entier $b \geq 2$. Mais que peut-on dire sur la représentation en base b des nombres irrationnels, et en particulier des irrationnels algébriques ? Cette question a été posée par Borel, qui conjectura en 1950 que la représentation de ces derniers doit suivre une loi déterminée [14]. En particulier, il est attendu que les irrationnels algébriques soient normaux en base b, pour toute base $b \geq 2$. Rappelons qu'un nombre α est *algébrique* s'il est racine d'un polynôme de $\mathbb{Z}[X]$, et qu'un nombre ξ est *normal en base* b si, pour tout entier naturel n, chacun des b^n blocs de longueur n de la représentation en base b de ξ apparaît avec une fréquence $\frac{1}{b^n}$. Un des premiers exemples de nombre normal en base 10,

$$C_{10} = 0,12345678910111213 14\cdots ,$$

a été donné par Champernowne [20] en 1933. Borel montra en 1909 qu'au sens de la mesure de Lebesgue, presque tout nombre est normal en base b, et ce pour toute base $b \geq 2$ [13], ce qui motive sa conjecture. Cependant, celle-ci est réputée hors d'atteinte, car il est en pratique très difficile de démontrer qu'un nombre quelconque est normal dans une base donnée.

Cette conjecture a suscité de nombreuses tentatives de formalisation de la complexité des représentations en base b des nombres réels, selon des points de vue variés. Une première approche, initiée par Borel, repose sur des considérations probabilistes et conduit à la notion de normalité présentée précédemment. D'autres cadres ont également été développés, notamment en dynamique symbolique avec les travaux de Morse et Hedlund [37], ou encore dans une perspective algorithmique à travers

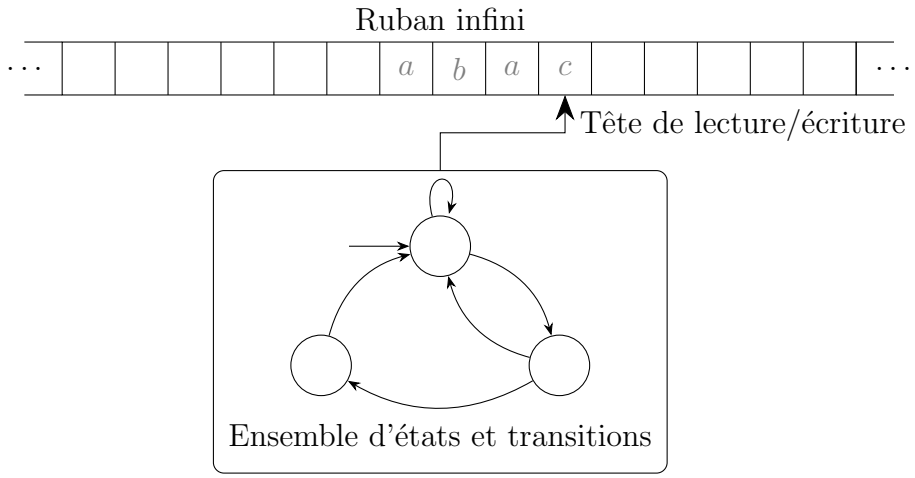


FIGURE 1.1 – Représentation schématique d'une machine de Turing

le modèle des machines de Turing introduit par Hartmanis et Stearns [29]. Dans ce mémoire, nous adoptons majoritairement ce dernier point de vue.

En 1965, Hartmanis et Stearns publient *On the computational complexity of algorithms* [29], article fondateur de la théorie de la complexité. Dans ce travail, les auteurs introduisent pour la première fois une formalisation du temps de calcul des algorithmes à l'aide du modèle de la machine de Turing. Ils définissent un cadre permettant de classer les fonctions calculables selon la croissance du nombre d'opérations nécessaires à leur calcul.

Une *machine de Turing* est un modèle abstrait de calcul formalisé par Turing dans les années 1930 [51]. Elle se compose d'un (ou plusieurs) ruban infini divisé en cases (chacune pouvant contenir un symbole d'un alphabet fini), d'une tête de lecture/écriture se déplaçant sur le ruban, et d'un ensemble fini d'états gouvernés par une table de transition. À chaque étape, en fonction de l'état courant et du symbole lu sur le ruban, la machine modifie le contenu de la case, ou se déplace à gauche ou à droite sur le ruban, et change d'état. Une fonction (ou suite) $f : \mathbb{N} \to \mathbb{N}$ est dite *calculable* s'il existe une machine de Turing qui, pour tout entier naturel m en entrée sur le ruban, termine son exécution et produit $f(m)$ en sortie.

Plus précisément, Hartmanis et Stearns introduisent la notion de *fonction temporelle* $T(n)$, une fonction croissante qui borne le nombre d'opérations nécessaires à une machine de Turing (à plusieurs rubans) pour produire les n premiers symboles d'une suite. Une suite est dite *T-calculable* si elle peut être générée dans cette borne de temps, et l'ensemble de toutes les suites T-calculables forme la *classe de complexité temporelle* S_T. Ils posent ainsi les bases de la classification des fonctions calculables selon leur complexité temporelle, inaugurant par la même occasion l'étude des classes de complexité temporelle.

Parmi les motivations de cette analyse, les auteurs s'intéressent à la classification des nombres selon la complexité temporelle de leur développement en base b. Autrement dit, ils considèrent le problème suivant : étant donnés un nombre réel $\xi \in \mathbb{R}$ et une base b, dans quelle classe de complexité se situe la suite des chiffres du développement en base b de ξ ? Plus précisément, peut-on produire les n premiers termes de cette suite en un nombre d'étapes linéaire par rapport à n ? Si

l'on peut répondre par l'affirmative pour les nombres rationnels, ainsi que pour certains nombres transcendants construits explicitement, les auteurs posent la question suivante à la fin de leur article :

Problème (Hartmanis–Stearns)**.** *Existe-t-il des nombres irrationnels algébriques dont la représentation en base b peut être générée par une machine de Turing à plusieurs rubans en $O(n)$ opérations ?*

Aujourd'hui, cette question reste ouverte et une réponse négative est généralement attendue, donnant lieu à *la conjecture de Hartmanis–Stearns*.

Le présent mémoire s'inscrit dans la continuité de cette conjecture et a pour objectif de présenter des résultats allant dans le sens de celle-ci. Il suit également une approche initiée par Cobham [22] pour aborder ce problème : l'on étudiera des restrictions de la question en changeant le modèle de calcul. Nous nous concentrerons sur des sous-classes strictes de machines de Turing, telles que les automates finis déterministes et les automates à pile, afin d'examiner dans quelle mesure ces modèles peuvent (ou ne peuvent pas) produire les développements en base entière de réels irrationnels algébriques.

Le chapitre 2 introduira les notions fondamentales nécessaires à la suite de ce travail. L'on y définira plus en détails les notions de représentation d'un réel en base entière, et de nombre algébrique. Nous évoquerons aussi le domaine de l'approximation diophantienne, c'est à dire l'approximation de nombres irrationnels par des rationnels, qui est étroitement lié à la conjecture. La lectrice ou le lecteur curieux d'approfondir ces notions peut consulter, entre autres, [27, 42] pour les notions de numération, et [8, 12, 16] pour les aspects de théorie des nombres et d'approximation diophantienne.

Ensuite, le chapitre 3 présentera plus en détails le cadre de la conjecture. L'on y définira les notions de machine de Turing et de fonction et de nombre calculables. L'on introduira ensuite la notion de classe de complexité temporelle définie par Hartmanis et Stearns dans [29] et l'on établira une première classification des nombres calculables. Des références qui traitent des notions de calculabilité sont [30, 49].

Une fois tout le cadre posé, le chapitre 4, basé principalement sur les travaux d'Adamczewski, Bugeaud et Luca [3, 4], sera consacré à l'étude du premier modèle de calcul que l'on considèrera, les automates finis déterministes. Nous y présenterons le premier résultat central de ce mémoire : ce modèle ne peut pas produire la b-représentation des nombres irrationnels algébriques. L'on étudiera ensuite les propriétés diophantiennes des suites automatiques, c'est à dire les suites produites par les automates finis déterministes, reprenant les travaux d'Adamczewski, Bugeaud et Cassaigne [2, 5]. Une référence classique qui porte sur les suites automatiques est [10].

Enfin, le chapitre 5 portera sur un modèle plus général, les automates à pile. L'on y présentera le deuxième résultat central de ce mémoire, basé sur les travaux d'Adamczewski, Cassaigne et Le Gonidec [6], à savoir que ce modèle n'est toujours pas suffisamment puissant pour produire la représentation en base entière des irrationnels algébriques. Nous étendrons ensuite cette étude aux machines à piles, qui généralisent à la fois les automates finis et les automates à pile, et pour lesquelles apparaît une distinction de puissance de calcul selon le nombre de piles considérées.

Nous introduirons un modèle particulier de machine de Turing, les machines unidirectionnelles, qui imposent des contraintes sur l'organisation de la mémoire et les déplacements de la tête de lecture, et pour lequel l'on dispose également d'un critère d'impossibilité de produire la b-représentation des irrationnels algébriques. Enfin, nous présenterons un modèle de calcul tout à fait différent des machines de Turing : les réseaux de réactions chimiques, qui nous permettra de mettre en perspective la conjecture.

Chapitre 2

Notions préliminaires

Ce chapitre est consacré à la présentation des notions fondamentales intervenant dans la question posée par Hartmanis et Stearns. Il a pour premier objectif de définir les termes de la question et de présenter les concepts centraux qui leur sont reliés. Il nous permettra également de fixer des notations qui seront utilisées tout au long de ce mémoire. Nous commencerons, d'une part, par présenter la notion de représentation d'un nombre et, d'autre part, par décrire les nombres algébriques, qui sont au cœur de la conjecture.

2.1 Notions de combinatoire des mots

Comme nous le verrons plus tard, la représentation d'un nombre est un mot sur un alphabet donné. Commençons alors par rappeler les notions fondamentales de combinatoire des mots qui seront utilisées tout au long de ce mémoire. Les éléments suivants sont principalement repris du cours *"Combinatorics on words"* [33] et du livre [9].

Définition 2.1.

- Un *alphabet* Σ est un ensemble fini et non-vide, dont les éléments sont appelés *lettres* ou *symboles*.
- Un *mot fini* sur l'alphabet Σ est une suite finie de symboles de Σ. Un *mot infini* $\boldsymbol{w}$ sur l'alphabet Σ est une application $\boldsymbol{w} : \mathbb{N} \to \Sigma$. Tout au long de ce mémoire, on utilisera les termes de suite et de mot de manière interchangeable.
- On note Σ^* l'ensemble des mots finis sur Σ et $\Sigma^{\mathbb{N}}$ l'ensemble des mots infinis sur Σ.
- Pour un mot $w \in \Sigma^*$, sa *longueur*, notée $|w|$, est le nombre de lettres qui le composent. On note Σ^n l'ensemble des mots finis écrits sur Σ et de longueur n, pour n un entier naturel. On a $\Sigma^* = \bigcup_{n\in\mathbb{N}} \Sigma^n$.
- On note ε le *mot vide*, c'est-à-dire l'unique mot de longueur 0.

Rappelons que l'ensemble Σ^* muni de l'opération de concaténation définie, pour tous mots finis u, v, par

$$\begin{array}{rcl} \cdot : \Sigma^* \times \Sigma^* & \longrightarrow & \Sigma^* \\ (u, v) & \longmapsto & uv, \end{array}$$

forme un monoïde libre, avec pour élément neutre ε.

Définition 2.2.

- Un *facteur* u d'un mot w est un mot tel qu'il existe $x, y \in \Sigma^*$ vérifiant $w = xuy$.
- Si $x = \varepsilon$, u est un *préfixe* de w. Si $y = \varepsilon$, u est un *suffixe* de w.
- On note $\mathrm{Fac}(w)$ l'ensemble des facteurs de w, et $\mathrm{Fac}_n(w)$ l'ensemble de ses facteurs de longueur n.
- Soit $w \in \Sigma^* \cup \Sigma^{\mathbb{N}}$. La *complexité en facteurs* de w est l'application

$$\begin{aligned} p_w : \mathbb{N} &\to \mathbb{N} \\ n &\mapsto \#\mathrm{Fac}_n(w) \end{aligned}$$

  qui associe à chaque entier n le nombre de facteurs distincts de longueur n présents dans w.

Notation.

- Pour tous $i, j \in \mathbb{N}$, on note w_i la i-ème lettre du mot w, et $w_{[i,j)} := w_i w_{i+1} \cdots w_{j-1}$ le facteur de w qui commence à la i-ème lettre et qui est de longueur $j - 1 - i$.
- Pour un entier naturel n, on note $w^n = w \cdots w$ la concaténation de w, n fois.
- Pour u un mot fini, on note $u^\omega = \lim_{n \to +\infty} u^n$ le mot infini composé des concaténations à l'infini de u.
- Pour un réel positif x, on note w^x le mot $w^{\lfloor x \rfloor} w'$, avec w' le préfixe de w de longueur $\lceil (x - \lfloor x \rfloor)|w| \rceil$.

Exemple 2.3. Soit $w = abac$ un mot fini de $\{a, b, c\}^*$. On a

- $w^3 = abacabacabac$,
- $w^\omega = abacabacabacabac \cdots$,
- $w^{\sqrt{2}} = abacab$, avec $w^{\lfloor \sqrt{2} \rfloor} = abac$ et $w' = ab$.

Terminons cette section par le rappel du théorème suivant [38] :

Théorème 2.4 (Morse-Hedlund)**.** *Soit $\boldsymbol{w} \in \Sigma^{\mathbb{N}}$ un mot infini, et $p_{\boldsymbol{w}} : \mathbb{N} \to \mathbb{N}$ sa fonction de complexité en facteurs. Alors $\boldsymbol{w}$ est ultimement périodique si et seulement si $p_{\boldsymbol{w}}$ est bornée.*

2.2 Représentation en base b

Dans cette section, nous rappelons la manière dont les nombres peuvent être représentés dans une base entière $b \geq 2$, comme c'est le cas dans l'écriture usuelle en base 10 ou en base 2, par exemple. Les sources qui ont été particulièrement utiles pour rédiger cette section sont [43, chap. 1], [27], ainsi que [33].

Nous fixons un entier $b \geq 2$, appelé *base*, et nous considérons l'*alphabet canonique* des chiffres associé, noté $\Sigma_b = \{0, 1, \ldots, b-1\}$. L'entier b, conjointement avec l'alphabet Σ_b, définit le *système de numération en base b*. Nous introduisons dans un premier temps la représentation en base b des entiers naturels, avant d'étendre cette notion au cas des réels.

2.2.1 Représentation des entiers naturels

Avec une base b fixée, l'on peut associer une valeur entière à chaque mot du langage Σ_b^*. En effet, pour tout $w \in \Sigma_b^*$, notons $[w]_b$ sa *valeur en base b*, définie de la manière suivante

$$w = w_n w_{n-1} \cdots w_1 w_0 \longmapsto [w]_b = \sum_{i=0}^{n} w_i b^i.$$

Par convention, nous écrivons donc les entiers en plaçant le chiffre le plus significatif à gauche, conformément à l'usage courant, bien que ce choix soit en réalité purement arbitraire.

Exemple 2.5. Considérons le mot $w = 1010$.

— En base 2, bien connue des informaticiens, on a $[w]_2 = 2^3 + 2^1 = 17$.

— En base 5, on a $[w]_5 = 5^3 + 5^1 = 126$.

— En base 10, que l'on utilise au quotidien, on a bien évidemment $[w]_{10} = 1010$.

Lemme 2.6. *Pour tout $n \in \mathbb{N}$, l'application $[\cdot]_b$ est injective sur Σ_b^n.*

Démonstration. Soient $u = u_{n-1}u_{n-2}\cdots u_1u_0$ et $v = v_{n-1}v_{n-2}\cdots v_1v_0$ deux mots différents de Σ_b^*, tous deux de longueur n et tels que $[u]_b = [v]_b$. Alors

$$\sum_{i=0}^{n-1} u_i b^i - \sum_{i=0}^{n-1} v_i b^i = 0 \text{ et donc } P(X) = \sum_{i=0}^{n-1} (u_i - v_i) X^i$$

est un polynôme non-nul de $\mathbb{Z}[X]$ qui possède b pour racine. En réalisant la division euclidienne de P par $X - b$, l'on peut le réexprimer sous la forme

$$P(X) = (X - b) \cdot Q(X),$$

pour Q un polynôme de $\mathbb{Z}[X]$. En évaluant P en 0, on obtient

$$P(0) = u_0 - v_0 = -b \cdot Q(0).$$

Ceci implique que b divise $u_0 - v_0$, ce qui est une contradiction car $|u_0 - v_0| < b$. □

L'application $[\cdot]_b$ n'est pas injective sur le langage Σ_b^* tout entier. En effet, il est évident que $[0^h w]_b = [w]_b$ pour tout $w \in \Sigma_b^*$ et $h \in \mathbb{N}$. De plus, le lemme 2.6 nous assure que rajouter des 0 de tête est la seule possibilité d'avoir deux mots de Σ_b^* qui ont même valeur :

$$[u]_b = [v]_b \text{ et } |u| > |v| \implies u = 0^h v \text{ avec } h = |u| - |v|.$$

Réciproquement, l'on peut associer à chaque entier $N \in \mathbb{N}$ une représentation dans Σ_b^*. Cette représentation est unique sous la condition qu'elle ne commence pas par le symbole 0, et peut-être calculée via l'*algorithme glouton*.

Algorithme glouton. Soit $N \in \mathbb{N}$. Il existe un unique k entier tel que $b^k \leq N < b^{k+1}$. Notons $N_k = N$, et pour tout $i = k$ jusqu'à $i = 0$, l'on pose

$$a_i = \left\lfloor \frac{N_i}{b^i} \right\rfloor \text{ et } N_{i-1} = N_i - a_i b^i.$$

On a que $a_i \in \Sigma_b$, $a_k \neq 0$ et $N_i < b^i$. Finalement :

$$N = \sum_{i=0}^{k} a_i b^i = [a_k a_{k-1} \cdots a_1 a_0]_b$$

Cet algorithme calcule les chiffres de la gauche vers la droite, conformément à notre représentation choisie, c'est à dire en notant le chiffre le plus significatif à gauche. Il nous permet donc d'énoncer le théorème suivant :

Théorème 2.7. *Tout entier naturel $N \in \mathbb{N}$ possède une unique b-représentation qui ne commence pas par un 0.*

Cette b-représentation est dite *canonique*, et on la note $\langle N \rangle_b$. Dans la suite de ce travail, sauf mention explicite du contraire, toute b-représentation d'un entier naturel N sera implicitement supposée canonique. Notons que la b-représentation canonique de 0 est toujours ε, et que le langage des b-représentations canoniques est

$$\mathcal{C}_b = \{\langle N \rangle_b \mid N \in \mathbb{N}\} = \Sigma_b^* \setminus 0\, \Sigma_b^* = \{\Sigma_b \setminus 0\}\Sigma_b^* \cup \{\varepsilon\}.$$

Remarque 2.8. Bien que nous définissions la b-représentation d'un entier naturel pour une base $b \geq 2$, il est également possible de la définir pour $b = 1$. Dans ce cas, l'on parle de représentation en base *unaire*, et un entier $n \in \mathbb{N}$ est représenté par un mot de la forme $1^n = \underbrace{111\cdots1}_{n \text{ fois}}$ (ou en fait n'importe quel autre symbole que 1, répété n fois).

2.2.2 Représentation des réels

Là où les entiers naturels sont représentés par une suite finie de symboles, les nombres réels sont représentés par une suite infinie de symboles, c'est à dire par un mot de $\Sigma_b^{\mathbb{N}}$.

Soit x un réel positif. Étant donné que nous avons déjà traité la représentation d'un entier naturel, une première approche pour donner une b-représentation de x serait de traiter séparément sa partie entière $\lfloor x \rfloor$ de sa partie décimale $\{x\} = x - \lfloor x \rfloor$. De manière similaire, définissons une application $[\cdot]_b : \Sigma_b^{\mathbb{N}} \to [0, 1]$ qui représente la valeur en base b d'un réel de $[0, 1]$, de la manière suivante :

$$\boldsymbol{w} = w_1 w_2 w_3 \cdots \longmapsto [\boldsymbol{w}]_b = \sum_{i=1}^{+\infty} w_i b^{-i}.$$

Par convention, et par souci de simplicité, nous commençons l'indexation des symboles des mots infinis à 1. Tout mot de $\Sigma_b^{\mathbb{N}}$ sera noté $\boldsymbol{w} = (w_i)_{i \geq 1}$, avec tous les $w_i \in \Sigma_b$.

Rappelons que $\Sigma_b^{\mathbb{N}}$ est un espace métrique muni de la distance d définie de la manière suivante :

$$\begin{array}{rcl} d : \Sigma_b^{\mathbb{N}} \times \Sigma_b^{\mathbb{N}} & \longrightarrow & [0,1] \\ (\boldsymbol{u}, \boldsymbol{v}) & \longmapsto & \begin{cases} 0 & \text{si } \boldsymbol{u} = \boldsymbol{v}, \\ 2^{-|\boldsymbol{u} \wedge \boldsymbol{v}|} & \text{si } \boldsymbol{u} \neq \boldsymbol{v} \end{cases} \end{array}$$

avec $\boldsymbol{u} \wedge \boldsymbol{v}$ désignant le plus long commun préfixe de $\boldsymbol{u}$ et $\boldsymbol{v}$. De plus, $\Sigma_b^{\mathbb{N}}$ est totalement ordonné par l'ordre lexicographique : l'on a $\boldsymbol{u} < \boldsymbol{v}$ si et seulement si, avec $w = \boldsymbol{u} \wedge \boldsymbol{v}$, $\boldsymbol{u} = wa\boldsymbol{u'}$ et $\boldsymbol{v} = wb\boldsymbol{v'}$ et $a, b \in \Sigma_b$, on a $a < b$.

Proposition 2.9. *L'application $[\cdot]_b : \Sigma_b^{\mathbb{N}} \to [0,1]$ est continue et préserve l'ordre. De plus, pour des mots $\boldsymbol{u}$ et $\boldsymbol{v}$ de $\Sigma_b^{\mathbb{N}}$ tels que $\boldsymbol{u} < \boldsymbol{v}$, $w = \boldsymbol{u} \wedge \boldsymbol{v}$, on a $[\boldsymbol{u}]_b = [\boldsymbol{v}]_b$ si et seulement si $\boldsymbol{u} = wa(b-1)^\omega$ et $\boldsymbol{v} = w(a+1)0^\omega$, avec $a \in \Sigma_b$.*

Démonstration. Soient $\boldsymbol{u}$ et $\boldsymbol{v}$ des mots de $\Sigma_b^{\mathbb{N}}$. On a

$$\begin{aligned} |[\boldsymbol{u}]_b - [\boldsymbol{v}]_b| &= \left| \sum_{i=1}^{+\infty} (u_i - v_i) b^{-i} \right| = \left| \sum_{i=|u \wedge v|+1}^{+\infty} (u_i - v_i) b^{-i} \right| \\ &\leq b^{-|u \wedge v|} \\ &\leq 2^{-|u \wedge v|} = d(\boldsymbol{u}, \boldsymbol{v}). \end{aligned}$$

Ainsi, l'application $[\cdot]_b$ est lipschitzienne, donc continue. Remarquons que, si pour tout i, $u_i \leq v_i$, et s'il existe un j tel que $u_j \neq v_j$, alors $[\boldsymbol{u}]_b < [\boldsymbol{v}]_b$. De plus, si l'on considère la valeur en base b de l'élément maximal de $\Sigma_b^{\mathbb{N}}$, on a :

$$\sum_{i=1}^{+\infty} (b-1) b^{-i} = (b-1) \sum_{i=1}^{+\infty} b^{-i} = (b-1) \frac{\frac{1}{b}}{1 - \frac{1}{b}} = 1.$$

Ceci signifie en particulier que l'image de $\Sigma_b^{\mathbb{N}}$ par $[\cdot]_b$ est incluse dans $[0,1]$. De plus, supposons $\boldsymbol{u} < \boldsymbol{v}$, et soit $k \in \mathbb{N}$ minimal tel que $u_k \neq v_k$, c'est à dire $u_k \leq v_k - 1$. Posons alors les mots infinis

$$\boldsymbol{u'} = u_1 u_2 \cdots u_{k-1} u_k (b-1)(b-1) \cdots \text{ et } \boldsymbol{v'} = v_1 v_2 \cdots v_{k-1} (u_k + 1) 000 \cdots$$

On a alors $[u']_b = [v']_b$. En effet,

$$\begin{aligned} [\boldsymbol{u'}]_b &= \sum_{i=1}^{k-1} u'_i b^{-i} + u_k b^{-k} + \sum_{i=k+1}^{+\infty} (b-1) b^{-i} \\ &= \sum_{i=1}^{k-1} u'_i b^{-i} + u_k b^{-k} + (b-1) \frac{\frac{1}{b^{k+1}}}{1 - \frac{1}{b}} \\ &= \sum_{i=1}^{k-1} u'_i b^{-i} + u_k b^{-k} + b^{-k} \\ &= \sum_{i=1}^{k-1} v'_i b^{-i} + (u_k + 1) b^{-k} \\ &= [\boldsymbol{v'}]_b \end{aligned}$$

De plus, si $\boldsymbol{u} \neq \boldsymbol{u'}$, alors $[\boldsymbol{u}]_b < [\boldsymbol{u'}]_b$, et si $\boldsymbol{v} \neq \boldsymbol{v'}$, alors $[\boldsymbol{v'}]_b < [\boldsymbol{v}]_b$. On obtient bien que $[\cdot]_b$ conserve l'ordre. □

D'une manière similaire au cas des entiers naturels, il existe un algorithme glouton permettant de calculer la b-représentation d'un réel x dans $[0, 1)$, qui la produit en placant le chiffre le plus significatif à gauche.

Algorithme glouton. Soit $x \in [0, 1)$. Posons $r_0 = x$, et pour tout $i \geq 1$, soient

$$w_i = \lfloor b r_{i-1} \rfloor \text{ et } r_i = \{b r_{i-1}\}.$$

Chaque w_i appartient à Σ_b, et l'on a

$$x = w_1 b^{-1} + r_1 b^{-1} = w_1 b^{-1} + w_2 b^{-2} + r_2 b^{-2} = \cdots = \sum_{i=1}^{+\infty} w_i b^{-i}.$$

Ainsi, l'algorithme produit un mot infini $\boldsymbol{w} = w_1 w_2 w_3 \cdots \in \Sigma_b^{\mathbb{N}}$ qui est une b-représentation de x, que l'on note $\langle x \rangle_b$. Par convention, l'on dira qu'un mot $\boldsymbol{w} \in \Sigma_b^{\mathbb{N}}$ est une b-représentation *finie* si $\boldsymbol{w}$ est de la forme $u0^\omega$, avec $u \in \Sigma_b^*$. Un réel $x \in [0, 1)$ est dit *b-décimal* si sa b-représentation est finie, c'est à dire si et seulement si x est la forme $\frac{p}{b^k}$ avec p et k des entiers et $k \geq 1$ (On se convaincra assez facilement qu'appliquer l'algorithme glouton sur un réel de cette forme produira un mot qui se termine par une infinité de 0).

Corollaire 2.10. *L'application $[\cdot]_b : \Sigma_b^{\mathbb{N}} \to [0, 1]$ est surjective. Un réel $x \in [0, 1)$ a plus d'une b-représentation si et seulement si il est b-décimal. Dans ce cas, il en a exactement deux, une finie et une infinie.*

Dans le cas, où x est b-décimal, nous considérerons uniquement sa b-représentation finie, qui est la plus grande des deux dans l'ordre lexicographique, et celle qui est calculée par l'algorithme glouton.

Exemple 2.11. En base 10, le nombre $x = \frac{1}{10}$ possède deux représentations :

— 09^ω, l'on noterait alors $x = 0,099999\ldots$,

— 10^ω, qui correspond à l'écriture décimale usuelle, $x = 0,1$.

Nous noterons $\langle x \rangle_b = 10^\omega$.

Finalement, pour un réel x positif, si l'on concatène la représentation de sa partie réelle avec celle de sa partie décimale, l'on obtient une représentation de la forme suivante :

$$\langle x \rangle_b = \boldsymbol{w} = w_{-k} w_{-k+1} \cdots w_{-1} w_0 \bullet w_1 w_2 \cdots \longmapsto [\boldsymbol{w}]_b = \sum_{i=-k}^{+\infty} w_i b^{-i} = x,$$

avec le symbole $\bullet$ délimitant la partie entière de la partie décimale. Dans la suite de ce travail, nous nous intéresserons uniquement aux réels compris dans l'intervalle $[0, 1)$, pour des raisons de simplicité et ce, sans perte de généralité. En effet, omettre la partie entière des réels ne retire qu'un nombre fini de symboles à leur

b-représentation. De plus, pour tout $x > 1$ l'on peut toujours se ramener au cas d'un réel dans l'intervalle $[0, 1)$: si l'on considère k l'unique entier naturel tel que $b^{k-1} \leq x < b^k$, alors le réel $y = \frac{x}{b^k}$ appartient à $[0, 1)$. Si $\langle y \rangle_b = w_1 w_2 w_3 \cdots$, alors l'on obtiendra la b-représentation de x en replaçant au bon endroit la virgule séparant la partie entière de la partie décimale : $\langle x \rangle_b = w_1 w_2 \cdots w_k \bullet w_{k+1} \cdots$. Enfin, le cas des réels négatifs ne nécessite pas d'étude particulière, leur représentation usuelle ne différant de celle de leur valeur absolue que par le signe.

L'ensemble des b-représentations des réels de $[0, 1)$ est donc le langage suivant :

$$\mathcal{R}_b = \{\langle x \rangle_b \mid x \in [0, 1)\} = \Sigma_b^{\mathbb{N}} \setminus \Sigma_b^*(b-1)^\omega.$$

Concluons par mentionner les liens entre certaines propriétés arithmétiques des nombres réels et la structure de leur représentation en base entière. Le résultat suivant est un exemple classique, qui établit l'équivalence entre le caractère rationnel d'un nombre et la périodicité de sa b-représentation [28, Chap. 9, Theorem 135].

Théorème 2.12. *Un nombre est rationnel si et seulement si sa b-représentation est ultimement périodique.*

Les nombres irrationnels sont donc précisément ceux dont la représentation en base b n'est pas ultimement périodique. C'est le comportement de ces développements non périodiques qui constituera le coeur de notre étude. Nous nous intéresserons plus particulièrement aux irrationnels *algébriques*, qui font l'objet de la section suivante.

2.3 Nombres algébriques

Dans la mesure où la question de Hartmanis et Stearns porte sur les nombres algébriques, il convient d'en rappeler la définition. Plus largement, nous évoquerons le domaine des mathématiques qu'est l'approximation diophantienne, étroitement lié à l'étude de ceux-ci, et qui apparaîtra à plusieurs reprises au cours de ce mémoire. Les sources qui ont été particulièrement utiles pour rédiger cette section sont [8, Chap. 5], [9, Chap. 2] et [16, Chap. 1-3].

2.3.1 Nombres algébriques et transcendants

Définition 2.13. Soit $x \in \mathbb{C}$.

— On dit que x est un nombre *algébrique* s'il est solution d'une équation de la forme

$$a_d X^d + a_{d-1} X^{d-1} + \cdots + a_1 X + a_0 = 0, \tag{2.1}$$

avec tous les $a_0, a_1, \ldots, a_d \in \mathbb{Z}$ et $a_d \neq 0$. Autrement dit, x est algébrique s'il est racine d'un polynôme de $\mathbb{Z}[X]$. On note $\overline{\mathbb{Q}}$ l'ensemble des nombres algébriques.

— En particulier, les nombres rationnels sont algébriques.

— Si $a_d = 1$, c'est à dire si x est racine d'un polynôme unitaire, on dit que c'est un *entier algébrique*.

— On dit qu'un nombre est *transcendant* s'il n'est pas algébrique.

Exemple 2.14.

— $\sqrt{2}$ est algébrique, il est racine du polynôme $X^2 - 2 \in \mathbb{Z}[X]$.

— Tout nombre rationnel $\frac{p}{q} \in \mathbb{Q}$ est racine du polynôme $qX - p \in \mathbb{Z}[X]$.

— Le nombre complexe i est également algébrique, il est racine du polynôme $X^2 + 1 \in \mathbb{Z}[X]$.

— Certaines constantes usuelles telles que π et e sont transcendantes [1].

Remarques 2.15.

— L'on pourrait également définir un nombre algébrique comme solution d'une équation de la forme (2.1), avec $a_0, a_1, \ldots, a_d \in \mathbb{Q}$ plutôt que $\mathbb{Z}$. En effet, tout polynôme de $\mathbb{Z}[X]$ est aussi trivialement dans $\mathbb{Q}[X]$. Réciproquement, pour $P \in \mathbb{Q}[X]$, en notant k le plus petit commun multiple des dénominateurs des coefficients de P, on obtient que kP est dans $\mathbb{Z}[X]$ et possède les mêmes racines que P.

— La notation $\overline{\mathbb{Q}}$ se réfère au fait que l'ensemble des nombres algébriques forme un corps particulier : c'est la *clôture algébrique* de $\mathbb{Q}$, c'est à dire le plus petit corps $\mathbb{K}$ qui contient toutes les racines des polynômes de $\mathbb{K}[X]$, et qui contient le corps $\mathbb{Q}$.

Théorème 2.16. *L'ensemble $\overline{\mathbb{Q}}$ des nombres algébriques est dénombrable.*

Démonstration. Soit $x \in \overline{\mathbb{Q}}$ un nombre algébrique. Alors x est solution d'une équation de la forme

$$a_d X^d + a_{d-1} X^{d-1} + \cdots + a_1 X + a_0 = 0.$$

Sans perte de généralité, on peut supposer que les $a_0, a_1, \ldots, a_d \in \mathbb{Z}$. Définissons alors le rang de cette équation comme $d + |a_0| + |a_1| + \cdots + |a_d|$. Pour chaque rang, il y a un nombre fini d'équations, chacune ayant un nombre fini de solutions. L'on peut donc établir une bijection entre $\mathbb{N}$ et l'ensemble des nombres algébriques $\overline{\mathbb{Q}}$ en listant les solutions des équations pour chaque rang. □

Corollaire 2.17. *Presque tous les nombres réels sont transcendants (au sens de la mesure de Lebesgue).*

Démonstration. L'ensemble $\mathbb{R} \cap \overline{\mathbb{Q}}$ étant dénombrable, il est de mesure nulle. □

Le caractère dénombrable de l'ensemble des réels algébriques constitue non seulement une première preuve de l'existence de nombres transcendants, mais montre également que, d'un point de vue de la mesure, presque tous les réels le sont. Ainsi, les nombres transcendants constituent la quasi-totalité de $\mathbb{R}$, rendant les nombres algébriques (et en particulier, les irrationnels algébriques) particulièrement rares. L'idée de la preuve du théorème 2.16 remonte à Cantor [18], bien qu'elle ne soit

1. La lectrice ou le lecteur curieux peut, entre autres, se référer à [9, chapitre 2.2] pour la preuve de l'irrationnalité de π et e, et à [12, chapitre 1.2] pour la preuve de leur transcendance.

pas historiquement la première démonstration de l'existence de nombres transcendants (nous reviendrons sur cette autre approche dans la suite). Enfin, malgré cette "abondance" des nombres transcendants, il est paradoxalement souvent difficile, en pratique, d'établir le caractère transcendant d'un réel donné. La sous-section suivante nous permettra d'étudier plus précisément cette question.

2.3.2 Approximation diophantienne

L'approximation diophantienne est le domaine des mathématiques qui s'intéresse à l'approximation des réels irrationnels par des rationnels, et en particulier au degré de précision de cette approximation. La question centrale de l'approximation diophantienne est la suivante : dans quelle mesure un irrationnel ξ peut-il être approché par des nombres rationnels $\frac{p}{q}$? C'est à dire, à quel point peut-on rendre la différence $\left|\xi - \frac{p}{q}\right|$ petite en faisant varier $\frac{p}{q}$? Cette sous-section et la suivante constituent une brève introduction à cette très vaste question. Le lecteur ou la lectrice souhaitant approfondir le sujet pourra consulter, entre autres, le travail de Bugeaud [16], qui a grandement contribué à la rédaction de celles-ci.

Tout nombre réel ξ peut être exprimé comme une limite d'une suite de nombres rationnels (et ce, d'une infinité de manières). De plus, pour tout entier $q > 0$, il est assez facile de voir qu'il existe un entier p tel que $\left|\xi - \frac{p}{q}\right| < \frac{1}{2q}$. En effet, en notant $x = \lfloor \xi q \rfloor$, il suffit de choisir $p = x$ si $|\xi q - x| \leq \frac{1}{2}$, et $p = x + 1$ sinon, pour obtenir le résultat voulu. L'on peut cependant espérer une meilleure approximation de ξ par un rationnel $\frac{p}{q}$. Quand ξ est irrationnel, un premier théorème nous donne en effet une approximation plus satisfaisante.

