La propagation de retenue de la fonction successeur

Auteur : Kreczman, Savinien
Promoteur(s) : Rigo, Michel
Faculté : Faculté des Sciences
Diplôme : Master en sciences mathématiques, à finalité approfondie
Année académique : 2021-2022
URI/URL : http://hdl.handle.net/2268.2/14655

Faculté des Sciences
Département de Mathématique

La propagation de retenue de la fonction successeur

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

Année académique 2021-2022

Auteur :
Savinien Kreczman

Promoteur :
Michel Rigo

Table des matières

Remerciements

Je tiens à remercier Naïm Zenaïdi, Didier Kreczman, Anne-Marie Longrée et Sébastien Van Dhelsen pour leur relecture attentive et leur soutien moral.

Bien entendu, je ne peux suffisamment remercier mon promoteur Michel Rigo pour ses conseils judicieux, son aide et sa disponibilité tout au long de la rédaction de ce mémoire.

1 Introduction

Le sujet de ce mémoire est un article écrit par V.Berthé, C.Frougny, M.Rigo et J.Sakarovitch et intitulé "The Carry Propagation of the Successor Function" [5]. Dans la suite, nous posons le problème étudié dans cet article, développons les définitions et arguments utilisés par les auteurs pour le résoudre ainsi que les résultats obtenus. Nous fournissons également des compléments pour certaines bases théoriques qui sont supposées connues par les auteurs de l'article.

Il n'est malheureusement pas possible de reprendre toutes les bases depuis le début. Si ce mémoire tente de se suffire à lui-même le plus souvent possible, nous supposons tout de même que le lecteur a suivi un bachelier en sciences mathématiques et est donc familier avec un certain nombre de résultats qui seront alors admis. Dans ces cas, nous renverrons le lecteur vers des ouvrages de référence.

Le problème étudié est celui de la *propagation de retenue*. Une addition écrite ne peut pas toujours s'effectuer chiffre par chiffre : pour ajouter 1 à 99, il faut par deux fois reporter une retenue à la position suivante. Ces reports sont inévitables quelle que soit la façon dont les nombres sont représentés, puisque tous les nombres ne peuvent pas avoir des représentations de même longueur. Pire, le nombre de positions sur lesquelles la retenue peut se propager est illimité.

Une des conséquences de ce fait est que des processeurs travaillant sur des grands nombres entiers doivent attendre le résultat d'opérations précédentes pour déterminer s'il y a une retenue à prendre en compte ou pas, et ne peuvent pas additionner des nombres en parallèle, bloc de chiffres par bloc de chiffres. Comprendre et maîtriser les endroits où des retenues apparaissent permettrait ainsi de mieux gérer le temps passé sur une opération arithmétique par un processeur.

Le problème qui nous intéresse est un cas très restreint du cadre ci-dessus. Nous nous intéressons uniquement à l'opération successeur, qui ajoute 1 à son argument, et nous nous intéressons à la quantité moyenne de chiffres modifiée par cette opération.

La *propagation de retenue* d'un système de numération est une mesure du nombre de chiffres qui changent en moyenne entre la représentation d'un nombre et celle de son successeur.

Considérons un exemple. Dans la numération en base 10, le chiffre des unités change toujours et il est le seul à changer dans 90% des cas : quand le nombre auquel on ajoute 1 ne se finit pas par le chiffre 9. Deux chiffres changent dans 9% des cas, quand le chiffre des unités est 9 mais pas celui des dizaines. Trois chiffres changent dans $0,9\%$ des cas, quatre dans $0,09\%$ des cas... et on peut donc s'attendre à ce que le nombre "moyen" de chiffres changeant à chaque addition d'une unité soit

$$\frac{9}{10} + 2 \cdot \frac{9}{100} + 3 \cdot \frac{9}{1000} + \ldots = \frac{10}{9}.$$

C'est effectivement ce résultat qu'on obtient après calculs (proposition 3.2.12).

La valeur moyenne cherchée dépend du système utilisé pour représenter les nombres : dans le système binaire, cette valeur sera 2 plutôt que $\frac{10}{9}$. Le problème posé est alors de déterminer, en fonction du système choisi pour représenter les nombres, que vaudra ce nombre moyen de chiffres sur lesquels la retenue se propage.

En fait, déterminer si la quantité que nous étudions, qui est une limite, existe effectivement est l'étape la plus compliquée. En effet, si elle existe, la propagation de retenue est toujours de la forme $\frac{y}{y-1}$, où y est le *taux de croissance* du langage de la numération, qui est une constante facile à déterminer une fois le système fixé (corollaire 3.3.9). Par exemple, la propagation de retenue de la numération de Zeckendorf est $\frac{\phi}{\phi-1}$ où ϕ est le nombre d'or.

Ce mémoire est organisé en suivant globalement l'article source [5]. Nous avons décidé de déplacer les introductions d'autres résultats en annexe pour ne pas interrompre le flux des résultats qui s'appliquent directement au problème étudié.

- La section 2 introduit la théorie des systèmes de numération. Nous définissons les systèmes de numérations positionnels, les systèmes de numérations abstraits et la p/q-numération, trois types de numérations qui pourront demander différentes techniques dans la détermination de la propagation de retenue.
- La section 3 définit formellement le problème qui nous intéresse (définition 3.1.4) et utilise ensuite des arguments "élémentaires" (ne nécessitant pas de vastes bases théoriques) pour obtenir l'existence de la propagation de retenue dans les cas où la signature du langage est ultimement périodique (proposition 3.4.5). Ces cas recouvrent notamment le cas des bases entières, et celui des numérations en base rationnelle. De plus, on obtient un résultat permettant de calculer la propagation de retenue quand celle-ci existe (proposition 3.3.9).
- La section 4 s'intéresse au cas spécifique des langages rationnels. Dans ce cadre, on peut utiliser toute la théorie associée aux séries formelles et aux séries rationnelles pour exprimer plus finement les quantités en oeuvre et obtenir des résultats plus précis. Le théorème 4.3.2 nous permet de mieux comprendre quels

langages rationnels possèdent ou non une propagation de retenue. En particulier, des hypothèses peuvent être nécessaires sur les quotients du langage.

— Enfin, l'annexe regroupe les résultats empruntés à d'autres branches des mathématiques et utilisés dans les sections précédentes. Ceci comprend notamment des rappels sur la théorie des automates et des langages formels ainsi qu'une longue introduction aux séries formelles.

Le lecteur familier de notre source ([5]) constatera que nous avons choisi, avec l'accord de notre promoteur, de passer sous silence tout un pan de l'article, qui approche le problème via l'angle de la théorie ergodique. La raison de ce choix est que nous estimions que cette approche nécessitait trop de prérequis théoriques. Elle se base en effet sur des notions de topologie et de théorie ergodique, puis nécessite l'introduction de techniques récentes du ressort de la théorie ergodique pour contourner certaines des difficultés qui apparaissent. Par conséquent, nous avons jugé que cette section de l'article n'entrait pas dans le cadre de ce mémoire.

L'objectif de ce mémoire est double. Tout d'abord, il répond à une question précise, celle de déterminer des classes de langages pour lesquels la propagation de retenue existe. Le processus de réponse à cette question nous fera alors parcourir plusieurs domaines attenants aux systèmes de numération. Le deuxième objectif de ce mémoire est d'être une introduction guidée à ces domaines : langages formels, séries formelles et rationnelles, en plus d'un tour d'horizon des différentes familles de systèmes de numération.

Le problème de déterminer la propagation de retenue d'un langage peut être intéressant en lui-même, en tant que marchepied pour le problème plus compliqué de la propagation de retenue de l'opération d'addition. Une application possible est la conception de systèmes de numération dans lesquels l'addition de deux nombres peut être faite en parallèle sur des blocs de longueur fixée, ce qui permet de gagner du temps dans la manipulation de grands nombres. Cela dit, répétons que l'objectif de ce mémoire est double et que le voyage est aussi important que la destination de cette aventure.

2 Systèmes de numération

Un nombre n'est pas sa représentation : le nombre quarante-deux, égal à six fois sept, est un objet mathématique distinct du couple formé par le chiffre 4 et le chiffre 2. Nous utilisons couramment le système décimal, qui n'est rien de plus qu'une façon d'associer à chaque nombre une suite de chiffres qui le décrit complètement. Le système décimal n'est pas la seule façon de faire cela. Par exemple, le système binaire, utilisé en informatique, se base sur seulement deux chiffres et ainsi associe à quarante-deux la suite de chiffres $1, 0, 1, 0, 1, 0$. D'autres systèmes plus compliqués peuvent aussi être considérés. Ces systèmes peuvent vérifier ou non plusieurs propriétés intéressantes.