Théorème 2.18 (Dirichlet [16, Theorem 1.1])**.** *Soient ξ et Q des nombres réels tels que $Q \geq 1$. Il existe un nombre rationnel $\frac{p}{q}$, avec $1 \leq q \leq Q$, tel que*

$$\left|\xi - \frac{p}{q}\right| < \frac{1}{qQ}.$$

De plus, si ξ est irrationnel, alors il existe une infinité de nombres rationnels $\frac{p}{q}$ tels que

$$\left|\xi - \frac{p}{q}\right| < \frac{1}{q^2}.$$

Démonstration. Commençons par montrer la première assertion, et posons $t = \lfloor Q \rfloor$. Si $\xi = \frac{a}{b} \in \mathbb{Q}$ alors poser $p := a$ et $q := b$ suffit. Sinon, considérons les $t + 2$ points $0, \{\xi\}, \{2\xi\}, \ldots, \{t\xi\}, 1$. Ils sont deux-à-deux distincts et divisent l'intervalle $[0, 1]$ en $t + 1$ intervalles, dont au moins l'un d'eux a au maximum pour longueur $\frac{1}{t+1}$ (par le principe des tiroirs). Ainsi, il existe des entiers k, ℓ, m_k, m_ℓ, tels que $0 \leq k < \ell \leq t$ et

$$|(\ell\xi - m_\ell) - (k\xi - m_k)| \leq \frac{1}{t+1} < \frac{1}{Q}.$$

Il nous suffit alors de poser $p := m_\ell - m_k$ et $q := \ell - k$, qui conviennent car l'on a bien $1 \leq q \leq t \leq Q$.

Supposons maintenant que ξ soit irrationnel pour la deuxième assertion, et posons Q_0 un entier positif. Il existe donc un rationnel $\frac{p}{q}$ tel que $1 \leq q \leq Q_0$ et

$$\left|\xi - \frac{p}{q}\right| < \frac{1}{qQ_0},$$

et supposons de plus que q soit minimal pour cette propriété. En réappliquant la première assertion avec $Q_1 := \frac{1}{\left|\xi - \frac{p}{q}\right|}$, l'on obtient un nombre rationnel $\frac{p'}{q'}$, tel que $1 \leq q' \leq \frac{1}{\left|\xi - \frac{p}{q}\right|}$ et

$$\left|\xi - \frac{p'}{q'}\right| < \frac{1}{q'Q_1} = \frac{1}{q'}\left|\xi - \frac{p}{q}\right| < \frac{1}{q'Q_0} < \frac{1}{q^2}.$$

Comme q est minimal, l'on a bien $q < q'$. En itérant cet argument avec $Q_2 := \frac{1}{\left|\xi - \frac{p'}{q'}\right|}$, et les suivants choisis de la même manière, l'on construit une suite $\left(\frac{p_n}{q_n}\right)_{n\geq 1}$ de rationnels tels que pour tout $n \geq 1$,

$$\left|\xi - \frac{p_n}{q_n}\right| < \frac{1}{q^2},$$

ce qui prouve la seconde assertion. □

Nous intéresser à l'approximation diophantienne va nous permettre d'établir de premiers critères de transcendance pour les nombres réels. Commençons par nous donner quelques définitions qui nous permettrons d'étudier plus précisément les nombres algébriques.

Proposition 2.19. *Pour tout nombre algébrique α, il existe un unique polynôme $m_\alpha \in \mathbb{Q}[X] \setminus \{0\}$, dont α est une racine, qui est unitaire et qui divise tous les autres polynômes qui ont α pour racine.*

Démonstration. Soit α un nombre algébrique, et soit $Q \in \mathbb{Q}[X]$ qui a α pour racine. Posons $I(\alpha)$ l'ensemble des polynômes qui ont α pour racine, c'est à dire

$$I(\alpha) = \{P \in \mathbb{Q}[X] \mid P(\alpha) = 0\}.$$

Remarquons que le polynôme nul appartient à $I(\alpha)$. De plus, $I(\alpha)$ est clairement stable par addition et par multiplication par un élément de $\mathbb{Q}[X]$, c'est donc un idéal de $\mathbb{Q}[X]$. De plus, comme $\mathbb{Q}[X]$ est un anneau principal (car $\mathbb{Q}$ est un corps), $I(\alpha)$ est un idéal principal, c'est à dire qu'il existe un polynôme m_α tel que

$$I(\alpha) = \langle m_\alpha \rangle. \tag{2.2}$$

L'on a $I(\alpha) \neq \langle 0 \rangle$, car $Q \in I(\alpha)$. Montrons que ce m_α est unique et supposons qu'il existe un autre polynôme P tel que $I(\alpha) = \langle P \rangle$. Alors $\langle m_\alpha \rangle = \langle P \rangle$ et donc $P(X) = q \cdot m_\alpha(X)$ avec $q \in \mathbb{Q} \setminus \{0\}$, car $\mathbb{Q}[X]$ est un anneau intègre. On peut alors supposer m_α unitaire, qui est alors uniquement déterminé par (2.2). □

Définition 2.20. Soit α un nombre algébrique. Son *polynôme minimal* est le polynôme m_α de $\mathbb{Q}[X] \setminus \{0\}$ dont α est une racine, de degré minimal et qui divise les autres polynômes qui ont α pour racine.

Exemple 2.21.

— Le polynôme minimal de $\sqrt{2}$ est trivialement $m_{\sqrt{2}}(X) = X^2 - 2$.

— De manière générale, pour tout $a \in \mathbb{N} \setminus \{0, 1\}$, le polynôme minimal de $\sqrt{a}$ est $m_{\sqrt{a}}(X) = X^2 - a$.

— Le polynôme minimal du nombre d'or $\varphi = \frac{1+\sqrt{5}}{2}$ est $m_\varphi(X) = X^2 - X - 1$.

Définition 2.22. Soit α un nombre algébrique. Son *degré* est le degré de son polynôme minimal.

Rappelons que les rationnels sont algébriques. En effet, tout $\frac{p}{q} \in \mathbb{Q}$ est trivialement racine de $X - \frac{p}{q}$, ce dernier étant son polynôme minimal. Les rationnels sont donc exactement les nombres algébriques de degré 1.

Définition 2.23. Un nombre réel ξ est *approximable à l'ordre* $n \in \mathbb{N}$ s'il existe un nombre rationnel $\frac{p}{q}$ tel que

$$\left|\xi - \frac{p}{q}\right| < \frac{1}{q^n}.$$

Un premier théorème important concernant l'approximation diophantienne des nombres algébriques est le suivant :

Théorème 2.24. *Un nombre algébrique irrationnel α de degré d n'est approximable au maximum qu'à l'ordre d, c'est à dire qu'il existe une constante positive C (dépendant de α) telle que*

$$\left|\alpha - \frac{p}{q}\right| \geq \frac{C}{q^d}$$

pour tout rationnel $\frac{p}{q}$.

Démonstration. Soit α algébrique de degré d, c'est à dire qu'il existe $P \in \mathbb{Q}[X]$ de degré d tel que

$$P(\alpha) = a_d\alpha^d + a_{d-1}\alpha^{d-1} + \cdots + a_1\alpha + a_0 = 0.$$

Quitte à multiplier les coefficients de P par le plus petit commun multiple de leurs dénominateurs, on peut supposer $P \in \mathbb{Z}[X]$. Le polynôme P étant continu et dérivable sur $[\alpha - 1, \alpha + 1]$, il existe un nombre M (dépendant de α) tel que

$$|P'(x)| < M \quad \text{pour tout} \quad \alpha - 1 < x < \alpha + 1.$$

Soit $\frac{p}{q}$ un rationnel tel que $\alpha - 1 < \frac{p}{q} < \alpha + 1$, et tel que $\frac{p}{q}$ soit plus proche de α que de toutes les autres racines de P. On a que $P\left(\frac{p}{q}\right) \neq 0$. Alors

$$\left|P\left(\frac{p}{q}\right)\right| = \frac{|a_d p^d + a_{d-1}p^{d-1}q + \cdots + a_1 pq^{d-1} + a_0 q|}{q^d} \geq \frac{1}{q^d},$$

et le théorème des accroissements finis donne

$$P\left(\frac{p}{q}\right) = P\left(\frac{p}{q}\right) - P(\alpha) = \left(\frac{p}{q} - \alpha\right) P'(x)$$

pour un certain x entre $\frac{p}{q}$ et α. L'on en déduit, avec $C = \frac{1}{M}$,

$$\left|\frac{p}{q} - \alpha\right| = \frac{\left|P\left(\frac{p}{q}\right)\right|}{|P'(x)|} \geq \frac{1}{Mq^d} = \frac{C}{q^d}.$$

□

Ce théorème nous donne un premier critère de transcendance : un nombre irrationnel algébrique n'est approximable par des rationnels qu'à un certain ordre (égal à son degré). Voyons un exemple de comment l'appliquer pour prouver la transcendance d'une certaine classe de nombres.

Corollaire 2.25 (Liouville, 1844)**.** *Le nombre* $\xi := \sum_{n\geq 1} 10^{-n!}$ *est transcendant.*

Démonstration. Comme la 10-représentation de ξ n'est clairement pas ultimement périodique, il est irrationnel. Pour tout entier $n \geq 2$, posons les suites $q_n = 10^{(n-1)!}$ et $p_n = q_n(10^{-1!} + \cdots + 10^{-(n-1)!})$. On a alors

$$\begin{aligned}\left|\xi - \frac{p_n}{q_n}\right| &= \sum_{n\geq 1} 10^{-n!} - (10^{-1!} + \cdots + 10^{-(n-1)!}) \\ &= \sum_{m\geq n} 10^{-m!} \leq \frac{2}{10^{n!}} = \frac{2}{q_n^n}.\end{aligned}$$

Nous venons d'exhiber une suite de rationnels $(\frac{p_n}{q_n})_{n\geq 2}$ qui approximent ξ à l'ordre n, pour tout entier $n \geq 2$. Le nombre ξ n'est donc ni algébrique de degré $n \geq 2$, ni rationnel, il est donc transcendant. □

Le nombre ξ défini au corollaire précédent correspond à la construction introduite par Joseph Liouville [34], et constitue historiquement le premier exemple de ce que l'on appelle un *nombre de Liouville*. Ce dernier fut le premier à démontrer l'existence de nombres transcendants et à en proposer une construction explicite. La définition d'un nombre de Liouville est la suivante :

Définition 2.26. Un *nombre de Liouville* λ est un réel qui est approximable à l'ordre n par un rationnel, pour tout entier positif n. Autrement dit, λ est un nombre de Liouville si pour tout entier positif n,

$$\left|\lambda - \frac{p}{q}\right| < \frac{1}{q^n}$$

possède une solution rationnelle $\frac{p}{q}$.

Il est donc clair par leur définition et par le théorème 2.24 que tout nombre de Liouville est transcendant. Cependant tous les nombres transcendants ne sont pas de Liouville, bien que ces derniers en forment une large classe :

Théorème 2.27. *L'ensemble des nombres de Liouville est non-dénombrable.*

Démonstration. Procédons de manière analogue à la preuve du corollaire 2.25, mais en construisant un nombre écrit dans une base entière $b \geq 2$ quelconque. Soit $\lambda := \sum_{n\geq 1} a_n b^{-n!}$, avec pour tout n, $a_n \in \Sigma_b = \{0, 1, \ldots, b-1\}$. Pour tout $n \geq 2$, posons les suites $q_n = b^{-(n-1)!}$ et $p_n = q_n \sum_{k=1}^{n-1} a_k b^{-k!}$. On a

$$\begin{aligned}\left|\lambda - \frac{p_n}{q_n}\right| &= \sum_{n\geq 1} b^{-n!} - \sum_{k=1}^{n-1} a_n b^{-k!} \\ &= \sum_{m\geq n} a_m b^{-m!} \leq (b-1) \cdot \sum_{m\geq n} b^{-m!} < (b-1) \cdot \frac{2}{b^{n!}} = \frac{2(b-1)}{q_n^n}.\end{aligned}$$

Ainsi, tout nombre de la forme $\sum_{n\geq 1} a_n b^{-n!}$, avec les $a_n \in \Sigma_b$ est approximable à l'ordre n, pour tout n, et est donc de Liouville. L'ensemble des suites $\Sigma_b^{\mathbb{N}}$ n'étant pas dénombrable, l'on obtient que l'ensemble des nombres de Liouville non plus n'est pas dénombrable. □

2.3.3 Mesure d'irrationalité et classification de Mahler

Une notion qui va nous permettre de généraliser le théorème 2.24 est celle de mesure d'irrationalité, dont la définition est la suivante.

Définition 2.28. Soit ξ un nombre réel. On dit que le réel μ est une *mesure d'irrationalité* pour ξ si, l'équation

$$\left|\xi - \frac{p}{q}\right| < \frac{1}{q^\mu}$$

possède une infinité de solutions rationnelles $\frac{p}{q}$. On dit que *la* mesure d'irrationalité de ξ, notée $\mu(\xi)$, est la borne supérieure des réels qui sont une mesure d'irrationalité pour ξ.

La mesure d'irrationalité de ξ permet donc de mesurer à quelle "précision" ce réel ξ peut-être approximé par des rationnels. Ainsi, le théorème 2.18 nous permet de déduire que les nombres qui ont pour mesure d'irrationalité 1 sont exactement les rationnels, et que la mesure d'irrationalité de tout nombre irrationnel est au moins 2. Le théorème 2.24 permet quant à lui d'affirmer que le degré d'un irrationnel algébrique α est une borne supérieure de sa mesure d'irrationalité pour $\mu(\alpha)$. Un résultat plus fort a cependant été obtenu par Roth en 1955.

Théorème 2.29 (Roth, [44])**.** *Soit α un nombre algébrique de degré $d \geq 2$. Alors, pour tout réel ε positif, il existe une constante C (dépendante de α et ε) telle que*

$$\left|\alpha - \frac{p}{q}\right| > \frac{C}{q^{2+\varepsilon}},$$

pour tout rationnel $\frac{p}{q}$, avec $q > 0$.

Donc, la mesure d'irrationalité d'un irrationnel algébrique est exactement 2. D'autre part, par conséquence de sa définition, tout nombre de Liouville λ vérifie $\mu(\lambda) = +\infty$. Les irrationnels algébriques apparaissent ainsi comme les nombres les moins bien approximables par des rationnels, tandis que les nombres de Liouville se situent à l'opposé, puisqu'ils admettent des approximations rationnelles aussi précises que l'on veut. Ces deux classes apparaissent alors comme aux deux extrêmes d'un même spectre, du point de vue de leur mesure d'irrationalité.

Distinguer alors simplement les nombres algébrique des transcendants peut sembler relativement insatisfaisant, dans la mesure où presque tout réel est transcendant. Une manière de raffiner cette classification consiste à généraliser cette notion d'approximation en considérant non plus seulement les approximations par des rationnels, c'est-à-dire des nombres algébriques de degré 1, mais plus généralement par des nombres algébriques de degré quelconque. C'est dans cet esprit que Mahler introduit en 1932 [36] une classification plus fine des nombres réels selon la précision de leur approximation par des nombres algébriques. Commençons par donner nous les définitions nécessaires à la description de ces classes.

Définition 2.30. Soit $P(X) = a_dX^d + a_{d-1}X^{d-1} + \cdots + a_1X + a_0$ un polynôme de $\mathbb{Z}[X]$. Sa *hauteur* est définie par

$$H(P) := \max\{|a_i| \mid 0 \leq i \leq d\},$$

c'est à dire $H(P)$ est le maximum des valeurs absolues des coefficients de P.

Définition 2.31.

— Soient $\xi \in \mathbb{R}$ et $d \in \mathbb{N}$. On définit $w_d(\xi)$ comme le supremum des réels w pour lesquels il existe une infinité de polynômes de $\mathbb{Z}[X]$ de degré inférieur ou égal à d qui satisfont

$$0 < |P(\xi)| < H(P)^{-w}.$$

— On définit

$$w(\xi) := \limsup_{d \to +\infty} \frac{w_d(\xi)}{d}.$$

Remarque 2.32. Remarquons que pour tout réel ξ, on a $w_1(\xi) = \mu(\xi) - 1$. En effet, sans perte de généralité, supposons que ξ soit dans $[0, 1)$. Alors pour une infinité de rationnels $\frac{p}{q}$ dans $[0, 1)$, l'on a

$$\left|\xi - \frac{p}{q}\right| \leq \frac{1}{q^{\mu(\xi)}}.$$

On a alors

$$|q\xi - p| \leq \frac{1}{q^{\mu(\xi)-1}}.$$

Le polynôme $P(X) = qX - p$ est de degré 1 et l'on a $H(P) = q$. Autrement dit, l'on a

$$0 < |P(\xi)| \leq H(P)^{-(\mu(\xi)-1)},$$

ce qui nous donne exactement la définition de $w_1(\xi)$.

Étudier le comportement la suite $(w_d(\xi))_{d\geq 1}$ est donc une généralisation de l'étude de la mesure d'irrationalité de ξ. Remarquons que, pour tout $d \geq 1$, l'on a $0 \leq w_d(\xi) \leq +\infty$. De plus, la suite $(w_d(\xi))_{n\geq 1}$ est clairement croissante : quand d augmente, l'on considère un plus grand nombre de polynômes. On ne peut donc qu'améliorer l'approximation de ξ.

L'on peut alors classer les nombres réels en 4 classes distinctes.

Définition 2.33 (Classification de Mahler)**.** Soit $\xi \in \mathbb{R}$. On dit que ξ est un

— *A-nombre* si $w(\xi) = 0$;
— *S-nombre* si $0 < w(\xi) < +\infty$;
— *T-nombre* si $w(\xi) = +\infty$ et $w_d(\xi) < +\infty$ pour tout $d \geq 1$;
— *U-nombre* si $w(\xi) = +\infty$ et $w_d(\xi) = +\infty$ pour un certain $d \geq 1$.

Les A-nombres correspondent ainsi aux réels les moins bien approximables par des nombres algébriques, tandis que les U-nombres sont ceux qui admettent les approximations algébriques les plus précises.

Théorème 2.34. *Les A-nombres sont exactement les nombres algébriques.*

Théorème 2.35. *Les nombres de Liouville sont des U-nombres.*

Démonstration. Soit λ un nombre de Liouville. De par leur définition (voir définition 2.26), il est clair que pour tout polynôme P de degré 1, l'on ait

$$\sup\{w \in \mathbb{R} : 0 < |P(\lambda)| \leq H(P)^{-w}\} = +\infty,$$

ce qui nous donne $w_1(\lambda) = +\infty$. Comme la suite $(w_n(\lambda))_{n\geq 1}$ est croissante, l'on en déduit que $w(\lambda) = +\infty$ et que λ est un U-nombre. □

Remarque 2.36. Les nombres de Liouville sont en fait les U-nombres *de type 1*. Le type d'un U-nombre ξ est défini comme le plus petit d tel que $w_d(\xi) = +\infty$. Les types sont définis pour les S, T, U-nombres et permettent une classification encore plus fine des nombres transcendants. Le lecteur ou la lectrice souhaitant approfondir cette notion peut consulter, entre autres, [16, Chap. 3].

Ainsi, en plus de distinguer les nombres algébriques des nombres transcendants, cette classification permet une distinction plus fine parmi ces derniers. De plus, les différentes classes possèdent une propriété de stabilité : deux nombres appartenant à la même classe sont *algébriquement dépendants*.

Définition 2.37. Deux nombres réels ξ et η sont *algébriquement dépendants* s'il un polynôme $P \in \mathbb{Z}[X, Y]$ tel que $P(\xi, \eta) = 0$. S'il n'existe pas de tel polynôme P, on dit alors que ξ et η sont *algébriquement indépendants*.

Proposition 2.38. *Deux nombres réels ξ et η algébriquement dépendants appartiennent à la même classe dans la classification de Mahler.*

Notons cependant qu'encore une fois, pour un réel donné, il est souvent difficile en pratique de déterminer à quelle classe il appartient. De plus, un résultat montré par Sprindžuk en 1965 [50] (et déjà conjecturé par Mahler) nous donne qu'en fait, presque tous les réels sont des S-nombres.

Théorème 2.39 (Sprindžuk)**.** *Presque tout réel ξ vérifie $w(\xi) = 1$ (au sens de la mesure de Lebesgue).*

Chapitre 3

Cadre du problème

Maintenant que nous avons défini la notion de b-représentation ainsi que celle de nombre algébrique, il s'agit d'en examiner la complexité du point de vue de la calculabilité. Pour cela, introduisons le cadre théorique dans lequel s'inscrit l'étude de Hartmanis et Stearns, fondé sur le modèle de la machine de Turing. Avant d'expliciter leurs travaux dans la deuxième section de ce chapitre, nous rappelons les principes de la théorie des machines de Turing et les différentes formulations équivalentes de ce modèle. Nous pourrons ainsi présenter la définition de machine de Turing utilisée par les auteurs de la conjecture, ainsi que les classes de complexité temporelle qu'ils ont définies afin de comparer la difficulté de calcul de différentes suites. Enfin, nous verrons dans quelle classe de complexité se situent certaines classes de nombres, ce qui nous permettra d'introduire la conjecture.

3.1 Machines de Turing et calculabilité

Introduite par Turing dans les années 1930, une machine de Turing peut être vue, de manière informelle, comme un automate fini muni d'un ruban infini sur lequel elle lit, écrit et déplace une tête de lecture, ce qui constitue un outil pour représenter les algorithmes et formaliser la notion de calcul effectif, et sur lequel va se baser notre étude de la complexité des b-représentations des nombres. Cette section a été rédigée principalement avec des éléments repris du cours "Algorithmique et calculabilité" [21], les autres source utiles à sa rédaction étant [29, 30, 49].

Nous nous réserverons un symbole particulier #, appelé le *symbole blanc*, ainsi que les symboles `L` et `R`, utilisés pour "left" et "right".

3.1.1 Définitions

Définition 3.1. Une *machine de Turing* (à une bande) est la donnée d'un quintuple $\mathcal{M} = (Q, q_0, h, \Sigma, \delta)$ où

- Q est un ensemble fini non-vide, appelé ensemble des *états*,
- q_0 et h sont des éléments de Q, appelés respectivement *état initial* et *état final*,
- Σ est un alphabet contenant le symbole blanc # mais ne contenant pas les symboles `L` et `R`,

— $\delta : Q \setminus \{h\} \times \Sigma \to Q \times (\Sigma \cup \{\mathtt{L}, \mathtt{R}\})$ est une fonction partielle, appelée *fonction de transition*.

Une manière pratique pour représenter une machine de Turing $\mathcal{M} = (Q, q_0, h, A, \delta)$ est en utilisant un graphe orienté, dont les sommets sont les états de la machine et les arêtes sont données par sa fonction de transition. Pour tous états $p, q \in Q$, et symboles $\sigma \in \Sigma$ et $x \in \Sigma \cup \{\mathtt{L}, \mathtt{R}\}$ tels que $\delta(p, \sigma) = (q, x)$, on dessine un arc de p vers q, étiqueté par σ, x. De plus, l'état initial q_0 est désigné par une flèche entrante, et l'état final h par un double cercle, similairement à la représentation d'un automate fini.

Exemple 3.2. Considérons la machine de Turing

$$(\{0, 1, 2, \ldots, 8\}, 0, 8, \{\#, \mathtt{u}\}, \delta)$$

dont la fonction de transition est donnée par la table suivante :

	#	u
0	(1, L)	(0, L)
1	(5, R)	(2, #)
2	(3, R)	/
3	(4, u)	/
4	(0, u)	(4, R)
5	(6, u)	/
6	(7, L)	(6, R)
7	/	(8, #)

TABLE 3.1 – Table de transition de δ.

Un graphe orienté pour représenter cette machine est le suivant :

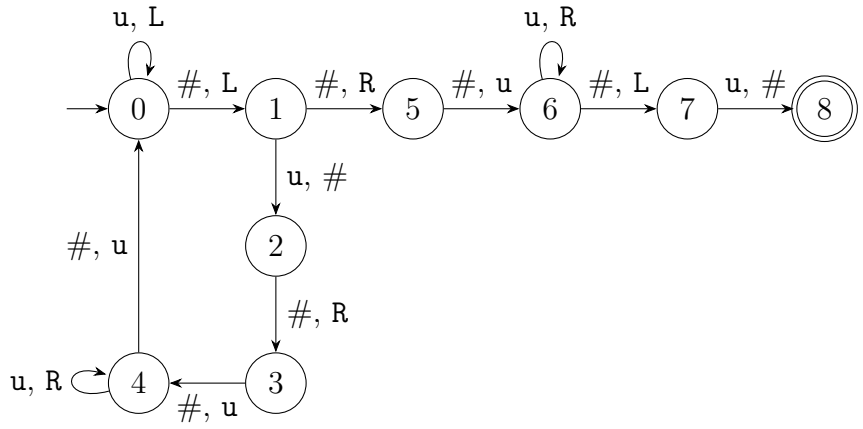


FIGURE 3.1 – Machine de Turing calculant la multiplication par 2 en unaire.

Cette machine prend en entrée sur le ruban un entier naturel n écrit en base unaire, c'est à dire un mot de la forme $\mathtt{u}^n$, et effectue sa multiplication par deux. Donc, quand elle arrive sur son état final, sur le ruban se trouve le mot $\mathtt{u}^{2n}$.

Définissons plus précisément comment une machine de Turing interagit avec le ruban :

Définition 3.3.

— Une *configuration mémoire* est un couple $(w, k) \in \Sigma^{\mathbb{N}} \times \mathbb{N}$ où le mot infini w ne contient qu'un nombre fini de fois le symbole blanc #. Le mot infini w est appelé le ruban mémoire et l'entier k est un pointeur qui pointe sur la k-ème lettre de w. On dit aussi que k désigne une cellule référencée qui contient le symbole w_k. Pour simplifier les écritures, on renseigne souvent la partie significative d'une configuration mémoire.

— La *partie significative* r *d'une configuration mémoire* (w, k) telle que $w_l \neq \#$ et $w_n = \#$ pour tout $n > l$ est définie par

$$r = \begin{cases} w_1 \cdots w_{k-1} \underline{w_k} w_{k+1} \cdots w_l & \text{si } l \geq k \\ w_1 \cdots w_l \#^{k-l-1} \underline{\#} & \text{si } l < k. \end{cases}$$

On souligne donc le symbole de la partie significative r pour indiquer où se trouve la tête de lecture/écriture de la machine.

— Une *configuration machine* est un triplet (p, w, k) où p est un état et (w, k) est une configuration mémoire. Une configuration machine (p, w, k) est souvent notée $p.r$ où r est la partie significative de la configuration mémoire (w, k). Une machine de Turing $\mathcal{M}$ agit sur les configurations machine de la manière suivante :

  — Si on se trouve dans la configuration machine (p, w, k) et que $\delta(p, w_k) = (q, \sigma)$ avec $\sigma \in \Sigma$, alors on bascule dans la configuration machine (q, w', k) où

$$w'_n = \begin{cases} w_n & \text{si } n \neq k \\ \sigma & \text{si } n = k. \end{cases}$$

  — Si on se trouve dans la configuration machine (p, w, k) et que $\delta(p, w_k) = (q, \mathtt{R})$, alors on bascule dans la configuration machine $(q, w, k + 1)$.

  — Si on se trouve dans la configuration machine (p, w, k) avec $k \geq 2$ et que $\delta(p, w_k) = (q, \mathtt{L})$, alors on bascule dans la configuration machine $(q, w, k - 1)$.

— La notation $C \vdash C'$ signifie que C' est une configuration machine atteignable depuis C. On note $C \vdash^* C'$ s'il existe $j \geq 0$, des configurations machine $C_0, \ldots, C_j$ telles que l'on ait

  — $C = C_0$;

  — $C' = C_j$;

  — $C_i \vdash C_{i+1}$ pour tout $i \in \{0, 1, \ldots, j - 1\}$.

— Une *configuration pendante* est une configuration machine (p, w, k) avec $p \neq h$, depuis laquelle aucune configuration machine n'est atteignable. Cela peut se produire soit lorque la fonction de transition δ n'est pas définie en (p, w_k), soit lorsque $k = 1$ et $\delta(p, w_1) = (q, \mathtt{L})$. Enfin, si $d = h$, on parle de configuration d'arrêt.

Trois situations peuvent se produire lorsqu'on lance une machine de Turing à partir d'une configuration machine donnée :

1. La machine de Turing aboutit dans une configuration d'arrêt en un nombre fini de transitions, auquel cas on dit que la machine de Turing s'arrête.
2. La machine de Turing atteint une configuration pendante.
3. Une infinité de transitions successives sont possibles, auquel cas on dit que la machine de Turing ne s'arrête pas.

3.1.2 Fonctions calculables par machines de Turing

Maintenant que nous avons un aperçu du fonctionnement des machines de Turing, nous décrivons la notion de fonction calculable par ces machines.

Définition 3.4. Soient $A_1, \dots, A_{d+1}$ des alphabets ne contenant pas le symbole blanc #. Une fonction $f : A_1^* \times \cdots \times A_d^* \to A_{d+1}^*$ est *calculable* s'il existe une machine de Turing $\mathcal{M} = (Q, q_0, h, \Sigma, \delta)$ telle que $\bigcup_{i=1}^{d+1} A_i \subseteq \Sigma$ et telle que pour tout $(w_1, \dots, w_d) \in A_1^* \times \cdots \times A_d^*$ on ait

$$q_0 \cdot \#w_1\# \cdots \#w_d\underline{\#} \vdash^* h \cdot \#f(w_1, \dots, w_d)\underline{\#}.$$

On note $\mathcal{C}$ l'ensemble des fonctions calculables.

Dans ce mémoire, nous allons uniquement nous intéresser aux fonctions de $\mathbb{N}$ dans $\mathbb{N}$ (nous utiliserons les termes de "fonction" et de "suite" de manière interchangeable), et en particulier aux suites de $\Sigma_b^{\mathbb{N}}$. Définissons alors un codage des entiers naturels qui nous permettra de nous raccrocher à la définition précédente. Celui que nous utiliserons ici est celui en base unaire (bien qu'il en existe d'autres, c'est un choix arbitraire). On se réserve un symbole spécial $\mathtt{u}$, et l'on code tout $n \in \mathbb{N}$ par le mot $\mathtt{u}^n$.

Exemple 3.5. Montrons que la machine de Turing de l'exemple précédent calcule en effet la fonction $f : \mathbb{N} \to \mathbb{N}, m \mapsto 2m$. En effet, pour tous $m, n \in \mathbb{N}$ avec $m \geq 1$, on a

$$\begin{aligned}
0.\#\mathtt{u}^m\underline{\#}\mathtt{u}^n &\vdash 1.\#\mathtt{u}^{m-1}\underline{\mathtt{u}}\#\mathtt{u}^n \\
&\vdash 2.\#\mathtt{u}^{m-1}\underline{\#}\#\mathtt{u}^n \\
&\vdash 3.\#\mathtt{u}^{m-1}\#\underline{\#}\mathtt{u}^n \\
&\vdash 4.\#\mathtt{u}^{m-1}\#\underline{\mathtt{u}}\mathtt{u}^n \\
&\vdash^* 4.\#\mathtt{u}^{m-1}\#\mathtt{u}^{n+1}\underline{\#} \\
&\vdash 0.\#\mathtt{u}^{m-1}\#\mathtt{u}^{n+1}\underline{\mathtt{u}} \\
&\vdash^* 0.\#\mathtt{u}^{m-1}\underline{\#}\mathtt{u}^{n+2}.
\end{aligned}$$

En itérant cet argument, on obtient que pour tout $m \in \mathbb{N}$, on a

$$0.\#\mathtt{u}^m\underline{\#} \vdash^* 0.\#\#\underline{\#}\mathtt{u}^{2m}.$$

De plus, on a

$$\begin{aligned} 0.\#\underline{\#}\mathtt{u}^{2m} &\vdash 1.\underline{\#}\#\mathtt{u}^{2m} \\ &\vdash 5.\#\underline{\#}\mathtt{u}^{2m} \\ &\vdash 6.\#\underline{\mathtt{u}}\mathtt{u}^{2m} \\ &\vdash^* 6.\#\mathtt{u}^{2m+1}\underline{\#} \\ &\vdash 7.\#\mathtt{u}^{2m}\underline{\mathtt{u}} \\ &\vdash 8.\#\mathtt{u}^{2m}\underline{\#} \end{aligned}$$

Donc, avec entrée un mot de la forme $\mathtt{u}^m$ sur le ruban, pour tout symbole $\mathtt{u}$ lu, la machine le supprime et en écrit 2 autres. Puis elle déplace la tête de lecture à la fin du mot nouvellement créé, qui est de la forme $\mathtt{u}^{2m}$. Elle calcule bien la fonction annoncée.

Même si les machines de Turing peuvent à première vue représenter un outil de calcul relativement puissant, certaines fonctions restent néanmoins hors de leur portée.

Proposition 3.6. *Il existe des fonctions de* $\mathbb{N}$ *dans* $\mathbb{N}$ *qui ne sont pas calculables.*

Démonstration. L'ensemble des machines de Turing est dénombrable. En effet, pour tout $n \in \mathbb{N}$, il n'existe qu'un nombre fini de machines à n états. Ainsi, on peut établir une bijection entre $\mathbb{N}$ et l'ensemble des machines de Turing en listant chaque machine à n états, pour tout $n \in \mathbb{N}$. Ainsi, $\mathbb{N}^{\mathbb{N}} \cap \mathcal{C}$ est dénombrable. L'ensemble des fonctions de $\mathbb{N}$ dans $\mathbb{N}$ n'est cependant pas dénombrable. Il existe donc des fonctions de $\mathbb{N}^{\mathbb{N}}$ non-calculables. □

Remarquons que l'argument utilisé ici, consistant à énumérer les fonctions calculables, est similaire à celui utilisé par Cantor dans la preuve du théorème 2.16. Cette preuve n'exhibe cependant pas explicitement de fonction non-calculable, alors que certaines sont connues à ce jour, citons par exemple le "Castor affairé" (ou Busy Beaver) [41].

3.1.3 Variantes de machines de Turing

La définition de machine de Turing que nous avons adoptée jusqu'à présent diffère de celle utilisée par Hartmanis et Stearns [29]. En effet, la notion de machine de Turing s'avère relativement souple : de nombreuses variantes du modèle ont été proposées, modifiant par exemple le nombre de rubans, le ruban lui-même, ou la nature des mouvements possibles. Cependant, ces différences ne changent pas la puissance de calcul du modèle, toutes ces définitions restant équivalentes du point de vue des fonctions calculables [21, Chap. 2.12]. Cette sous-section est donc consacrée à la présentation brève de quelques variations de la définition standard de machine de Turing.

Définition 3.7. Une *machine de Turing à ruban bi-infini* $\mathcal{M} = (Q, q_0, h, \Sigma, \Delta)$ est similaire à une machine de Turing standard, à ceci près que le ruban mémoire est un mot bi-infini écrit sur Σ.

Les machines à ruban bi-infini n'ont donc pas de contrainte de déplacement de la tête de lecture sur le ruban. Elle peut librement se déplacer à l'infini que ce soit vers la gauche ou vers la droite. Cependant, "doubler" la taille du ruban ne permet pas à ces machines de calculer plus de fonctions que le modèle standard.

Théorème 3.8. *Les machines de Turing à ruban bi-infini calculent les mêmes fonctions que les machines de Turing standards.*

Une autre variation possible est d'autoriser la machine à travailler sur plusieurs rubans en parallèle.

Définition 3.9. Une *machine de Turing à r bandes* $\mathcal{M} = (Q, q_0, h, \Sigma, \Delta)$ est similaire à une machine de Turing standard, à ceci près que la mémoire est constituée de r rubans.

Le graphe d'une machine de Turing à r bandes est adapté comme suit : pours tous $p, q \in Q, \sigma_1, \ldots, \sigma_r \in \Sigma$ et $x_1, \ldots, x_r \in \Sigma \cup \{\mathtt{L}, \mathtt{R}\}$ tels que $\delta(p, a_1, \cdots, a_r) = (q, x_1, \ldots, x_r)$, on dessine un arc de p vers q étiqueté par $(a_1, \ldots, a_r), (x_1, \ldots, x_r)$.

Exemple 3.10. La machine suivante est une machine de Turing à 2 bandes, qui effectue la multiplication par 2.

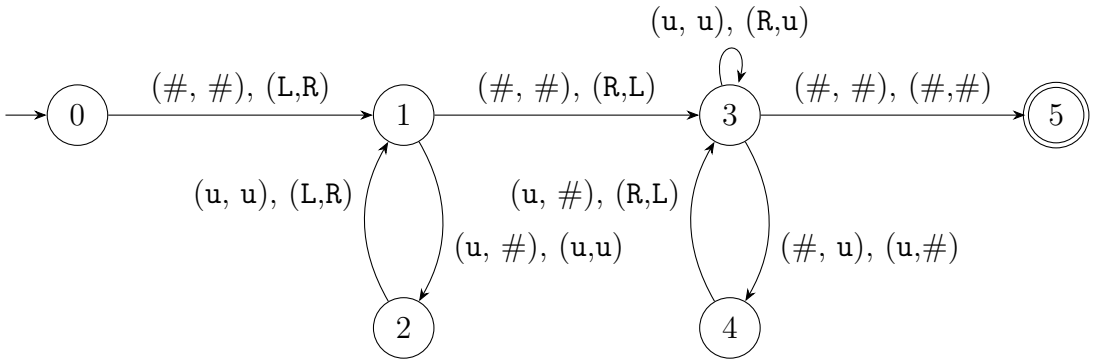


FIGURE 3.2 – Une machine de Turing à 2 bandes, qui calcule $n \mapsto 2n$.

De manière similaire, augmenter le nombre de bandes n'augmente pas la puissance de calcul des machines de Turing.

Théorème 3.11. *Les machines de Turing multibandes calculent les mêmes fonctions que les machine de Turing standards.*

Plutôt que modifier la forme ou le nombre des rubans mémoire, l'on peut s'intéresser aux modifications de la fonction de transition. Considérons maintenant une fonction de transition de la forme

$$\delta : Q \setminus \{h\} \times \Sigma \to Q \times \Sigma \times \{\mathtt{L}, \mathtt{R}, \mathtt{S}\},$$

avec un nouveau symbole réservé $\mathtt{S}$, qui codera l'instruction "stationnaire". Une machine de Turing avec une fonction de transition de la sorte peut à chaque étape écrire

un symbole sur le ruban et se déplacer, là où une machine standard ne peut effectuer qu'une seule de ces deux actions. L'on se convaincra assez facilement que cette définition de machine de Turing n'accorde pas plus de puissance que la définition standard.

L'équivalence de puissance de calcul est permise grâce à une classe particulière de machines, les *machines de Turing universelles*, déjà conceptualisées par Turing lui-même dans son article originel [51]. En se donnant un codage c d'une machine et de ses configurations mémoire, l'on peut concevoir une machine universelle $\mathcal{U}$ qui aura le code stocké en mémoire, et qui pourra simuler la machine codée [1].

Proposition 3.12. *Pour toute machine de Turing $\mathcal{M}$, on peut construire des machines de Turing réalisant les actions suivantes :*

- $q_0.\underline{\#} \vdash^* h.\#c(\mathcal{M})\underline{\#}$
- $q_0.\#w\underline{\#} \vdash^* h.\#c(\mathcal{M})c(w)\underline{\#}$

quelque soit le mot d'entrée w ne contenant pas le symbole $\#$.

Théorème 3.13. *Il existe une machine de Turing $\mathcal{U}$ telle que pour toute machine de Turing $\mathcal{M}$ et toute configuration mémoire r de $\mathcal{M}$,*

- *$\mathcal{U}$ atteint une configuration d'arrêt à partir de $q_0.\#c(\mathcal{M})c(w)\underline{\#}$ si $\mathcal{M}$ en atteint une à partir de $q_0.\#w\underline{\#}$, auquel cas, la configuration d'arrêt de $\mathcal{U}$ est $h.\#\rho(u\underline{a}v)\underline{\#}$ si $h.u\underline{a}v$ est la configuration d'arrêt de $\mathcal{M}$.*
- *$\mathcal{U}$ atteint une configuration pendante à partir de $q_0.\#c(\mathcal{M})c(w)\underline{\#}$ si $\mathcal{M}$ en atteint une à partir de $q_0.\#w\underline{\#}$.*
- *$\mathcal{U}$ ne s'arrête pas à partir de $q_0.\#c(\mathcal{M})c(w)\underline{\#}$ si $\mathcal{M}$ ne s'arrête pas partir de $q_0.\#w\underline{\#}$.*

Nous ne détaillerons pas ici la construction d'une telle machine de Turing universelle, mais la lectrice ou le lecteur peut se référer à [21, Chap 2.9] pour un exemple de codage c. De nombreux auteurs ont proposé différents codages de machines, et donc différentes machines de Turing universelles. Citons par exemple les travaux de Shannon [47], qui a proposé une machine de Turing universelle à 2 états seulement.

Les variations de machine présentées ici sont en fait celles qui différencient le modèle utilisé par Hartmanis et Stearns du modèle standard que nous avons utilisé juqu'à maintenant. L'on peut cependant imaginer bien d'autres variations de la définition : rubans multidimensionnels, plusieurs têtes de lecture/écriture, machines non-déterministes... En fait, dès lors que la machine que l'on décrit possède un nombre fini d'états et travaille avec un alphabet fini, réalise une quantité finie de travail à chaque opération et possède un accès constant à de la mémoire infinie, elle possède la même puissance de calcul qu'une machine de Turing standard [2].

1. Comme le montrent Hopcroft et Ullman [30, Chap. 8.6] , les machines de Turing sont également capables de simuler n'importe quel ordinateur. Cela signifie que certaines fonctions seront toujours inaccessibles à ces derniers, quelle que soit leur vitesse de calcul.

2. La lectrice ou le lecteur curieux peut consulter, entre autres, [30, Chap. 8.4, 8.5] et [29], qui listent des modèles seulement évoqués ou non-mentionnés ici.

3.2 Problème de Hartmanis–Stearns

En 1965, Hartmanis et Stearns publient l'article *On the computational complexity of algorithms* [29], ce qui marque la naissance de la théorie de la complexité. Prolongeant le cadre posé par Turing pour la notion de calculabilité, ils introduisent pour la première fois une formalisation du temps de calcul des machines de Turing. Leur objectif n'est pas seulement de distinguer les fonctions calculables de celles qui ne le sont pas, mais de proposer une classification plus fine des fonctions calculables en fonction de la croissance du nombre d'opérations nécessaires à leur calcul. Ils introduisent alors les *classes de complexité temporelle*, définies à partir de fonctions bornant le temps requis pour produire les premiers symboles d'une suite.

Dans ce cadre, l'étude des nombres réels calculables devient donc un cas particulier de l'étude de la complexité temporelle des suites calculables. Un nombre réel est alors considéré d'autant plus simple que son développement en base b peut être généré rapidement par une machine. Comme nous le verrons à la fin de cette section, les nombres algébriques sont calculables, bien qu'ils ne tombent pas tous dans la même classe de complexité temporelle. Un autre résultat surprenant est que certains nombres transcendants peuvent, admettre des représentations simples du point de vue algorithmique. Ce sont ces deux résultats qui donnent lieu à la conjecture de Hartmanis et Stearns.

Dans cette section, nous commencerons par présenter le modèle de machine de Turing adopté par Hartmanis et Stearns dans leurs travaux, et par montrer son équivalence en puissance de calcul avec le modèle standard de Turing. Nous introduirons ensuite la notion de nombre calculable par une machine de Turing. Nous étudierons enfin les classes de complexité temporelle définies par les auteurs, avant d'en tirer une classification des nombres réels selon la complexité de leur b-représentation. Les principales sources utilisées pour la rédaction de cette section sont [21, 24, 25, 26, 29].

3.2.1 Modèle de machine utilisé dans le problème

Le modèle utilisé par Hartmanis et Stearns est une machine de Turing multi-bande, dont les rubans sont bi-infinis, et qui peut, à chaque étape, écrire un symbole et se déplacer sur chaque ruban. Parmi les rubans, l'un est distingué et désigné comme ruban de sortie. Sur ce ruban, la tête de lecture/écriture ne peut pas se déplacer à gauche, elle peut seulement rester stationnaire ou aller à droite. Les symboles écrits sur le ruban de sortie, une fois dépassés par la tête de lecture, sont donc définitifs et ne peuvent plus intervenir dans les calculs ultérieurs. La définition suivante est adaptée de l'article original [29], et consiste en la définition formelle de leur modèle.

Définition 3.14. Une *machine de Turing à r bandes* $\mathcal{M}$ est la donnée d'un quadruplet (Q, q_0, Σ, δ) où

— Q, l'ensemble des états, est fini et non-vide,

— $q_0 \in Q$ est l'état initial,

— Σ est un alphabet fini,

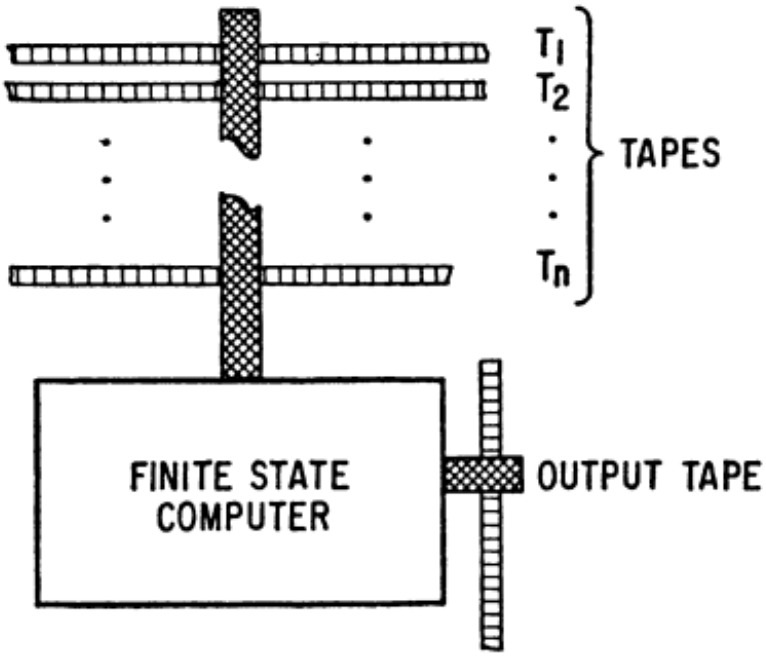


FIGURE 3.3 – Schéma du modèle de machine utilisé par Hartmanis et Stearns dans [29]

— la fonction de transition δ est de la forme

$$\delta : Q \times \Sigma^r \longrightarrow \Sigma^{r+1} \times \{\mathtt{R}, \mathtt{S}\} \times \{\mathtt{L}, \mathtt{R}, \mathtt{S}\}^r \times Q.$$

Un élément de la fonction de transition δ est donc un $(3r + 4)$-uple de la forme

$$(q_i, S_{i_1}, \ldots, S_{i_r}, S_{j_0}, \ldots, S_{j_r}, m_0, \ldots, m_r, q_j)$$

où :

— q_i est l'état actuel,
— $S_{i_1}, \ldots, S_{i_r}$ sont les symboles lus sur les bandes, respectivement $T_1, \ldots, T_r$,
— $S_{j_0}, \ldots, S_{j_r}$ sont les nouveaux symboles qui seront écrits,
— $m_0, \ldots, m_n$ les mouvements du curseur sur chaque bande (`L`, `R` ou `S`),
— q_j est le nouvel état.

T_0 est la bande de sortie de la machine, sur laquelle le curseur ne peut se déplacer qu'à droite, c'est à dire $m_0 \in \{\mathtt{R}, \mathtt{S}\}$. L'on suppose de plus que la machine commence son calcul avec tous ses rubans vides.

Remarquons que la machine n'a pas d'état final. En effet, contrairement au modèle standard présenté plus tôt (qui suit la définition originale de Turing), qui calcule une valeur $f(k)$ à partir d'une entrée unique de taille k puis s'arrête, la machine introduite par Hartmanis et Stearns est conçue pour démarrer sans entrée sur les rubans, et pour produire successivement les valeurs $f(1), f(2), f(3), \ldots$ à l'infini sur son ruban de sortie. Notons également que la bande de sortie n'est pas

comptée dans les r bandes. Ainsi, une machine de Turing avec 0 bandes possède tout de même une bande de sortie, et s'apparente à un automate fini déterministe avec sortie (qui fera par ailleurs l'objet du prochain chapitre de ce mémoire).

Encore une fois, ce changement de point de vue n'altère pas la puissance de calcul : les machines au sens de Hartmanis et Stearns calculent les mêmes fonctions que les machines qui suivent la définition de Turing.

Proposition 3.15. *Soit Σ un alphabet fini et $\boldsymbol{a} = (a_n)_{n\geq 1}$ une suite de $\Sigma^{\mathbb{N}}$. Les deux propositions suivantes sont équivalentes :*

- *Il existe une machine de Turing multibande $\mathcal{M}$ qui, pour tout $n \in \mathbb{N}$, avec pour entrée $\mathtt{u}^n$, s'arrête et renvoie le mot $a_1 \cdots a_n$.*
- *Il existe une machine de Turing multibande $\mathcal{M}_\infty$ qui, avec toutes ses bandes vides au départ, ne s'arrête jamais et produit la suite $a_1 a_2 a_3 \cdots$ sur sa bande de sortie.*

Démonstration. Supposons d'abord avoir une machine de Turing $\mathcal{M}_\infty$ au sens de Hartmanis et Stearns, qui produit à l'infini les éléments de $\boldsymbol{a}$. L'on se donne alors une machine de Turing universelle $\mathcal{U}$, qui prend en entrée le mot $\mathtt{u}^n$, et qui simule la machine $\mathcal{M}_\infty$. La machine peut alors compter à chaque fois que $\mathcal{M}_\infty$ produit un élément de la suite (en décrémentant de 1 symbole le mot d'entrée), et s'arrêter quand elle a produit n éléments (quand le mot d'entrée est vide).

Supposons maintenant avoir une machine de Turing multibande $\mathcal{M}$ à r bandes, qui prend un mot de la forme $\mathtt{u}^n$ en entrée, produit le mot $a_1 \cdots a_n$ et s'arrête. La machine $\mathcal{M}_\infty$ au sens d'Hartmanis et Stearns possède alors $r+1$ bandes (sans compter la bande de sortie). La première bande sert de compteur, et contient un mot de la forme $\mathtt{u}^n$. Pour tout $n \in \mathbb{N}$, la machine simule $\mathcal{M}$ sur ses r autres bandes, et copie le dernier symbole a_n produit sur sa bande de sortie. Elle vide ensuite les r bandes utilisées pour simuler $\mathcal{M}$, incrémente de 1 sa première bande et recommence la simulation avec $\mathtt{u}^{n+1}$. Elle procède ainsi à l'infini, sans jamais s'arrêter.

□

3.2.2 Nombres calculables

Dans la suite de ce travail, nous nous intéresserons plus particulièrement aux suites de $\Sigma_b^{\mathbb{N}}$, qui correspondent aux développements en base b des nombres réels. Ainsi, du point de vue de la calculabilité, la complexité d'un nombre réel est liée à la complexité de sa b-représentation. La notion de nombre calculable remonte en fait déjà aux travaux fondateurs de Turing [51], qui propose une première distinction entre les nombres réels accessibles aux machines de Turing et ceux qui ne le sont pas.

Définition 3.16. Un nombre réel x est *calculable* s'il existe une machine de Turing qui produit sa b-représentation.

Cette définition est indépendante de la base b choisie. En effet, une machine de Turing qui produit la représentation d'un réel x de $[0,1)$ en base $b \geq 2$ nous donne en fait une approximation (à une précision arbitraire) de x par un rationnel de la

forme $\frac{p}{b^n}$, avec $p, n \in \mathbb{N}$. En appliquant l'algorithme glouton dans une autre base entière $c \geq 2$ à ce rationnel $\frac{p}{b^n}$, l'on obtient le début de la c-représentation de x. L'on peut donc produire autant de chiffres de la c-représentation de x que l'on veut, pourvu que l'on produise assez de décimales de sa b-représentation.

Proposition 3.17. *Il existe des nombres non-calculables.*

Démonstration. L'ensemble des suites calculables $\mathcal{C}$ est dénombrable, mais $\Sigma_b^{\mathbb{N}}$ ne l'est pas. L'ensemble $\mathcal{C} \cap \Sigma_b^{\mathbb{N}}$ des nombres calculables est donc dénombrable également. Il existe ainsi des nombres non-calculables. □

Un fait qui peut sembler surprenant est que la grande majorité des nombres réels ne sont pas calculables. On peut cependant s'interroger sur le caractère calculable de certains ensembles de nombres. Un premier résultat à ce sujet est que les nombres algébriques sont calculables.

Proposition 3.18. *Les nombres algébriques sont calculables.*

Démonstration. Soit $\alpha = \sum_{i \geq 1} \alpha_i b^{-i}$ un nombre algébrique, tel que $0 \leq \alpha < 1$, et soit le polynôme $P \in \mathbb{Z}[X]$, de degré minimal d et qui possède α pour racine. Posons le rationnel $a_n \in \mathbb{Q}$, qui est b-décimal et égal à α jusqu'à la n-ème décimale, c'est à dire $a_n = \sum_{i \geq 1} a_{n,i} b^{-i}$ avec

$$a_{n,i} = \begin{cases} \alpha_i & \text{si } 1 \leq i \leq n \\ 0 & \text{sinon.} \end{cases}$$

L'on fixe n suffisamment grand pour que a_n soit plus proche de α que des autres racines de P. Sans perte de généralité, l'on peut supposer

$$P(a_n) \leq 0 \quad \text{et} \quad P(a_n + b^{-n}) > 0,$$

car $a_n \leq \alpha < a_n + b^{-n}$. Décrivons la machine de Turing $\mathcal{M}$ qui va produire les décimales de α, en calculant des approximations successives. La machine commence par produire le développement de a_n, qui est une suite finie de chiffres. Pour les décimales suivantes, la machine procède par étapes. L'on se réserve d bandes qui servent à stocker les puissances de l'approximation courante a_n. Sur ces bandes, au début de chaque étape, les têtes de lecture se trouvent à la position nd. Pour déterminer le symbole suivant, et donc a_{n+1}, la machine essaye toutes les valeurs possibles en commençant par la plus élevée. Elle calcule donc

$$c_{(b-1)} := P(a_n + (b-1)b^{-(n+1)}).$$

Pour cela, elle calcule les puissances de $a_n + (b-1)b^{-(n+1)}$ sur d autres bandes réservées et place les têtes de lecture en position $(n+1)d$, puis opère les multiplications et additions adéquates.

— Si $c_{(b-1)} \leq 0$, alors $a_{n+1,n+1} = b - 1$. Les puissances de a_{n+1} sont alors déjà calculées et la machine efface des d autres bandes les puissances de a_n.

— Si $c_{(b-1)} > 0$, alors la machine réitère l'opération avec

$$c_{(b-2)} := P(a_n + (b-2)b^{-(n+1)}),$$

et ainsi de suite.

— Si pour tout $i \in \{1, 2, \ldots, b-1\}$, l'on a $c_i > 0$, alors $a_{n+1,n+1} = 0$, c'est à dire $a_{n+1} = a_n$, et la machine n'a pas d'autre calcul à effectuer. Elle place les têtes de lecture sur les d bandes réservées à la position $(n+1)d$ et répète le processus pour la décimale suivante, et ce à l'infini.

□

De plus, bien que les nombres transcendants soient "rares" du point de vue de la calculabilité, certaines constantes transcendantes usuelles sont calculables.

Proposition 3.19. *Le nombre π est calculable.*

Démonstration. Une formule donnée par Bailey, Borwein et Plouffe dans [11] pour exprimer π est la suivante :

$$\pi = \sum_{n=0}^{+\infty} \frac{1}{16^n} \left(\frac{4}{8n+1} - \frac{2}{8n+4} - \frac{1}{8n+5} - \frac{1}{8n+6} \right)$$

La somme, la multiplication et l'exponentiation sont toutes calculables, l'on peut donc se donner une machine de Turing qui calcule cette formule. De plus, l'algorithme donné par les auteurs permet d'extraire les décimales de la représentation en base 16 de π [3]. □

Si la notion de calculabilité permet de distinguer les nombres accessibles aux machines de Turing de ceux qui ne le sont pas, elle reste trop peu précise pour rendre compte de la difficulté effective du calcul. En pratique, tous les nombres calculables ne se valent pas : certains possèdent un b-développement très simple à produire, tandis que d'autres requièrent des ressources de calcul plus importantes. La sous-section suivante est consacrée à l'étude de cette complexité, à travers l'analyse du temps nécessaire à une machine de Turing pour produire les premiers symboles d'une suite, et à la classification correspondante introduite par Hartmanis et Stearns.

3.2.3 Classes de complexité

Dans leurs travaux [29], Hartmanis et Stearns considèrent le nombre d'opérations $T(n)$ nécessaire à une machine de Turing pour produire les n premiers éléments d'une suite. Ils définissent ainsi les notions de *fonction temporelle* et de *classe de complexité*.

3. Les auteurs calculent dans ce travail les décimales en base 16 d'autres constantes transcendantes comme π^2 ou $\log(2)$. L'algorithme proposé permet en fait de calculer la b-représentation de tout nombre pouvant s'exprimer sous la forme $\sum_{n\geq 0} \frac{P(n)}{b^{cn}Q(n)}$, avec c un entier positif et P et Q des polynômes de $\mathbb{Z}[X]$. Les travaux récents de Plouffe [40] donnent également une formule qui permet d'approximer π très précisément, et donc de déduire un très grand nombre de ses décimales en base 10.

Définition 3.20. Soit $T : \mathbb{N} \to \mathbb{N}$ une fonction calculable croissante telle que pour tout $n \in \mathbb{N}, T(n) \geq \frac{n}{k}$, pour un certain $k \in \mathbb{N}$.

- On dit que la suite $\boldsymbol{a} = (a_n)_{n\geq 1}$ est *T-calculable* si et seulement si il existe une machine de Turing $\mathcal{M}$ qui produit les n premiers termes de la suite en moins de $T(n)$ opérations.
- On dit alors que T est une *fonction temporelle*.
- On note S_T la classe des suites T-calculables, appelée *classe de complexité*.

La condition $T(n) \geq \frac{n}{k}$ dans la définition précédente découle de la définition de la machine de Turing. Les auteurs autorisent la machine à écrire un nombre fini de symboles sur une même case d'un ruban mémoire, ce qui ne change pas la puissance de calcul, quitte à considérer les symboles composites comme faisant partie d'un alphabet plus grand. Ainsi, pour une machine $\mathcal{M}$ produisant une suite, il existe un entier naturel k tel qu'il y a au maximum k symboles sur une case d'un ruban de $\mathcal{M}$. Si l'on a une suite $(a_n)_{n\geq 1}$ calculée par $\mathcal{M}$ et qui appartient à une classe de complexité S_T, alors $\mathcal{M}$ peut produire au maximum kn symboles après n opérations. Alors, $T(kn) \geq n$, ou en substituant, $T(n) \geq \frac{n}{k}$. Cette condition permet également de ne pas considérer l'ensemble vide comme une classe de complexité. En effet, la condition $T(n) \geq \frac{n}{k}$ implique que toute suite constante est dans S_T, car une machine peut écrire k fois le même symbole pour chaque opération, ce qui signifie que S_T est non-vide.

Remarque 3.21. Les classes de complexité présentées ici sont des classes de fonctions, et non des classes de langages, elles ne correspondent donc pas tout à fait à des classes couramment utilisées dans la littérature actuelle comme P et NP.

Proposition 3.22. *L'ensemble des classes de complexité est dénombrable.*

Démonstration. L'ensemble des fonctions calculables étant dénombrable (car les machines de Turing sont dénombrables), l'ensemble des fonctions temporelles l'est aussi. □

Théorème 3.23. *Toute classe de complexité S_T est récursivement énumérable, c'est à dire qu'il existe une machine de Turing $\mathcal{E}$ qui produit une énumération des éléments de S_T.*

Démonstration. L'ensemble des machines de Turing étant dénombrable, l'ensemble des machines de Turing qui suivent le modèle de la définition 3.14 l'est également. Il existe une machine de Turing $\mathcal{E}$ qui, conformément à la proposition 3.12, produit une énumération $(\mathcal{M}_n)_{n\geq 1}$ de ces machines. Comme la fonction temporelle T est calculable, l'on peut modifier chaque machine $\mathcal{M}_i$ en une autre machine $\mathcal{M}'_i$ de la manière suivante : tant que $\mathcal{M}_i$ produit les n premiers symboles de la suite qu'elle calcule en $T(n)$ opérations, alors le n-ème symbole produit par $\mathcal{M}'_i$ est égal au n-ème produit par $\mathcal{M}_i$. S'il existe un n tel que $\mathcal{M}_i$ produit son n-ème symbole en plus de $T(n)$ opérations, alors la machine $\mathcal{M}'_i$ produit une suite constante de symboles à partir de n (disons de symboles 0). La nouvelle énumération $(\mathcal{M}'_n)_{n\geq 1}$, uniquement composée des machines qui produisent les suites de S_T, peut de manière similaire être produite par une machine $\mathcal{E}'$. □

Corollaire 3.24. *Il n'existe pas de fonction temporelle T telle que S_T est la classe de toute les suites calculables.*

Démonstration. Comme S_T est récursivement énumérable, en suivant une construction diagonale de Cantor, l'on peut construire une machine $\mathcal{D}$, qui produit une suite $(a_n)_{n\geq 1}$ de la manière suivante. Pour tout $n \geq 1$, $\mathcal{D}$ calcule le n-ème terme $\sigma \in \Sigma_b$ de la machine $\mathcal{M}'_n$ de l'énumération construite lors du théorème précédent, et produit le symbole $\sigma + 1 \mod b$. La machine $\mathcal{D}$ produit alors une suite différente de toute suite de S_T, et n'est donc pas dans S_T. □

Corollaire 3.25. *Pour toute fonction temporelle T, il existe une fonction temporelle U telle que $S_T \subset S_U$. Il existe donc des suites $(T_n)_{n\geq 1}$ de fonctions temporelles telles que les classes de complexités forment une suite strictement croissante*

$$S_{T_1} \subsetneq S_{T_2} \subsetneq S_{T_3} \subsetneq \cdots$$

de classes distinctes.

Démonstration. Soient T une fonction temporelle et $\mathcal{M}$ une machine qui calcule une suite $\boldsymbol{a} = (a_n)_{n\geq 1}$ qui n'est pas dans S_T. Soit V une fonction telle que, pour tout $n \in \mathbb{N}, V(n)$ est le nombre d'opérations nécessaires à $\mathcal{M}$ pour produire les n premiers symboles de $\boldsymbol{a}$. La fonction V est clairement calculable (il suffit de construire une machine de Turing qui simule $\mathcal{M}$ et qui compte et renvoie le nombre d'opérations effectuées pour produire n symboles), et $\boldsymbol{a} \in S_V$. Soit la fonction U définie par $U(n) = \max\{T(n), V(n)\}$ pour tout entier naturel n. La fonction U est une fonction temporelle, et l'on a

$$S_T \subsetneq S_U.$$

En effet, comme $\boldsymbol{a} \in S_U$ et $\boldsymbol{a} \notin S_T$, on a bien que l'inclusion précédente est stricte. En itérant cette construction, l'on obtient bien des suites strictement croissantes de classes de complexité. □

Le prochain théorème est fondamental en théorie de la complexité, et nous permet d'affirmer qu'un changement linéaire d'une fonction temporelle ne modifie pas sa classe de complexité. Autrement dit, l'on peut modifier la vitesse de calcul d'une machine de Turing par un facteur constant.

Théorème 3.26. *Si $\boldsymbol{a} = (a_n)_{n\geq 1}$ est une suite T-calculable et k est un nombre réel positif calculable, alors $\boldsymbol{a}$ est $\lceil kT \rceil$-calculable, c'est à dire*

$$S_T = S_{\lceil kT \rceil}.$$

Démonstration. Remarquons d'abord que si k est calculable, alors $\lceil kT \rceil$ reste une fonction temporelle. Commençons par montrer le résultat pour $k = \frac{1}{2}$. Soit $\boldsymbol{a}$ une suite de S_T et $\mathcal{M}$ la machine de Turing à r bandes qui la produit en temps T. L'on cherche donc à décrire une machine $\mathcal{M}'$ qui produit $\boldsymbol{a}$ deux fois plus vite que $\mathcal{M}$. L'idée générale est de construire $\mathcal{M}'$ de sorte à ce qu'elle ait accès à plus d'information à chaque transition : d'une part en codant une partie de l'information dans ses états, d'autre part en travaillant sur un alphabet plus grand. Si la machine $\mathcal{M}'$ a accès à tout moment à

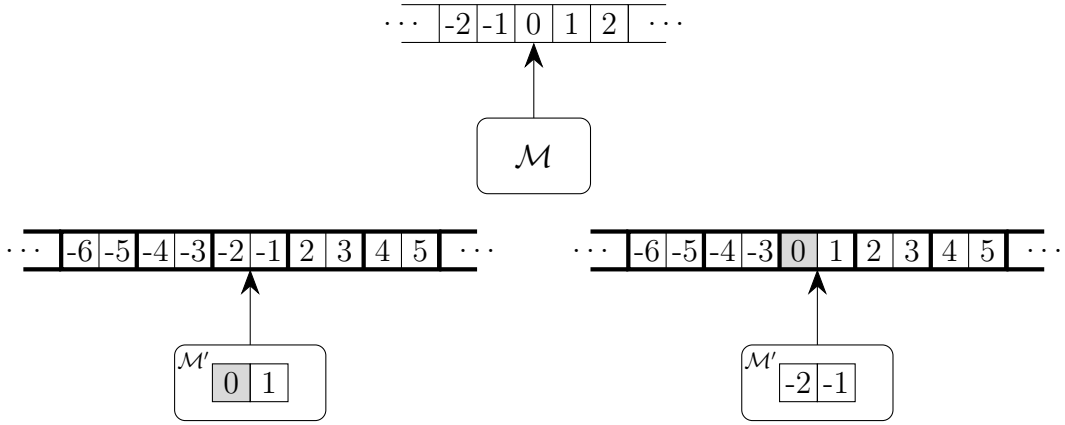


FIGURE 3.4 – Exemples de configurations de $\mathcal{M}'$ pour une configuration d'une machine $\mathcal{M}$ à une bande, dont la tête pointe sur la case 0.

— l'état q de $\mathcal{M}$,
— pour tout $1 \leq i \leq r$, le symbole σ_i lu par la tête de $\mathcal{M}$ sur la i-ème bande,
— les deux symboles adjacents à σ_i sur le ruban,
— et l'un des deux symboles adjacents à ce triplet,

alors elle peut effectuer en une opération ce que fait $\mathcal{M}$ en deux opérations. Notons Q l'ensemble des états de $\mathcal{M}$ et Σ l'alphabet sur lequel elle travaille, et décrivons comment fonctionne $\mathcal{M}'$.

— $\mathcal{M}'$ a le même nombre de bandes que $\mathcal{M}$.
— L'alphabet de $\mathcal{M}'$ est inclus dans Σ^2. Chaque case d'un ruban mémoire de $\mathcal{M}'$ contient l'information de deux cases d'un ruban de $\mathcal{M}$.
— L'on stocke également deux valeurs de chaque ruban de $\mathcal{M}$ dans les états de $\mathcal{M}'$, ainsi que l'information représentant où se trouve la tête de $\mathcal{M}$ sur le ruban. L'ensemble des états de $\mathcal{M}'$ est inclus dans

$$Q \times (\Sigma^2)^r \times \{\texttt{TL}, \texttt{TR}, \texttt{SL}, \texttt{SR}\}^r.$$

L'ajout de l'ensemble $\{\texttt{TL}, \texttt{TR}, \texttt{SL}, \texttt{SR}\}$ permet d'encoder l'information de la position de la tête de lecture sur chaque ruban de $\mathcal{M}$: soit sur l'une des deux cases du ruban correspondant à la case lue par la tête de $\mathcal{M}'$ (TL ou TR, pour "tape left" et "tape right", respectivement) soit sur une des deux cases stockées dans l'état courant de $\mathcal{M}'$ (SL ou SR, pour "state left" et "state right", respectivement).

La figure 3.4 illustre deux configurations possibles dans lesquels $\mathcal{M}'$ peut se trouver avec une machine $\mathcal{M}$ à une seule bande dont la tête de lecture est la position 0 sur le ruban. Les nombres représentés sur les rubans ne correspondent pas au contenu des cases, mais à leur position sur le ruban de $\mathcal{M}$. La case où se trouve la tête de lecture de $\mathcal{M}$ sur son ruban est représentée grisée. La figure 3.5 illustre des transitions et configurations possibles de $\mathcal{M}'$ selon les transitions de $\mathcal{M}$. Dans chaque cas, lorsque $\mathcal{M}$ opère deux transitions, la machine $\mathcal{M}'$ peut effectuer le même travail en une seule transition. Si $\mathcal{M}$ modifie le contenu de son ruban, $\mathcal{M}'$ peut faire de même, soit

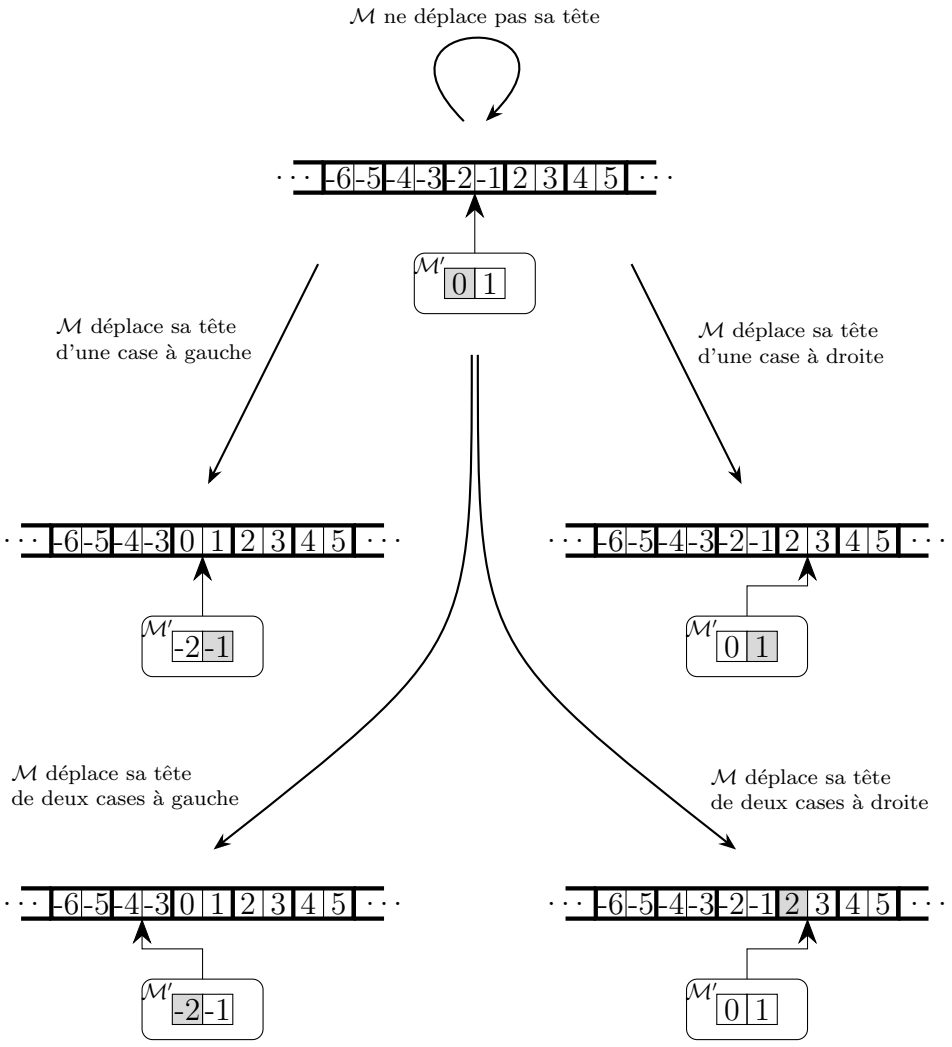


FIGURE 3.5 – Configurations possibles de $\mathcal{M}'$ après deux transitions de $\mathcal{M}$ (et donc une seule transition de $\mathcal{M}'$).

en écrivant sur son ruban, soit en modifiant la valeur stockée dans les états, c'est à dire en atteignant l'état adéquat.