Dans ce chapitre, nous introduisons les bases de la théorie des systèmes de numération, et nous décrivons plusieurs familles de systèmes de numération que nous étudierons par la suite. Les sections suivantes donneront en effet des méthodes permettant de déduire l'existence de la propagation de retenue d'un langage et sa valeur, mais toutes les méthodes ne s'appliqueront pas à tous les langages. Il convient donc d'avoir un répertoire de systèmes de numération en tête en tant qu'exemples.

Ce chapitre utilise le vocabulaire de la théorie des langages formels et des automates. Nous renvoyons le lecteur qui ne serait pas familier de ce formalisme à l'annexe A où nous donnons les définitions de base : mots, langages, automates,... Nous définissons également les opérations usuelles sur le monoïde $A^\star$. Le lecteur pourra également consulter [6], [20] ou [21].

2.1 Bases

On fixe un alphabet A qui sera appelé l'*alphabet de la numération*. On peut supposer que cet alphabet est un tronçon initial des naturels, de la forme $\{0, 1, \ldots, r-1\}$. Donner un système de numération sur cet alphabet revient alors à décrire la correspondance entre les nombres et leurs écritures dans le système.

Définition 2.1.1. Un *système de numération* (de nombres naturels) sur A est la donnée de deux fonctions :

— Une injection de $\mathbb{N}$ dans $A^\star$, notée $\langle\cdot\rangle_L$, d'image $L \subset A^\star$, appelée *représentation*. Le langage L est appelé *langage de la numération*.

— Une surjection de M dans $\mathbb{N}$, notée π_L, avec $L \subseteq M \subseteq A^\star$ et vérifiant

$$\pi_L(\langle n\rangle) = n \quad \forall n \in \mathbb{N} \text{ et } \langle \pi_L(w)\rangle = w \quad \forall w \in L,$$

appelée *évaluation*.

Exemple 2.1.2. Considérons le système décimal usuel. L'alphabet est $A_{10} = \{0, \ldots, 9\}$. L'évaluation est donnée par la fonction

$$\pi_{10} : A_{10} \to \mathbb{N} : w_{l-1} \ldots w_0 \mapsto \sum_{i=0}^{l} w_i 10^i.$$

Le langage de la numération est $A_{10}^\star \setminus 0A_{10}^\star$, mais l'évaluation est définie sur $A_{10}^\star$. Remarquons qu'on prend la convention que le nombre 0 est représenté par le mot vide ε plutôt que par le mot d'une lettre 0.

La fonction de représentation peut être définie de différentes manières. En commençant par le chiffre le moins significatif, on peut considérer l'algorithme suivant. Pour trouver la représentation du nombre n, on pose $n_0 = n$ puis, si n_i est défini et strictement positif, on effectue la division euclidienne

$$n_i = 10n_{i+1} + a_i,\ a_i \in \{0, \ldots 9\}.$$

Si n_i est nul, l'algorithme s'arrête et la représentation de n est alors $a_{i-1} \ldots a_0$. On peut alors vérifier qu'on a bien

$$\sum_{j=0}^{i-1} a_j 10^j = n.$$

Une autre façon de procéder est d'obtenir les chiffres en commençant par les plus significatifs. Dans ce cas, on trouve d'abord $l = \lfloor \log_{10}(n) \rfloor + 1$, qui vérifie que $10^{l-1} \leq n < 10^l$. Puis, on définit $n_{l-1} = n$ et, si n_i est défini et non-nul, on choisit a_i tel que

$$a_i 10^i \leq n_i < (a_i + 1)10^i$$

et on pose $n_{i-1} = n_i - a_i 10^i$. On vérifie alors que $n_{-1} = 0$ et qu'à nouveau

$$\sum_{i=0}^{l-1} a_i 10^i = n.$$

On peut constater que les fonctions de représentation et d'évaluation données ci-dessus sont inverses l'une de l'autre, à condition de restreindre l'évaluation au langage annoncé.

Ces procédés se généralisent à n'importe quelle base entière. A titre d'exemple, nous pouvons décomposer 100 en base 8. Si l'on utilise le premier procédé ci-dessus, on a

$$\begin{aligned} 100 &= 8 * 12 + 4 \\ 12 &= 8 * 1 + 4 \\ 1 &= 8 * 0 + 1 \end{aligned}$$

et la représentation de 100 en base 8 est donc 144.

Pour l'autre méthode, on constate que $64 \leq 100 < 512$, et on a

$$\begin{aligned} 100 &= 64 * 1 + 36 \\ 36 &= 8 * 4 + 4 \\ 4 &= 1 * 4 + 0 \end{aligned}$$

et on retrouve la même représentation.

En fait, ces deux procédés sont assez généraux ; nous retrouverons le premier dans la sous-section 2.4 et le second dans la sous-section 2.2.

La définition 2.1.1 accomplit l'objectif d'associer à chaque nombre un mot qui le représente, via la fonction $\langle\cdot\rangle_L$. L'intérêt de définir l'évaluation sur un sur-ensemble de L est d'autoriser un nombre à posséder plusieurs représentations, tout en en canonisant une. En décimal par exemple, 42 et 042 sont deux représentations du même nombre mais 42 est la représentation canonique. Notre définition permet de retenir quelle valeur associer à la représentation 042 tout en sachant que ce n'est pas la bonne.

Le choix d'afficher une dépendance en L vient de ce que les systèmes de numération seront souvent confondus avec leur langage, pour des raisons qui apparaîtront dans la sous-section sur 2.3 les systèmes de numération abstraits.

Un autre exemple moins trivial est celui de la base de Fibonacci, qui a été étudiée par Edouard Zeckendorf dans [26].

Exemple 2.1.3. La suite de Fibonacci est définie par

$$U_0 = 1, U_1 = 2 \text{ et } U_{i+2} = U_{i+1} + U_i, \forall i \in \mathbb{N}\,.$$

On peut définir un système de numération basé sur la suite de Fibonacci via un algorithme glouton : si n est le nombre à représenter, soit l le naturel vérifiant $U_l \leq n < U_{l+1}$ (la représentation de 0 est toujours ε). Alors, on définit les nombres $a_l, \ldots, a_0$ et $n_l, \ldots, n_0$ via

$$n_l = n, a_i = \begin{cases} 1 & \text{si } n_i \geq F_i \\ 0 & \text{sinon} \end{cases}, \text{ et } n_{i-1} = n_i - a_i F_i.$$

par exemple, la représentation de $42 = 34 + 8$ est 10010000. On peut alors vérifier que

$$n = \sum_{i=0}^{l} a_i F_i$$

et qu'aucune représentation ne contient deux chiffres 1 consécutifs. En effet, si l'on avait $a_{i-1} = a_i = 1$, et en supposant $a_{i+1} = 0$ quitte à augmenter i, cela voudrait dire que

$$F_{i+1} = F_i + F_{i-1} \leq n_i = n_{i+1},$$

ce qui contredit le fait que $a_{i+1} = 0$. En fait, on peut montrer que tout mot qui ne contient pas deux 1 consécutifs et ne commence pas par 0 est la représentation d'un naturel.

Passons en revue quelques systèmes de numération classiques.

2.2 Systèmes de numération positionnels

L'idée des systèmes de numération de position est de donner à chaque position dans l'écriture des nombres une valeur fixée, puis de décomposer chaque nombre en somme de ces valeurs via un algorithme glouton. Cette idée généralise les bases entières, où les positions ont pour valeur les puissances de la base, ainsi que la numération de Zeckendorf, où les positions ont pour valeur la suite de Fibonacci. Des bases entières variées ont été utilisées à travers l'histoire, mais l'intérêt porté aux numérations positionnelles générales date des travaux de Cobham ([7] et [8]).

Définition 2.2.1. Le système de numération positionnel basé sur la suite $(U_i)_{i\in\mathbb{N}}$ est défini comme suit.

— La suite $(U_i)_{i\in\mathbb{N}}$ doit vérifier

$$U_0 = 1 \text{ et } U_i < U_{i+1}\, \forall i \in \mathbb{N} \text{ et } \exists C \in \mathbb{N} : \sup_{i\in\mathbb{N}} \frac{U_{i+1}}{U_i} \leq C.$$

— On considère alors l'alphabet $A_C = \{0, \dots, C-1\}$ pour le plus petit C vérifiant l'inégalité ci-dessus.