En itérant cette construction, pour tout $m \in \mathbb{N}$, l'on peut ainsi construire une machine $\mathcal{M}'$ qui fonctionne en $\left\lceil \frac{1}{2^m} T \right\rceil$, et donc pour tout $k \leq 1$. Pour les valeurs de $k > 1$, il est facile de concevoir une machine de Turing $\mathcal{M}''$ qui simule $\mathcal{M}$, mais ajoutant autant d'opérations que nécessaire, sans changer le contenu des rubans, entre chaque opération de $\mathcal{M}$ pour produire $\boldsymbol{a}$ en temps $\lceil kT \rceil$. □

Exemple 3.27. Par exemple, décrivons une transition de $\mathcal{M}'$ dans le cas où $\mathcal{M}$ est une machine à une bande, et en 2 opérations déplace sa tête de lecture de deux cases à droite sur son ruban de travail et sur son ruban de sortie. La figure 3.6 illustre ce cas. En notant δ la fonction de transition, q_1, q_2 et q_3 des états de Q, et les symboles $\tau_1, \tau_2, \sigma_1, \sigma_2, \sigma_3, \sigma_4$ écrits sur le ruban de $\mathcal{M}$, supposons qu'elle soit à l'état q_1 et que l'on ait les transitions suivantes

$$\delta(q_1, \sigma_1) = (\gamma_1, \sigma_1^*, \mathtt{R}, \mathtt{R}, q_2) \text{ et } \delta(q_2, \sigma_2) = (\gamma_2, \sigma_2^*, \mathtt{R}, \mathtt{R}, q_3),$$

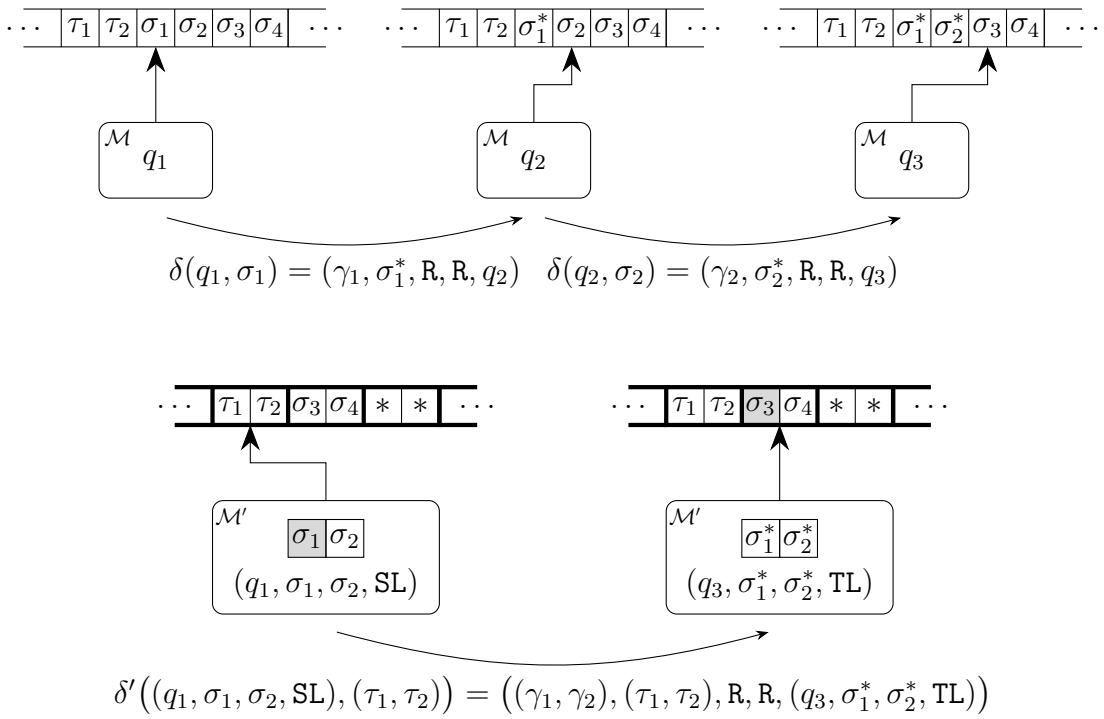


FIGURE 3.6 – Illustration d'une transition de $\mathcal{M}'$ quand $\mathcal{M}$ se déplace de deux cases à droite sur le ruban en deux transitions.

avec γ_1, γ_2 les symboles écrits sur le ruban de sortie et σ_1^*, σ_2^* les symboles écrits après chaque transition. Dans ce cas, un état de $\mathcal{M}'$ correspondant à l'état q_1 de $\mathcal{M}$ est $(q_1, \sigma_1, \sigma_2, \mathtt{SL})$, et sa fonction de transition δ' est définie comme

$$\delta'\big((q_1, \sigma_1, \sigma_2, \mathtt{SL}), (\tau_1, \tau_2)\big) = \big((\gamma_1, \gamma_2), (\tau_1, \tau_2), \mathtt{R}, \mathtt{R}, (q_3, \sigma_1^*, \sigma_2^*, \mathtt{TL})\big).$$

Ce théorème, nommé "speed-up theorem" par les auteurs (ou "théorème d'accélération" en français), permet donc de justifier l'usage de la notation $O(T(n))$, habituelle en algorithmique, puisque les classes de complexité peuvent être définies à un facteur constant près.

Corollaire 3.28. *Soient T et U des fonctions temporelles.*

— *Si*
$$\liminf_{n\to+\infty} \frac{T(n)}{U(n)} > 0$$
alors $S_U \subseteq S_T$.

— *Si*
$$\limsup_{n\to+\infty} \frac{T(n)}{U(n)} < +\infty$$
alors $S_T \subseteq S_U$.

Démonstration. Montrons le premier point. Comme l'on a $\liminf_{n\to+\infty} \frac{T(n)}{U(n)} > 0$, il existe $k > 0$ tel que $kU(n) \leq T(n)$, pour des n assez grands, ce qui donne $S_{\lceil kU \rceil} \subseteq S_T$. Or, l'on a $S_U = S_{\lceil kU \rceil}$ par le théorème 3.26, ce qui permet de conclure. L'on montre le deuxième point par un raisonnement analogue. □

Corollaire 3.29. *Soient T et U des fonctions temporelles. Si*

$$0 < \lim_{n\to+\infty} \frac{T(n)}{U(n)} < \infty,$$

alors $S_U = S_T$.

Corollaire 3.30. *Si T est une fonction temporelle, alors $S_n \subseteq S_T$. Autrement dit, S_n est la plus petite classe de complexité.*

Démonstration. Comme T est une fonction temporelle, il existe $k \in \mathbb{N}$ tel que $T(n) \geq \frac{n}{k}$. Alors

$$\liminf_{n\to+\infty} \frac{T(n)}{n} \geq \frac{1}{k} > 0,$$

et donc $S_n \subseteq S_T$. □

L'on peut ainsi en déduire la hiérarchie de classes de complexité usuelle en algorithmique :

$$S_n \subset S_{n^2} \subset S_{n^3} \subset \cdots$$

Rappelons que les auteurs autorisent les machines à écrire plusieurs symboles sur une même case de leur ruban de sortie (voir la définition 3.20), c'est à dire à avoir un alphabet de sortie plus grand que Σ_b. Nous allons voir qu'imposer aux machines d'avoir Σ_b pour alphabet de sortie, et donc de ne produire qu'un seul symbole à la fois sur leur bande de sortie, n'affecte pas les classes de complexités que nous avons définies. L'on nommera de telles machines des machines de Turing *à sortie b-aire*.

Proposition 3.31. *Si T est une fonction temporelle telle que*

$$T(n) \geq n \text{ et } \liminf_{n\to+\infty} \frac{T(n)}{n} > 1,$$

alors pour toute suite $\boldsymbol{a}$ dans S_T, il existe une machine de Turing multibande $\mathcal{M}$ à sortie b-aire et qui produit le n-ème terme de $\boldsymbol{a}$ en maximum $T(n)$ opérations.

Démonstration. La fonction temporelle T satisfait $\liminf_{n\to+\infty} \frac{T(n)}{n} > 1$, c'est à dire pour tout $\varepsilon > 0$, il existe $N \in \mathbb{N}$ tel que pour tout $n > N$, l'on ait

$$\frac{T(n)}{n} > 1 + \varepsilon.$$

Soit $\varepsilon > 0$. Posons $\varepsilon' = \frac{\varepsilon}{1+\varepsilon}$. Alors, il existe $N \in \mathbb{N}$ tel que, pour tout $n > N$,

$$(1 - \varepsilon')T(n) > n, \text{ c'est à dire } T(n) > \varepsilon' T(n) + n.$$

Soit $\boldsymbol{a}$ une suite de S_T. Par le théorème 3.26, il existe une machine $\mathcal{M}'$ qui produit $\boldsymbol{a}$ en temps $\lceil \varepsilon' T \rceil$. L'on peut la modifier en une machine $\mathcal{M}''$ à sortie b-aire, suspendant ses calculs le temps nécessaire à la production des symboles sur sa bande de sortie. $\mathcal{M}''$ produit donc les n premiers symboles de $\boldsymbol{a}$ en temps $\lceil \varepsilon' T(n) \rceil + n$, ce qui est inférieur à $T(n)$ pour tous les $n > N$. L'on modifie ainsi $\mathcal{M}''$ en une machine $\mathcal{M}$ en encodant les N premiers symboles de $\boldsymbol{a}$ dans ses transitions, qui seront donc produits en temps $n < T(n)$. □

Corollaire 3.32. *Soit T une fonction temporelle et une suite $\boldsymbol{a} \in S_T$. Alors pour tout $\varepsilon > 0$, il existe une machine de Turing multibande à sortie b-aire qui produit $\boldsymbol{a}$ en temps $\lceil (1+\varepsilon)T \rceil$.*

Démonstration. Soit $\varepsilon > 0$. Alors par le théorème 3.26, il existe une machine de Turing qui produit $\boldsymbol{a}$ en temps $\lceil (1+\varepsilon)T \rceil$. L'on a pour tout $n \in \mathbb{N}$, $(1+\varepsilon)T(n) \geq T(n) \geq n$. De plus,

$$\liminf_{n \to +\infty} \frac{\lceil (1+\varepsilon)T(n) \rceil}{n} \geq 1 + \varepsilon > 1.$$

Ainsi, la proposition précédente nous donne une machine de Turing multibande à sortie b-aire, qui produit $\boldsymbol{a}$ en temps $\lceil (1+\varepsilon)T \rceil$. □

Théorème 3.33. *Soient deux suites $\boldsymbol{u}$ et $\boldsymbol{v}$ de $\Sigma_b^{\mathbb{N}}$ qui diffèrent en un nombre fini d'éléments. Alors pour toute fonction temporelle T, l'on a*

$$\boldsymbol{u} \in S_T \iff \boldsymbol{v} \in S_T.$$

Démonstration. Soit $\mathcal{M}$ la machine de Turing qui produit $\boldsymbol{u}$ en temps T. L'on peut aisément construire une machine $\mathcal{M}'$ qui produit $\boldsymbol{v}$ en temps T en codant les éléments qui diffèrent dans les états et transitions de $\mathcal{M}$. □

3.2.4 Classification des nombres calculables

Maintenant que nous avons introduit les classes de complexité temporelle définies par Hartmanis et Stearns, une question consiste à déterminer dans quelle mesure ces classes permettent de distinguer différents ensembles de nombres réels. Plus précisément, pour un réel calculable donné, il s'agit de comprendre dans quelle classe de complexité se situe sa b-représentation, et donc à quel vitesse une machine de Turing peut la produire.

Commençons par noter que la b-représentation d'un nombre rationnel $\frac{p}{q}$ est calculable en temps linéaire. En effet, la b-représentation de $\frac{p}{q}$ est ultimement périodique, conformément au théorème 2.12, et peut donc être générée par un automate déterministe muni d'une sortie, c'est-à-dire par une machine de Turing à zéro bandes de travail. Nous définirons plus précisément ce type d'automate au chapitre suivant. Il en résulte que tout nombre rationnel est dans la classe de complexité S_n. Certains nombres transcendants sont également dans S_n.

Théorème 3.34. *Il existe des nombres transcendants dans S_n.*

Pour démontrer cela, il nous suffit d'exhiber un nombre transcendant calculable en temps $O(n)$. Considérons le nombre λ défini de la manière suivante :

$$\lambda = \sum_{n \geq 1} \frac{1}{b^{n!}},$$

pour $b \geq 2$ un entier naturel. Le nombre λ est de Liouville (voir définition 2.26), il est donc transcendant. Sa représentation en base b correspond exactement à la suite χ caractéristique de la factorielle, définie de la manière suivante :

$$\chi(n) = \begin{cases} 1 & \text{s'il existe } m \in \mathbb{N} \text{ tel que } m! = n \\ 0 & \text{sinon.} \end{cases}$$

Il nous suffit donc de montrer que cette suite est calculable en $O(n)$ opérations pour prouver notre assertion. Nous allons en fait démontrer un résultat plus fort, qui nous demandera quelques définitions et propositions supplémentaires : la factorielle est *comptable en temps réel*.

Définition 3.35. Une fonction $f : \mathbb{N} \to \mathbb{N}$ est dite *comptable en temps réel* s'il existe une machine de Turing multibande $\mathcal{M}$ qui à sa n-ème opération produit sur sa bande de sortie

$$\begin{cases} 1 & \text{s'il existe } m \in \mathbb{N} \text{ tel que } f(m) = n \\ 0 & \text{sinon.} \end{cases}$$

On dit aussi que $\mathcal{M}$ est un *compteur de* f, ou un f-compteur.

Il est clair que les suites produites par les compteurs de fonctions sont dans S_n, elles sont calculées exactement en temps $T(n) = n$.

Les résultats présentés dans la suite de cette section sont dûs initialement à Yamada [52]. N'ayant toutefois pas eu accès à cette source originale, les démonstrations proposées ici s'appuient sur les travaux ultérieurs de Fischer, Meyer et Rosenberg [24, 26]. Ces derniers utilisent des machines à compteurs manipulant des entiers naturels, que la machine peut incrémenter ou décrémenter de 1 à chaque opération, au lieu de bandes sur lesquelles écrire des symboles. Une machine à compteur est facilement simulable par une machine de Turing standard : l'on imite le comportement d'un compteur en travaillant en unaire sur une bande. Les machines utilisées dans les propositions suivantes suivent donc le modèle standard de Turing et travaillent principalement en unaire.

Définition 3.36. Une fonction $f : \mathbb{N} \to \mathbb{N}$ (ou suite) est dite *auto-calculable* si

- Pour tout $n \in \mathbb{N}, f(n) \geq n$,
- Il existe $k > 0$ tel que f est calculable en temps kf, c'est à dire en maximum $k \cdot f(n)$ opérations.

Proposition 3.37. *Toute fonction f telle que pour tout $n \in \mathbb{N}, f(n) \geq n$ et comptable en temps réel est auto-calculable.*

Démonstration. Soit f une fonction comptable en temps réel, et soit $\mathcal{M}$ son compteur. Soient $m, n \in \mathbb{N}$ tels que $f(m) = n$. Alors $\mathcal{M}$ produit 1 sur sa bande de sortie à sa n-ème opération. L'on peut aisément concevoir une machine $\mathcal{M}'$ qui compte le nombre d'opérations n depuis le lancement de $\mathcal{M}$ et écrit n en unaire sur sa bande de sortie quand $\mathcal{M}$ renvoie 1. La fonction f est alors calculable en temps f, c'est à dire auto-calculable pour $k = 1$. □

Tout fonction comptable en temps réel f est en fait calculable en temps $O(f(n))$.

Définition 3.38. Une fonction $f : \mathbb{N} \to \mathbb{N}$ (ou suite) est *fortement auto-calculable* si

- Pour tout $n \in \mathbb{N}, f(n) \geq n$,
- Il existe $k > 0$ tel que f est calculable en **exactement** $k \cdot f(n)$ opérations.

S'il est évident que toute fonction fortement auto-calculable est auto-calculable, nous allons voir qu'en fait ces deux notions coïncident. Plus précisément, si f est auto-calculable et calculée par une machine $\mathcal{M}$, il est possible de construire une machine $\mathcal{N}$ qui, pour tout $n \in \mathbb{N}$, calcule $f(n)$ en exactement $k \cdot f(n)$ opérations, pour un certain k. L'idée consiste à simuler $\mathcal{M}$ en ajoutant des bandes supplémentaires servant de compteur et de mémoire tampon. La construction de machines qui comportent plus bandes est un outil que nous exploiterons à plusieurs reprises dans les preuves qui suivent.

Lemme 3.39. *Soit $f : \mathbb{N} \to \mathbb{N}$ calculable. Alors f est auto-calculable si et seulement si elle est fortement auto-calculable.*

Démonstration. La condition nécessaire étant évidente, montrons la condition suffisante. Soit f une fonction auto-calculable, et $\mathcal{M}$ une machine de Turing à r bandes qui, pour tout $n \in \mathbb{N}$, calcule $f(n)$ en maximum $kf(n)$ opérations, pour un $k > 0$. Décrivons une machine de Turing $\mathcal{N}$ à $r+2$ bandes qui calcule $f(n)$ en exactement $2kf(n)$ opérations. La machine $\mathcal{N}$ fonctionne en plusieurs étapes et commence par simuler $\mathcal{M}$ sur ses r premières bandes. Sur la $(r+1)$-ème bande, elle compte (en unaire) le nombre d'opérations depuis son lancement. Au bout d'un certain temps $t \leq kf(n)$, quand $\mathcal{M}$ arrête son calcul, il y a

— $f(n)$ écrit sur sa bande de sortie (disons la r-ème bande),

— t symboles sur la $(r+1)$-ème bande,

— rien sur la $(r+2)$-ème bande.

Ensuite, $\mathcal{N}$ copie le contenu de la r-ème bande sur la $(r+2)$-ème bande. Pour chaque symbole copié, elle retire k symboles de la $(r+1)$-ème bande. Si cette bande venait à être vide (c'est à dire si le nombre encodé sur cette bande venait à être négatif), la machine recommence à écrire des symboles au lieu de les effacer (cela revient à considérer que la bande vide représente la valeur $-t$ au lieu de 0). A la fin de cette étape, la machine $\mathcal{N}$ a fait exactement $t + kf(n)$ opérations. Le contenu de ses rubans est le suivant :

— rien sur la r-ème bande,

— $kf(n) - t$ sur la $(r+1)$-ème bande,

— $f(n)$ sur la $(r+2)$-ème bande.

La dernière étape pour $\mathcal{N}$ consiste à vider un par un les symboles sur la $(r+1)$-ème bande. Au total, elle a fait exactement $t + kf(n) + kf(n) - t = 2kf(n)$ opérations. La fonction f est ainsi calculable en exactement $2kf(n)$ opérations pour tout $n \in \mathbb{N}$, et est donc fortement auto-calculable. □

Rappelons que l'on peut également encoder de l'information directement dans les états d'une machine de Turing. En effet, de manière similaire à la construction de la machine présentée dans la démonstration du théorème 3.26, quitte à augmenter le nombre d'états et à adapter les transitions d'une machine, l'on peut stocker une valeur finie dans ses états. Ceci permet, pour toute machine de Turing $\mathcal{M}$ et tout nombre positif c, de construire une machine qui calcule c fois plus vite, comme l'énonce le lemme suivant.

Lemme 3.40. *Pour toute machine $\mathcal{M}$ qui calcule une fonction f en temps kf, pour un k positif, et pour tout entier positif c, il existe une machine $\mathcal{N}$ qui fonctionne de la manière suivante :*

- *$\mathcal{N}$ commence son calcul avec $\left\lfloor \frac{n}{c} \right\rfloor$ sur sa bande d'entrée, et commence à l'état correspondant à la valeur $n - c\left\lfloor \frac{n}{c} \right\rfloor$.*
- *$\mathcal{N}$ termine son calcul en $\left\lceil k\frac{f(n)}{c} \right\rceil$ opérations, à l'état correspondant à la valeur $f(n) - c\left\lfloor \frac{f(n)}{c} \right\rfloor$, et produit $\left\lfloor \frac{f(n)}{c} \right\rfloor$ sur sa bande de sortie.*

Démonstration. Soient un réel calculable $k > 0$, c un entier positif, et $\mathcal{M}$ la machine qui calcule f en temps kf. D'une manière similaire au théorème 3.26, décrivons la machine $\mathcal{N}$ qui simule $\mathcal{M}$, mais réalise une seule opération pour c opérations de $\mathcal{M}$. Elle a encodés dans ses états autant d'entiers que $\mathcal{M}$ a de bandes, qui prennent des valeurs entre 0 et c. Elle compte les symboles produits par $\mathcal{M}$ (via un des compteurs encodés dans les états), et produit un symbole tous les c symboles produits par $\mathcal{M}$. Si $\mathcal{M}$ prend $n \in \mathbb{N}$ pour entrée, alors $\mathcal{N}$ prend $\left\lfloor \frac{n}{c} \right\rfloor$ en entrée et commence à l'état correspondant à la valeur $n - c\left\lfloor \frac{n}{c} \right\rfloor$. Comme $\mathcal{M}$ produit $f(x)$ en $kf(x)$ opérations, et que $\mathcal{N}$ calcule c fois plus vite, elle produit $\left\lfloor \frac{f(n)}{c} \right\rfloor$ en $\left\lceil k\frac{f(x)}{c} \right\rceil$ opérations. □

Notation.

- On note $f^{(c)}$ la fonction calculée par la machine $\mathcal{N}$ du lemme précédent.
- On pose, pour toute fonction f calculable, $\Delta f(n) = f(n+1) - f(n)$.

Théorème 3.41. *Soit $f : \mathbb{N} \to \mathbb{N}$ calculable, strictement croissante et telle que Δf est auto-calculable. Alors f est comptable en temps réel.*

Démonstration. Soit $\mathcal{M}$ qui calcule le n-ème terme de Δf en exactement $k \cdot \Delta f(n)$ opérations, pour un k positif, conformément au lemme 3.39. Le lemme 3.40 nous donne une machine $\mathcal{N}$ à r bandes qui calcule $\Delta f^{(2k)}$. Construisons une machine $\mathcal{P}$ qui compte f en temps réel. Sa structure est la suivante :

- Elle possède $r+2$ bandes en plus de sa bande de sortie. Les r premières bandes serviront à simuler $\mathcal{N}$ et les deux autres serviront de compteur.
- Elle a un entier codé dans les états, qui peut prendre des valeurs de 0 à $2k$.

La machine $\mathcal{P}$ fonctionne en étapes. Au début de chaque étape q, elle a $\lfloor \frac{q}{2k} \rfloor$ écrit sur la première et $(r+1)$-ème bande, et 0 sur les autres bandes, et a la valeur $q - 2k\lfloor \frac{q}{2k} \rfloor$ stockée dans les états. Elle simule la machine $\mathcal{N}$ sur ses r premières bandes jusqu'à la fin du calcul de cette dernière. Elle émet un 0 sur sa bande de sortie à chaque opération en parallèle de la simulation, à la fin de laquelle se sont déroulées $\lceil k\frac{\Delta f(q)}{2k} \rceil = \lceil \frac{\Delta f(q)}{2} \rceil$ opérations. Il y a

- 0 sur les bandes 1 à $r-1$,
- $\lfloor \frac{\Delta f(q)}{2k} \rfloor$ sur la r-ème bande,
- $\lfloor \frac{q}{2k} \rfloor$ sur la $(r+1)$-ème bande,
- et 0 sur la $(r+2)$-ème bande.

La machine $\mathcal{P}$ vide alors la r-ème bande, la décrémentant de 1 toutes les k opérations. En même temps, elle copie le contenu de la bande $r+1$ sur la première et $(r+2)$-ème bande, tout en effaçant le contenu de la bande $r+1$. Ce processus demande $k\lfloor\frac{\Delta f(q)}{2k}\rfloor$ opérations. Le temps de calcul total de l'étape q s'élève donc à $\Delta f(q)$ opérations, quitte à retarder d'au maximum k opérations la production du symbole 1 à la fin de l'étape.

Ainsi, par induction, si $\mathcal{P}$ produit 1 à la $f(q)$-ème opération, elle produit le 1 suivant à la $f(q) + \Delta f(q) = f(q+1)$-ème opération. Autrement dit $\mathcal{P}$ est un f-compteur, et f est comptable en temps réel. □

Ce critère d'auto-calculabilité de la fonction Δf s'avère très utile en pratique pour montrer la comptabilité en temps réel d'une fonction f.

Corollaire 3.42. *Soit f une fonction calculable croissante. Si Δf est comptable en temps réel, alors f l'est aussi.*

Corollaire 3.43. *Soit f une fonction auto-calculable croissante et telle que, pour un réel $a \geq 2$, $f(n+1) \geq a \cdot f(n)$, pour tout $n \in \mathbb{N}$. Alors f est comptable en temps réel.*

Démonstration. Soit f croissante et auto-calculable, c'est à dire qu'il existe $k > 0$ tel que pour tout $n \in \mathbb{N}, f(n)$ est calculée en au plus $kf(n)$ opérations. Alors calculer $f(n), f(n+1)$ et faire la différence du second par le premier demandent au plus $3k \cdot f(n+1)$ opérations. Or, par hypothèse, il existe un réel $a \geq 2$ tel que $f(n+1) \geq a \cdot f(n)$. Alors

$$\Delta f(n) = f(n+1) - f(n) \geq f(n+1) - \frac{f(n+1)}{a} = \frac{a-1}{a} f(n+1),$$

d'où $f(n+1) \leq \frac{a}{a-1} \cdot \Delta f(n)$. L'on obtient alors que

- $\Delta f(n)$ est calculable en au plus $\lceil\frac{3ka}{a-1}\rceil \Delta f(n)$ opérations et
- $\Delta f(n) \geq (a-1) f(n) \geq f(n) \geq n$.

Δf est alors une fonction auto-calculable, ce qui nous donne que f est comptable en temps réel par le théorème 3.41. □

La notion d'auto-calculabilité est en fait stable par plusieurs opérations arithmétiques usuelles. En particulier, la somme et le produit conservent cette propriété.

Proposition 3.44. *Si f et g sont auto-calculables alors les fonctions suivantes le sont aussi :*

- $f + g$
- $f \cdot g$
- $\prod_{i=0}^{n} f(i)$

Démonstration. Soient f et g deux fonctions telles que pour tout $n \in \mathbb{N}, f(n)$ et $g(n)$ sont respectivement calculables en au plus $k_1 f(n)$ et $k_2 g(n)$ opérations.

— Additionner les nombres $f(n)$ et $g(n)$ écrits en unaire se fait en maximum $f(n) + g(n)$ opérations. Ainsi le nombre d'opérations nécessaire au calcul de $(f + g)(n)$ est

$$\begin{aligned} k_1 f(n) + k_2 g(n) + f(n) + g(n) &\leq k_1 \left(f(n) + g(n)\right) \\ &\quad + k_2 \left(f(n) + g(n)\right) + f(n) + g(n) \\ &\leq (k_1 + k_2 + 1)(f(n) + g(n)). \end{aligned}$$

La fonction $f + g$ est donc auto-calculable.

— Après le calcul de $f(n)$ et de $g(n)$, l'on peut effectuer le produit en copiant la valeur de $g(n)$ et ce $f(n)$ fois. Soustraire 1 à $f(n)$ et copier $g(n)$ se fait en au maximum $g(n) + 1$ étapes. Ainsi le nombre d'opérations nécessaire au calcul de $(f \cdot g)(n)$ est

$$\begin{aligned} k_1 f(n) + k_2 g(n) + f(n) \cdot (g(n) + 1) &\leq k_1 f(n) g(n) + k_2 f(n) g(n) + 2 f(n) g(n) \\ &\leq (k_1 + k_2 + 2)(f(n) g(n)). \end{aligned}$$

La fonction $f \cdot g$ est donc auto-calculable.

— Remarquons d'abord que si $f(0) = 0$, alors $\prod_{i=0}^{n} f(i) = 0$ pour tout $n \in \mathbb{N}$, et est donc calculable en temps constant. L'on peut donc supposer $f(0) \neq 0$. De plus, comme f est auto-calculable, on a $f(n) \geq n$ pour tout $n \in \mathbb{N}$. Calculer $\prod_{i=0}^{n} f(i)$ se fait en au plus $k_1 \sum_{i=0}^{n} f(i)$ opérations. Or, comme pour tout $0 \leq i \leq n$, l'on a $f(i) \geq i$, l'on en déduit $k_1 \sum_{i=0}^{n} f(i) \leq k_1 \prod_{i=0}^{n} f(i)$, d'où le résultat annoncé.

□

Corollaire 3.45. *La factorielle est comptable en temps réel.*

Démonstration. Avec f la fonction factorielle, montrons que $\Delta f(n) = n \cdot n!$ est auto-calculable. L'identité est trivialement auto-calculable, et la proposition précédente nous permet d'affirmer que la factorielle l'est également. De plus, l'auto-calculabilité est stable par le produit. On a bien que Δf est auto-calculable, donc que la factorielle est comptable en temps réel par le théorème 3.41. □

Remarque 3.46. L'auto-calculabilité est en fait stable par d'autres opérations, non utilisées ici. En effet, si f et g sont auto-calculables alors les fonctions suivantes le sont aussi :

— $g \circ f$,
— f^g,
— $\sum_{i=0}^{n} f(i)$.

L'auto-calculabilité de ces fonctions se montre de manière similaire aux preuves présentées dans la proposition 3.44.

Les propriétés de stabilité des opérations présentées nous donnent donc qu'un grand nombre de fonctions usuelles sont auto-calculables (en particulier il est facile de voir que les polynômes de $\mathbb{N}[X]$ le sont), et laissent donc penser qu'il existe un grand nombre de fonctions comptables en temps réel. En particulier, la factorielle, bien qu'elle ne soit trivialement pas dans S_n est elle aussi comptable en temps réel. Nous pouvons finalement revenir sur la démonstration du théorème 3.34.

Démonstration du théorème 3.34. La b-représentation de λ est exactement la suite χ caractéristique de la factorielle, et est donc dans S_n. Nous avons bien exhibé un nombre transcendant calculable en temps linéaire. □

Notons qu'altérer un nombre fini de symboles de la b-représentation de λ n'altère ni son aspect transcendant, ni la classe de complexité dans laquelle il se trouve. Cela nous donne donc un sous ensemble dense de nombres transcendants qui sont des éléments de S_n.

Mais qu'en est-il alors des nombres algébriques ? Les auteurs montrent le résultat suivant.

Théorème 3.47. *Les nombres algébriques sont contenus dans S_{n^2}.*

Démonstration. Reprenons la machine $\mathcal{M}$ de la proposition 3.18, et montrons qu'elle opère son calcul en temps $O(n^2)$. La machine commence par produire une suite finie de symboles fixée, la b-représentation de a_n, ce qu'elle peut faire en $O(n)$ opérations. Pour les décimales suivantes, montrons que le calcul de chaque décimale se fait en $O(n)$ opérations.

- Additionner k nombres de longueur nd en commençant avec la tête de lecture sur le bit le moins significatif et la retourner ensuite à cette position après l'opération prend au maximum $2 \cdot (nd + k) = O(n)$ opérations.
- Multiplier par une constante h se fait aussi en temps linéaire : cela revient à additionner h fois le même nombre.
- Les différentes puissances

$$(a_n + b^{-n})^i = a_n^i + \binom{i}{1} a^{i-1} b^{-n} + \cdots + b^{-in}, \text{ pour } 1 \leq i \leq d,$$

  peuvent donc être calculées en temps linéaire, car cela revient à additionner et multiplier par des nombres stockés en mémoire.
- Les c_i, pour $1 \leq i \leq b-1$ sont également calculés en temps linéaire pour la même raison.
- Effacer les valeurs qui ne sont plus utiles au calcul se fait aussi en $O(n)$.

Ainsi, le calcul de chaque décimale se fait en $O(n)$ opérations, ce qui nous donne que la machine $\mathcal{M}$ calcule le nombre algébrique α en $O(n^2)$ opérations. Autrement dit, $\alpha \in S_{n^2}$. □

Comme l'on a $S_n \subset S_{n^2}$, la question est donc de savoir s'il existe des irrationnels algébriques qui sont dans S_n, ou autrement formulé :

Problème (Hartmanis-Stearns)**.** *Existe-t-il des nombres irrationnels algébriques dont la b-représentation peut être produite par une machine de Turing multibande en $O(n)$ opérations ?*

À l'heure actuelle, une réponse négative à ce problème est généralement conjecturée. En effet, l'existence d'un nombre irrationnel algébrique α appartenant à la classe S_n impliquerait l'existence d'une machine de Turing capable de produire chaque chiffre de la b-représentation de α en temps constant. Or, comme en témoigne

la démonstration de la proposition 3.18, l'on utilise, entre autres, des multiplications et exponentiations pour calculer les décimales de α. Ces opérations induisent un coût au moins linéaire, ce qui rend peu plausible l'appartenance d'un irrationnel algébrique à la classe S_n. Cependant, si la conjecture venait à être confirmée, on obtiendrait une classification relativement surprenante des nombres réels, dans laquelle certains nombres transcendants posséderaient une b-représentation plus simple que celle de tous les irrationnels algébriques.

Une approche pour aborder ce problème, initiée par Cobham [22], consiste à restreindre la puissance du modèle de calcul utilisé pour produire les représentations des nombres. Plutôt que de considérer l'ensemble des machines de Turing, on étudie des sous-classes plus faibles, telles que les automates finis ou les automates à pile, et l'on cherche à déterminer quels types de nombres peuvent être engendrés par ces modèles. Les chapitres suivants seront consacrés à l'étude de ces classes de machines, et montreront notamment que leur puissance de calcul est insuffisante pour produire la b-représentation d'un irrationnel algébrique, montrant ainsi la nécessité de la puissance de calcul des machines de Turing dans le cadre du problème.

Chapitre 4

La conjecture de Cobham

Comme évoqué précédemment, Cobham fut le premier à proposer une restriction du modèle utilisé dans la question d'Hartmanis et Stearns. Bien que formulée en 1968 en termes de *tag machines*, la question de Cobham, qui constitue ainsi une restriction de la question principale et dont il conjecturait une réponse négative, s'énoncerait aujourd'hui ainsi :

Problème (Cobham)**.** *Existe-t-il des nombres irrationnels algébriques dont la b-représentation peut être produite par un automate fini déterministe avec sortie ?*

Ce chapitre est donc d'abord consacré à la réponse à cette question. Nous commencerons par présenter la classe restreinte de machines de Turing que sont les automates finis déterministes, ainsi que les suites produites par ces derniers. Nous rappellerons une équivalence importante associée à ce modèle de calcul, à savoir celle entre les suites automatiques et les suites produites par des morphismes uniformes. Ensuite, nous verrons un premier résultat central de ce mémoire, reprenant les travaux d'Adamczewski, Bugeaud et Luca [3, 4], qui confirme la conjecture de Cobham : les automates finis déterministes ne peuvent pas produire la b-représentation des irrationnels algébriques. Pour démontrer ce résultats, les auteurs ont établi un critère combinatoire de transcendance qui sera présenté dans la deuxième section, et la troisième section en présentera une reformulation. Les deux dernières sections de ce chapitre, qui reprennent les travaux d'Adamczewski et Bugeaud [2, 5], permettront d'établir plus en détails les propriétés diophantiennes des nombres produits par les automates finis.

4.1 Automates finis et morphismes

4.1.1 Automates finis déterministes

Définition 4.1. Un *k-automate fini déterministe avec sortie* (abrégé en k-AFDS) est la donnée d'un sextuple $\mathcal{A} = (Q, \Sigma_k, \delta, q_0, \Delta, \tau)$, où

— Q est un ensemble fini, appelé *ensemble d'états*,

— Σ_k est l'alphabet $\{0, 1, 2, \ldots, k-1\}$,

— $\delta : Q \times \Sigma_k \to Q$ est appelée la *fonction de transition*,

— $q_0 \in Q$ est *l'état initial*,
— Δ est un alphabet fini appelé *l'alphabet de sortie* et
— $\tau : Q \to \Delta$ est la *fonction de sortie*.

Pour donner du sens au calcul $\tau(\delta(q_0, w))$, pour tout $w \in \Sigma_k^*$, l'on peut étendre la fonction de transition δ à $Q \times \Sigma_k^*$ en posant $\delta(q, wa) = \delta(\delta(q, w), a)$ pour tous $q \in Q, w \in \Sigma_k^*$ et $a \in \Sigma_k$. Ceci signifie en particulier que l'automate lit les mots de la gauche vers la droite. Là où un automate fini déterministe *sans sortie* est utilisé en tant qu'accepteur des langages réguliers, les automates finis déterministes *avec sortie* peuvent servir à produire des suites. Ils fonctionnent de manière similaire à une machine de Turing avec zéro bande de travail et une bande de sortie.

Définition 4.2. Une suite $\boldsymbol{a} \in \Sigma_b^{\mathbb{N}}$ est *k-automatique* s'il existe un k-AFDS $\mathcal{A} = (Q, \Sigma_k, \delta, q_0, \Sigma_b, \tau)$ tel que $\boldsymbol{a} = (\tau(\delta(q_0, \langle n \rangle_k)))_{n \geq 0}$. On dit aussi que la suite $\boldsymbol{a}$ est *produite par l'automate $\mathcal{A}$*. De manière générale, une suite est *automatique* si elle est k-automatique pour un certain k.

Pour une présentation plus en détails des suites automatiques, le lecteur ou la lectrice peut se référer à [9, Chap. 5, 6]

Similairement à une machine de Turing, une manière pratique de représenter un k-AFDS $\mathcal{A} = (Q, \Sigma_k, \delta, q_0, \Delta, \tau)$ est avec un graphe orienté, dont les sommets sont les états de Q, et les arêtes sont données par la fonction de transition δ de la manière suivante : pour tous états q_i, q_j et symbole $\sigma \in \Sigma_k$, tels que $\delta(q_i, \sigma) = q_j$, on trace une flèche du sommet q_i vers le sommet q_j, étiquetée par σ. L'état initial q_0 est symbolisé par une flèche entrante sur le sommet correspondant, et la fonction de sortie τ est représentée par une flèche sortante de chaque état q_i, et étiquetée par le symbole correspondant $\tau(q_i)$.

Exemple 4.3. La suite de Thue-Morse $\boldsymbol{t} = (t_n)_{n \geq 0} \in \Sigma_2^{\mathbb{N}}$ définie de la manière suivante : pour tout $n \geq 0$,

$$t_n := s_2(n) \mod 2,$$

avec $s_2(n)$ la fonction qui associe la somme des chiffres en base 2. Le 2-AFDS qui produit $(t_n)_{n \geq 0}$, représenté à la figure 4.1, est le suivant :

$$(\{q_0, q_1\}, \Sigma_2, \delta, q_0, \Sigma_2, \tau),$$

avec

$$\delta(q_0, 0) = q_0, \delta(q_0, 1) = q_1, \delta(q_1, 0) = q_1, \delta(q_0, 1) = q_0,$$

et

$$\tau(q_0) = 0, \tau(q_1) = 1.$$

Les automates avec sorties permettent naturellement de produire des nombres réels, étant donné qu'ils peuvent produire des suites de $\Sigma_b^{\mathbb{N}}$. De tels réels sont appelés les *nombres automatiques*.

Définition 4.4. Un réel $\xi \in [0, 1)$ est *(b, k)-automatique* si sa b-représentation est une suite k-automatique, c'est à dire s'il existe un k-AFDS $\mathcal{A} = (Q, \Sigma_k, \delta, q_0, \Sigma_b, \tau)$ tel que $\langle \xi \rangle_b = (\tau(\delta(q_0, \langle n \rangle_k)))_{n \geq 0}$.