— La fonction d'évaluation est donnée par $\pi_U : A_C^\star \to \mathbb{N} : w_n \cdots w_0 \mapsto \sum_{i=0}^{n} w_i U_i$.

— Le langage de la numération est donné par les conditions gloutonnes

$$L_U = \{w_n \cdots w_0 \in A_C^\star | w_n \neq 0 \wedge \forall m \leq n, \sum_{i=0}^{m} w_i U_i < U_{m+1}.\} \tag{1}$$

Exemple 2.2.2. La numération usuelle en base 10 correspond à la suite $U_i = 10^i$, avec $\sup_{i\in\mathbb{N}} \frac{U_{i+1}}{U_i} = 10$, d'où l'alphabet minimal $\{0, \dots, 9\}$. Plus généralement, la numération en base p correspond à la suite $U_i = p^i$ avec l'alphabet $\{0, \dots, p-1\}$.

La numération de Zeckendorf correspond à la suite de Fibonacci. Respecter les conditions de (1) est équivalent à ne pas contenir deux chiffres 1 consécutifs. Un mot qui contient deux chiffres 1 consécutifs aux positions i et $i-1$ ne respecte pas la condition $\sum_{j=0}^{i} w_j U_j < U_{i+1}$. D'autre part, si un mot ne contient pas deux chiffres 1 consécutifs, alors la valeur maximale de $\sum_{j=0}^{i} w_j U_j$ est

$$U_i + U_{i-2} + U_{i-4} + \dots$$

et on peut montrer par récurrence sur i que cette quantité vaut $U_{i+1} - 1$. Toutes les conditions gloutonnes sont donc respectées.

Avant de passer à un exemple nouveau, faisons quelques remarques sur la définition 2.2.1.

Les conditions gloutonnes rendent la définition indépendante du choix de C, puisqu'un mot contenant n'importe quel chiffre supérieur ou égal à $\sup_{i\in\mathbb{N}} \frac{U_{i+1}}{U_i}$ violera toujours ces conditions.

D'autre part, forcer $U_0 = 1$ garantit que π_U soit une surjection. Elle n'est pas une bijection, puisque différentes combinaisons entières des éléments de U peuvent avoir la même valeur. Les conditions gloutonnes servent à sélectionner une représentation canonique, appelée *représentation gloutonne*. Sur le langage de la numération, l'évaluation est alors bien injective, et reste surjective, comme le garantit la proposition suivante.

Proposition 2.2.3. *La fonction π_U est bijective entre L_U et $\mathbb{N}$, i.e. pour tout naturel n il existe un unique mot $w_l \dots w_0$ vérifiant les conditions* (1) *et tel que $\sum_{i=0}^{l} w_i U_i = n$.*

Démonstration. Nous commençons par l'injectivité de la fonction π_U sur L_U. Supposons qu'il existe deux mots $a_l \dots a_0$ et $b_k \dots b_0$ tels que $\pi_U(a_l \dots a_0) = \pi_U(b_k \dots b_0) = n$. Alors, $k = l$. En effet, si l'on avait par exemple $k < l$, on aurait

$$n \geq U_l > \sum_{i=0}^{k} b_i U_i = n$$

où l'inégalité du milieu vient de la condition (1), ce qui est une contradiction. Donc $k \geq l$, et $k \leq l$ par symétrie.

Supposons maintenant de plus que n soit le plus petit naturel qui admette deux représentations dans L_U. Nous montrons que $a_l = b_l$, ce qui constituera une contradiction car alors $n - a_l$ admettrait lui aussi deux représentations, à savoir $a_{l-1} \dots a_0$ et $b_{l-1} \dots b_0$, et serait plus petit que n. Par l'absurde, supposons donc que $a_l > b_l$. On a alors

$$n \geq (a_l - 1)U_l + U_l > b_l U_l + \sum_{i=0}^{l-1} b_i U_i = n,$$

ce qui est à nouveau une contradiction. Ceci conclut le raisonnement, l'injectivité de π_U est démontrée.

Pour la surjectivité, nous allons exhiber algorithmiquement une représentation dans L_U pour chaque naturel n, via un algorithme glouton. Si n est un naturel non nul (0 est représenté par ε), soit l le naturel vérifiant $U_l \leq n < U_{l+1}$. On pose $n_l = n$ puis on définit $a_n, \dots, a_0$ et $n_{l-1}, \dots, n_{-1}$ via les égalités

$$a_i = \left\lfloor \frac{n_i}{U_i} \right\rfloor \text{ et } n_{i-1} = n_i - a_i U_i.$$

Remarquons qu'on a $n = n_i + \sum_{j=i+1}^{l} a_j U_j$ pour tout j entre -1 et l, ce qui se montre par récurrence descendante sur j. Comme on a aussi $n_{i-1} < U_i$ pour tout i, et donc $n_{-1} = 0$, la suite $a_l \dots a_0$ est bien une représentation de n. Il reste à montrer que c'est la représentation canonique, donc qu'elle vérifie les conditions de (1). On a clairement $a_l \neq 0$. Remarquons que $n_i = \sum_{j=0}^{i} a_i U_i$ pour tout i entre -1 et l. Comme $n_i < U_{i+1}$, on obtient

$$\sum_{j=0}^{i} a_i U_i < U_{i+1} \quad \forall i \in \{0, \dots, n\},$$

ce qui est l'égalité cherchée. Ainsi, la représentation obtenue via l'algorithme glouton est dans L_U, ce dont on déduit la surjectivité de π_U. □

Ainsi, les conditions gloutonnes permettent bien de canoniser exactement une représentation de chaque nombre, qui peut être obtenue explicitement quand la suite U est connue. La bijection π_U jouit même d'une propriété supplémentaire. Nous rappelons que les mots peuvent être ordonnés, et que l'ordre radiciel, défini en A.16, est un bon ordre.

Proposition 2.2.4. *La fonction π_U de L_U dans $\mathbb{N}$ est strictement croissante pour l'ordre radiciel de L_U et l'ordre usuel de $\mathbb{N}$.*

Démonstration. Considérons deux mots $a = a_l \dots a_0$ et $b = b_k \dots b_0$ de L_U, supposons $a \sqsubset b$ et montrons $\pi_U(a) < \pi_U(b)$. Il y a deux possibilités : soit $l < k$, soit $l = k$ et dans ce cas il existe $j \in \{0, \dots, l\}$ tel que $a_l = b_l, \dots a_{j+1} = b_{j+1}$ et $a_j < b_j$.

Dans le cas où $l < k$, on a

$$\pi_U(a) = \sum_{i=0}^{l} a_i U_i < U_{i+1} \leq \sum_{i=0}^{k} b_i U_i = \pi_U(b),$$

où l'inégalité stricte vient des conditions (1) appliquées à a.

Dans le second cas, on a

$$\begin{aligned}
\pi_U(a) &= \sum_{i=0}^{j-1} a_i U_i + a_j U_j + \sum_{i=j+1}^{l} a_i U_i \\
&< U_j + + a_j U_j + \sum_{i=j+1}^{l} b_i U_i \\
&\leq \sum_{i=j}^{l} b_i U_i \\
&\leq \pi_U(b).
\end{aligned}$$

Ainsi, on a bien obtenu la croissance stricte de la fonction π_U. □

Ainsi, les systèmes de numération positionnels *préservent l'ordre* : à un mot plus petit (pour l'ordre radiciel) correspond un nombre plus petit. Remarquons qu'il n'y a qu'une seule bijection croissante entre L_U et $\mathbb{N}$. Que notre fonction d'évaluation soit cette bijection est une propriété désirable, qui témoigne du bien-fondé de la définition 2.2.1.

Illustrons ces concepts par un exemple nouveau.

Exemple 2.2.5. La suite $1, 3, 8, 21, 55, \dots$ obtenue en prenant un terme sur deux dans la suite de Fibonacci est également une suite linéaire récurrente à coefficients constants : elle est définie par $U_{n+2} = 3U_{n+1} - U_n$ pour tout n naturel, ce qui peut se montrer par simple manipulation des égalités définissant la suite de Fibonacci.

La numération de Fina [1] est le système de numération associé, sur l'alphabet $A_3\{0,1,2\}$. Ici, la fonction d'évaluation est donc

$$w_l \dots w_0 \mapsto \sum_{i=0}^{l} w_i U_i.$$

Par contre, connaître le langage L_U de la numération n'est pas évident a priori. Par exemple, les mots 2000 et 1212 ont tous deux pour valeur 42, mais 1212 ne respecte pas les conditions (1) et n'est pas dans le langage. Nous aimerions une condition permettant de distinguer au premier coup d'oeil un mot faisant partie du langage d'un mot n'en faisant pas partie. Nous allons à présent montrer que cette condition est d'appartenir au langage des mots ne commençant pas par 0 et ne contenant pas de facteur de la forme 21^n2,

$$M_U = A_3^\star \setminus \left(0A_3^\star \cup A_3^\star 21^\star 2 A_3^\star\right).$$

Dans un premier temps, nous montrons que $L_U \subset M_U$. Pour cela, montrons que si $i < j$, alors

$$2U_i + U_{i+1} + \ldots + U_{j-1} + 2U_j \geq U_{j+1}.$$

D'abord, remarquons que, en posant $U_{-1} = 0$,

$$2U_i + 2U_{i+1} = 3U_{i+1} - U_i - U_{i+1} + 3U_i = U_{i+2} + U_{i-1}$$

pour tout naturel i, et donc $2U_i + 2U_{i+1} \geq U_{i+2}$. Dès lors, on a

$$\begin{aligned}
&0 \leq 2U_i + 2U_{i+1} - U_{i+2} \\
\Leftrightarrow\ &0 \leq (2U_i + 2U_{i+1} - U_{i+2}) + (-U_{i+1} + 3U_{i+2} - U_{i+3}) \\
&\quad = 2U_i + U_{i+1} + 2U_{i+2} - U_{i+3} \\
\Leftrightarrow\ &0 \leq 2U_i + U_{i+1} + U_{i+2} + 2U_{i+3} - U_{i+4} \\
&\quad \vdots \\
\Leftrightarrow\ &U_{j+1} \leq 2U_i + U_{i+1} + \ldots + 2U_j
\end{aligned}$$

Dès lors, un mot contenant un facteur de la forme 21^n2 ne saurait pas vérifier les conditions gloutonnes. Ainsi, M_U^c est inclus dans L_U^c, et donc L_U est inclus dans M_U comme prévu.

Comme π_U est surjective sur L_U, elle l'est donc également sur M_U. Dès lors, pour tout naturel n il existe au moins un mot de M_U dont l'évaluation est n. Remarquons que si $U_{N-1} \leq n < U_N$, alors un tel mot a pour longueur au plus N. Pour montrer que $L_U = M_U$, nous montrons que pour tout naturel il existe un unique mot w de M_U tel que $\pi_U(w) = n$. Pour ce faire, nous prouvons que $\left|M_u \cap A_3^{\leq N}\right| = U_N$. Cela montrera en effet que tout naturel de $\{0, \ldots, U_{N-1}\}$ possède exactement un antécédent par π_U dans $M_u^{\leq N}$, donc que $M_U = L_U$.

Pour calculer $\left|M_u \cap A_3^{\leq N}\right|$ et montrer l'égalité à U_N, nous employons une technique qui nous resservira à la section 4, se basant sur la matrice d'adjacence d'un automate reconnaissant le langage. Un automate reconnaissant le langage M_U est donné à la figure 1. Si nous voyons cet automate comme un graphe, sa matrice d'adjacence est

$$A = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 0 & 2 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 3 \end{pmatrix}.$$

Soit t_m le nombre de mots de longueur m dans M_U. On sait que t_m est le nombre de chemins de longueur m reliant l'état initial à un état final du graphe que constitue notre automate. On sait que si le graphe G a pour matrice d'adjacence M_G, le nombre de chemins de longueur m du sommet i vers le sommet j dans G est donné par $(M_G^m)_{ij}$. Ici, la quantité t_m est donc égale à $(A^m)_{11} + (A^m)_{12} + (A^m)_{13}$. Comme l'état 4 est un puits, on peut effectuer les calculs avec la sous-matrice formée des colonnes et des lignes 1 à 3 de A, que nous notons B.

1. Cette notation n'est pas standard. Nous n'en sommes pas responsables, il s'agit de la notation de [5].

Or, on sait par le théorème de Cayley-Hamilton qu'une matrice est annulée par son polynôme caractéristique. Calculons donc le polynôme caractéristique de A. On a

$$\begin{aligned}\det(B-\lambda I) &= \begin{vmatrix} -\lambda & 1 & 1 \\ 0 & 2-\lambda & 1 \\ 0 & 1 & 1-\lambda \end{vmatrix} \\ &= (-\lambda)\left((2-\lambda)(1-\lambda)-1\right) \\ &= -\lambda^3+3\lambda^2-\lambda.\end{aligned}$$

Dès lors, on a $B^3 = 3B^2 - B$, et, en considérant cette égalité sur les coefficients de la première ligne, on trouve $t_{m+3} = 3t_{m+2} - t_{m+1}$ pour tout naturel m. On trouve les premiers termes de la suite $(t_m)_m$ en examinant le langage M_U, et ces premiers termes sont $1, 2, 5, 13, \ldots$.

Si l'on pose maintenant

$$v_m = \begin{cases} 1 \text{ si } m = 0 \\ U_m - U_{m-1} \text{ si } m > 0 \end{cases},$$

alors la suite $(v_m)_m$ vérifie la relation $v_{m+3} = 3v_{m+2} - v_{m+1}$ pour tout naturel m car $(U_m)_m$ la vérifie également, et les premiers termes de cette suite sont $1, 2, 5, 13, \ldots$. Ainsi, on a $t_m = v_m$ pour tout naturel m. De là, le nombre de mots de longueur au plus m dans M_U est

$$t_0 + \ldots + t_m = v_0 + \ldots + v_m = U_0 + (U_1 - U_0) + \ldots + (U_m - U_{m-1}) = U_m.$$

Ceci permet d'obtenir la conclusion attendue : nous avons montré que les naturels strictement inférieurs à U_m étaient en bijection avec les mots de $M_U \cap A_3^{\leq m}$. Le langage M_U est donc bien le langage de la numération de Fina.

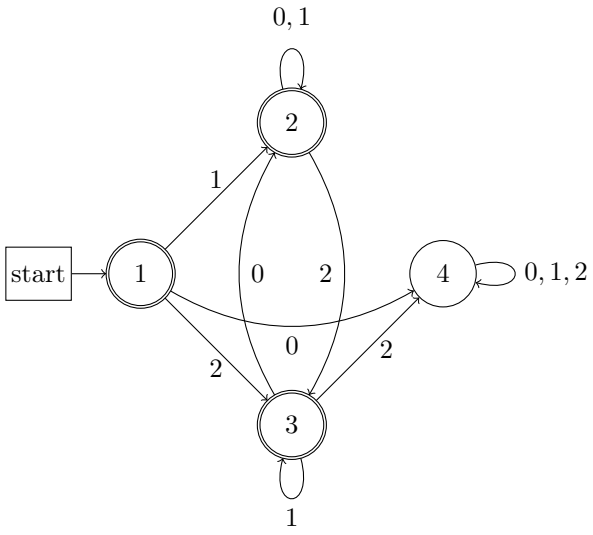

FIGURE 1 – Un automate acceptant le langage M_U.

Dans les trois exemples ci-dessus, le langage de la numération est rationnel. Ce n'est cependant pas toujours le cas, comme on peut le constater sur un exemple, présenté par J.Shallit dans [23].

Exemple 2.2.6. Considérons la suite donnée par $U_n = (n+1)^2$ pour tout n naturel. Cette suite est une suite linéaire récurrente, car on a $U_{n+3} = 3U_{n+2} - 3U_{n+1} + U_n$, comme on peut le voir en développant les deux membres.