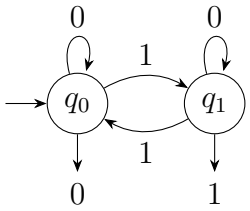


FIGURE 4.1 – 2-AFDS qui génère la suite de Thue-Morse

Exemple 4.5. Le nombre de Thue-Morse

$$\tau = \sum_{n \geq 1} t_n 2^{-n}$$

dont la 2-représentation est la suite de Thue-Morse $(t_n)_{n\geq 0}$, est un nombre transcendant [35] (voir aussi [9, Chap. 13.4]).

La question est donc de savoir quels nombres peuvent être produits par les k-AFDS. Un premier résultat concernant les suites automatiques est le suivant.

Théorème 4.6 ([9, Theorem 5.4.2])**.** *Toute suite de $\Sigma_b^{\mathbb{N}}$ ultimement périodique est k-automatique, pour tout $k \geq 2$ et tout $b \geq 2$.*

Ainsi, tout nombre rationnel peut être produit par un k-AFDS, et ce pour tout $k \geq 2$ [1]. En particulier, comme en témoigne l'exemple 4.5 certains nombres transcendants sont égalements automatiques [2]. Comme mentionné en introduction, nous allons voir que l'on ne peut pas en dire autant pour les irrationnels algébriques. Avant cela, voyons une équivalence importante : les suites automatiques coïncident avec une certaine classe de suites produites par des morphismes.

4.1.2 Morphismes

Définition 4.7. Soient Σ et Γ deux alphabets. Un *morphisme* de Σ^* dans Γ^* est une application $\varphi : \Sigma^* \to \Gamma^*$ telle que :

— pour tous $u, v \in \Sigma^*$, l'on a $\varphi(uv) = \varphi(u)\varphi(v)$,

— $\varphi(\varepsilon) = \varepsilon$.

Remarque 4.8. Comme $\varphi(uv) = \varphi(u)\varphi(v)$ pour tous $u, v \in \Sigma^*$, il n'est suffisant de définir φ que sur les lettres de Σ.

Définition 4.9. Soit Σ et Γ des alphabets, $\sigma \in \Sigma$ et $k \in \mathbb{N}\backslash\{0\}$.

— Un morphisme $\varphi : \Sigma^* \to \Gamma^*$ est *non-effaçant* si $\varphi(\sigma) \neq \varepsilon$ pour tout $\sigma \in \Sigma$.

1. L'on peut étendre la notion de suite k-automatique pour $k = 1$. Un 1-AFDS travaille donc avec des mots unaires, de la forme 1^n avec n un entier naturel, et les suites 1-automatiques sont en fait exactement les suites ultimement périodiques [9, Theorem 5.7.1].

2. Un résultat d'Adamczeswki et Faverjon [7] nous donne qu'un nombre irrationnel ne peut pas être automatique à la fois en base b_1 et b_2, pour b_1 et b_2 des entiers naturels multiplicativement indépendants.

— Un morphisme $\varphi : \Sigma^* \to \Gamma^*$ est dit *k-uniforme* si pour tout $\sigma \in \Sigma, |\varphi(\sigma)| = k$.
— Plus généralement, on dit que φ est *uniforme* s'il est k-uniforme pour un certain k. On dit que φ est un *codage* si $k = 1$.
— Un morphisme $\varphi : \Sigma^* \to \Sigma^*$ est dit *prolongeable en* σ si $\varphi(\sigma) = \sigma u$, avec u un mot de $\Sigma^* \setminus \{\varepsilon\}$.

Définition 4.10. Soit $\varphi : \Sigma^* \to \Sigma^*$ un morphisme non-effaçant et prolongeable en $\sigma \in \Sigma$, avec $\varphi(\sigma) = \sigma u$. On note

$$\varphi^\omega(\sigma) = \sigma u \varphi(u)\varphi^2(u)\varphi^3(u)\cdots = \lim_{n\to+\infty} \varphi^n(\sigma)$$

le mot de $\Sigma^{\mathbb{N}}$ produit par l'application de φ une infinité de fois.

Proposition 4.11. *Si $\varphi : \Sigma^* \to \Sigma^*$ est non-effaçant et prolongeable en $\sigma \in \Sigma$, alors le mot $\varphi^\omega(\sigma)$ est un point fixe du morphisme φ.*

Démonstration. L'on a d'une part

$$\varphi^\omega(\sigma) = \lim_{n\to+\infty} \sigma u \varphi(u)\varphi^2(u)\cdots\varphi^n(u).$$

D'autre part, l'on a pour tout $n \geq 1$

$$\varphi(\sigma u \varphi(u)\varphi^2(u)\cdots\varphi^n(u)) = \sigma u \varphi(u)\varphi^2(u)\cdots\varphi^n(u)\varphi^{n+1}(u)$$

et

$$\lim_{n\to+\infty} \sigma u \varphi(u)\varphi^2(u)\cdots\varphi^n(u)\varphi^{n+1}(u) = \varphi^\omega(\sigma).$$

L'on en déduit $\varphi^\omega(\sigma) = \varphi(\varphi^\omega(\sigma))$, et donc que $\varphi^\omega(\sigma)$ est bien un point fixe du morphisme φ. □

Définition 4.12. Soient Σ et Γ des alphabets, $\sigma \in \Sigma$, $\gamma \in \Gamma$, $k \in \mathbb{N}\setminus\{0\}$, et soit un mot $w \in \Sigma^{\mathbb{N}}$. On dit que w est :

— *purement morphique* s'il existe $\varphi : \Gamma^* \to \Gamma^*$ un morphisme prolongeable en γ tel que $w = \varphi^\omega(\gamma)$,
— *morphique* s'il existe $\varphi : \Gamma^* \to \Gamma^*$ un morphisme prolongeable en γ, et $\tau : \Gamma^* \to \Sigma^*$ un codage tels que $w = \tau(\varphi^\omega(\sigma))$

Exemple 4.13. La suite de Thue-Morse est produite par le morphisme $\varphi : \Sigma_2 \to \Sigma_2$ défini par $\varphi(0) = 01$ et $\varphi(1) = 10$, qui est 2-uniforme et prolongeable en 0. La suite

$$(t_n)_{n\geq 0} = \varphi^\omega(0)$$

est donc purement morphique.

La suite de Thue-Morse est donc à la fois une suite automatique et morphique. Elle ne relève en fait pas de l'exception : en effet, les mots produits par les automates finis déterministes et les mots produits par les morphismes uniformes coïncident.

Théorème 4.14 (Cobham)**.** *Les mots produits par les morphismes k-uniformes coïncident avec les mots produits par les k-AFDS.*

La démonstration de ce théorème se fait en construisant explicitement les automates et morphismes adéquats. L'exemple suivant présente brièvement la construction, mais le lecteur ou la lectrice souhaitant examiner celle-ci en détails peut consulter le théorème 6.3.2 de [9], et plus généralement le chapitre 6 du même livre.

Exemple 4.15. Illustrons la construction dans le cas de la suite de Thue-Morse $\boldsymbol{t}$.

Pour une suite morphique $\boldsymbol{a} = \tau(\varphi^\omega(\sigma_0))$, avec $\varphi : \Sigma^* \to \Sigma^*$, k-uniforme et prolongeable en $\sigma_0 \in \Sigma$, et un codage $\tau : \Sigma \to \Delta$, l'on construit l'automate

$$\mathcal{A} = (\Sigma, \Sigma_k, \delta, \sigma_0, \Delta, \tau)$$

dont les états sont les lettres de Σ et l'état initial est σ_0. Sa fonction de transition δ est donnée par $\delta(\sigma, i) = \varphi(\sigma)_i$ (c'est à dire la i-ème lettre de $\varphi(\sigma)$). Dans le cas de la suite de Thue-Morse, l'on a $\varphi : \Sigma_2 \to \Sigma_2$, défini par $\varphi(0) = 01$ et $\varphi(1) = 10$, et $\tau : \Sigma_2 \to \Sigma_2$ l'identité. L'on se donne donc l'automate $\mathcal{A} = (\Sigma_2, \Sigma_2, \delta, 0, \Sigma_2, \tau)$, avec

$$\delta(0,0) = 0, \delta(0,1) = 1, \delta(1,0) = 1 \text{ et } \delta(1,1) = 0,$$

qui est exactement l'automate de l'exemple 4.3.

À partir d'un k-automate $\mathcal{A} = (Q, \Sigma_k, \delta, q_0, \Delta, \tau)$ l'on construit le morphisme $\varphi : Q^* \to Q^*$ défini par $\varphi(q) = \delta(q,0)\delta(q,1)\cdots\delta(q,k-1)$ pour tout $q \in Q$. La suite produite par $\tau(\varphi^\omega(q_0))$ est exactement celle produite par l'automate $\mathcal{A}$. Avec l'automate qui produit la suite de Thue-Morse, cela nous donne exactement le morphisme φ avec $\varphi(0) = 01$ et $\varphi(1) = 10$.

Les mots produits par les k-AFDS ont une propriété particulière qui nous sera utile et que nous étudierons plus tard : ils sont de *complexité sous-linéaire*, c'est à dire que leur complexité en facteurs vérifie $p(n) = O(n)$.

Proposition 4.16. *La fonction de complexité en facteurs d'un mot automatique w vérifie $p_w(n) = O(n)$.*

Démonstration. Soit $\varphi : \Sigma^* \to \Sigma^*$ un morphisme k-uniforme et prolongeable en $\sigma \in \Sigma$, et soit $w := \varphi^\omega(\sigma)$. Montrons que la fonction de complexité en facteurs $p_w : \mathbb{N} \to \mathbb{N}$ vérifie, pour tout $n \geq 1$, $p_w(n) \leq (\#\Sigma)^2 kn$. Soit $n \geq 1$. Il existe un entier naturel d tel que $k^d \leq n < k^{d+1}$. Comme w est point fixe de φ, on a $w = \varphi^{d+1}(w)$. Pour tout facteur u de longueur n, il existe des lettres $\gamma, \eta \in \Sigma$ et un entier naturel $i < k^{d+1}$ tels que u est un facteur de $\varphi^{d+1}(\gamma\eta)$ avec $u = (\varphi^{d+1}(\gamma\eta))_{[i,i+n)}$. Le facteur u est entièrement déterminé par γ, η et i. Cela donne $(\#\Sigma)^2 k^{d+1}$ facteurs u possibles. Autrement dit,

$$p_w(n) \leq (\#\Sigma)^2 k^{d+1} = (\#\Sigma)^2 k^d k \leq (\#\Sigma)^2 kn.$$

Pour tout codage $\tau : \Sigma \to \Gamma$, il est clair que $p_{\tau(w)}(n) \leq p_w(n)$ pour tout $n \geq 1$, ce qui donne bien $p_{\tau(w)}(n) = O(n)$. □

Historiquement, Cobham a formulé sa conjecture en termes de *tag machines* : un modèle spécifique de machine de Turing qui comporte deux bandes et trois têtes de lecture/écriture. Ce modèle n'est plus vraiment utilisé dans la littérature actuelle, qui utilise plutôt les morphismes présentés plus tôt pour décrire les suites produites par les tag machines. Sa définition est la suivante.

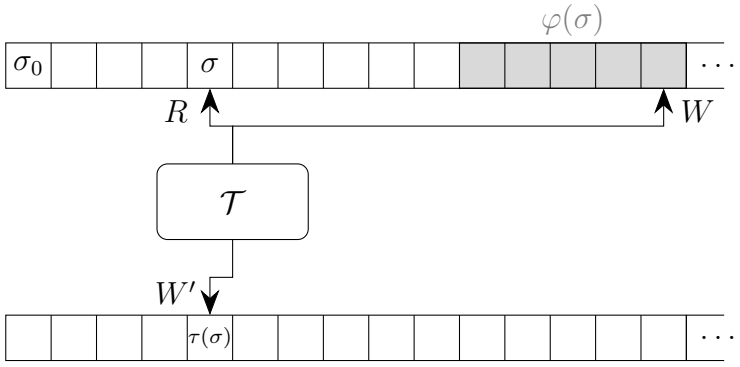


FIGURE 4.2 – Représentation schématique d'une tag machine

Définition 4.17. Une *tag machine* est la donnée d'un quintuple $\mathcal{T} = (\Sigma, \sigma_0, \varphi, \Delta, \tau)$ où

— Σ est un alphabet fini, appelé *l'alphabet interne*,

— $\sigma_0 \in \Sigma$ est *le symbole de départ*,

— $\varphi : \Sigma^* \to \Sigma^*$ est un morphisme prolongeable en σ,

— Δ est un alphabet fini, appelé *alphabet de sortie* et

— $\tau : \Sigma \to \Delta$ est un codage.

Une tag machine $\mathcal{T}$, représentée à la figure 4.2, fonctionne de la manière suivante. Elle a une bande de travail qui comporte deux têtes : une tête R de lecture uniquement, une autre W qui peut uniquement écrire. Sa deuxième bande est la bande de sortie, sur laquelle se trouve une tête W' qui ne peut pas se déplacer à gauche et ne peut qu'écrire.

Quand la machine $\mathcal{T}$ commence son calcul, les deux têtes R et W sont positionnées tout à gauche de la bande de travail, et W écrit le mot $\varphi(\sigma_0)$ sur le ruban. Ensuite, R lit le premier symbole σ écrit sur le ruban, et W écrit $\varphi(\sigma)$. Quand W termine d'écrire, R lit le symbole suivant et l'opération se répète à l'infini, à moins que la tête de lecture R rattrape la tête d'écriture W. En parallèle, pour chaque symbole σ lu par R, la tête W' écrit le symbole $\tau(\sigma)$ sur la bande de sortie. Si le morphisme φ est uniforme, on dit alors que $\mathcal{T}$ est uniforme. Il est assez facile de se convaincre que la machine $\mathcal{T}$ produit la suite $\tau(\varphi^\omega(\sigma_0))$, qui peut donc également produite par un automate fini, par le théorème de Cobham 4.14.

4.2 Problème de Hartmanis–Stearns pour les k-AFDS

Comme annoncé plus tôt, ce chapitre est dédié à la confirmation de la conjecture de Cobham. Nous allons donc montrer que, contrairement à une machine de Turing, un k-AFDS n'est pas un modèle de calcul suffisamment puissant pour produire la b-représentation des irrationnels algébriques. Cette section est reprise des travaux d'Adamczewski, Bugeaud et Luca [3, 4]. Nous en reprenons les principaux résultats et arguments, en apportant par endroits des détails supplémentaires dans certaines

démonstrations. Nous allons commencer par nous donner un puissant critère combinatoire de transcendance.

4.2.1 Critère combinatoire de transcendance

Définition 4.18. Soit $\boldsymbol{a} = (a_n)_{n\geq 1}$ une suite d'éléments de Σ_b. On dit que la suite **a** *satisfait la condition* $(*)$ si elle n'est pas ultimement périodique et s'il existe deux suites de mots finis $(U_n)_{n\geq 1}$ et $(V_n)_{n\geq 1}$ et un réel $w > 1$ tels que :

— $\forall n \geq 1, U_n V_n^w$ est un préfixe de $\boldsymbol{a}$,

— la suite $\left(\frac{|U_n|}{|V_n|}\right)_{n\geq 1}$ est bornée,

— la suite $(|V_n|)_{n\geq 1}$ est strictement croissante.

On dit alors que $\boldsymbol{a}$ est une "stammering sequence", ou *suite bègue* en français.

Le nom de suite bègue vient du fait que les préfixes de la suite $\boldsymbol{a}$ peuvent être mis sous la forme $U_n V_n^w$, ressemblant à une suite ultimement périodique. Dans le cas où $w > 1$ n'est pas un entier naturel, la suite "répète" un même motif avant de le couper, ce qui évoque un phénomène de bégaiement.

Nous allons voir qu'en fait, toute suite de $\Sigma_b^{\mathbb{N}}$ satisfaisant la condition $(*)$ est la b-représentation d'un nombre transcendant. Ce résultat se base sur un théorème énoncé par Schmidt en 1972.

Notation. Pour un vecteur $x = (x_1, \ldots, x_m) \in \mathbb{R}^m$, on note

$$||x||_\infty = \sup_{1\leq i\leq m} |x_i|_\infty$$

sa norme infinie.

Théorème 4.19 (du sous-espace, Schmidt [46])**.** *Soient $L_1, \ldots, L_m$ des formes linéaires à m variables à coefficients algébriques, linéairement indépendantes, et soit $\varepsilon > 0$ un réel. Alors les $x \in \mathbb{Z}^m$ non-nuls tels que*

$$|L_1(x) \cdots L_m(x)| < ||x||_\infty^{-\varepsilon}$$

sont contenus dans un nombre fini de sous-espaces vectoriels propres de $\mathbb{Q}^m$.

Remarque 4.20. Notons que le théorème du sous-espace est une généralisation du théorème de Roth 2.29. En effet, pour tout réel algébrique $\xi \in [0, 1)$ et tout $\varepsilon > 0$, posons les formes linéaires $L_1(x, y) = \xi x - y$ et $L_2(x, y) = x$, qui sont clairement linéairement indépendantes. Alors, en appliquant le théorème du sous-espace, l'on a que toutes les solutions entières non nulles $(p, q) \in \mathbb{Z}^2$ à l'inéquation

$$|q| \cdot |q\xi - p| < |q|^{-\varepsilon}, \tag{4.1}$$

ou, de manière équivalente, à l'inéquation

$$\left|\xi - \frac{p}{q}\right| < |q|^{-2-\varepsilon},$$

sont dans une union finie de sous-espaces vectoriels propres de $\mathbb{Q}^2$. Ces sous-espaces vectoriels propres de $\mathbb{Q}^2$ étant forcément de dimension 1, il existe un nombre fini d'équations

$$\begin{cases} a_1x + b_1y = 0 \\ a_2x + b_2y = 0 \\ \quad\vdots \\ a_nx + b_ny = 0 \end{cases}$$

telles que chaque solution (p, q) de (4.1) vérifie $a_kp + b_kq = 0$ avec $1 \leq k \leq n$ et $a_k, b_k \in \mathbb{Z}$. L'on en déduit que $\frac{p}{q} = -\frac{b_k}{a_k}$. Comme l'on a qu'un nombre fini de rationnels $-\frac{b_k}{a_k}$, il n'y a qu'un nombre fini de solutions rationnelles à l'inéquation (4.1), ce qui est exactement le théorème de Roth. En effet, notons S l'ensemble des solutions $\frac{p}{q}$ de (4.1), et notons $M := \min\left\{\left|\xi - \frac{p}{q}\right| \cdot |q|^{2+\varepsilon} : \frac{p}{q} \in S\right\}$. L'on a $M > 0$ et tout $C \in]0, M[$ convient pour le théorème 2.29.

Une généralisation du théorème du sous-espace a été obtenue par Schlickewei en 1976. Donnons nous quelques définitions avant de l'énoncer.

Définition 4.21. Soit p un nombre premier et $x \in \mathbb{Q}$. La *valeur absolue p-adique* de x est

$$|x|_p = p^{-v_p(x)},$$

avec $v_p(x)$ la *valuation p-adique* de x, c'est à dire le plus grand entier relatif n tel que $x = p^n\frac{a}{b}$, avec $a, b \in \mathbb{Z}$ premiers entre eux, et p ne divisant ni a ni b.

Notation. On notera $|x|_\infty$ la valeur absolue usuelle de $x \in \mathbb{R}$, c'est à dire

$$|x|_\infty = \begin{cases} x & \text{si } x \geq 0 \\ -x & \text{si } x < 0. \end{cases}$$

De la même manière que $\mathbb{R}$ est la complétion du corps $\mathbb{Q}$ via les suites de Cauchy avec la norme usuelle $|\cdot|_\infty$, l'on peut construire d'autres extensions de $\mathbb{Q}$ en utilisant les normes $|\cdot|_p$, ce qui donne les corps de *nombres p-adiques* $\mathbb{Q}_p$. L'on définit de manière similaire $\overline{\mathbb{Q}_p}$ la clôture algébrique du corps $\mathbb{Q}_p$, c'est à dire l'ensemble des nombres algébriques sur le corps $\mathbb{Q}_p$. Ainsi, un nombre p-adique est *algébrique* sur $\mathbb{Q}_p$ s'il est racine d'un polynôme de $\mathbb{Q}_p[X]$.

Théorème 4.22 (Généralisation p-adique, Schlickewei [45])**.** *Soit $\mathcal{P}$ un ensemble fini de nombres premiers, et soit $\mathcal{S} = \mathcal{P} \cup \{\infty\}$. Pour chaque $p \in \mathcal{S}$, soient $L_{1,p}, \ldots, L_{m,p}$ des formes linéaires linéairement indépendantes à m variables, et à coefficients algébriques sur $\mathbb{Q}_p$ (ou sur $\mathbb{Q}$ pour $p = \infty$). Alors pour tout $\varepsilon > 0$, les $x \in \mathbb{Z}^m$ non-nuls tels que*

$$\prod_{p\in\mathcal{S}}\prod_{i=1}^{m} |L_{i,p}(x)|_p < ||x||_\infty^{-\varepsilon}$$

sont contenus dans un nombre fini de sous-espaces vectoriels propres de $\mathbb{Q}^m$.

C'est cette version du théorème, qui utilise les normes p-adiques, qui va nous permettre de montrer finalement que la condition $(*)$ est bien un critère de transcendance.

Théorème 4.23. *Soit un entier $b \geq 2$ et soit $\boldsymbol{a} = (a_n)_{n\geq 1}$ une suite de $\Sigma_b^{\mathbb{N}}$, qui satisfait la condition $(*)$. Alors le nombre réel*

$$\alpha := \sum_{n=1}^{+\infty} a_n b^{-n}$$

est transcendant.

Démonstration. Comme $\boldsymbol{a}$ satisfait la condition $(*)$, on considère fixés les paramètres $(U_n)_{n\geq 1}$, $(V_n)_{n\geq 1}$ et w. Posons

— es suites $(r_n)_{n\geq 1}$ et $(s_n)_{n\geq 1}$ telles que pour tout $n \geq 1, r_n = |U_n|$ et $s_n = |V_n|$,

— la suite $(b_k^{(n)})_{k\geq 1}$ avec $\begin{cases} b_k^{(n)} = a_k & \text{si } k \leq r_n \\ b_{r_n+k+js_n}^{(n)} = a_{r_n+k} & \text{si } 1 \leq k \leq s_n \text{ et } j \geq 0, \end{cases}$

— le nombre $\alpha_n = \sum_{k=1}^{+\infty} \frac{b_k^{(n)}}{b^k}$.

Par définition, $(b_k^{(n)})_{k\geq 1}$ est ultimement périodique, de pré-période U_n et de période V_n. On peut alors réexprimer α_n :

$$\alpha_n = \sum_{k=1}^{r_n} \frac{a_k}{b^k} + \sum_{k=r_n+1}^{+\infty} \frac{b_k^{(n)}}{b^k} = \sum_{k=1}^{r_n} \frac{a_k}{b^k} + \frac{1}{b^{r_n}} \sum_{k=1}^{+\infty} \frac{b_{r_n+k}^{(n)}}{b^k}$$

Par périodicité de $(b_k^{(n)})_{k\geq 1}$, on déduit :

$$\begin{aligned} \alpha_n &= \sum_{k=1}^{r_n} \frac{a_k}{b^k} + \frac{1}{b^{r_n}} \sum_{k=1}^{s_n} \left[\frac{a_{r_n+k}}{b^k} \left(\sum_{j=0}^{+\infty} \left(\frac{1}{b^{s_n}} \right)^j \right) \right] = \sum_{k=1}^{r_n} \frac{a_k}{b^k} + \frac{1}{b^{r_n}} \sum_{k=1}^{s_n} \frac{a_{r_n+k}}{b^k - b^{k-s_n}} \\ &= \sum_{k=1}^{r_n} \frac{a_k}{b^k} + \frac{1}{b^{r_n}} \sum_{k=1}^{s_n} \frac{a_{r_n+k}}{b^{k-s_n}(b^{s_n}-1)} = \frac{p_n}{b^{r_n}(b^{s_n}-1)} \end{aligned}$$

en posant $p_n := \sum_{k=1}^{r_n} a_k b^{r_n-k}(b^{s_n}-1) + \sum_{k=1}^{s_n} a_{r_n+k} b^{s_n-k}$. Notons que p_n est un entier.

Remarquons que α_n est un rationnel qui approxime α, leur développement en base b partageant un préfixe de longueur au moins $r_n + ws_n$. Cela nous donne l'inégalité suivante :

$$|\alpha - \alpha_n| < \frac{1}{b^{r_n+ws_n}}.$$

Comme $(s_n)_{n\geq 1}$ est strictement croissante, l'on obtient que la suite $(\alpha_n)_{n\geq 1}$ tend vers α. Par l'absurde, supposons α algébrique, et notons $\mathcal{P}$ l'ensemble des diviseurs premiers de b. Afin d'appliquer le théorème 4.22, posons les formes linéaires suivantes, à coefficients algébriques et à 3 variables :

$$\begin{aligned} L_{1,\infty}(x,y,z) &= \alpha x + \alpha y + z \\ L_{2,\infty}(x,y,z) &= y \\ L_{3,\infty}(x,y,z) &= z \end{aligned}$$

et pour tout $p \in \mathcal{P}$ posons

$$\begin{aligned} L_{1,p}(x,y,z) &= x \\ L_{2,p}(x,y,z) &= y \\ L_{3,p}(x,y,z) &= z. \end{aligned}$$

Les formes $L_{1,\infty}$, $L_{2,\infty}$, et $L_{3,\infty}$, d'une part, et $L_{1,p}$, $L_{2,p}$ et $L_{3,p}$, d'autre part, sont linéairement indépendantes. Considérons la suite de vecteurs $((b^{r_n+s_n}, -b^{r_n}, -p_n))_{n\geq 1}$ de $\mathbb{Z}^3$. On en déduit les estimations suivantes :

$$\begin{aligned} |L_{1,\infty}(b^{r_n+s_n}, -b^{r_n}, -p_n)|_\infty &= |\alpha b^{r_n+s_n} - \alpha b^{r_n} - p_n| \\ &= \left|\frac{\alpha(b^{r_n}(b^{s_n}-1)) - p_n}{b^{r_n}(b^{s_n}-1)}\right| |b^{r_n}(b^{s_n}-1)| \\ &= |\alpha - \alpha_n||b^{r_n}(b^{s_n}-1)| < \frac{b^{r_n}(b^{s_n}-1)}{b^{r_n+ws_n}} < \frac{1}{b^{(w-1)s_n}} \\ |L_{2,\infty}(b^{r_n+s_n}, -b^{r_n}, -p_n)|_\infty &= b^{r_n} \\ |L_{3,\infty}(b^{r_n+s_n}, -b^{r_n}, -p_n)|_\infty &= p_n \end{aligned}$$

D'autre part, l'on a

$$\prod_{p\in\mathcal{P}} |L_{1,p}(b^{r_n+s_n}, -b^{r_n}, -p_n)|_p = \frac{1}{b^{r_n+s_n}}$$

$$\prod_{p\in\mathcal{P}} |L_{2,p}(b^{r_n+s_n}, -b^{r_n}, -p_n)|_p = \frac{1}{b^{r_n}}$$

$$\prod_{p\in\mathcal{P}} |L_{3,p}(b^{r_n+s_n}, -b^{r_n}, -p_n)|_p \leq 1, \text{ car pour tout } p \in \mathcal{P}, |p_n|_p \leq 1$$

On obtient enfin :

$$\begin{aligned} \prod_{i=1}^{3} |L_{i,\infty}(b^{r_n+s_n}, -b^{r_n}, -p_n)|_\infty \cdot \prod_{i=1}^{3}\prod_{p\in\mathcal{P}} |L_{i,p}(b^{r_n+s_n}, -b^{r_n}, -p_n)|_p &< \frac{p_n}{b^{(w-1)s_n} b^{r_n+s_n}} \\ &\leq \frac{\alpha_n}{b^{(w-1)s_n}} \\ &\leq \frac{1}{b^{(w-1)s_n}} \end{aligned}$$

De plus, la condition $(*)$ nous donne que la suite $\left(\frac{r_n}{s_n}\right)_{n\geq 1}$ est bornée, c'est à dire qu'il existe $K > 0$ tel que pour tout $n \geq 1$, $r_n < Ks_n$, d'où $b^{r_n+s_n} < b^{(K+1)s_n}$ pour tout $n \geq 1$. En posant $\delta := \frac{w-1}{K+1}$, l'on a

$$(b^{r_n+s_n})^\delta < (b^{(K+1)s_n})^\delta = b^{(w-1)s_n},$$

et l'on en déduit

$$\frac{1}{b^{(w-1)s_n}} < (b^{r_n+s_n})^{-\delta}.$$

Comme $||(b^{r_n+s_n}, -b^{r_n}, -p_n)||_\infty = b^{r_n+s_n}$, on en tire finalement :

$$\prod_{i=1}^{3} |L_{i,\infty}(b^{r_n+s_n}, -b^{r_n}, -p_n)|_\infty \cdot \prod_{i=1}^{3}\prod_{p\in\mathcal{P}} |L_{i,p}(b^{r_n+s_n}, -b^{r_n}, -p_n)|_p \leq \left\|(b^{r_n+s_n}, -b^{r_n}, -p_n)\right\|_\infty^{-\delta}$$

Alors, le théorème du sous-espace 4.22 impose aux vecteurs de la suite $((b^{r_n+s_n}, -b^{r_n}, -p_n))_{n\geq 1}$ d'appartenir à un nombre fini de sous-espaces vectoriels propres de $\mathbb{Q}^3$. Ainsi, par le principe des tiroirs, au moins un de ces sous-espaces contient une infinité de ces vecteurs. Notons E ce sous-espace, et fixons (x_0, y_0, z_0) un vecteur non-nul de $E^\perp$, le complément orthogonal de E. On obtient alors par le produit scalaire d'un vecteur de E et de (x_0, y_0, z_0) :

$$x_0 b^{r_n+s_n} - y_0 b^{r_n} - z_0 p_n = 0,$$

ou autrement dit

$$x_0 - \frac{y_0}{b^{s_n}} - \frac{z_0 p_n}{b^{r_n+s_n}} = 0.$$

On a $\lim_{n\to\infty} \frac{p_n}{b^{r_n+s_n}} = \alpha$ et $\lim_{n\to\infty} \frac{1}{b^{s_n}} = 0$ (car $(s_n)_{n\geq 1}$ est strictement croissante). Ainsi, en passant à la limite, selon la sous-suite des vecteurs contenus dans E, on obtient :

$$x_0 = \alpha z_0$$

Or, $x_0, z_0 \in \mathbb{Q}$ et $\alpha \in \mathbb{R} \setminus \mathbb{Q}$, car son développement en base b n'est pas ultimement périodique. On obtient ainsi $x_0 = y_0 = z_0 = 0$, ce qui est une contradiction, car le vecteur (x_0, y_0, z_0) est non-nul. Donc α est transcendant. □

4.2.2 Premier résultat d'impossibilité

Maintenant munis du critère de transcendance qu'est la condition $(*)$ nous pouvons finalement démontrer la conjecture de Cobham. Nous allons en fait montrer un résultat sensiblement plus fort : la b-représentation d'un irrationnel algébrique n'est pas de complexité sous-linéaire.

Théorème 4.24. *La fonction de complexité de la b-représentation d'un nombre irrationnel algébrique satisfait*

$$\liminf_{n\to\infty} \frac{p(n)}{n} = +\infty.$$

Démonstration. Soit α un irrationnel, $\langle\alpha\rangle_b = (a_n)_{n\geq 1}$ sa représentation en base b, et $p : \mathbb{N} \to \mathbb{N}$ la fonction de complexité en facteurs de $\langle\alpha\rangle_b$. Par contraposée, supposons que $p(n) = O(n)$, c'est à dire qu'il existe $\kappa \in \mathbb{N}_0$ tel que pour tout $n \in \mathbb{N}, p(n) \leq \kappa n$. Montrons que α est transcendant, en exhibant les suites de mots $(U_n)_{n\geq 1}$ et $(V_n)_{n\geq 1}$, et $w > 1$, qui conviennent pour que $\langle\alpha\rangle_b$ satisfasse $(*)$.

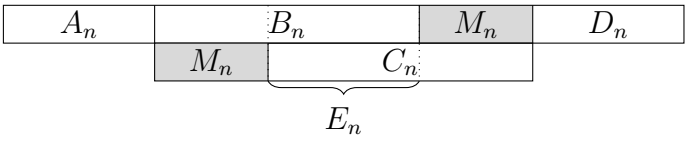
Soient $w = 1 + \frac{1}{\kappa}$ et $n \in \mathbb{N}$. Pour $\ell \in \mathbb{N}$, notons $\mathrm{Pr}(\ell)$ le préfixe de $\langle\alpha\rangle_b$ de longueur ℓ. Alors, par le principe des tiroirs, pour tout $n \geq 1$ il existe un mot M_n

de longueur n tel que M_n apparaît au moins deux fois dans $\Pr((\kappa+1)n)$. Ainsi, il existe des mots $A_n, B_n, C_n, D_n \in \Sigma_b^*$ (possiblement vides) tels que

$$\Pr((\kappa+1)n) = A_nB_nM_nD_n = A_nM_nC_nD_n, \text{ avec } |B_n| \geq 1.$$

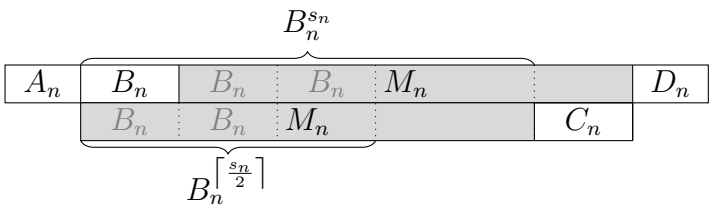
Comme $|M_n| = n$, on a $|A_n| \leq \kappa n$. Distinguons 2 cas :

— Si $|B_n| \geq |M_n|$:

$\Pr((\kappa+1)n)$ 

Notons $E_n \in \Sigma_b^*$ le mot tel que $\Pr((\kappa+1)n) = A_nM_nE_nM_nD_n$. On a $|E_n| \leq (\kappa-1)n = (\kappa-1)|M_n|$. Alors, $A_n(M_nE_n)^{1+\frac{1}{\kappa}}$ est un préfixe de $\langle\alpha\rangle_b$. De plus, on a $|M_nE_n| \geq |M_n| \geq \frac{|A_n|}{\kappa}$. On pose $U_n = A_n$ et $V_n = M_nE_n$.

— Si $|B_n| < |M_n|$:

$\Pr((\kappa+1)n)$ 

Il existe alors $s_n > 1$ tel que $B_n^{s_n} = M_n$. Dans ce cas, comme $|M_n| = n$, l'on a $\frac{n}{2} \leq |B_n^{\lceil \frac{s_n}{2} \rceil}| \leq n$, et $A_n(B_n^{\lceil \frac{s_n}{2} \rceil})^{1+\frac{1}{\kappa}}$ est un préfixe de $\langle\alpha\rangle_b$. L'on pose $U_n = A_n$ et $V_n = B_n^{\lceil \frac{s_n}{2} \rceil}$.

Dans les deux cas, on a construit U_n et V_n tels que, pour tout $n \geq 1$, $U_nV_n^{1+\frac{1}{\kappa}}$ est préfixe de $\langle\alpha\rangle_b$, et $|U_n| \leq \kappa n$, $|V_n| \geq \frac{n}{2}$, d'où $\frac{|U_n|}{|V_n|} \leq 2\kappa$. De plus, comme $|V_n| \geq \frac{n}{2}$, l'on peut supposer $(|V_n|)_{n\geq 1}$ strictement croissante, quitte à en extraire une sous-suite. Alors, $\langle\alpha\rangle_b$ satisfait $(*)$, donc α est transcendant. □

Ce résultat, couplé à la proposition 4.16, nous donne donc directement que les suites automatiques ne peuvent pas être des b-représentations de nombres irrationnels algébriques, ce qui confirme la conjecture de Cobham, et nous permet d'énoncer le théorème suivant. L'on peut cependant, en utilisant les mêmes idées que dans la preuve précédente, construire des suites pour appliquer la condition $(*)$ depuis la caractérisation des suites automatiques via les morphismes.

Théorème 4.25 (Adamczewski-Bugeaud [3])**.** *La b-représentation d'un nombre irrationnel algébrique ne peut pas être produite par un morphisme uniforme, ou de manière équivalente, par un automate fini.*

Démonstration. Soit $\alpha \in [0,1)$ un irrationnel algébrique (b,k)-automatique, et soit $\langle\alpha\rangle_b \in \Sigma_b^{\mathbb{N}}$ sa b-représentation. Il existe alors un morphisme k-uniforme $\varphi : \Sigma^* \to \Sigma^*$ avec $\Sigma = \{1, 2, \ldots, r\}$, prolongeable en $\sigma \in \Sigma$, et un codage $\tau : \Sigma \to \Sigma_b$ tels que

$$\langle\alpha\rangle_b = \tau(\varphi^\omega(\sigma)).$$

Notons $w = \varphi^\omega(\sigma)$. Par le principe des tiroirs, le préfixe de longueur $r+1$ de w peut s'écrire sous la forme $w_1cw_2cw_3$ avec $c \in \Sigma$, et w_1, w_2, w_3 des mots (possiblement vides) de Σ^*.