Le langage L_U de la numération associée n'est pas rationnel : s'il l'était, alors le langage

$$L_U \cap 10^\star 10^\star$$

serait lui aussi rationnel puisque les langages rationnels sont stables par intersection. Or, pour qu'un mot de la forme 10^a10^b soit dans L_U, il doit respecter les conditions gloutonnes, qui se résument ici à avoir

$$\begin{aligned} U_{a+1+b} + U_b < U_{a+b+2} &\Leftrightarrow (a+b+2)^2 + (b+1)^2 < (a+b+3)^2 \\ &\Leftrightarrow (a+b)^2 + 4(a+b) + 4 + b^2 + 2b + 1 < (a+b)^2 + 6(a+b) + 9 \\ &\Leftrightarrow b^2 < 2a + 4 \end{aligned}$$

et on a donc

$$L_U \cap 10^\star 10^\star = \left\{10^a 10^b : b^2 < 2a+4\right\}.$$

Si le langage du membre de droite était rationnel, alors $M = \left\{10^a 10^b : b^2 \geq 2a+4\right\}$ le serait également, à nouveau en utilisant différentes stabilités des langages rationnels.

Or, ce dernier langage n'est pas rationnel : s'il l'était, par le lemme de la pompe (A.14) il devrait exister un N naturel tel que, pour tout mot de la forme 10^a10^b avec $a+b+2 > N$, on puisse trouver x, y, z avec $xyz = 10^a10^b$, $|xy| < N$ et xy^nz reste dans M quelque soit le naturel n. C'est une contradiction : si un tel N existait, on pourrait considérer le mot $10^N 10^{2N+4}$. Ce mot est dans le langage, et les x, y, z qui doivent exister vérifieront $xy = 10^j$ pour un naturel j. Si y contient un 1, alors xy^2z contient trois chiffres 1 et n'est donc pas dans M. Sinon, en prenant des valeurs de n de plus en plus grandes, on peut exhiber des mots de M de la forme 10^J10^N pour des valeurs arbitrairement grandes de J, ce qui est impossible. Donc le langage M n'est pas rationnel, et par suite L_U n'est pas rationnel non plus.

Que le langage d'une numération soit rationnel est une propriété souhaitable. Non seulement on peut dans ce cas tester efficacement l'appartenance d'un mot à ce langage, mais c'est également une condition nécessaire pour que la *normalisation* soit réalisable par un automate fini. Cette opération consiste à remplacer une représentation quelconque par la représentation canonique de même valeur, i.e. à calculer la fonction

$$w \mapsto \pi_L(\langle w \rangle_L).$$

Pouvoir effectuer la normalisation efficacement permet alors d'utiliser des algorithmes qui travaillent chiffre par chiffre sur leurs entrées puis de normaliser le résultat obtenu, plutôt que de considérer les représentations dans leur entièreté.

Malheureusement, nous venons d'en voir un exemple, cette propriété n'est pas toujours vérifiée en pratique, et il n'existe pas à ce jour de critère théorique permettant de déterminer exactement quand une suite donnée fournit un langage de numération rationnel. J.Shallit a montré en 1994 que la suite de départ devait être récurrente linéaire à coefficients constants pour que le langage puisse être rationnel, mais ce n'est pas suffisant vu l'exemple ci-dessus. En 1998, Hollander a apporté des conditions nécessaires et des conditions suffisantes dans le cas où $\lim_{i\to\infty} \frac{U_{i+1}}{U_i}$ existe, mais les conditions dans le cas contraire ne sont toujours pas pleinement comprises.

2.3 Systèmes de numération abstraits

Nous venons de voir qu'il est difficile de déterminer en toute généralité si un système de numération positionnel correspond à un langage rationnel. Les systèmes de numération abstraits (ANS), introduits en 2001 par P.Lecomte et M.Rigo dans [16], permettent de contourner cette difficulté. L'idée est de prendre le problème à l'envers, et de définir directement à partir d'un langage L un système de numération dont le langage sera L.

Pour cela, on suppose que A est muni d'un ordre total $\leq$. Cet ordre peut être étendu de A à $A^\star$ de deux manières différentes : l'ordre radiciel et l'ordre lexicographique (voir annexe A, définition A.16).

On peut énumérer tous les éléments de L par ordre radiciel croissant, en définissant $w_0 = \min L$, $L_0 = L \setminus w_0$, puis en posant inductivement $w_i = \min L_{i-1}$ et $L_i = L_{i-1} \setminus \{w_i\}$ si $i \geq 1$ et si L_{i-1} est défini. Si L est infini, il est dénombrable et on obtient une énumération de la forme

$$L = \{w_0, w_1, w_2, \ldots\} \text{ avec } w_i \sqsubseteq w_{i+1} \forall i \in \mathbb{N}.$$

Alors, la fonction $\langle\cdot\rangle_L : \mathbb{N} \to L : n \mapsto w_n$ est une bijection entre $\mathbb{N}$ et L, qui est de plus croissante pour l'ordre usuel sur $\mathbb{N}$ et l'ordre radiciel sur L. On peut prendre pour π_L la bijection inverse, qui envoie w_n sur n, et on a défini ainsi un système de numération dont le langage est L. En résumé :

Définition 2.3.1. Si L est un langage énuméré sous la forme

$$L = \{w_i : i \in \mathbb{N}, w_i \sqsubset w_{i+1} \, \forall i \in \mathbb{N}\},$$

alors le *système de numération abstrait* associé à L est celui qui a pour langage L, pour fonction d'évaluation

$$\pi_L : L \to \mathbb{N} : w_i \mapsto i$$

et pour fonction de représentation

$$\langle\cdot\rangle_L : \mathbb{N} \to L : i \mapsto w_i.$$

Bien sûr, il est ici facile de voir quand le langage de la numération est rationnel, puisqu'il est donné au départ.

Les exemples abordés ci-dessus peuvent être revus dans ce contexte. Nous avons vu à la proposition 2.2.4 que la fonction d'évaluation π_U d'un système de numération de position associé à la suite U est croissante pour l'ordre radiciel sur L_U et l'ordre usuel sur $\mathbb{N}$. Vu la définition ci-dessus, la fonction d'évaluation π_L d'un système de numération abstrait associé au langage L est également croissante pour l'ordre radiciel sur L et l'ordre usuel sur $\mathbb{N}$. Dès lors, tout système de numération positionnel peut être vu comme un système de numération abstrait associé à son langage. Ceci justifie qu'on confonde souvent les notations pour un système de numération et son langage.

Exemple 2.3.2. Le système de numération abstrait associé au langage $\{0,\dots 9\}^\star \setminus 0\{0,\dots,9\}^\star$ est le système usuel en base 10 restreint aux seules représentations canoniques. De même, le système de numération abstrait associé au langage $L_F = \{0,1\}^\star\setminus(\{0,1\}^\star 11\{0,1\}^\star \cup 0\{0,1\}^\star)$ est la numération de Zeckendorf, et de même pour les exemples 2.2.5 et 2.2.6.

Cette façon de définir des systèmes de numération est plus générale que celle de la section précédente ; on peut trouver des systèmes de numération qui ne sont pas des numérations de position.

Exemple 2.3.3. On considère le langage rationnel $L = 0^\star 1^\star$ sur l'alphabet $\{0,1\}$. Son énumération par ordre radiciel est

$$L = \varepsilon, 0, 1, 00, 01, 11, 000, 001, 011, 111, 0000, 0001, 0011, \dots$$

et L contient $n+1$ mots de longueur n, pour chaque naturel n.

On a par exemple $\langle 0\rangle_L = \varepsilon$, $\langle 8\rangle_L = 011$ et $\pi_L(0011) = 12$. Cette numération ne peut pas être vue comme une numération de position, puisque le mot 0 n'a pas la valeur 0.

On peut, en cherchant un peu plus loin, trouver des exemples encore plus éloignés du cadre de la numération de position.

Exemple 2.3.4. Considérons le langage $a^\star b^\star a^\star$. L'unique mot de longueur inférieure ou égale à 3 qui n'en fait pas partie est *bab*. De là, on trouve

$$\pi(aa) = 3, \pi(ba) = 5, \pi(baa) = 11 \text{ et } \pi(bba) = 12.$$

On en déduit qu'il n'existe pas de fonction

$$\varphi : \{a,b\} \times \mathbb{N} \to \mathbb{N}$$

telle que π s'écrive comme

$$\pi(w_l \dots w_0) = \sum_{i=0}^{l} \varphi(w_i, i).$$

En effet, les égalités ci-dessus impliquent simultanément $\varphi(b,2) = 8$ et $\varphi(b,2) = 7$. Cette propriété nous éloigne plus des numérations de positions (où l'on peut poser $\varphi(w_i, i) = w_i U_i$) que l'exemple précédent, où l'on peut évaluer des mots chiffre par chiffre en se basant sur la fonction

$$\varphi(0,i) = i+1 \text{ et } \varphi(1,i) = i+2.$$

2.4 La p/q-numération

Bien que les ANS décrits à la sous-section précédente recouvrent les systèmes de numération dont nous traiterons dans la suite de ce mémoire, il est parfois utile d'avoir un autre éclairage sur un système de numération d'intérêt particulier, notamment quand on désire une formule close pour la fonction d'évaluation. C'est le cas de la numération en base rationnelle que nous décrivons maintenant. Cette numération, introduite par S.Akiyama, C.Frougny et J.Sakarovitch dans [2], et étudiée en profondeur par V.Marsault dans sa thèse de doctorat ([17]), généralise l'algorithme de division euclidienne vu à l'exemple 2.1.2 qui permet de déterminer les chiffres d'une représentation en base entière en commençant par le moins significatif.

Soient $p > q \geq 1$ deux entiers premiers entre eux et N l'entier à représenter. Posons $N_0 = N$ puis

$$qN_i = pN_{i+1} + a_i \quad , 0 \leq a_i < p.$$

Comme $q < p$, on a $N_{i+1} < N_i$ et cet algorithme se termine donc toujours, à un indice k tel que $N_k \neq 0$ et $N_{k+1} = 0$. On a la suite d'égalités

$$\begin{aligned} qN &= pN_1 + a_0 \\ q^2N &= p(pN_2 + a_1) + qa_0 \\ q^3N &= p^2(pN_3 + a_2) + pqa_1 + q^2a_0 \\ &\vdots \\ q^{k+1}N &= p^ka_k + \ldots + pq^{k-1}a_1 + q^ka_0 \end{aligned}$$

et on obtient ainsi l'égalité

$$N = \sum_{i=0}^{k} \frac{a_i}{q} \left(\frac{p}{q}\right)^i.$$

Alors, la représentation de l'entier N est le mot $\langle N \rangle_{\frac{p}{q}} = a_k \cdots a_0$, calculé à partir du chiffre le moins significatif. On pose donc

$$L_{\frac{p}{q}} = \{\langle N \rangle_{\frac{p}{q}} \mid N \in \mathbb{N}\},$$

l'alphabet minimal est $\{0, \ldots, p-1\}$ et la fonction d'évaluation est

$$\pi_{\frac{p}{q}} : a_k \cdots a_0 \mapsto \sum_{i=0}^{k} \frac{a_i}{q} \left(\frac{p}{q}\right)^i. \tag{2}$$

Remarquons que si $q = 1$, on retrouve la numération usuelle en base p. Dans la suite de cette section, on suppose $q \geq 2$.

Exemple 2.4.1. Pour illustrer les développements qui viennent d'être faits, calculons le développement en base $\frac{3}{2}$ de 17 : on pose $N_0 = 17$, puis on obtient successivement

$$\begin{aligned} 2 * 17 = 34 &= 11 * 3 + 1 \text{ d'où } a_0 = 1, N_1 = 11 \\ 2 * 11 = 22 &= 7 * 3 + 1 \text{ d'où } a_1 = 1, N_2 = 7 \\ 2 * 7 = 14 &= 4 * 3 + 2 \text{ d'où } a_2 = 2, N_3 = 4 \\ 2 * 4 = 8 &= 2 * 3 + 2 \text{ d'où } a_3 = 2, N_4 = 2 \\ 2 * 2 = 4 &= 1 * 3 + 1 \text{ d'où } a_4 = 1, N_5 = 1 \\ 2 * 1 = 2 &= 0 * 3 + 2 \text{ d'où } a_5 = 2, N_6 = 0 \end{aligned}$$

et l'algorithme s'arrête alors. la représentation de 17 en base $\frac{3}{2}$ est donc 212211. On peut vérifier que

$$\frac{1}{2}\left(2 * \frac{243}{32} + 1 * \frac{81}{16} + 2 * \frac{27}{8} + 2 * \frac{9}{4} + 1 * \frac{3}{2} + 1 * 1\right) = 17.$$

Le langage de la numération semble plus compliqué à décrire, et nous verrons qu'il ne s'agit pas que d'une impression. Dans un premier temps, examinons quelles sont les ambiguïtés possibles : à un nombre naturel N, peut-on associer d'autres mots que celui obtenu par l'algorithme ci-dessus qui, évalués via (2), donnent tout de même N ? Nous allons montrer qu'il n'y a aucune autre ambiguïté possible que l'ajout de zéros de tête : les mots qui ne sont pas obtenus par l'algorithme ci-dessus (éventuellement avec ajout de zéros de tête) n'ont pas une évaluation entière.

Proposition 2.4.2. *Pour tout naturel k, la fonction $\pi_{\frac{p}{q}}$ restreinte à $\{0, \ldots, p-1\}^k$ est injective.*

Démonstration. On procède par récurrence sur $k \in \mathbb{N}$, l'initialisation à $k = 0$ est triviale.

Supposons que $\pi_{\frac{p}{q}}(a_{k-1}\cdots a_0) = \pi_{\frac{p}{q}}(b_{k-1}\cdots b_0)$. Dans ce cas, on a

$$\begin{aligned}
\sum_{i=0}^{k-1} \frac{a_i}{q}\left(\frac{p}{q}\right)^i &= \sum_{i=0}^{k-1} \frac{b_i}{q}\left(\frac{p}{q}\right)^i \\
\Leftrightarrow \sum_{i=0}^{k-1} \frac{a_i - b_i}{q}\left(\frac{p}{q}\right)^i &= 0 \\
\Leftrightarrow \sum_{i=0}^{k-1} (a_i - b_i) q^{k-1-i} p^i &= 0.
\end{aligned}$$

Dès lors, p est une racine entière du polynôme en x : $\sum_{i=0}^{k-1}(a_i - b_i)q^{k-1-i}x^i$, et doit diviser son terme indépendant $(a_0 - b_0)q^{k-1}$. Comme p est premier avec q, on trouve $p \mid a_0 - b_0$. Comme $0 \leq a_0, b_0 < p$, on a $a_0 = b_0$. De là, on trouve

$$\begin{aligned}
\sum_{i=1}^{k-1} (a_i - b_i) q^{k-1-i} p^i &= 0 \\
\Leftrightarrow \sum_{i=1}^{k-1} \frac{a_i - b_i}{q} \frac{p^{i-1}}{q^{i-1}} &= 0 \\
\Leftrightarrow \pi_{\frac{p}{q}}(a_{k-1}\cdots a_1) &= \pi_{\frac{p}{q}}(b_{k-1}\cdots b_1)
\end{aligned}$$

et on conclut que $a = b$ par l'hypothèse de récurrence. □

Corollaire 2.4.3. *Si $\pi_{\frac{p}{q}}(u) = \pi_{\frac{p}{q}}(v)$ et $|u| \geq |v|$, alors $u = 0^{|u|-|v|}v$. En particulier, les seuls mots de $\{0, \ldots, p-1\}^\star$ qui ont une valeur entière sont ceux de $0^\star L_{\frac{p}{q}}$.*

Démonstration. Si $\pi_{\frac{p}{q}}(u) = \pi_{\frac{p}{q}}(v)$, alors $\pi_{\frac{p}{q}}(u) = \pi_{\frac{p}{q}}(0^{|u|-|v|}v)$. Par le résultat précédent, on a le premier point.

Pour le deuxième point, si $\pi_{\frac{p}{q}}(u) = n$ pour un naturel n, comme $\langle n \rangle_{\frac{p}{q}}$ ne commence pas par 0 (puisque cela contredirait que l'algorithme décrit ci-dessus s'arrête dès que $N_k + 1 = 0$) , on a forcément $u \in 0^\star \langle n \rangle_{\frac{p}{q}}$. □

Ceci justifie qu'on ne puisse pas étendre le langage de la numération plus qu'en ajoutant des zéros de tête, contrairement au cas des numérations positionnelles.

Nous allons maintenant examiner les propriétés de $L_{\frac{p}{q}}$ du point de vue de la théorie des langages formels. Dans un premier temps, montrons que ce langage est PCE, ce qui permettra d'appliquer les résultats développés dans la section 3.

Proposition 2.4.4. *Le langage $L_{\frac{p}{q}}$ est clos par préfixe et extensible à droite.*

Démonstration. Soit $w = a_k \cdots a_0$ la représentation d'un naturel N tel que la première égalité de l'algorithme décrit ci-dessus soit $qN = pN_1 + a_0$. Alors, on peut vérifier que $a_k \cdots a_1$ est la représentation de N_1. Le langage est donc clos par préfixe.