Pour tout entier $n \geq 1$, posons $U_n = \tau(\varphi^n(w_1)), V_n = \tau(\varphi^n(cw_2))$ et $V'_n = \tau(\varphi^n(c))$. Comme φ est k-uniforme et τ est un codage, l'on a pour tout $n \geq 1$, $|U_n| = |w_1| \cdot k^n$ et $|V_n| = (|w_2|+1) \cdot k^n$. L'on a clairement $(|V_n|)_{n\geq 1}$ strictement croissante et $\left(\frac{|U_n|}{|V_n|}\right)_{n\geq 1}$ bornée. En effet, pour tout $n \geq 1$,

$$\frac{|U_n|}{|V_n|} = \frac{|w_1|}{|w_2|+1}.$$

De plus, pour tout n, V'_n est un préfixe de V_n, et l'on a $|V'_n| = \frac{|V_n|}{|w_2|+1} \geq \frac{1}{r}|V_n|$. L'on obtient donc que $U_n V_n^{1+\frac{1}{r}}$ est préfixe de $\langle\alpha\rangle_b$ pour tout $n \geq 1$. Ainsi, $\langle\alpha\rangle_b$ vérifie le critère $(*)$ et donc α est transcendant. □

4.3 Exposant diophantien

Dans des travaux ultérieurs, Adamczewski et Bugeaud [1] ont introduit un exposant combinatoire qui permet de déduire des propriétés diophantiennes des nombres selon la structure de leur b-représentation. Cette section est donc dédiée à la présentation de ce dernier, l'*exposant diophantien* d'une suite de $\Sigma_b^{\mathbb{N}}$.

Définition 4.26. Soit $\rho \geq 1$. La suite $\boldsymbol{a} = (a_n)_{n\geq 1} \in \Sigma_b^{\mathbb{N}}$ satisfait la condition $(*)_\rho$ si il existe des suites de mots finis $(U_n)_{n\geq 1}$ et $(V_n)_{n\geq 1}$ et une suite de réels $(w_n)_{n\geq 1}$ telles que :

1. $\forall n \geq 1, U_n V_n^{w_n}$ est préfixe de $\boldsymbol{a}$,
2. $\forall n \geq 1, \frac{|U_n V_n^{w_n}|}{|U_n V_n|} \geq \rho$,
3. $(|V_n^{w_n}|)_{n\geq 1}$ est strictement croissante.

On pose alors $\mathrm{Dio}(\boldsymbol{a}) := \sup\{\rho \mid \boldsymbol{a} \text{ satisfait } (*)_\rho\}$, l'*exposant diophantien* de $\boldsymbol{a}$.

Cet exposant s'interprète comme une mesure de la périodicité de la suite $\boldsymbol{a}$. Remarquons que pour toute suite $\boldsymbol{a}$, l'on a toujours

$$1 \leq \mathrm{Dio}(\boldsymbol{a}) \leq +\infty,$$

et que toute suite ultimement périodique a un exposant diophantien infini. Toute suite vérifie trivialement $(*)_\rho$ avec $\rho = 1$, et une suite $\boldsymbol{a}$ pour laquelle $\mathrm{Dio}(\boldsymbol{a}) = 1$ ne comporte ainsi pas de motif périodique.

Bien que l'exposant diophantien d'une suite $\boldsymbol{a} \in \Sigma_b^{\mathbb{N}}$ soit défini de manière purement combinatoire, il possède un lien étroit avec l'approximation diophantienne du nombre dont $\boldsymbol{a}$ est la b-représentation, d'où sa dénomination. Nous allons en particulier voir que, pour tout irrationnel ξ, l'exposant diophantien de $\langle\xi\rangle_b$ est toujours inférieur ou égal à la mesure d'irrationalité de ξ.

Proposition 4.27. *Soit ξ un irrationnel. L'on a alors*

$$\mathrm{Dio}(\langle\xi\rangle_b) \leq \mu(\xi).$$

Démonstration. Soit $\rho \geq 1$ et soient les suites $(U_n)_{n\geq 1}, (V_n)_{n\geq 1}$ et $(w_n)_{n\geq 1}$ telles que $\langle \xi \rangle_b$ vérifie $(*)_\rho$ et pour tout $n \geq 1$, $U_n V_n^{w_n}$ est préfixe de $\langle \xi \rangle_b$. Procédons de manière similaire à la démonstration du théorème 4.24, et posons pour tout n, ξ_n le rationnel dont le b-développement est $U_n V_n^\omega$. On peut réexprimer ξ_n sous la forme

$$\frac{p_n}{q_n} = \frac{p_n}{b^{|U_n|}(b^{|V_n|}-1)}, \text{ avec } p_n \in \mathbb{N}.$$

Comme les b-représentations de ξ et ξ_n partagent un préfixe de longueur au moins $|U_n V_n^{w_n}|$, l'on a

$$|\xi - \xi_n| < \frac{1}{b^{|U_n V_n^{w_n}|}}.$$

L'on en déduit alors

$$\left|\xi - \frac{p_n}{q_n}\right| = |\xi - \xi_n| < \frac{1}{b^{\rho(|U_n|+|V_n|)}} < \frac{1}{(b^{|U_n|}(b^{|V_n|}-1))^\rho} = \frac{1}{q_n^\rho}.$$

L'on a $\rho \leq \mathrm{Dio}(\langle \xi \rangle_b)$ par définition, et l'on en déduit bien

$$\mathrm{Dio}(\langle \xi \rangle_b) \leq \mu(\xi)$$

par définition de mesure d'irrationalité. □

Si l'on peut toujours minorer la mesure d'irrationalité $\mu(\xi)$ d'un irrationnel ξ via l'exposant diophantien de sa b-représentation, nous verrons plus tard que ce dernier intervient également dans la majoration de $\mu(\xi)$ pour certaines classes de nombres.

Le critère $(*)$ de la définition 4.18, apparu dans [3, 4], a été reformulé dans les travaux ultérieurs d'Adamczewski et Bugeaud en terme d'exposant diophantien. Montrons maintenant que les suites bègues, c'est à dire celles qui vérifient le critère $(*)$, ont un exposant diophantien strictement plus grand que 1.

Lemme 4.28. *Pour tout mot fini w et tout $x \geq 0$, on a $x|w| \leq |w^x| < x|w| + 1$.*

Démonstration. Soient un mot fini w et $x \in \mathbb{R}$. On a $|w^x| = \lfloor x \rfloor |w| + \lceil (x - \lfloor x \rfloor)|w| \rceil$. Or,

$$(x - \lfloor x \rfloor)|w| \leq \lceil (x - \lfloor x \rfloor)|w| \rceil < (x - \lfloor x \rfloor)|w| + 1.$$

L'on en déduit

$$x|w| = \lfloor x \rfloor |w| + (x - \lfloor x \rfloor)|w| \leq \lfloor x \rfloor |w| + \lceil (x - \lfloor x \rfloor)|w| \rceil = |w^x|$$

et

$$|w^x| = \lfloor x \rfloor |w| + \lceil (x - \lfloor x \rfloor)|w| \rceil \leq \lfloor x \rfloor |w| + (x - \lfloor x \rfloor)|w| + 1 = x|w| + 1.$$

□

Proposition 4.29. *Si la suite $\boldsymbol{a} \in \Sigma_b^{\mathbb{N}}$ satisfait $(*)$, alors* $\mathrm{Dio}(\boldsymbol{a}) > 1$.

Démonstration. Soient le réel $w > 1$, et les suites $(U_n)_{n\geq 1}$ et $(V_n)_{n\geq 1}$ telles que la suite $\boldsymbol{a} \in \Sigma_b^{\mathbb{N}}$ satisfait $(*)$. On pose la suite $(w_n)_{n\geq 1}$ telle que pour tout $n \geq 1, w_n = w$. On a bien que $U_n V_n^w$ est préfixe de $\boldsymbol{a}$ pour tout $n \geq 1$. De plus, on a, pour tout n,

$$\begin{aligned}\frac{|U_n V_n^w|}{|U_n V_n|} &\geq \frac{|U_n| + w|V_n|}{|U_n| + |V_n|} = 1 + \frac{(w-1)|V_n|}{|U_n| + |V_n|} \\ &\geq 1 + \frac{(w-1)|V_n|}{(C+1)|V_n|} = 1 + \frac{w-1}{C+1},\end{aligned}$$

avec C une constante qui majore $\left(\frac{|U_n|}{|V_n|}\right)_{n\geq 1}$. Enfin, comme $(|V_n|)_{n\geq 1}$ est strictement croissante on a bien $(|V_n^w|)_{n\geq 1}$ strictement croissante également. On a donc que $\mathbf{a}$ satisfait la condition $(*)_\rho$ avec $\rho = 1 + \frac{w-1}{C+1} > 1$, d'où $\mathrm{Dio}(\boldsymbol{a}) > 1$. □

Proposition 4.30. *Si $\boldsymbol{a}$ n'est pas ultimement périodique et* $\mathrm{Dio}(\boldsymbol{a}) > 1$*, alors $\boldsymbol{a}$ satisfait* $(*)$.

Démonstration. Soient un $\rho > 1$, une suite $\boldsymbol{a}$ non ultimement périodique, et les suites $(U_n)_{n\geq 1}, (V_n)_{n\geq 1}$ et $(w_n)_{n\geq 1}$ telles que $\boldsymbol{a}$ satisfait $(*)_\rho$. Comme l'on a

$$\frac{|U_n V_n^{w_n}|}{|U_n V_n|} \geq \rho > 1, \tag{4.2}$$

l'on en déduit que $w_n > 1$ pour tout $n \geq 1$. En posant

$$w := \inf\{w_n \mid n \geq 1\},$$

l'on a bien que $U_n V_n^w$ est un préfixe de $\boldsymbol{a}$ pour tout $n \geq 1$.

Montrons par l'absurde que l'on a $w > 1$ et supposons qu'il existe une sous-suite $(w_{k(n)})_{n\geq 1}$ de $(w_n)_{n\geq 1}$ (avec $k : \mathbb{N} \to \mathbb{N}$ strictement croissante) telle que $\lim_{n\to+\infty} w_{k(n)} = 1$. Pour tout n assez grand, l'on a ainsi $w_{k(n)} < 2$, et donc $|V_{k(n)}^{w_{k(n)}}| \leq 2|V_{k(n)}|$. Comme la suite $(|V_n^{w_n}|)_{n\geq 1}$ est strictement croissante, la sous-suite $(|V_{k(n)}^{w_{k(n)}}|)_{n\geq 1}$ l'est également, et l'on en déduit que la suite $(|V_{k(n)}|)_{n\geq 1}$ n'est pas bornée. Ainsi, quitte à passer à une nouvelle sous-suite, l'on peut supposer

$$\lim_{n\to+\infty} |V_{k(n)}| = +\infty. \tag{4.3}$$

Notons, pour tout n, $w_{k(n)} = 1 + \varepsilon_{k(n)}$. L'on a donc $\varepsilon_{k(n)} > 0$ pour tout n et

$$\lim_{n\to+\infty} \varepsilon_{k(n)} = 0 \tag{4.4}$$

De plus, l'on a $\lfloor w_{k(n)} \rfloor = 1$ et $\varepsilon_{k(n)} < 1$ pour tout n assez grand. L'on obtient alors

$$\begin{aligned}1 < \rho \leq \frac{\left|U_{k(n)} V_{k(n)}^{w_{k(n)}}\right|}{|U_{k(n)} V_{k(n)}|} &= \frac{|U_{k(n)} V_{k(n)}| + \lceil \varepsilon_{k(n)} |V_{k(n)}| \rceil}{|U_{k(n)} V_{k(n)}|} \\ &< 1 + \frac{\varepsilon_{k(n)} |V_{k(n)}| + 1}{|U_{k(n)} V_{k(n)}|} \\ &\leq 1 + \varepsilon_{k(n)} + \frac{1}{|U_{k(n)} V_{k(n)}|}\end{aligned}$$

L'on en tire par (4.3) et (4.4) que

$$\lim_{n\to+\infty} \frac{\left|U_{k(n)}V_{k(n)}^{w_{k(n)}}\right|}{|U_{k(n)}V_{k(n)}|} = 1$$

ce qui est une contradiction car $\rho > 1$. L'on a donc bien $w > 1$.

S'il existe une sous-suite $\left(\frac{|U_{k(n)}|}{|V_{k(n)}|}\right)_{n\geq 1}$ bornée, alors la proposition est démontrée. Sinon, cela signifie

$$\lim_{n\to+\infty} \frac{|U_n|}{|V_n|} = +\infty. \tag{4.5}$$

Construisons alors une suite de mots $(V'_n)_{n\geq 1}$ et un réel $w' > 1$ qui vérifient :

— $(V'_n)^{w'}$ est un préfixe de $V_n^{w_n}$ pour tout $n \geq 1$,

— la suite $\left(\frac{|U_n|}{|V'_n|}\right)_{n\geq 1}$ est bornée,

— la suite $(V'_n)_{n\geq 1}$ est strictement croissante.

Posons alors, pour tout $n \geq 1$,

$$t_n := \left\lfloor \frac{\rho-1}{\rho}\frac{|U_n|}{|V_n|} \right\rfloor, V'_n := V_n^{t_n} \text{ et } w' = \inf\left\{ \frac{w_n}{t_n} \mid n \geq 1 \right\}.$$

L'on a $\lim_{n\to+\infty} t_n = +\infty$. Comme pour tout $n \geq 1$, l'on a $|V'_n| = t_n|V_n|$, l'on peut alors supposer $(V'_n)_{n\geq 1}$ strictement croissante, quitte à en extraire une sous-suite. D'autre part, l'on déduit de (4.2)

$$\rho|U_nV_n| \leq |U_n| + |V_n^{w_n}|,$$

et l'on en tire

$$(\rho-1)|U_n| + \rho|V_n| < w_n|V_n| + 1 \leq (w_n+1)|V_n|,$$

car V_n n'est pas le mot vide. L'on en déduit alors

$$0 < (\rho-1)\frac{|U_n|}{|V_n|} < (\rho-1)\left(\frac{|U_n|}{|V_n|} + 1\right) < w_n.$$

D'où, pour tout $n \geq 1$, $\rho t_n < w_n$ et donc $\frac{w_n}{t_n} > \rho$. Donc $w' \geq \rho > 1$. De plus, comme $t_n > \frac{\rho-1}{\rho}\frac{|U_n|}{V_n} - 1$, l'on a

$$\begin{aligned}\frac{|U_n|}{|V'_n|} = \frac{|U_n|}{|V_n^{t_n}|} = \frac{|U_n|}{t_n|V_n|} &\leq \frac{|U_n|}{\frac{\rho-1}{\rho}|U_n| - |V_n|} \\ &= \frac{1}{\frac{\rho-1}{\rho} - \frac{|V_n|}{|U_n|}}.\end{aligned}$$

L'on tire de (4.5) que $\lim_{n\to+\infty} \frac{|V_n|}{|U_n|} = 0$, et l'on a donc bien que la suite $\left(\frac{|U_n|}{|V'_n|}\right)_{n\geq 1}$ est bornée. Finalement, l'on a que

$$(V'_n)^{w'} = (V_n^{t_n})^{w'} = V_n^{t_nw'}$$

est bien un préfixe de $V_n^{w_n} = (V'_n)^{\frac{w_n}{t_n}}$, car $w'_n \leq \frac{w_n}{t_n}$ pour tout $n \geq 1$.

□

Ces deux formulations sont donc presque équivalentes, à ceci près que le critère $(*)$ impose à la suite de ne pas être ultimement périodique. Cependant, $\mathrm{Dio}(\boldsymbol{a}) > 1$ englobe les suites ultimement périodiques, pour lesquelles $\mathrm{Dio}(\boldsymbol{a}) = +\infty$. L'on peut donc ainsi reformuler le théorème 4.23 de la manière suivante.

Théorème 4.31. *Soit ξ un nombre réel et soit $\langle\xi\rangle_b \in \Sigma_b^{\mathbb{N}}$ sa b-représentation. Si $\mathrm{Dio}(\langle\xi\rangle_b) > 1$ alors ξ est rationnel ou transcendant.*

Les irrationnels algébriques ont ainsi un exposant diophantien égal à 1, et il n'y a donc aucune périodicité dans leur b-représentation.

4.4 Propriétés diophantiennes des nombres automatiques

Les k-AFDS produisent donc nécessairement des nombres rationnels ou transcendants. Dans le second cas, une question naturelle à se poser est celle des propriétés diophantiennes de tels nombres automatiques, et en particulier à quelle classe de la classification de Mahler ceux-ci appartiennent. Il est aujourd'hui conjecturé que les nombres automatiques irrationnels sont tous des S-nombres.

Conjecture (Becker)**.** Tout nombre irrationnel automatique est un S-nombre.

La conjecture de Becker est une version plus forte d'une conjecture énoncée par Shallit en 1993, qui affirme qu'un nombre de Liouville ne peut être automatique [2]. Rappelons que pour un irrationnel ξ, l'on définit $w_n(\xi)$ comme le supremum des réels w pour lesquels il existe une infinité de polynômes de $\mathbb{Z}[X]$ de degré au plus n pour lesquels

$$0 < |P(\xi)| < H(P)^{-w},$$

avec $H(P)$ la hauteur du polynôme P. La quantité $w_n(\xi)$ mesure à quelle "précision" l'on peut approcher l'irrationnel ξ par des nombres algébriques de degré au plus d. De plus l'on pose

$$w(\xi) = \limsup_{d\to+\infty} \frac{w_d(\xi)}{d},$$

et les S-nombres sont les irrationnels ξ tels que $0 < w(\xi) < +\infty$ (voir les définitions 2.30, 2.31 et 2.33).

Cette section ainsi que la suivante vont présenter des résultats allant dans le sens de cette conjecture. En effet, nous concluerons que tout nombre irrationnel automatique est ou bien un S-nombre ou bien un T-nombre. Adamczewski et Cassaigne [5] ont montré que la mesure d'irrationnalité d'un nombre automatique (voir définition 2.28), ainsi que l'exposant diophantien de sa b-représentation sont tous deux finis. Ceci signifie en particulier que les k-AFDS ne peuvent pas produire de nombre de Liouville, et confirme la conjecture de Shallit. La présente section reprend ainsi leurs travaux et est dédiée à la démonstration de ces deux résultats. Commençons par nous donner une définition préliminaire.

Définition 4.32. Soit $\boldsymbol{a} = (a_n)_{n\geq 1} \in \Sigma_b^{\mathbb{N}}$. Son *k-noyau* est l'ensemble des sous-suites de $\boldsymbol{a}$ de la forme

$$N_k(\boldsymbol{a}) = \{(a_{k^p n+r})_{n\geq 1} \mid p \in \mathbb{N}\setminus\{0\}, 0 \leq r < k^p\}.$$

Exemple 4.33. La suite de Thue-Morse $(t_n)_{n\geq 0}$ est 2-automatique. L'on a, pour tout $n \geq 1$, $t_{2n} = t_n$ et $t_{2n+1} = (t_n + 1) \mod 2$. Le 2-noyau de cette suite est donc composé des deux sous-suites

$$\{(t_n)_{n\geq 0}, (\bar{t}_n)_{n\geq 0}\},$$

avec $\bar{t}_n = (t_n + 1) \mod 2$ pour tout $n \geq 0$.

Cette définition est utile car une caractérisation des suites k-automatiques est que leur k-noyau est fini.

Théorème 4.34 ([9, Theorem 6.6.2])**.** *Une suite $\boldsymbol{a}$ de $\Sigma_b^{\mathbb{N}}$ est k-automatique si et seulement si son k-noyau $N_k(\boldsymbol{a})$ est fini.*

Nous allons voir que si ξ est un nombre automatique, le cardinal du k-noyau de sa b-représentation intervient dans la majoration de sa mesure d'irrationalité et de son exposant diophantien.

Proposition 4.35. *Soit $\boldsymbol{a} \in \Sigma_b^{\mathbb{N}}$ une suite non-périodique et k-automatique. Soient $U \in \Sigma_b^*$, $V \in \Sigma_b^* \setminus \{\varepsilon\}$ et $s \in \mathbb{R}$ tels que UV^s est un préfixe de $\boldsymbol{a}$. Soit $m = \#N_k(\boldsymbol{a})$. Alors*

$$\frac{|UV^s|}{|UV|} < k^m.$$

Démonstration. Un triplet $(h, p, \ell) \in \mathbb{N}^3$ est dit admissible pour la suite $\boldsymbol{a}$ si

— $1 \leq p \leq h \leq \ell$,

— pour tout entier n tel que $h \leq n < \ell$, $a_{n-p} = a_n$,

— $a_{\ell-p} \neq a_\ell$.

Commençons par montrer que pour tout triplet (h, p, ℓ) admissible pour $\boldsymbol{a}$, on a $\ell < hk^m$. Procédons par l'absurde et supposons qu'il existe (h, p, ℓ) tel que $\ell \geq hk^m$. Pour tout entier i tel que $0 \leq i \leq m$, nous allons construire un triplet (h_i, p, ℓ_i) admissible pour une suite $\boldsymbol{a}^{(i)} \in N_k(\boldsymbol{a})$. Commençons par $\boldsymbol{a}^{(0)} = \boldsymbol{a}$, avec $h_0 = h$ et $\ell_0 = \ell$. Ensuite, pour un $\boldsymbol{a}^{(i)} = (a_n^{(i)})_{n\geq 1}$ et un triplet (h_i, p, ℓ_i), on pose $r_i = \ell_i \mod k$, et l'on définit $\boldsymbol{a}^{(i+1)} = (a^{(i)}_{kn+r_i})_{n\geq 1}$. De plus, l'on pose $\ell_{i+1} = \left\lfloor \frac{\ell_i}{k} \right\rfloor$ et $h_{i+1} = \ell_{i+1} + p - \left\lfloor \frac{\ell_i + p - h_i}{k} \right\rfloor$. Par récurrence sur i, montrons que pour tout $0 \leq i \leq m$, l'on a

a) $\boldsymbol{a}^{(i)} \in N_k(\boldsymbol{a})$,

b) $h_i \leq h_0$,

c) $\ell_i \geq h_0 k^{m-i}$,

d) (h_i, p, ℓ_i) est admissible pour $\boldsymbol{a}^{(i)}$.

Le cas $i = 0$ est vérifié par hypothèse. Supposons maintenant toutes ces propriétés vraies pour $\boldsymbol{a}^{(i)}$ et montrons qu'elles le sont pour $\boldsymbol{a}^{(i+1)}$.

a) Par construction, il est clair que $\boldsymbol{a}^{(i+1)} \in N_k(\boldsymbol{a})$.

b) De plus l'on a $h_{i+1} \leq h_i$. En effet,

$$\begin{aligned} h_i - h_{i+1} &= h_i - p - \ell_{i+1} + \left\lfloor \frac{\ell_i + p - h_i}{k} \right\rfloor \\ &= \left\lfloor \frac{k(h_i - p - \ell_{i+1}) + \ell_i + p - h_i}{k} \right\rfloor \\ &= \left\lfloor \frac{r_i + (k-1)(h_i - p)}{k} \right\rfloor \geq 0 \end{aligned}$$

car $r_i \geq 0, k \geq 2$ et $h_i \geq p$ par hypothèse. On a donc $h_{i+1} \leq h_i \leq h_0$.

c) Par hypothèse, l'on a $\ell_i \geq h_0 k^{m-i}$. Alors

$$\ell_{i+1} = \left\lfloor \frac{\ell_i}{k} \right\rfloor \geq \left\lfloor \frac{h_0 k^{m-i}}{k} \right\rfloor = k^{m-i-1} h_0.$$

d) Montrons que (h_{i+1}, p, ℓ_{i+1}) est admissible pour $\boldsymbol{a}^{(i+1)}$. Comme $p - h_i \leq 0$, l'on a

$$p - h_{i+1} = p - \ell_{i+1} - p + \left\lfloor \frac{\ell_i + p - h_i}{k} \right\rfloor \leq \left\lfloor \frac{\ell_i}{k} \right\rfloor - \ell_{i+1} = 0.$$

De plus, par le point c), on a

$$\ell_{i+1} - h_{i+1} = \left\lfloor \frac{\ell_i + p - h_i}{k} \right\rfloor - p \geq \left\lfloor \frac{h_0 k^{m-i} + p - h_i}{k} \right\rfloor - p,$$

et comme l'on a $i \leq m$ et $h_i \leq h_0$ par le point b), l'on a

$$\begin{aligned} \ell_{i+1} - h_{i+1} &\geq \left\lfloor \frac{h_i k^{m-i} + p - h_i}{k} \right\rfloor - p = \left\lfloor \frac{h_i(k^{m-i} - 1) - (k-1)p}{k} \right\rfloor \\ &\geq \left\lfloor \frac{(h_i - p)(k-1)}{k} \right\rfloor \geq 0. \end{aligned}$$

On a donc $1 \leq p \leq h_{i+1} \leq \ell_{i+1}$. Par hypothèse, l'on a pour tout n tel que $h_i \leq n < \ell_i$, $a^{(i)}_{n-p} = a^{(i)}_n$, et donc $a^{(i)}_{n-jp} = a^{(i)}_n$, pour tous n, j tels que $j \geq 0$ et $h_i + (j-1)p \leq n < \ell_i$. D'autre part, l'on a

$$kh_{i+1} + r_i = \ell_i + kp - k \left\lfloor \frac{\ell_i + p - h_i}{k} \right\rfloor \geq h_i + (k+1)p.$$

Donc, $a^{(i)}_{n-kp} = a^{(i)}_n$ pout tout n tel que $kh_{i+1} + r_i \leq n < \ell_i$, et donc $a^{(i)}_{k(n-p)+r_i} = a^{(i)}_{kn+r_i}$ pour tout n tel que $kh_{i+1} + r_i \leq kn + r_i < k\ell_{i+1} + r_i$. Ceci signifie que l'on a

$$a^{(i+1)}_{n-p} = a^{(i+1)}_n$$

pour tout n tel que $h_{i+1} \leq n < \ell_{i+1}$. Comme $k\ell_{i+1} + r_i = \ell_i$, l'on a $a^{(i+1)}_{\ell_{i+1}} = a^{(i)}_{\ell_i}$ et $a^{(i+1)}_{\ell_{i+1}-p} = a^{(i)}_{\ell_i - kp}$. De plus, $\ell_{i+1} \geq h_{i+1}$ nous donne que

$$\left\lfloor \frac{\ell_i + p - h_i}{k} \right\rfloor \geq p,$$

et donc que $\ell_i \geq h_i + (k-1)p$. On a alors $\ell_i - p \geq h_i + (k-2)p$ et l'on obtient

$$a^{(i+1)}_{\ell_{i+1}-p} = a^{(i)}_{\ell_i - kp} = a^{(i)}_{(\ell_i - p)-(k-1)p} \neq a^{(i)}_{\ell_i} = a^{(i+1)}_{\ell_{i+1}}.$$

L'on obtient bien que le triplet (h_{i+1}, p, ℓ_{i+1}) est admissible pour $\boldsymbol{a}^{(i+1)}$.

Levons maintenant une contradiction. Comme le noyau $N_k(\boldsymbol{a})$ contient exactement m éléments, au moins deux suites $\boldsymbol{a}^{(i)}$ et $\boldsymbol{a}^{(j)}$, avec $0 \leq i < j \leq m$ doivent être égales. Ceci signifie que deux triplets différents (h_i, p, ℓ_i) et (h_j, p, ℓ_j) sont admissibles pour une même suite, et que pour tout n positif, $a^{(i)}_n = a^{(j)}_n$. De plus, l'on a $\ell_j \geq h_0 k^{m-j} \geq h_0 \geq h_i$, tandis que $\ell_j = \left\lfloor \frac{\ell_i}{k^{j-i}} \right\rfloor < \ell_i$. Alors, l'on a $h_i \leq \ell_j < \ell_i$, et $a^{(i)}_{\ell_j - p} = a^{(i)}_{\ell_j}$. D'autre part, l'on a

$$a^{(i)}_{\ell_j - p} = a^{(j)}_{\ell_j - p} \neq a^{(j)}_{\ell_j} = a^{(i)}_{\ell_j},$$

ce qui est une contradiction. L'on a donc bien $\ell < hk^m$ pour tout triplet (h, p, ℓ) admissible pour $\boldsymbol{a}$.

Soient U, V des mots de Σ_b^* et $s \in \mathbb{R}$ tels que UV^s est un préfixe de $\boldsymbol{a}$. Comme $\boldsymbol{a}$ est non périodique, il existe un $s' \in \mathbb{R}$ maximal tel que $UV^{s'}$ est un préfixe de $\boldsymbol{a}$, et le triplet $(|UV|, |V|, |UV^{s'}|)$ est admissible pour $\boldsymbol{a}$. L'on a donc $|UV^{s'}| < k^m |UV|$, et par conséquent $|UV^s| < k^m |UV|$ car s' est maximal. Autrement dit, l'on a bien

$$\frac{|UV^s|}{|UV|} < k^m$$

pour tout préfixe UV^s de $\boldsymbol{a}$. □

De par la définition d'exposant diophantien, l'on en déduit directement le corollaire suivant.

Corollaire 4.36. *L'exposant diophantien d'une suite k-automatique $\boldsymbol{a}$ vérifie*

$$\mathrm{Dio}(\boldsymbol{a}) \leq k^m,$$

avec $m = \#N_k(\boldsymbol{a})$.

Montrons maintenant que la mesure d'irrationalité d'un nombre automatique est également finie.

Lemme 4.37. *Soient $U, V \in \Sigma_b^*$, avec $|U| = r, |V| = s$, et soit $\frac{p}{q} \in \mathbb{Q}$ tel que*

$$\left\langle \frac{p}{q} \right\rangle_b = UV^\omega = a_1 a_2 a_3 \cdots .$$

Soit $\xi \in [0, 1)$ avec $\langle \xi \rangle_b = b_1 b_2 b_3 \cdots$ tel qu'il existe $j \geq r + 1$ avec

- *$a_n = b_n$ pour $1 \leq n < j$ et*
- *$a_j \neq b_j$.*

Alors

$$\left| \xi - \frac{p}{q} \right| > \frac{1}{b^{j+s}}.$$

Démonstration. Notons $\ell := a_j$ et $m := b_j$. Commençons par supposer $\ell > m$. L'on en déduit $\ell \geq 1$ et

$$\frac{p}{q} > 0.a_1a_2\cdots a_{j-1}\ell\underbrace{00\cdots0}_{s-1 \text{ fois}}\ell,$$

tandis que $\xi \leq 0.a_1a_2a_3\cdots a_{j-1}(m+1) \leq 0.a_1a_2a_3\cdots a_{j-1}\ell$. L'on en tire $\frac{p}{q}-\xi > \frac{1}{b^{j+s}}$.

Si maintenant l'on suppose $\ell < m$, l'on a $\ell \leq b-2$ et

$$\begin{aligned}\frac{p}{q} &< 0.a_1a_2a_3\cdots a_{j-1}\ell\underbrace{(b-1)(b-1)\cdots(b-1)}_{s-1 \text{ fois}}(\ell+1)\\ &\leq 0.a_1a_2a_3\cdots a_{j-1}\ell\underbrace{(b-1)(b-1)\cdots(b-1)}_{s \text{ fois}},\end{aligned}$$

alors que $\xi \geq 0.a_1a_2a_3\cdots a_{j-1}(\ell+1)$. L'on a alors $\xi - \frac{p}{q} > \frac{1}{b^{j+s}}$, et le lemme est démontré. □

Théorème 4.38. *Soit $\boldsymbol{a} \in \Sigma_b^{\mathbb{N}}$ k-automatique, avec $\boldsymbol{a} = \tau(\varphi^\omega(\sigma))$, pour un codage $\tau : \Sigma \to \Sigma_b$, un morphisme k-uniforme $\varphi : \Sigma^* \to \Sigma^*$ et prolongeable en $\sigma \in \Sigma$. Soient $m := \#N_k(\boldsymbol{a})$ et $d := \#\Sigma$. Alors la mesure d'irrationalité $\mu(\xi)$ du nombre*

$$\xi := \sum_{n=1}^{+\infty} a_n b^{-n}$$

satisfait

$$\mu(\xi) \leq dk(k^m+1).$$

Démonstration. Sans perte de généralité, l'on peut supposer que la suite $\boldsymbol{a}$ n'est pas ultimement périodique. En effet, dans ce cas, l'on aurait que le nombre ξ est rationnel et donc $\mu(\xi) = 1 \leq dk(k^m+1)$. Soient un réel $\delta > 0$ et un couple d'entiers non-nuls (p,q). Nous allons montrer que pour q assez grand, l'on a

$$\left|\xi - \frac{p}{q}\right| \geq \frac{1}{q^{M+\delta}} \tag{4.6}$$

avec $M = dk(k^m+1)$. Ceci nous donnera $\mu(\xi) \leq M$.

Commençons par construire une suite de rationnels qui approximent ξ, en procédant de manière similaire à la preuve du théorème 4.25. Notons $\boldsymbol{u} = \varphi^\omega(\sigma)$. Par le principe des tiroirs, il existe une lettre $c \in \Sigma$ qui apparait au moins deux fois dans le préfixe de $\boldsymbol{u}$ de longueur $d+1$, noté $\Pr(d+1)$. Autrement dit, l'on peut écrire $\Pr(d+1) = UcVcW$, avec U, V, W des mots de Σ^* possiblement vides et $|U| + |V| \leq d-1$. Ceci signifique que pour tout $n \geq 1$, $\varphi^n(UcVc)$ est un préfixe de $\boldsymbol{u}$, et donc $\tau(\varphi^n(UcVc))$ est un préfixe de $\boldsymbol{a}$. Posons alors pour tout $n \geq 1$ les suites $U_n = \tau(\varphi^n(U))$, $r_n = |U_n|$, $V_n = \tau(\varphi^n(cV))$, $s_n = |V_n|$, et $t_n = |\tau(\varphi^n(c))|$. Comme φ est k-uniforme et τ est un codage, l'on a $r_n = k^n|U|$, $s_n = k^n(|V|+1)$, et $t_n = k^n$. De manière similaire à la construction dans la preuve de la proposition 4.24, l'on pose pour tout $n \geq 1$, $q_n = b^{r_n}(b^{s_n}-1)$ et l'entier p_n tel que

$$\left\langle \frac{p_n}{q_n} \right\rangle_b = U_n V_n^\omega.$$

Comme $\tau(\varphi^n(UcVc))$ est un préfixe commun de $\boldsymbol{a}$ et $U_nV_n^\omega$, l'on a

$$\left|\xi - \frac{p_n}{q_n}\right| < \frac{1}{b^{r_n+s_n+t_n}}.$$

De plus, remarquons que

$$\frac{t_n}{r_n+s_n} = \frac{k^n}{k^n|UcV|} = \frac{1}{|UcV|} \geq \frac{1}{d}.$$

Ceci nous donne

$$\frac{1}{b^{r_n+s_n+t_n}} = \frac{1}{(b^{r_n+s_n})^{1+\frac{t_n}{r_n+s_n}}} \leq \frac{1}{q_n^{1+\frac{1}{d}}},$$

et donc

$$\left|\xi - \frac{p_n}{q_n}\right| < \frac{1}{q_n^{1+\frac{1}{d}}}. \tag{4.7}$$

De plus, comme $r_{n+1} = kr_n$ et $s_{n+1} = ks_n$, l'on en déduit que

$$q_{n+1} = b^{kr_n}(b^{ks_n} - 1) < (b^{r_n}(b^{s_n} - 1))^{k+\varepsilon} = q_n^{k+\varepsilon} \tag{4.8}$$

pour tout $\varepsilon > 0$ et n assez grand. En effet, l'on a toujours $(b^{s_n})^k - 1 \geq (b^{s_n} - 1)^k$ pour tout n. Or, comme

$$\lim_{n\to+\infty} \frac{(b^{s_n})^k - 1}{(b^{s_n} - 1)^k} = 1 \text{ et } \lim_{n\to+\infty} s_n = +\infty,$$

l'on en déduit, pour tout $\varepsilon > 0$,

$$b^{\varepsilon s_n} > \frac{(b^{s_n})^k - 1}{(b^{s_n} - 1)^k},$$

ou autrement dit

$$b^{ks_n} - 1 < (b^{s_n} - 1)^{k+\varepsilon}$$

pour tout n assez grand. L'on en déduit que pour tout $\varepsilon > 0$, l'on a

$$b^{kr_n}(b^{ks_n} - 1) < (b^{r_n}(b^{s_n} - 1))^{k+\varepsilon},$$

et donc $q_{n+1} < q_n^{k+\varepsilon}$ pour tout n assez grand.