De même, wa est la représentation d'un entier M si et seulement si $qM = pN + a$. Comme $p > q$, il existe un naturel b entre 0 et $p-1$ tel que $pN + a$ soit divisible par q, et $wb \in L$ puisque c'est la représentation de $\frac{pN+b}{q}$. Le langage est donc extensible à droite. □

Nous allons à présent montrer que $L_{\frac{p}{q}}$ n'est pas un langage rationnel. Pour cela, nous allons supposer qu'il l'est et obtenir une contradiction avec le lemme de la pompe (propriété A.14). Nous avons d'abord besoin d'un lemme décrivant des conditions nécessaires à l'apparition de grandes puissances d'un mot comme facteurs d'un mot de $L_{\frac{p}{q}}$.

Lemme 2.4.5. *Si $w \in L_{\frac{p}{q}}$ et $w = uv$ avec $u, v \neq \varepsilon$, alors*

$$uv^k \in L_{\frac{p}{q}} \Leftrightarrow q^{(k-1)|v|} \mid \pi_{\frac{p}{q}}(w) - \pi_{\frac{p}{q}}(u).$$

Dès lors, il existe $K \in \mathbb{N}$ tel que $uv^k \notin L_{\frac{p}{q}}$ pour tout $k > K$.

Démonstration. Comme $w \in L_{\frac{p}{q}}$, on a que u ne commence pas par 0, donc $uv^k \in L_{\frac{p}{q}} \Leftrightarrow \pi_{\frac{p}{q}}(uv^k) \in \mathbb{N}$. Si $uv^k \in L$ et $k \geq 1$, alors $uv^{k-1} \in L$ car L est clos par préfixe.

Remarquons de plus que

$$\begin{aligned}\pi_{\frac{p}{q}}(xv) &= \sum_{i=0}^{|v|-1} \frac{v_i}{q}\left(\frac{p}{q}\right)^i + \sum_{j=0}^{|x|-1} \frac{x_j}{q}\left(\frac{p}{q}\right)^{j+|v|} \\ &= \pi_{\frac{p}{q}}(v) + \left(\frac{p}{q}\right)^{|v|} \pi_{\frac{p}{q}}(x).\end{aligned}$$

Dès lors, on a

$$\begin{aligned}uv^k \in L &\Rightarrow \pi_{\frac{p}{q}}(uv^k) - \pi_{\frac{p}{q}}(uv^{k-1}) \in \mathbb{N} \\ &\Leftrightarrow \pi_{\frac{p}{q}}(v) + \left(\frac{p}{q}\right)^{|v|} \pi_{\frac{p}{q}}(uv^{k-1}) - \pi_{\frac{p}{q}}(v) + \left(\frac{p}{q}\right)^{|v|} \pi_{\frac{p}{q}}(uv^{k-2}) \in \mathbb{N} \text{ si } k \geq 2 \\ &\Leftrightarrow \left(\frac{p}{q}\right)^{|v|} (\pi_{\frac{p}{q}}(uv^{k-1}) - \pi_{\frac{p}{q}}(uv^{k-2})) \\ &\Leftrightarrow \cdots \\ &\Leftrightarrow \left(\frac{p}{q}\right)^{(k-1)|v|} \pi_{\frac{p}{q}}(uv) - \left(\frac{p}{q}\right)^{(k-1)|v|} \pi_{\frac{p}{q}}(u) \in \mathbb{N}\end{aligned}$$

et donc $uv^k \in L \Rightarrow q^{(k-1)|v|} \mid \pi_{\frac{p}{q}}(uv) - \pi_{\frac{p}{q}}(u)$ puisque p et q sont premiers entre eux, d'où l'égalité annoncée.

Le deuxième point du lemme résulte de ce que $\pi_{\frac{p}{q}}(uv) - \pi_{\frac{p}{q}}(u)$ est une quantité fixe et n'est donc pas divisible par des puissances arbitrairement grandes de q, si $q \geq 2$. □

Proposition 2.4.6. *Si $q \geq 2$, le langage $L_{\frac{p}{q}}$ n'est pas rationnel.*

Démonstration. Procédons par l'absurde et supposons qu'il soit rationnel. Dans ce cas, le lemme de la pompe indique l'existence d'un N naturel tel que pour tout $w \in L_{\frac{p}{q}}$, $|w| > N$, on puisse factoriser w en xyz avec $y \neq \varepsilon$ de sorte que $xy^n z \in L_{\frac{p}{q}}$ pour tout naturel n. Mais alors, comme $L_{\frac{p}{q}}$ est clos par préfixe, xy^n serait dans $L_{\frac{p}{q}}$ pour tout n, ce qui contredit le lemme ci-dessus. □

Remarque 2.4.7. Un raisonnement similaire utilisant le même lemme permet de montrer que L n'est pas non plus un langage algébrique (nous renvoyons le lecteur à [20] pour les définitions).

Ainsi, le langage $L_{\frac{p}{q}}$ n'est pas rationnel. Il n'est donc pas "simple" du point de vue de la théorie des langages formels. Malgré cette absence de rationalité, le langage $L_{\frac{p}{q}}$ possède tout de même des propriétés intéressantes, dont celle qui justifie son introduction. Nous verrons alors qu'il est tout de même possible de décrire simplement ce langage, malgré que la description en question ne soit pas du ressort de la théorie des langages formels. Pour développer cette propriété, nous avons besoin du concept de signature d'un langage.

Définition 2.4.8. La *signature* d'un langage L est la suite s_L définie par

$$(s_L)_n = \delta_{n,0} + \mid \{a : \langle n \rangle_L a \in L\} \mid .$$

où $\delta_{n,0}$ est le delta de Kronecker, qui vaut 1 si $n = 0$ et 0 sinon.

Intuitivement, le n-ième terme de la signature est le nombre de façons d'étendre à droite la représentation de n, où l'on ajoute la convention que ε s'étend lui-même à gauche (ce qui est nécessaire si l'on veut par exemple que la signature du langage associé à la base p soit p^ω). En prenant un peu d'avance sur les notions de la section suivante, on peut illustrer ce concept.

La figure 2 représente les premiers niveaux de l'arbre du langage $L_{\frac{3}{2}}$, qui sera défini à la section 3.2. Chaque naturel est l'étiquette d'un noeud, et l'étiquetage du chemin du noeud 0 vers le noeud N donne la représentation du naturel N. Par exemple, le chemin reliant 0 à 17 est étiqueté par 212211, qui avait été obtenu dans l'exemple ci-dessus. Sur la figure, la signature du langage est alors la suite dont le n-ième terme est le nombre d'arêtes sortant du noeud n.

Le langage $L_{\frac{p}{q}}$ possède alors la propriété suivante :

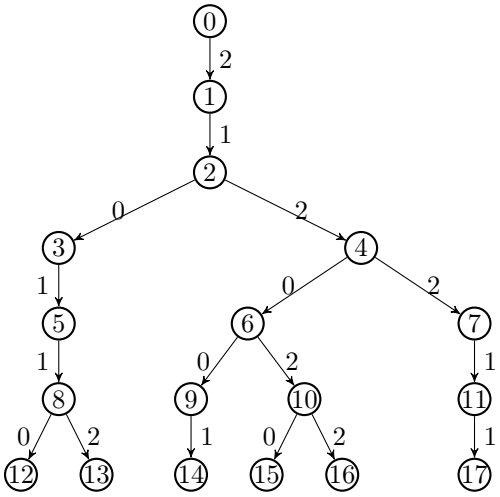

FIGURE 2 – Les premiers niveaux de l'arbre du langage $L_{\frac{3}{2}}$

Proposition 2.4.9. *La signature de $L_{\frac{p}{q}}$ est périodique de période q et la somme des q éléments d'une période vaut p.*

D'après la figure 2, la signature de $L_{\frac{3}{2}}$ semble être $(21)^\omega$, qui est bien périodique de période 2, et pour laquelle la somme des éléments de la période est 3.