Posons maintenant un ε suffisamment petit pour que

$$d\varepsilon + d\varepsilon^2 + \varepsilon dk^m + \varepsilon dk < \frac{\delta}{2}. \tag{4.9}$$

Soit $n_0 \geq 1$ tel que l'équation (4.8) est vraie pour tout $n \geq n_0$. La proposition 4.35 nous donne que pour tout préfixe de $\boldsymbol{a}$ la forme $U_nV_n^s$ avec $s \in \mathbb{R}$, l'on a $|U_nV_n^s| < |U_nV_n|k^m$. Ceci signifie que $\langle\xi\rangle_b$ et $\langle\frac{p}{q}\rangle_b$ ont un préfixe commun de longueur au plus $(r_n+s_n)k^m-1$. Alors, les conditions du lemme précédent sont vérifiées pour un entier j tel que $r_n + s_n + t_n < j \leq (r_n + s_n)k^m$. Cela nous donne

$$\left|\xi - \frac{p_n}{q_n}\right| \geq \frac{1}{b^{(r_n+s_n)k^m+s_n}} > \frac{1}{(b^{r_n}(b^{s_n} - 1))^{k^m+1+\varepsilon}} = \frac{1}{q_n^{k^m+1+\varepsilon}} \tag{4.10}$$

pour tout n assez grand. Il existe ainsi un entier $n_1 \geq n_0$ tel que

$$\left|\xi - \frac{p_n}{q_n}\right| > \frac{1}{q_n^{k^m+1+\varepsilon}}$$

pour tout $n \geq n_1$.

Montrons maintenant que l'inégalité (4.6) est vérifiée pour un q assez grand. Posons un rationnel $\frac{p}{q}$ avec q qui vérifie les inégalités suivantes :

$$2q \geq q_{n_1+1}^{\frac{1}{d}}, q \geq 2^{1+\frac{2M}{\delta}} \text{ et } q \geq 2^{1+dk+\frac{\delta}{2}}.$$

L'on suppose également que $\frac{p}{q}$ vérifie

$$\left|\xi - \frac{p}{q}\right| < \frac{1}{(2q)^{1+d(k+\varepsilon)}}. \tag{4.11}$$

Remarquons que si cette dernière inégalité n'est pas satisfaite, alors (4.6) l'est. En effet, l'on a toujours $dk^{m+1} > 2$ et (4.9) implique $d\varepsilon < \frac{\delta}{2}$. L'on a donc

$$\begin{aligned} 2 + d(k+\varepsilon) = 2 + dk + d\varepsilon &< 2 + dk + \frac{\delta}{2} \\ &< dk^{m+1} + dk + \frac{\delta}{2} \\ &< dk(k^{m+1}+1) + \delta = M + \delta, \end{aligned}$$

et comme $q \geq 2^{1+dk+\frac{\delta}{2}} > 2^{1+d(k+\varepsilon)}$, l'on a

$$(2q)^{1+d(k+\varepsilon)} < q^{2+1d(k+\varepsilon)} < q^{M+\delta}$$

et (4.6) est satisfaite par $\frac{p}{q}$.

Par hypothèse l'on a $2q \geq q_{n_1+1}^{\frac{1}{d}}$, et (4.8) nous donne l'existence d'un unique entier naturel $n_2 > n_1 \geq n_0$ tel que

$$q_{n_2-1} < (2q)^d < q_{n_2} \leq q_{n_2-1}^{k+\varepsilon}. \tag{4.12}$$

L'on a l'inégalité triangulaire suivante

$$\left|\xi - \frac{p_{n_2}}{q_{n_2}}\right| \geq \left|\left|\xi - \frac{p}{q}\right| - \left|\frac{p_{n_2}}{q_{n_2}} - \frac{p}{q}\right|\right|. \tag{4.13}$$

Remarquons que si l'on a $\frac{p_{n_2}}{q_{n_2}} \neq \frac{p}{q}$, alors

$$\left|\frac{p}{q} - \frac{p_{n_2}}{q_{n_2}}\right| = \left|\frac{pq_{n_2} - qp_{n_2}}{qq_{n_2}}\right| \geq \frac{1}{qq_{n_2}}.$$

De plus, comme l'on a $q_{n_2} \leq q_{n_2-1}^{k+\varepsilon} \leq (2q)^{d(k+\varepsilon)}$, l'on déduit par (4.11)

$$\left|\frac{p}{q} - \frac{p_{n_2}}{q_{n_2}}\right| \geq \frac{1}{qq_{n_2}} \geq \frac{1}{q(2q)^{d(k+\varepsilon)}} = \frac{2}{(2q)^{1+d(k+\varepsilon)}} > 2\left|\xi - \frac{p}{q}\right|.$$

L'on a alors que, si $\frac{p_{n_2}}{q_{n_2}} \neq \frac{p}{q}$,

$$\left|\xi - \frac{p_{n_2}}{q_{n_2}}\right| > \left|\frac{p}{q} - \frac{p_{n_2}}{q_{n_2}}\right| - \left|\xi - \frac{p}{q}\right| > \frac{1}{qq_{n_2}} - \left|\xi - \frac{p}{q}\right| > \frac{1}{2qq_{n_2}}.$$

Comme l'on a $2q < q_{n_2}^{\frac{1}{d}}$, il en résulte

$$\left|\xi - \frac{p_{n_2}}{q_{n_2}}\right| > \frac{1}{2qq_{n_2}} > \frac{1}{q_{n_2}^{\frac{1}{d}} q_{n_2}} = \frac{1}{q_{n_2}^{1+\frac{1}{d}}},$$

ce qui est une contradiction avec (4.7). En supposant (4.11) l'on a donc forcément $\frac{p}{q} = \frac{p_{n_2}}{q_{n_2}}$. Dans ce cas, comme l'on a $n_2 \geq n_1$, l'on tire de (4.10)

$$\left|\xi - \frac{p}{q}\right| = \left|\xi - \frac{p_{n_2}}{q_{n_2}}\right| \geq \frac{1}{q_{n_2}^{k^m+1+\varepsilon}},$$

et comme $(2q)^{d(k+\varepsilon)} \geq q_{n_2}$, l'on a

$$\left|\xi - \frac{p}{q}\right| \geq \frac{1}{(2q)^{d(k+\varepsilon)(k^m+1+\varepsilon)}}.$$

Finalement, de l'hypothèse (4.9) sur ε et comme l'on a également supposé $q \geq 2^{1+\frac{2M}{\delta}}$, l'on déduit

$$\begin{aligned}(2q)^{d(k+\varepsilon)(k^m+1+\varepsilon)} &= (2q)^{dk^{m+1}+dk+dk\varepsilon+d\varepsilon k^m+d\varepsilon+d\varepsilon^2} \\ &\leq (2q)^{dk^{m+1}+dk+\frac{\delta}{2}} = (2q)^{dk(k^m+1)+\frac{\delta}{2}} = (2q)^{M+\frac{\delta}{2}} \\ &\leq q^{M+\frac{\delta}{2}} 2^{M+\frac{\delta}{2}} \\ &\leq q^{M+\delta}.\end{aligned}$$

Ceci signifie que l'on a donc

$$\left|\xi - \frac{p}{q}\right| \geq \frac{1}{q^{M+\delta}}$$

pour tout $q \geq \max\left\{\frac{q_{n_1+1}^{1/d}}{2}, 2^{1+\frac{2M}{\delta}}, 2^{1+dk+\frac{\delta}{2}}\right\}$, et le théorème est démontré. □

Les nombres automatiques, ayant une mesure d'irrationalité finie ne sont donc pas de Liouville. Remarquons par ailleurs que la borne de la mesure d'irrationalité d'un nombre ne dépend pas de la base b de sa représentation.

Corollaire 4.39. *Un nombre de Liouville ne peut pas être produit par un automate fini déterministe.*

La prochaine section, qui reprend les travaux d'Adamczewski et Bugeaud [2], et qui se place dans un cadre plus général que les nombres automatiques, va nous donner d'autres bornes pour les exposants diophantiens et mesures d'irrationalité d'une certaine classe de nombres. Nous pourrons en déduire certaines propriétés diophantiennes des nombres automatiques.

4.5 Nombres de complexité sous-linéaire

L'on peut en réalité reformuler le théorème 4.24 dans un cadre plus général :

Théorème 4.40 (Adamczewski-Bugeaud)**.** *Tout nombre réel de complexité sous-linéaire est ou bien rationnel, ou bien transcendant.*

Les nombres automatiques sont en fait un cas particulier de nombres de complexité sous-linéaire. Ainsi, l'étude de ces derniers s'inscrit dans une perspective plus large que la question de Cobham. Nous noterons

$$\mathcal{CL} := \{\xi \in \mathbb{R} \mid p_{\langle \xi \rangle_b}(n) = O(n), \text{pour un entier } b \geq 2\}$$

la classe des nombres de complexité sous-linéaire. Remarquons qu'en fait, étudier le comportement de la fonction de complexité en facteurs des b-représentations des nombres réels permet de définir d'autres classes de complexité que les classes S_T définies au chapitre 3 (voir définition 3.20), et donne donc lieu à une autre classification, a priori très différente de celle d'Hartmanis et Stearns. Par exemple, la classe $\mathcal{CL}$ contient une infinité non-dénombrable de nombres (notamment tous les nombres de Liouville de la forme $\sum_{n\geq 1} a_n b^{-n!}$, avec $(a_n)_{n\geq 1} \in \Sigma_b^{\mathbb{N}}$), et par conséquent contient des nombres non-calculables.

L'on peut alors naturellement encore une fois se demander quelles sont les propriétés diophantiennes des nombres de complexité sous-linéaire. Tout d'abord, l'on peut facilement distinguer les transcendants des rationnels, ces derniers ayant une b-représentation ultimement périodique. Il s'avère cependant que la classe $\mathcal{CL}$ contient des nombres irrationnels qui ont un comportement très variable par rapport à l'approximation rationnelle. Bugeaud [17] a montré que pour tout réel τ, les nombres de la forme

$$\xi = 2 \cdot \sum_{n \geq n_0(\tau,\lambda)} 3^{-\lfloor \lambda \tau^n \rfloor},$$

avec $\lambda > 0$, vérifient $\mu(\xi) = \tau$. Pour tout réel τ, l'on peut donc trouver une infinité de nombres de $\mathcal{CL}$ qui ont τ pour mesure d'irrationalité. Il semble donc compliqué de trouver une mesure d'irrationalité générale pour les nombres de complexité sous-linéaire.

Les travaux d'Adamczewski, Cassaigne et Bugeaud [2, 5] permettent cependant d'encadrer la mesure d'irrationalité d'un élément de $\mathcal{CL}$ via l'exposant diophantien de sa b-représentation. Rappelons que pour tout irrationnel ξ, l'on a $\mathrm{Dio}(\langle \xi \rangle_b) \leq \mu(\xi)$ (voir proposition 4.27). Quand ξ est de complexité sous-linéaire, l'on peut également majorer sa mesure d'irrationalité via son exposant diophantien.

Théorème 4.41. *Soit un réel $\xi \in \mathbb{R}$ et soit un entier $\kappa > 1$ tel que $p : \mathbb{N} \to \mathbb{N}$ la fonction de complexité en facteurs de $\langle \xi \rangle_b$ vérifie $p(n) \leq \kappa n$ pour tout entier n assez grand. Si l'exposant diophantien de $\langle \xi \rangle_b$ est fini, alors le nombre ξ vérifie*

$$\max\{2, \mathrm{Dio}(\langle \xi \rangle_b)\} \leq \mu(\xi) \leq (2\kappa + 1)^3 (\mathrm{Dio}(\langle \xi \rangle_b) + 1).$$

Ce théorème étend donc le théorème 4.38 aux nombres de complexité sous-linéaire, et nous donne un critère combinatoire pour distinguer les nombres de Liouville parmi ceux-ci.

Corollaire 4.42. *Soit ξ un nombre irrationnel de complexité sous-linéaire en base b. Alors ξ est un nombre de Liouville si, et seulement si,* $\mathrm{Dio}(\langle\xi\rangle_b)$ *est infini.*

Ainsi, assez surprenamment, la lecture de la b-représentation d'un nombre réel de complexité sous-linéaire nous permet toujours de déterminer s'il s'agit ou non d'un nombre de Liouville. L'on peut de façon plus générale se demander dans quelle classe de la classification de Mahler tombent ces nombres.

Théorème 4.43. *Soit un irrationnel $\xi \in \mathcal{CL}$. Alors, ou bien ξ est un nombre de Liouville, ou bien il existe une constante c indépendante de d et de b, telle que*

$$w_d(\xi) \leq (2d)^{c(\log 3d)(\log\log 3d)},$$

pour tout entier $d \geq 1$. Dans ce cas, ξ est un S-nombre ou un T-nombre.

La démonstration de ce théorème repose sur une version quantitative du théorème du sous-espace de Schmidt obtenue par Evertse et Schlickewei [2, Théorème ES]. De plus, sa démonstration ainsi que celle du théorème 4.41 utilisent également une construction d'une suite $\left(\frac{p_n}{q_n}\right)_{n\geq 1}$ avec un dénominateur de la forme $q_n = b^{r_n}(b^{s_n}-1)$ (et qui vérifie des propriétés supplémentaires), d'une manière similaire aux preuves des théorèmes 4.24 et 4.38.

Dans le cas des irrationnels de complexité sous-linéaire, le théorème précédent demande donc de distinguer parmi ceux-ci les nombres de Liouville de ceux qui n'en sont pas. Nous avons déjà fait cette distinction pour le cas des nombres automatiques, ces derniers n'étant jamais de Liouville. L'on en déduit donc le résultat suivant.

Théorème 4.44. *Tout nombre irrationnel automatique est ou bien un S-nombre, ou bien un T-nombre.*

Aucun irrationnel automatique ne peut donc être un U-nombre, et en particulier ne peut être de Liouville. Ce résultat va ainsi dans le sens de la conjecture de Becker.

Suivant l'idée de Cobham, un résultat plus général que le théorème 4.25 a été obtenu dans les travaux d'Adamczewski, Cassaigne et Le Gonidec [6], au sens où la b-représentation des irrationnels algébriques ne peut pas être produite par un modèle de calcul plus général que les k-AFDS, *les automates à pile*. C'est le sujet du prochain chapitre.

Chapitre 5

Vers des modèles de calcul plus puissants

Dans le chapitre précédent, nous avons étudié les automates finis déterministes, qui constituent une classe très restreinte de machines de calcul. Nous avons en particulier vu qu'ils ne peuvent pas produire la b-représentation des irrationnels algébriques, ce qui constitue le premier résultat central de ce mémoire. Ce chapitre est d'abord consacré à la présentation des automates à pile, qui forment un modèle de calcul plus puissant que les automates finis. Après avoir introduit ce modèle, nous montrerons le deuxième résultat central de ce mémoire : il ne permet pas non plus de produire la b-représentation des irrationnels algébriques. Nous étendrons ensuite cette analyse à des modèles plus généraux, notamment aux machines à plusieurs piles ainsi qu'à un modèle particulier de machines de Turing : les k-machines unidirectionnelles. Les définitions et résultats présentés dans ce chapitre sont repris des travaux d'Adamczewski, Cassaigne et Le Gonidec [6].

5.1 Automates à pile

Décrit de façon informelle, un automate à pile n'est en fait autre qu'un automate fini muni d'une mémoire organisée sous forme de pile. De manière similaire à une machine de Turing, chaque transition se fait selon l'état courant de l'automate, et selon le symbole lu en mémoire sur la pile. Cependant, là où une machine de Turing peut se déplacer "sans contrainte" sur son ruban, un automate à pile n'a accès qu'au symbole le plus en haut de sa pile, et doit nécessairement l'effacer pour pouvoir modifier les autres symboles en dessous de celui-ci. On parle de structure LIFO : Last In First Out (dernier entré, premier sorti). Ce modèle de calcul se trouve entre les automates finis et les machines de Turing dans la hiérarchie de Chomsky. Là où les automates finis (sans pile) sont utilisés pour accepter les langages réguliers, les automates à pile sans sortie sont utilisés pour accepter une famille plus large, les langages hors-contexte [1].

1. Pour plus de détails sur les différents modèles de calcul de la hiérarchie de Chomsky, et sur les langages hors-contexte, la lectrice ou le lecteur curieux peut consulter [30], en particulier aux chapitres 5, 6 et 7.

Définition 5.1. Un *k-automate à pile fini déterministe avec sortie* (abrégé en k-APDS) est la donnée d'un septuple $\mathcal{M} = (Q, \Sigma_k, \Gamma, \delta, q_0, \Delta, \tau)$, où

- Q est l'ensemble fini d'états,
- Σ_k est l'alphabet $\{0, 1, \ldots, k-1\}$,
- Γ est un alphabet des *symboles de pile*. Par convention, l'on notera $\#$ le mot vide de l'ensemble Γ^* des mots de la pile. L'ensemble $Q \times \Gamma^*$ est appelé l'ensemble des *configurations internes de* $\mathcal{M}$.
- $\delta : E \subset Q \times (\Gamma \cup \{\#\}) \times (\Sigma_k \cup \{\varepsilon\}) \to Q \times \Gamma^*$ est la *fonction de transition*,
- $q_0 \in Q$ est *l'état initial* et $(q_0, \#)$ est appelée la *configuration initiale*.
- Δ est *l'alphabet de sortie* et
- $\tau : Q \times (\Gamma \cup \{\#\}) \to \Delta$ est la *fonction de sortie*.

De plus, la fonction de transition δ satisfait les conditions suivantes :

- *Déterminisme* : Si il existe $(q, s, \varepsilon) \in E$ pour un certain $(q, s) \in Q \times (\Gamma \cup \{\#\})$, alors il n'existe pas de $(q, s, a) \in E$ avec $a \in \Sigma_k$. On dit alors que $\mathcal{M}$ a une ε-transition depuis la configuration interne (q, s). De plus, définir δ comme une fonction et non une relation fait également partie de l'hypothèse de déterminisme.
- *Complétude* : Si $(q, s, \varepsilon) \notin E$ pour un certain $(q, s) \in Q \times (\Gamma \cup \{\#\})$, alors $\{q\} \times \{s\} \times \Sigma_k \subset E$. Autrement dit, depuis la configuration (q, s), ou bien $\mathcal{M}$ possède une ε-transition depuis cette configuration, ou bien il possède une transition définie pour chaque lettre de Σ_k.
- *Finitude sur les ε-transitions* : Il n'existe pas de suite infinie de configurations internes $(q_i, S_i)_{i \geq 1}$ telle que $\delta(q_i, s_i, \epsilon) = (q_{i+1}, X_i)$ pour tout $i \geq 1$, où s_i est le premier symbole de S_i, ou $\#$ si S_i est vide, $S_i = S'_i s_i$, et $S_{i+1} = S'_i X_i$. C'est en particulier satisfait lorsque toutes les ε-transitions réduisent la taille de la pile.

De la même manière que dans le cas des k-AFDS, l'on peut étendre la fonction de transition δ pour pour donner du sens au calcul $\tau(\delta(q_0, \#, w))$ pour tout mot $w \in \Sigma_k^*$. L'on peut commencer par l'étendre à un sous-ensemble de $Q \times \Gamma^* \times (\Sigma_k \cup \{\varepsilon\})$ en posant pour tout $S \in \Gamma^*$,

$$\delta(q, Ss, a) = (q', SX)$$

avec $\delta(q, s, a) = (q', X)$ et $s \neq \#$. De plus, après avoir lu un symbole a, l'automate à pile pourrait atteindre une configuration (q, S) à partir de laquelle il pourrait effectuer des ε-transitions. Dans ce cas l'hypothèse de déterminisme force l'automate à effectuer toutes les ε-transitions possibles avant de lire un nouveau symbole. L'hypothèse de finitude nous assure qu'il n'y a qu'un nombre fini de telles transitions. Pour respecter ces hypothèses, l'on peut étendre la fonction de transition δ à $\bar{\delta}$ en posant :

$$\begin{cases} \bar{\delta}(q, S, a) = \bar{\delta}(\delta(q, S, a), \varepsilon) & \\ \bar{\delta}(q, S, \varepsilon) = (q, S) & \text{si } (q, S, \varepsilon) \notin E \\ \bar{\delta}(q, S, \varepsilon) = \bar{\delta}(\delta(q, S, \varepsilon), \varepsilon) & \text{si } (q, S, \varepsilon) \in E. \end{cases}$$

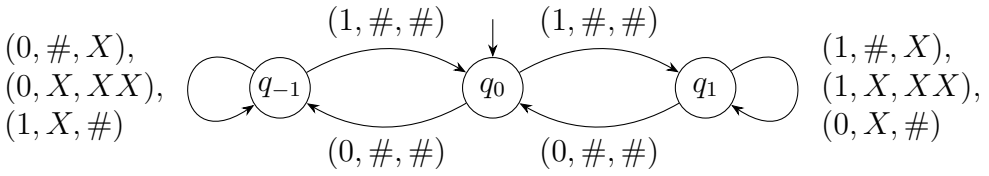


FIGURE 5.1 – 2-automate à pile $\mathcal{A}$ qui produit la suite $\boldsymbol{a}$.

Finalement, l'on peut étendre $\bar{\delta}$ à $Q \times \Gamma^* \times \Sigma_k^*$ en posant

$$\bar{\delta}(q, S, wa) = \bar{\delta}(\bar{\delta}(q, S, w), a) \text{ pour tous } w \in \Sigma_k^* \text{ et } a \in \Sigma_k.$$

En particulier, ceci signifique que $\mathcal{M}$ lit les mots de gauche à droite. L'on étend aussi la fonction de sortie τ à l'ensemble $Q \times \Gamma^*$ en posant $\tau(q, Ss) = \tau(q, s)$ pour $q \in Q, s \in \Gamma$ et $S \in \Gamma^*$.

Munis d'une sortie, les automates à piles produisent une famille plus large de suites que les suites automatiques, les *suites hors-contexte*. Le lecteur ou la lectrice souhaitant plus de détails sur les suites hors-contexte peut consulter [19].

Définition 5.2. Soit $\mathcal{M} = (Q, \Sigma_k, \Gamma, \delta, q_0, \Delta, \tau)$ un k-automate à pile. La suite $(\tau(\bar{\delta}(q_0, \#, \langle n \rangle_k)))_{n \geq 0}$ est *la suite produite par* $\mathcal{M}$.

L'on peut représenter un automate à pile par un graphe orienté dont les sommets sont les états de Q et les arêtes sont données par la fonction de transition δ et sont étiquetées de la manière suivante. Pour tous états $q, q' \in Q$ tels que $\delta(q, s, a) = (q', X)$, l'on dessine une flèche du sommet q vers le sommet q' étiquetée par (a, s, X). L'état initial q_0 est désigné par une flèche entrante.

Exemple 5.3. La suite binaire

$$\boldsymbol{a} := 11101110011010000111111011101000000010110\cdots$$

dont le n-ème terme vaut 1 si la différence entre le nombre de 0 et de 1 est d'au plus 1 dans la 2-représentation de n, et 0 sinon, est produite par le k-APDS $\mathcal{A}$, représenté à la figure 5.1 et défini de la manière suivante :

$$\mathcal{A} = (\{q_0, q_1, q_{-1}\}, \Sigma_2, \{X\}, \delta, q_0, \Sigma_2, \tau),$$

avec δ définie par

$$\begin{array}{lll}
\delta(q_0, \#, 0) = (q_{-1}, \#), & \delta(q_1, \#, 0) = (q_0, \#), & \delta(q_{-1}, \#, 0) = (q_{-1}, X), \\
\delta(q_0, \#, 1) = (q_1, \#), & \delta(q_1, \#, 1) = (q_1, X), & \delta(q_{-1}, \#, 1) = (q_0, \#), \\
\delta(q_0, X, 0) = (q_0, \#), & \delta(q_1, X, 0) = (q_1, \#), & \delta(q_{-1}, X, 0) = (q_{-1}, XX), \\
\delta(q_0, X, 1) = (q_0, \#), & \delta(q_1, X, 1) = (q_1, XX), & \delta(q_{-1}, X, 1) = (q_{-1}, \#),
\end{array}$$

et la fonction de sortie τ définie par

$$\begin{aligned}
\tau(q_0, \#) = \tau(q_1, \#) = \tau(q_{-1}, \#) = 1 \\
\tau(q_0, X) = \tau(q_1, X) = \tau(q_{-1}, X) = 0.
\end{aligned}$$

Cet automate fonctionne de la manière suivante. Il se trouve dans l'état q_0 quand il a lu le même nombre de 1 et de 0, et dans l'état q_1 (resp. q_{-1}) quand il a lu plus de 1 que de 0 (resp. plus de 0 que de 1) dans la 2-représentation de l'entrée n. Une fois dans l'état q_1 (resp. q_{-1}), il empile un symbole X à chaque fois qu'il lit un symbole 1 (resp. 0) ou l'efface s'il a lu un 0 (resp. 1). S'il n'a plus de symbole à effacer, il repasse dans l'état q_0. Quand il a lu tous les symboles du mot d'entrée, avoir une pile vide signifie que la différence entre le nombre de 1 et de 0 dans la 2-représentation de n est égale à 1 ou moins (l'automate produit 1 via sa fonction de sortie τ), tandis qu'avoir un symbole X au-dessus de la pile signifie que la différence est d'au moins 2 (l'automate produit 0). Notons que les deux transitions $\delta(q_0, X, 0)$ et $\delta(q_0, X, 1)$ sont définies pour respecter l'hypothèse de complétude, bien qu'elles ne soient jamais utilisées dans le calcul. Elles ne sont pas représentées dans la figure 5.1.

L'automate lit les mots de gauche à droite, ce qui correspond à notre manière usuelle de lire les nombres. Dans le cas d'un k-AFDS, lire de gauche à droite ou de droite à gauche ne change rien, les deux étants équivalents. L'on ne peut cependant pas en dire autant pour les k-APDS, car l'ensemble des langages hors-contexte n'est pas fermé pour le miroir. De façon similaire aux machines de Turing, il existe souvent plusieurs automates à piles qui produisent une même suite. En particulier, il est possible d'en choisir un qui n'a qu'un seul état, en encodant les différents états dans la pile. L'automate donné en exemple ici n'est donc certainement pas le plus petit (en termes d'états) qui produise cette suite.

Lorsque l'alphabet de sortie du k-APDS est Σ_b, la suite produite est ainsi la b-représentation d'un nombre réel.

Définition 5.4. Un nombre réel $\xi \in [0, 1)$ est *produit par un k-automate à pile* s'il existe un k-APDS $\mathcal{M} = (Q, \Sigma_k, \Gamma, \delta, q_0, \Sigma_b, \tau)$ tel que $\langle \xi \rangle_b = (\tau(\delta(q_0, \#, \langle n \rangle_k)))_{n \geq 0}$.

Exemple 5.5. Le nombre $\xi_{\boldsymbol{a}}$ dont la 2-représentation est la suite $\boldsymbol{a}$ de l'exemple précédent est produit par le 2-automate à pile $\mathcal{A}$. Comme nous le verrons dans la section suivante, c'est un nombre transcendant.

5.2 Problème de Hartmanis–Stearns pour les k-APDS

Nous avons montré au chapitre précédent que les b-représentations des irrationnels algébriques ont une complexité en facteur plus que linéaire, et que par conséquent les k-AFDS ne peuvent pas produire de tels nombres, car ils ne produisent que des suites de complexité sous-linéaire. Le cas des suites hors-contexte est bien différent : pour tout entier $d \geq 1$, il existe une suite produite par un k-APDS avec une complexité en facteurs en $O(n^d)$ [39]. Pour montrer que les automates à pile ne sont pas non plus un modèle de calcul suffisamment puissant pour produire les irrationnels algébriques, l'on ne peut donc pas se reposer sur des arguments similaires de complexité en facteurs, celle-ci étant très variable. Pour cela, l'on va introduire une relation d'équivalence sur les configurations internes d'un k-APDS, très similaire à la relation de Myhill-Nerode [30, section 4.4].

Notation. Une configuration interne sur $\mathcal{M}$ est un couple (q, S), avec q un état de Q et S le mot de Γ^* dans la pile. On note $C_{\mathcal{M}}(w)$ la configuration interne atteinte par $\mathcal{M}$ avec pour entrée le mot w, c'est à dire : $C_{\mathcal{M}}(w) = \bar{\delta}(q_0, \#, w)$. Quand cela n'apporte aucune confusion, l'on se permet d'omettre l'indice et l'on note $C(w)$. De plus, quand n est un entier, et l'alphabet d'entrée de $\mathcal{M}$ est Σ_k, l'on simplifie la notation $C(\langle n\rangle_k)$ en $C(n)$.

Définition 5.6. Soit $\mathcal{M}$ un k-automate à pile. On définit la relation d'équivalence $\sim$ sur les configurations internes de $\mathcal{M}$ comme suit : pour C_1 et C_2 des configurations internes de $\mathcal{M}$, on a $C_1 \sim C_2$ si et seulement si, pour tout $w \in \Sigma_k^*, \tau(\bar{\delta}(C_1, w)) = \tau(\bar{\delta}(C_2, w))$. Autrement dit, on considère les configurations internes C_1 et C_2 équivalentes si les deux mènent toujours aux mêmes sorties, quel que soit le mot lu à partir de celles-ci.

Il est clair que la relation $\sim$ est une relation d'équivalence. Assez surprenamment, pour un nombre irrationnel ξ produit par un automate à pile $\mathcal{M}$, avoir deux configurations internes de $\mathcal{M}$ équivalentes sous cette relation suffit à établir la transcendance de ξ. En effet, dans ce cas, nous allons montrer que l'on a $\mathrm{Dio}(\langle\xi\rangle_b) > 1$.

Proposition 5.7. *Soit $\xi \in [0, 1)$ un nombre généré par un k-APDS $\mathcal{M}$. Si il existe deux entiers distincts n et n' tels que $C_{\mathcal{M}}(n) \sim C_{\mathcal{M}}(n')$, alors ξ est rationnel ou transcendant.*

Démonstration. Soit $\boldsymbol{a} = (a_n)_{n\geq 0}$ la suite produite par $\mathcal{M}$ telle que $\langle\xi\rangle_b = a_1a_2\ldots$. Supposons qu'il existe $n < n' \in \mathbb{N}$ tels que $C(n) \sim C(n')$. Posons $w_n = \langle n\rangle_k$ et $w_{n'} = \langle n'\rangle_k$. Alors,

$$a_{[w_n w]_k} = a_{[w_{n'} w]_k}$$

pour tout $w \in \Sigma_k^*$, par définition de la relation d'équivalence $\sim$. Remarquons que si $[w]_k = i$ et $|w| = m$, alors l'on a $[w_n w]_k = k^m n + i$ et $[w_{n'} w]_k = k^m n' + i$. Ceci signifie donc que pour tout $m \in \mathbb{N}$, on a en particulier :

$$\forall i \in \{0, 1, 2, \ldots, k^m - 1\}, a_{k^m n+i} = a_{k^m n'+i}.$$

Posons $U_m = a_1a_2\ldots a_{k^m n-1}$ et $V_m = a_{k^m n}a_{k^m n+1}\ldots a_{k^m n'-1}$. Alors, le mot

$$U_m V_m^{1+\frac{1}{n'-n}} = a_1a_2\ldots a_{k^m n-1}a_{k^m n}a_{k^m n+1}\ldots a_{k^m n'-1}a_{k^m n}\ldots a_{k^m n+k^m-1}$$

est un préfixe de $\boldsymbol{a}$. On a $|U_m| = k^m n - 1$ et $|V_m| = k^m(n' - n)$, et l'on en déduit

$$\frac{|U_m V_m^{1+\frac{1}{n'-n}}|}{|U_m V_m|} \geq \frac{k^m n - 1 + k^m(n' - n)\cdot(1 + \frac{1}{n'-n})}{k^m n - 1 + k^m(n' - n)} = 1 + \frac{1}{n' - \frac{1}{k^m}} \geq 1 + \frac{1}{n'} > 1.$$

De plus, comme la suite $(|V_m|)_{m\geq 0}$ est strictement croissante, $(|V_m^{1+\frac{1}{n'-n}}|)_{m\geq 0}$ l'est également. L'on en déduit alors que $\boldsymbol{a}$ satisfait la condition $(*)_\rho$ pour $\rho = 1 + \frac{1}{n'}$, et donc

$$\mathrm{Dio}(\boldsymbol{a}) \geq 1 + \frac{1}{n'} > 1,$$

ce qui implique que ξ est rationnel ou transcendant, par le théorème 4.31. □

Remarquons que cette proposition nous donne une démonstration encore plus simple du théorème 4.25. En effet, un automate fini n'est jamais qu'un automate à pile qui n'utilise pas sa mémoire. Une configuration interne d'un tel automate est donc toujours de la forme $(q, \#)$ et dépend uniquement de l'état de l'automate. Comme un k-AFDS n'a qu'un nombre fini d'états, il ne possède qu'un nombre fini de configurations internes. Par le principe des tiroirs, il existe donc deux entiers positifs n et n' tels que $C(n) = C(n')$, et l'on conclut par la proposition précédente. Nous pouvons finalement montrer le deuxième résultat central de ce mémoire : les automates à piles ne peuvent pas produire la b-représentation d'un irrationnel algébrique.

Théorème 5.8 (Adamczewski-Cassaigne-Le Gonidec [6])**.** *La b-représentation d'un nombre algébrique irrationnel ne peut pas être produite par un k-automate à pile déterministe.*

Démonstration. Soit $\xi \in [0, 1)$ dont la b-représentation est produite par le k-automate à pile $\mathcal{M} = (Q, \Sigma_k, \Gamma, \delta, q_0, \Sigma_b, \tau)$. Notons

- $q_w \in Q$ l'état atteint par $\mathcal{M}$ en lisant un mot $w \in \Sigma_k^*$ et en partant de $(q_0, \#)$,
- $S(w) \in \Gamma^*$ le contenu de la pile de $\mathcal{M}$ en lisant le mot w et en partant de $(q_0, \#)$ et
- $H(w)$ la hauteur de la pile correspondante, c'est à dire la longueur du mot $S(w)$.

De plus, notons les ensembles

- $\mathcal{C}_k = (\Sigma_k \backslash \{0\})\Sigma_k^*$ le langage des k-représentations canoniques des entiers naturels et
- $\mathcal{H}_m = \{w \in \mathcal{C}_k : H(w) \leq m\}$, pour tout entier positif m, l'ensemble des mots $w \in \mathcal{C}_k$ pour lesquels la pile de $\mathcal{M}$ est au plus de taille m durant le calcul, avec w en entrée.

Remarquons que pour tout $m \geq 1, \mathcal{H}_m \subset \mathcal{H}_{m+1}$, et $\mathcal{C}_k = \bigcup_{m=1}^{\infty} \mathcal{H}_m$.

Supposons d'abord qu'il existe $m \in \mathbb{N}$ tel que $\mathcal{H}_m$ est infini. Or, pour tout $w \in \mathcal{H}_m$, la configuration interne $C(w) = (q_w, S(w))$ appartient à l'ensemble fini $Q \times \Gamma^{\leq m}$. Par le principe des tiroirs, il existe donc deux mots distincts w et w' de $\mathcal{H}_m$ tels que $C(w) = C(w')$. En posant alors $n = [w]_k$ et $n' = [w']_k$, on a $C(n) = C(n')$, et en particulier $C(n) \sim C(n')$. On conclut par la proposition 5.7.

Supposons maintenant que tous les $\mathcal{H}_m$ sont finis, et pour tout $m \geq 1$, prenons un mot $v_m \in \mathcal{H}_m$ de longueur maximale. On alors que $|v_m| \leq |v_{m+1}|$ pour tout m, et l'ensemble $\{v_m : m \geq 1\}$ est infini. En effet, s'il était fini, l'on aurait un $m \in \mathbb{N}$ tel que $\mathcal{H}_m$ est infini, ce qui correspond à l'hypothèse précédente. Comme v_m est de longueur maximale, l'on a

$$\forall w \in \Sigma_k^* \setminus \{\varepsilon\},\ H(v_m) < H(v_m w). \tag{5.1}$$

Autrement dit, dans le calcul $(q_{v_m}, S(v_m)) \overset{w}{\vdash} (q_{v_m w}, S(v_m w))$, la taille de la pile augmente nécessairement. De plus, pour m assez grand, on a $|v_m| > |v_1|$, et donc $H(v_m) > 1$. Pour un tel m, décomposons le mot $S(v_m)$ en

$$S(v_m) = X_m z_m$$

où $z_m \in \Gamma$ est le symbole au dessus de la pile. Alors pour tout $w \in \Sigma_k^* \setminus \{0\}$, le mot X_m est est un préfixe du mot $S(v_m w)$, par l'inégalité précédente (5.1). Autrement dit, la partie de la pile correspondant au mot X_m n'est pas modifiée, ni même lue, durant le calcul $(q_{v_m}, S(v_m)) \overset{w}{\vdash} (q_{v_m w}, S(v_m w))$, ce qui signifie

$$(q_{v_m}, S(v_m)) \sim (q_{v_m}, z_m)$$

On a $(q_{v_m}, z_m) \in Q \times \Gamma$, qui est un ensemble fini, alors que $\{v_m : m \geq 1\}$ est infini. Ainsi, par le principe des tiroirs, il existe m et m' tels que $v_m \neq v_{m'}$ et $C(v_m) \sim C(v_{m'})$. En posant $n = [v_m]_k$ et $n' = [v_{m'}]_k$, on a $n \neq n'$ et $C(n) \sim C(n')$. On conclut en utilisant la proposition 5.7. □

Comme annoncé, les automates à pile ne constituent pas un modèle de calcul suffisamment puissant pour produire la b-représentation des irrationnels algébriques. L'utilisation du modèle de la machine de Turing semble donc indispensable dans le cadre du problème d'Hartmanis et Stearns. Intéressons-nous alors à certaines variantes de machines de Turing. En fait, les théorèmes 4.25 et 5.8 fournissent une réponse partielle à ce problème pour une classe particulière de machines de Turing : les machines à piles.

5.3 Machines à piles

Définition 5.9. Une *machine à piles* $\mathcal{M}$ est similaire à une machine de Turing standard, à ceci près que la mémoire est constituée de piles au lieu de rubans. Plus précisément, elle possède

- une bande d'entrée, sur laquelle la tête ne peut que lire et ne peut se déplacer vers la gauche,
- une mémoire de travail organisée sous forme de piles, pour lesquelles chaque tête de lecture se trouve toujours au dessus de la pile,
- et une bande de sortie, sur laquelle la tête ne peut qu'écrire et ne peut rien effacer.

De manière similaire à un automate à pile, chaque transition de la machine est donnée par l'état courant, du symbole lu sur la bande d'entrée, et du symbole lu sur le dessus de chaque pile. Une machine à pile peut également faire une ε-transition, c'est à dire une transition pour laquelle la tête sur la bande de lecture ne se déplace pas. De plus, une telle machine ne peut terminer son calcul si une ε-transition est possible. Avec un mot d'entrée w, et après avoir terminé son calcul, la machine produit un symbole $a(w)$ sur sa bande de sortie. Pour un entier $k \geq 2$, l'on peut définir une sous-classe spécifique de machines à piles.

Définition 5.10. Soit un entier $k \geq 2$.

- Une *k-machine à piles* $\mathcal{M}$ est une machine à piles qui prend en entrée les mots de $\mathcal{C}_k$, l'ensemble des représentations canoniques des entiers.
- La *suite produite par la k-machine à piles* $\mathcal{M}$ est la suite $(a(\langle n \rangle_k))_{n \geq 0}$.

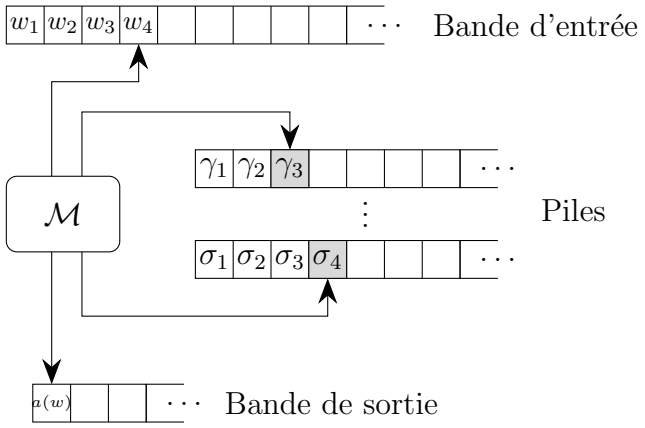


FIGURE 5.2 – Représentation schématique d'une machine à piles $\mathcal{M}$

Il est clair de par sa définition qu'une k-machine à une pile est en fait un k-automate à pile, et qu'une k-machine à zéro pile est un k-automate fini déterministe. Similairement aux modèles précédents, quand l'alphabet de sortie d'une k-machine à piles $\mathcal{M}$ est Σ_b, alors la suite $(a(\langle n \rangle_k))_{n\geq 0}$ produite par $\mathcal{M}$ est la b-représentation d'un nombre réel.

Définition 5.11. Un nombre réel $\xi \in [0, 1)$ est *produit par une k-machine à piles* s'il existe une k-machine à piles $\mathcal{M}$ telle que $\langle \xi \rangle_b = (a(\langle n \rangle_k)_{n\geq 0}$.

L'on a ainsi une description plus précise du problème d'Hartmanis et Stearns pour cette sous-classe de machines, en fonction du nombre de piles. En effet, étant équivalente à un automate à pile, une machine à une seule pile ne peut pas produire la b-représentation d'un irrationnel algébrique. De plus, dès que l'on s'autorise une machine à deux piles ou plus, l'on a autant de puissance de calcul qu'avec une machine de Turing standard, et l'on peut donc produire tout nombre calculable, et en particulier tout irrationnel algébrique, a priori en temps plus que linéaire.

Proposition 5.12. *Une machine à deux piles calcule les mêmes fonctions qu'une machine à une bande.*

Démonstration. Une machine à pile n'étant qu'une machine à bande dont la tête de lecture/écriture ne se déplace pas librement sur sa bande, montrons l'autre sens. Donnons nous une machine à une bande $\mathcal{M}$ et construisons une machine à deux piles $\mathcal{M}'$ qui fonctionne comme $\mathcal{M}$, avec deux piles P et P'. Les états de $\mathcal{M}'$ sont les mêmes que ceux de $\mathcal{M}$. La partie significative de la bande de $\mathcal{M}$ est stockée sur les piles de $\mathcal{M}'$ comme suit. La pile P stocke la même information que ce qui se trouve à gauche de la tête de lecture/écriture sur la bande de $\mathcal{M}$. Le symbole lu par la tête de $\mathcal{M}$ se trouve au dessus de la pile P', et les symboles dessous sont les mêmes que ceux qui se trouvent à droite de la tête de $\mathcal{M}$. La figure 5.3 illustre la manière dont $\mathcal{M}'$ simule le contenu de la bande de $\mathcal{M}$ sur ses deux piles. Si $\mathcal{M}$ modifie un symbole sur sa bande, $\mathcal{M}'$ modifie le symbole sur le dessus de la pile P'. Si $\mathcal{M}$ déplace sa tête à droite sur sa bande, $\mathcal{M}'$ dépile le premier symbole de P' et l'empile sur P, et inversement si $\mathcal{M}$ déplace sa tête à gauche. □

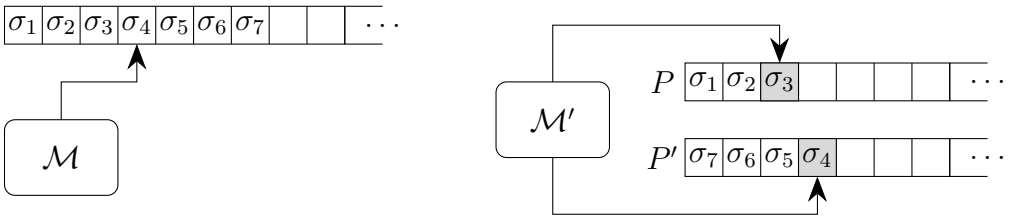


FIGURE 5.3 – La machine $\mathcal{M}'$ stocke autant d'information sur ses deux piles que $\mathcal{M}$ sur son ruban.

L'on peut alors reformuler le théorème 5.8 en termes de machines à piles, de la manière suivante.

Théorème 5.13 (Adamczewski-Cassaigne-Le Gonidec)**.** *La b-représentation d'un nombre irrationnel algébrique ne peut pas être produite par une machine à une pile (resp. à zéro pile).*

5.4 Machines unidirectionnelles

Dans leurs travaux [6], Adamczewski, Cassaigne et Le Gonidec proposent un changement de point de vue dans la question d'Hartmanis et Stearns. Plutôt que d'imposer une contrainte de complexité temporelle sur la machine, l'on considère des machines soumises à une contrainte d'organisation interne, les *k-machines unidirectionnelles*. Cette contrainte va nous permettre d'étendre naturellement la relation d'équivalence $\sim$ de la définition 5.6.

Définition 5.14. Soit $k \geq 2$. Une *k-machine unidirectionnelle* est la donnée d'un 6-uple $\mathcal{M} = (\Sigma_k, \mathcal{C}, \delta, C_0, \Delta, \tau)$ où

- Σ_k est l'alphabet $\{0, 1, \ldots, k-1\}$,
- $\mathcal{C}$ est un ensemble dénombrable, appelé ensemble des configurations,
- $\delta : \mathcal{C} \times \Sigma_k \to \mathcal{C}$ est la fonction de transition,
- C_0 est la configuration initiale,
- Δ est l'alphabet de sortie et
- $\tau : \mathcal{C} \to \Delta$ est la fonction de sortie.

De plus, l'on suppose que la machine est complète, c'est à dire que δ est définie sur l'entièreté de $\mathcal{C} \times \Sigma_k$. Cela nous permet d'étendre la définition de δ à tout Σ_k^* en posant, pour tout mot $w \in \Sigma_k^*$ et lettre $i \in \Sigma_k$, $\delta(C, wi) = \delta(\delta(C, w), i)$. L'on note $C_{\mathcal{M}}(n) := \delta(C_0, \langle n \rangle_k)$ la configuration atteinte par $\mathcal{M}$ avec le mot $\langle n \rangle_k$ en entrée. L'on se permettra d'alléger la notation en $C(n)$ quand cela n'apporte aucune confusion.

Les k-machines unidirectionnelles fonctionnent de manière similaire aux k-machines à piles. Elles possèdent une bande d'entrée sur laquelle la tête ne peut que lire et se déplacer sur la droite, et une bande de sortie sur laquelle la tête ne

peut qu'écrire et ne peut rien effacer ou modifier. Cependant, l'on ne précise pas la manière dont leur mémoire de travail est organisée : cela peut être sous forme de bandes, de piles, ou autre. Les seules contraintes imposées sont celles sur la bande d'entrée et sur la bande de sortie.

Chaque configuration $C \in \mathcal{C}$ d'une machine $\mathcal{M}$ est donc donnée par l'état courant et par le mot écrit sur chaque bande (ou pile, ou autre) de travail. Ce modèle de calcul est aussi puissant qu'une machine de Turing standard, et peut produire n'importe quelle suite calculable de $\Sigma_b^{\mathbb{N}}$, pour tout $b \geq 2$.

Définition 5.15. Soit $\mathcal{M} = (\Sigma_k, \mathcal{C}, \delta, C_0, \Delta, \tau)$ une k-machine unidirectionnelle. La *suite produite par* $\mathcal{M}$ est la suite $(\tau(C(n)))_{n\geq 0}$.

Pour une entrée $n \in \mathbb{N}$, la sortie $\tau(C(n))$ de la machine ne dépend que d'une quantité finie d'information de $C(n)$: l'état de l'automate pour un k-AFDS ou l'état de l'automate et le symbole au dessus de la pile dans le cas d'un k-APDS, ou encore l'état courant et le symbole sur le dessus de chaque pile pour une k-machine à pile. Il en va de même pour une machine dont la mémoire serait organisée sous forme de rubans. L'on peut alors étendre la relation d'équivalence $\sim$ de la définition 5.6 aux machines unidirectionnelles.

Définition 5.16. Soit $\mathcal{M} = (\Sigma_k, \mathcal{C}, \delta, C_0, \Delta, \tau)$ une k-machine unidirectionnelle. Pour deux configurations C_1 et C_2 de $\mathcal{M}$, l'on dit qu'elles sont équivalentes, et notons $C_1 \sim C_2$, si pour tout mot $w \in \Sigma_k$, $\tau(\delta(C_1, w)) = \tau(\delta(C_2, w))$.

Définition 5.17. Un nombre réel $\xi \in [0, 1)$ est *produit par une k-machine unidirectionnelle* $\mathcal{M} = (\Sigma_k, \mathcal{C}, \delta, C_0, \Sigma_b, \tau)$ si $\langle \xi \rangle_b = (\tau(C(n)))_{n\geq 0}$.

L'on peut alors énoncer la proposition 5.7 dans le cas plus général des k-machines unidirectionnelles.

Proposition 5.18. *Soit $\xi \in [0, 1)$ un nombre réel produit par une k-machine unidirectionnelle $\mathcal{M}$. S'il existe deux entiers $n \neq n'$ tels que $C(n) \sim C(n')$, alors ξ est rationnel ou transcendant.*

Cette proposition se démontre exactement de la même manière que dans le cas des automates à pile. Il suffit de remarquer que la démonstration de la proposition 5.7 ne fait pas intervenir directement le mot écrit sur la pile de l'automate. Le résultat tient tant que la machine considérée stocke et modifie une quantité finie de mémoire à chaque transition, peu importe comment sa mémoire de travail est organisée, ne modifie jamais le contenu de sa bande d'entrée et n'efface jamais le contenu de sa bande de sortie. Ce résultat permet donc de montrer la transcendance de nombres qui ne peuvent pas être produits par des automates à pile.

Exemple 5.19. La suite

$$\boldsymbol{a} = 0110120110010110100101100110100110010110011010010112\cdots$$

dont le n-ème terme est 2 si $\langle n \rangle_2$ est un mot de la forme $1^j0^j1^j$ avec $j \in \mathbb{N} \setminus \{0\}$, 1 si le nombre de 1 dans $\langle n \rangle_2$ est impair, et 0 sinon, ne peut pas être produite par un automate à pile. En effet, le langage $\{0^j1^j0^j, j \in \mathbb{N} \setminus \{0\}\}$ n'est pas un langage

hors-contexte, et ne peut être reconnu par un automate à pile. Cependant, pour tout $b \geq 3$, le nombre ξ tel que $\langle \xi \rangle_b = \boldsymbol{a}$ peut être produit par une 2-machine unidirectionnelle. Décrivons cette machine $\mathcal{M}$.

L'ensemble des états de la machine est $Q = \{i, q, r, s, t\} \times \{0, 1\}$, et possède deux piles avec un seul symbole de pile X. Par souci de simplicité, l'on notera les configurations des piles (X^j, X^k) par des couples $(j, k) \in \mathbb{N}^2$. L'ensemble des configurations de $\mathcal{M}$ est en fait $\mathcal{C} = Q \times \{(j, k) \in \mathbb{N}^2 : j \leq k\}$ (par construction de la machine, l'on aura toujours $j \leq k$). La configuration initiale est $C_0 = (i, 0, 0, 0)$. La fonction de transition $\delta : \mathcal{C} \times \Sigma_2 \to \mathcal{C}$ est définie de la manière suivante. L'on pose $\delta((i, 0, 0, 0), 1) = (q, 1, 1, 1)$. Pour tout $(e, j, k) \in \{0, 1\} \times \{(j, k) \in \mathbb{N}^2 : j \leq k\}$, l'on pose

$$\begin{aligned}
&\delta((q, e, j, j), 1) = (q, 1 - e, j + 1, j + 1), && \delta((r, e, 0, k), 0) = (t, e, 0, k),\\
&\delta((s, e, 0, 0), 1) = (t, 1 - e, 0, 0), && \delta((s, e, 0, k), 0) = (t, e, 0, k),\\
&\delta((t, e, j, k), 1) = (t, 1 - e, j, k), && \delta((t, e, j, k), 0) = (t, e, j, k),
\end{aligned}$$

et si j et k sont positifs, l'on pose

$$\begin{aligned}
&\delta((q, e, j, j), 0) = (r, e, j - 1, j), && \delta((r, e, j, k), 0) = (r, e, j - 1, k),\\
&\delta((r, e, j, k), 1) = (t, 1 - e, j, k), && \delta((r, e, 0, k), 1) = (s, 1 - e, 0, k - 1),\\
&\delta((s, e, 0, k), 1) = (s, 1 - e, 0, k - 1).
\end{aligned}$$

L'alphabet de sortie est $\Delta = \Sigma_3$. La fonction de sortie $\tau : \mathcal{C} \to \Delta$ est donnée par : $\tau(x, 1, j, k) = 1$ si $x \in \{i, q, r, s, t\}$ et $(j, k) \in \{(j, k) \in \mathbb{N}^2 : j \leq k\}$, $\tau(s, 0, 0, 0) = 2$ et $\tau(x, 0, j, k) = 0$ si $(x, j, k) \neq (s, 0, 0)$.

La machine $\mathcal{M}$ fonctionne de la manière suivante. Pour un état (x, e) avec $x \in \{i, q, r, s, t\}$ et $e \in \{0, 1\}$, la deuxième composante e représente la parité du nombre de 1 lus par la machine, tandis que la première composante x et les deux piles sont utilisées pour reconnaître les mots de la forme $0^j 1^j 0 j$. Plus précisément,

— dans un état (q, e) avec sur les piles (j, j), la machine a lu un préfixe de la forme 1^j,

— dans un état (r, e) avec sur les piles (j, k), la machine a lu un préfixe de la forme $1^j 0^{j-k}$,

— dans un état (s, e) avec sur les piles $(0, k)$, la machine a lu un préfixe de la forme $1^j 0^j 1^{j-k}$ et

— dans un état (t, e) la machine a lu un mot qui n'est pas un préfixe d'un mot de la forme $1^j 0^j 1^j$, pour tout $j \in \mathbb{N} \setminus \{0\}$.

Montrons maintenant que le nombre ξ produit par $\mathcal{M}$ est transcendant. Pour cela montrons que les deux configurations $C(10)$ et $C(20)$ de $\mathcal{M}$ sont équivalentes. L'on a $\langle 10 \rangle_2 = 1010$. Après avoir lu le mot 1010, $\mathcal{M}$ est dans la configuration $C(10) = (t, 0, 0, 1)$. En effet l'on a les transitions suivantes :

$$\begin{aligned}
\delta((i, 0, 0, 0), 1) &= (q, 1, 1, 1),\\
\delta((q, 1, 1, 1), 0) &= (r, 1, 0, 1),\\
\delta((r, 1, 0, 1), 1) &= (t, 0, 0, 1),\\
\delta((t, 0, 0, 1), 0) &= (t, 0, 0, 1).
\end{aligned}$$

De plus, l'on a $\langle 20 \rangle_2 = 10100$. L'on a donc

$$C(20) = \delta((t, 0, 0, 1), 0) = (t, 0, 0, 1) = C(10).$$

Ces deux configurations étant égales, elles sont en particulier équivalentes sous la relation $\sim$. Par la proposition 5.18, le nombre ξ est donc transcendant.

5.5 Introduction à d'autres modèles de calcul : ordinateurs analogiques et réseaux de réactions chimiques

Tout au long de ce mémoire, nous nous sommes intéressés à des modèles de calcul discrets, tels que les machines de Turing, dans lesquels l'information est représentée par des mots finis sur des alphabets et l'évolution du calcul par une suite de configurations discrètes. Nous terminons en ouvrant la discussion vers des modèles de calcul analogiques, dont l'état évolue de façon continue au cours du temps. L'exemple historique le plus emblématique est le *General Purpose Analog Computer*, régulièrement abrégé GPAC dans la littérature, et introduit par Shannon en 1941 [48]. Ce modèle de calcul est basé sur les ordinateurs analogiques utilisés dans les années 1930 pour résoudre des équations différentielles. À partir des années 1960, les ordinateurs analogiques ont progressivement laissé place aux ordinateurs numériques que l'on utilise aujourd'hui, et qui travaillent en binaire. La théorie de la calculabilité, fortement influencée par les travaux de Turing, s'est également d'avantage tournée vers les modèles de calculs discrets [15].

Nous nous intéresserons plus particulièrement aux *réseaux de réactions chimiques*, un modèle de calcul initialement utilisé pour décrire les interactions chimiques entre molécules, et qui s'avère relié au GPAC de Shannon, les deux étant décrits par des systèmes d'équations différentielles. En termes de puissance de calcul, les réseaux de réactions chimiques sont un modèle *Turing-complet* : ils sont capables de simuler les machines de Turing. Réciproquement, les réseaux de réactions chimiques déterministes peuvent eux-mêmes être simulés par machine de Turing [23]. Cette dernière section, qui se veut être une ouverture à ce mémoire et une mise en perspective de la conjecture de Hartmanis et Stearns, présente brièvement les travaux de Huang, Klinge, Lathrop, Li et Lutz [31, 32].

Définition 5.20. Un *réseau de réactions chimiques déterministe*, ou plus simplement réseau de réactions chimiques, est un couple $N = (S, R)$ où

- $S = \{Y_1, Y_2, \ldots, Y_n\}$ est un ensemble fini d'*espèces* (pouvant représenter des molécules, des atomes ou des ions) et
- R est un ensemble fini de *réactions* de la forme $\rho = (\boldsymbol{r}, \boldsymbol{p}, k) \in \mathbb{N}^n \times \mathbb{N}^n \times \mathbb{R}_{>0}$. Le vecteur $\boldsymbol{r}$ est appelé *vecteur des réactifs*, $\boldsymbol{p}$ est le *vecteur des produits* et k est la *constante de vitesse*. On notera $\boldsymbol{r}_i$ (resp. $\boldsymbol{p}_i$) la composante du vecteurs des réactifs (resp. du vecteurs des produits) associée à l'espèce Y_i.

Une espèce Y_i est *un réactif* si $\boldsymbol{r}_i > 0$, *un produit* si $\boldsymbol{p}_i > 0$ et *un catalyseur* si $\boldsymbol{r}_i = \boldsymbol{p}_i > 0$.

Exemple 5.21. Avec un ensemble d'espèces $S = \{X, Y, Z\}$, et une constante de vitesse k, considérons la réaction suivante entre les espèces de S,

$$X + Y \xrightarrow{k} X + 2Z,$$

que l'on représente par le triplet $\rho = ((1,1,0),(1,0,2),k)$. Dans cette réaction, les espèces X et Y sont les réactifs, X et Z sont les produits, et X est un catalyseur.

À un temps $t \in \mathbb{R}_{\geq 0}$, l'*état* d'un réseau de réactions chimiques $N = (S, R)$ est un vecteur $\boldsymbol{y}(t) = (y_1(t), y_2(t), \ldots, y_n(t)) \in (\mathbb{R}_{\geq 0})^n$, qui représente la concentration de chaque espèce chimique de S. Pour chaque espèce $Y_i \in S$, la composante $y_i(t)$, est appelée *la concentration de* Y_i au temps t.

Le *taux de réaction* d'une réaction $\rho = (\boldsymbol{r}, \boldsymbol{p}, k)$ au temps t est défini par

$$\text{rate}_\rho(t) = k \cdot \prod_{Y_i \in S} y_i(t)^{\boldsymbol{r}_i}.$$

Le *taux total de variation* d'une espèce $Y_i \in S$ dépend du taux de réaction de chaque réaction $\rho \in R$ du réseau dans laquelle Y_i intervient et vérifie l'équation différentielle

$$y_i'(t) = \sum_{\rho=(\boldsymbol{r},\boldsymbol{p},k)\in R} (\boldsymbol{p}_i - \boldsymbol{r}_i) \cdot \text{rate}_\rho(t). \tag{5.2}$$

Avec un état initial $\boldsymbol{y}_0 \in (\mathbb{R}_{\geq 0})^n$, le comportement du réseau de réactions chimiques $N = (S, R)$ est donc la solution du système d'équations différentielles

$$\begin{cases} y_1'(t) = \displaystyle\sum_{\rho=(\boldsymbol{r},\boldsymbol{p},k)\in R} (\boldsymbol{p}_1 - \boldsymbol{r}_1) \cdot \text{rate}_\rho(t). \\ \qquad\qquad \vdots \\ y_n'(t) = \displaystyle\sum_{\rho=(\boldsymbol{r},\boldsymbol{p},k)\in R} (\boldsymbol{p}_n - \boldsymbol{r}_n) \cdot \text{rate}_\rho(t). \end{cases}$$

avec pour conditions initiales

$$\boldsymbol{y}(0) = \boldsymbol{y}_0.$$

Ce système est un problème polynomial de Cauchy et le théorème de Cauchy-Lipschitz nous donne qu'il possède une unique solution $\boldsymbol{y} : \mathbb{R}_{\geq 0} \to (\mathbb{R}_{\geq 0})^n$.

Exemple 5.22. Considérons le réseau N constitué des trois espèces $S = \{X, Y, Z\}$ et de l'unique réaction présentée à l'exemple précédent. L'évolution de l'état du réseau au cours du temps est donnée par le système

$$\begin{cases} x'(t) = 0 \\ y'(t) = -kx(t)y(t) \\ z'(t) = 2kx(t)y(t), \end{cases}$$

où $x(t), y(t)$ et $z(t)$ sont les concentrations des espèces X, Y et Z respectivement au temps t.

L'on dit qu'un réseau N calcule un réel $\xi \in [0,1)$ si la concentration d'une des espèces de N tend vers ξ. Plus précisément, on introduit la notion de *calcul en temps réel*, qui quantifie la vitesse de convergence vers ξ.

Définition 5.23. Un nombre réel $\xi \in [0,1)$ est *calculable en temps réel par un réseau de réactions chimiques* s'il existe un réseau de réactions chimiques $N = (S, R)$ et une espèce $Y_j \in S$ qui vérifie les trois propriétés suivantes.

a) Pour toute réaction $(\boldsymbol{r}, \boldsymbol{p}, k) \in R$, $k \in \mathbb{N} \setminus \{0\}$.

b) Il existe une constante $\beta > 0$ telle que, si toutes les concentrations sont initialisées à 0, alors $y_i(t) < \beta$, pour toute espèce $Y_i \in S$ et tout temps $t > 0$.

c) Si toutes les concentrations sont initialisées à 0, alors pour tout $t \geq 1$,

$$|y_j(t) - \xi| \leq 2^{-t}.$$

L'hypothèse a) permet d'éviter d'encoder directement ξ dans les réactions et de s'assurer que le réseau soit encodé par un nombre fini d'informations. Sans cette hypothèse, tout nombre réel ξ peut être produit par un réseau avec une seule espèce X et avec les réactions

$$\emptyset \xrightarrow{\xi} X$$
$$X \xrightarrow{1} \emptyset.$$

Le taux total de variation de X vérifie alors l'équation différentielle

$$x' = \xi - x.$$

Avec $x(0) = 0$ pour condition initiale, la solution de cette équation est

$$x(t) = \xi(1 - e^{-t})$$

pour tout $t \geq 0$. L'hypothèse b) permet d'imposer une "limite de vitesse" au réseau, sans laquelle l'on pourrait calculer tout nombre arbitrairement vite. L'hypothèse de convergence c) impose au réseau N de calculer une approximation de ξ à $\lfloor t \rfloor$ décimales (binaires) à tout temps $t \geq 1$. Notons cependant que, contrairement à une machine de Turing, le réseau *ne produit pas explicitement les décimales de* ξ.

Rappelons brièvement les résultats connus dans le cadre des classes de complexité définies au chapitre 3 (voir définition 3.20). La classe de complexité S_n est l'ensemble des suites calculables en $O(n)$ opérations. Le théorème d'accélération 3.26 des machines de Turing nous permet d'affirmer que pour toute suite $\boldsymbol{a} \in S_n$ il existe une machine de Turing qui calcule les n premiers termes de $\boldsymbol{a}$ en exactement n transitions. Dans ce cas, l'on dit que $\boldsymbol{a}$ est *calculable en temps réel*. Rappelons alors que tout nombre rationnel est calculable en temps réel, et que certains nombres transcendants le sont également. L'on a des résultats similaires dans le cas des réseaux de réactions chimiques.

Théorème 5.24. *Tout nombre rationnel est calculable en temps réel par un réseau de réactions chimiques.*

Dans le cas des nombres transcendants, il est montré dans [31] que les nombres π et e sont calculables en temps réel par des réseaux de réactions chimiques.

Théorème 5.25. *Il existe des nombres transcendants calculables en temps réel par un réseau de réactions chimiques.*

Cependant, à ce jour, l'on ne connaît pas de nombre irrationnel algébrique qui soit calculable en temps réel par une machine de Turing, et la conjecture de Hartmanis-Stearns suggère qu'il n'en existe pas. Ces nombres sont cependant calculables en temps réel au sens des réseaux de réactions chimiques.

Théorème 5.26. *Tout irrationnel algébrique est calculable en temps réel par un réseau de réactions chimiques.*

La notion de calcul en temps réel d'un nombre dépend donc fortement du modèle de calcul considéré. Bien que les réseaux de réactions chimiques et les machines de Turing soient équivalents en terme de puissance de calcul, leurs classes de complexité temporelle ne semblent pas coïncider. Si un réseau de réactions chimiques N peut calculer un nombre ξ en temps réel, rien n'assure qu'une machine de Turing qui simule N calcule ξ en temps linéaire. Cette dépendance au modèle apparaît en fait déjà dans le cadre des machines de Turing : toute suite T-calculable par une machine de Turing à plusieurs bandes est T^2-calculable par une machine à une seule bande [29].

Bibliographie

[1] Boris Adamczewski et Yann Bugeaud. "Dynamics for β-shifts and Diophantine approximation". In : *Ergodic Theory Dynam. Systems* 27.6 (2007), p. 1695-1711.

[2] Boris Adamczewski et Yann Bugeaud. "Nombres réels de complexité sous-linéaire : mesures d'irrationalité et de transcendance". In : *J. Reine Angew. Math.* 658 (2011), p. 65-98.

[3] Boris Adamczewski et Yann Bugeaud. "On the complexity of algebraic numbers. I. Expansions in integer bases". In : *Ann. of Math. (2)* 165.2 (2007), p. 547-565.

[4] Boris Adamczewski, Yann Bugeaud et Florian Luca. "Sur la complexité des nombres algébriques". In : *C. R. Math. Acad. Sci. Paris* 339.1 (2004), p. 11-14.

[5] Boris Adamczewski et Julien Cassaigne. "Diophantine properties of real numbers generated by finite automata". In : *Compos. Math.* 142.6 (2006), p. 1351-1372.

[6] Boris Adamczewski, Julien Cassaigne et Marion Le Gonidec. "On the computational complexity of algebraic numbers : the Hartmanis-Stearns problem revisited". In : *Trans. Amer. Math. Soc.* 373.5 (2020), p. 3085-3115.

[7] Boris Adamczewski et Colin Faverjon. "Mahler's method in several variables and finite automata". In : *arXiv preprint arXiv :2012.08283* (2020).

[8] Saban Alaca et Kenneth S. Williams. *Introductory Algebraic Number Theory*. Cambridge University Press, 2003.

[9] Jean-Paul Allouche et Jeffrey Shallit. *Automatic sequences : theory, applications, generalizations*. Cambridge university press, 2003.

[10] Jean-Paul Allouche et Luca Q. Zamboni. "Algebraic irrational binary numbers cannot be fixed points of non-trivial constant length or primitive morphisms". In : *J. Number Theory* 69.1 (1998), p. 119-124.

[11] David Bailey, Peter Borwein et Simon Plouffe. "On the rapid computation of various polylogarithmic constants". In : *Math. Comp.* 66.218 (1997), p. 903-913.

[12] Alan Baker. *Transcendental number theory*. Cambridge university press, 2022.

[13] Émile Borel. “Les probabilités dénombrables et leurs applications arithmétiques”. In : *Rendiconti del Circolo Matematico di Palermo (1884-1940)* 27.1 (1909), p. 247-271.

[14] Émile Borel. “Sur les chiffres décimaux de $\sqrt{2}$ et divers problèmes de probabilités en chaîne”. In : *C. R. Acad. Sci. Paris* 230 (1950), p. 591-593.

[15] Olivier Bournez et Manuel L. Campagnolo. “A survey on continuous time computations”. In : *New computational paradigms*. Springer, New York, 2008, p. 383-423.

[16] Yann Bugeaud. *Approximation by Algebraic Numbers*. Cambridge Tracts in Mathematics. Cambridge University Press, 2004.

[17] Yann Bugeaud. “Diophantine approximation and Cantor sets”. In : *Math. Ann.* 341.3 (2008), p. 677-684.

[18] Georg Cantor. “Über eine Eigenschaft des Inbegriffes aller reellen algebraischen Zahlen”. In : *Über unendliche, lineare Punktmannigfaltigkeiten : Arbeiten zur Mengenlehre aus den Jahren 1872–1884*. Traduction anglaise disponible à l’adresse : https ://jamesrmeyer.com/infinite/cantor-1874-uncountability-proof. Springer, 1984, p. 19-24.

[19] Didier Caucal et Marion Le Gonidec. “Context-free sequences”. In : *Theoretical aspects of computing—ICTAC 2014*. T. 8687. Lecture Notes in Comput. Sci. Springer, Cham, 2014, p. 259-276.

[20] David G Champernowne. “The construction of decimals normal in the scale of ten”. In : *Journal of the London Mathematical Society* 1.4 (1933), p. 254-260.

[21] Emilie Charlier. “Algorithmique et calculabilité”. Notes de cours. Université de Liège, sept. 2023.

[22] Alan Cobham. “On the Hartmanis-Stearns problem for a class of tag machines”. In : *9th Annual Symposium on Switching and Automata Theory (swat 1968)*. IEEE, 1968, p. 51-60.

[23] François Fages et al. “Strong Turing completeness of continuous chemical reaction networks and compilation of mixed analog-digital programs”. In : *Computational methods in systems biology*. T. 10545. Lecture Notes in Comput. Sci. Springer, Cham, 2017, p. 108-127.

[24] Patrick C Fischer, Albert R Meyer et Arnold L Rosenberg. “Counter machines and counter languages”. In : *Mathematical systems theory* 2.3 (1968), p. 265-283.

[25] Patrick C Fischer, Albert R Meyer et Arnold L Rosenberg. “Real time counter machines”. In : *8th annual symposium on switching and automata theory (SWAT 1967)*. IEEE, 1967, p. 148-154.

[26] Patrick C. Fischer, Albert R. Meyer et Arnold L. Rosenberg. “Time-restricted sequence generation”. In : *Journal of Computer and System Sciences* 4.1 (1970), p. 50-73.

[27] Christiane Frougny et Jacques Sakarovitch. “Number representation and finite automata”. In : *Combinatorics, automata and number theory*. T. 135. Encyclopedia Math. Appl. Cambridge Univ. Press, Cambridge, 2010, p. 34-107.

[28] Godfrey Harold Hardy et Edward Maitland Wright. *An introduction to the theory of numbers*. Oxford university press, 1979.

[29] Juris Hartmanis et Richard E Stearns. “On the computational complexity of algorithms”. In : *Transactions of the American Mathematical Society* 117 (1965), p. 285-306.

[30] John E Hopcroft, Rajeev Motwani et Jeffrey D Ullman. “Introduction to automata theory, languages, and computation”. In : *Acm Sigact News* 32.1 (2001), p. 60-65.

[31] Xiang Huang, Titus H. Klinge et James I. Lathrop. “Real-time equivalence of chemical reaction networks and analog computers”. In : *DNA computing and molecular programming*. T. 11648. Lecture Notes in Comput. Sci. Springer, Cham, 2019, p. 37-53.

[32] Xiang Huang et al. “Real-time computability of real numbers by chemical reaction networks”. In : *Nat. Comput.* 18.1 (2019), p. 63-73.

[33] Julien Leroy. “Combinatorics on words”. Notes de cours. Université de Liège, sept. 2023.

[34] Joseph Liouville. “Sur des classes très-étendues de quantités dont la valeur n’est ni algébrique, ni même réductible à des irrationnelles algébriques”. In : *Journal de mathématiques pures et appliquées* 16 (1851), p. 133-142.

[35] Kurt Mahler. “Arithmetische eigenschaften der lösungen einer klasse von funktionalgleichungen”. In : *Mathematische Annalen* 101.1 (1929), p. 342-366.

[36] Kurt Mahler. “Zur approximation der exponentialfunktion und des logarithmus. Teil II.” In : (1932).

[37] Marston Morse et Gustav A Hedlund. “Symbolic dynamics”. In : *American Journal of Mathematics* 60.4 (1938), p. 815-866.

[38] Marston Morse et Gustav A Hedlund. “Symbolic dynamics II. Sturmian trajectories”. In : *American Journal of Mathematics* 62.1 (1940), p. 1-42.

[39] Yossi Moshe. “On some questions regarding k-regular and k-context-free sequences”. In : *Theoretical Computer Science* 400.1-3 (2008), p. 62-69.

[40] Simon Plouffe. “A formula for the n^{th} decimal digit or binary of π and powers of π”. In : *arXiv preprint arXiv :2201.12601* (2022).

[41] Tibor Rado. “On non-computable functions”. In : *Bell System Technical Journal* 41.3 (1962), p. 877-884.

[42] Michel Rigo. *Formal languages, automata and numeration systems*. eng. Networks and telecommunications series. Wiley-Blackwell, 2014.

[43] Michel Rigo. *Formal Languages, Automata and Numeration Systems 2 : Applications to Recognizability and Decidability*. eng. 1st edition. T. 9781848217881. Newark : John Wiley & Sons, Incorporated, 2014.

[44] Klaus Friedrich Roth. “Rational approximations to algebraic numbers”. In : *Mathematika* 2.1 (1955), p. 1-20.

[45] Hans Schlickewei. “On products of special linear forms with algebraic coefficients”. In : *Acta Arithmetica* 31.4 (1976), p. 389-398.

[46] Wolfgang M Schmidt. “Norm form equations”. In : *Annals of Mathematics* 96.3 (1972), p. 526-551.

[47] Claude E Shannon. “A universal Turing machine with two internal states”. In : *Automata studies* 34 (1956), p. 157-165.

[48] Claude E Shannon. “Mathematical theory of the differential analyzer”. In : *Journal of Mathematics and Physics* 20.1-4 (1941), p. 337-354.

[49] Michael Sipser. *Introduction to the theory of computation*. eng. Third edition. Boston, MA : Cengage Learning, 2013.

[50] Vladimir Sprindžuk. “A proof of Mahler’s conjecture on the measure of the set of S-numbers”. In : *Bulletin de l’Académie des sciences de l’URSS. Série mathématique.* 29 (1965), p. 379-436.

[51] Alan Mathison Turing et al. “On computable numbers, with an application to the Entscheidungsproblem”. In : *J. of Math* 58.345-363 (1936), p. 5.

[52] Hisao Yamada. “Real-Time Computation and Recursive Functions Not Real-Time Computable”. In : *IRE Transactions on Electronic Computers* EC-11.6 (1962), p. 753-760.