Démonstration. Nous avons établi à la proposition 2.4.4 que si w représente l'entier N, alors $wa \in L_{\frac{p}{q}} \Leftrightarrow q \mid pN+a$. On a donc, en omettant les indices pour la clarté des notations,

$$\begin{aligned}\langle N\rangle a \in L &\Leftrightarrow q \mid pN + a \\ &\Leftrightarrow q \mid p(N+q) + a \\ &\Leftrightarrow \langle N+q\rangle a \in L\end{aligned}$$

d'où la périodicité de période q de la signature du langage $L_{\frac{p}{q}}$. Calculons maintenant la somme des q valeurs de la période : on a

$$\begin{aligned}s_N + s_{N+1} + \ldots + s_{N+q-1} &= \sum_{i=0}^{q-1} |\{a \in \{0,\ldots,p-1\} : q \mid p(N+i)+a\}| \\ &= |\{n \in \{pN,\ldots,p(N+q)-1\} : q \mid n\}| \\ &= p\end{aligned}$$

et la conclusion est atteinte. □

Remarque 2.4.10. Cette preuve montre en fait un résultat plus fort : non seulement les représentations de N et $N+q$ peuvent être étendues à droite par le même nombre de chiffres, mais elles peuvent en fait être étendues à droite par les mêmes chiffres. Cela se voit sur la figure 2 où les étiquettes des arêtes elles-mêmes sont périodiques, de période $(0,2,1)$.

Ainsi, les p/q-numérations donnent des langages à signature périodique. En fait, on peut montrer qu'il s'agit même d'une caractérisation de ces numérations, mais la preuve dépasse le cadre de ce mémoire. Nous renvoyons le lecteur à ([18], théorème 32) pour les détails. Après les numérations en base entière, qui ont une signature constante, cette famille de langages est donc "la plus simple" du point de vue de la signature. Cependant, cette propriété sur la signature ne se traduit pas facilement en termes de rationalité, ni même d'algébricité du langage.

Ceci termine notre tour d'horizon de différentes familles de systèmes de numération. Dans la suite de ce mémoire, nous examinerons différentes techniques permettant de déterminer l'existence ou non de la propagation de retenue d'un système de numération, et nous pourrons comparer ces techniques par les résultats qu'elles permettent d'obtenir sur les différents exemples ci-dessus.

3 Approche combinatoire du problème

Dans cette section, nous introduisons formellement les objets liés au problème de la propagation de retenue ainsi que l'arbre d'un langage. Ensuite, nous utilisons un argument combinatoire pour réécrire la propagation de retenue sous une forme plus facilement utilisable.Nous utilisons cette forme pour obtenir quelques résultats généraux, et pour montrer l'existence de la propagation de retenue dans le cas des langages à signature périodique. Ce cas inclut notamment le cas des bases entières et celui des p/q-numérations.

3.1 Présentation du problème

Notre objectif est de déterminer, étant donné un système de numération fixé, "combien de chiffres changent en moyenne entre la représentation d'un nombre et celle de son successeur". Les définitions suivantes permettent de transcrire mathématiquement cette question.

Définition 3.1.1. Soient u et v deux mots de $A^\star$. On note $u \wedge v$ le plus long préfixe commun de ces deux mots. Si $|u| = |v|$, on note

$$\Delta(u, v) = |u| - |u \wedge v| = |v| - |u \wedge v|\,.$$

Si $|u| \neq |v|$, on note

$$\Delta(u, v) = \max\{|u|\,, |v|\}.$$

L'objectif de cette fonction Δ est de mesurer "le nombre de chiffres qui changent" à droite lorsqu'on passe de u à v. On a par exemple $\Delta(98, 99) = 1$, $\Delta(99, 100) = 3$ ou encore $\Delta(0111, 1111) = 4$.

Si L est un système de numération (représenté par son langage associé), on peut alors définir la *propagation de retenue* en un mot de L (ou, de manière équivalente, en un naturel).

Définition 3.1.2. Si A est un alphabet ordonné et L est un système de numération sur A, la *propagation de retenue* dans L d'un naturel i est la quantité

$$\mathrm{cp}_L(i) = \Delta(\langle i\rangle_L, \langle i+1\rangle_L),$$

le nombre de chiffres qui changent entre la représentation de i et celle de $i+1$. On notera aussi $\mathrm{cp}_L(w) = \mathrm{cp}_L(\pi_L(w))$ si w est dans le langage L. Remarquons que $\langle \pi_L(w) + 1\rangle_L$ n'est autre que le mot suivant w dans L dans l'ordre radiciel. On peut abandonner l'indice L si le contexte est clair.

Exemple 3.1.3. Dans la base 10 usuelle, on a $\mathrm{cp}(98) = \Delta(98, 99) = 1$ et $\mathrm{cp}(99) = \Delta(99, 100) = 3$. En base Fibonacci, on a $\mathrm{cp}(32) = \Delta(1010100, 1010101) = 1$ et $\mathrm{cp}(33) = \Delta(1010101, 10000000) = 8$.

On peut alors considérer la "valeur moyenne" de cette propagation de retenue, définie comme suit.

Définition 3.1.4. Soit L un système de numération sur A. On note $\mathrm{scp}_L(N)$ la quantité

$$\mathrm{scp}_L(N) = \sum_{i=0}^{N-1} \mathrm{cp}_L(i).$$

La propagation de retenue d'un langage L, notée CP_L, est la limite, si elle existe, de la moyenne de la propagation de retenue aux n premiers mots du langage, quand n croît vers l'infini :

$$\mathrm{CP}_L = \lim_{N\to\infty} \frac{1}{N} \sum_{i=0}^{N-1} \mathrm{cp}_L(i) \text{ si cette quantité existe.}$$

Nous verrons dans la suite que montrer l'existence de cette propagation de retenue est souvent la partie difficile, le calcul exact est facilement réalisable. Dans la suite de ce mémoire, nous exposons plusieurs approches permettant chacune de déterminer l'existence et la valeur de CP_L pour différentes catégories de langages. Avant d'introduire notre premier outil, donnons un exemple de calcul.

Exemple 3.1.5. Considérons la base 10 usuelle. On peut tenir le raisonnement intuitif suivant. Quand on passe de la représentation d'un nombre à celle de son successeur, le chiffre des unités change à chaque fois, le chiffre des dizaines change une fois sur 10, celui des centaines une fois sur 100...On s'attend donc à ce que la propagation de retenue du langage soit de

$$1 + \frac{1}{10} + \frac{1}{100} + \ldots = \sum_{i=0}^{\infty} (\frac{1}{10})^i = \frac{1}{1 - \frac{1}{10}} = \frac{10}{9}.$$

C'est ce que nous allons vérifier formellement.

Soit N un nombre représenté par $w_k \cdots w_0$. Comme ci-dessus, on peut considérer les contributions de chaque position dans la propagation de retenue plutôt que celles de chaque nombre.

Si l'on définit $\delta(u,v) = \{0, \ldots, |\Delta(u,v)| - 1\}$ l'ensemble des positions où la retenue est propagée entre u et v, on a

$$\begin{aligned}\mathrm{scp}_L(N) &= \sum_{i=0}^{N-1} \Delta(\langle i\rangle_L, \langle i+1\rangle_L) \\ &= \sum_{i=0}^{N-1} \sum_{j\in\{0,\ldots,N\}} \chi_{\delta(\langle i\rangle_L,\langle i+1\rangle_L)}(j) \\ &= \sum_{j\in\{0,\ldots,N\}} \sum_{i=0}^{N-1} \chi_{\delta(\langle i\rangle_L,\langle i+1\rangle_L)}(j)\end{aligned}$$

où la somme sur j est finie car on peut borner la longueur de la représentation d'un nombre par ce nombre dès que le langage est clos par préfixe.

On remarque alors que $\sum_{i=0}^{N-1} \chi_{\delta(\langle i\rangle_L,\langle i+1\rangle_L)}(j) = \left\lfloor \frac{N}{10^j} \right\rfloor$ puisque le chiffre à la position j change exactement quand N vaut -1 modulo 10^j. La quantité dont on doit déterminer la limite est donc

$$\frac{1}{N} \sum_{j=0}^{N} \left\lfloor \frac{N}{10^j} \right\rfloor,$$

où la somme devrait être infinie, mais où tous les termes suivants valent 0 vu la remarque ci-dessus. En fait, cette somme vaut même

$$\frac{1}{N} \sum_{j=0}^{\lfloor \log_{10}(N)\rfloor+1} \left\lfloor \frac{N}{10^j} \right\rfloor$$

où la borne supérieure de la somme n'est autre que le nombre de chiffres dans la représentation de N. Cela fait, on peut encadrer $\left\lfloor \frac{N}{10^j} \right\rfloor$ par $\frac{N}{10^j} - 1$ et $\frac{N}{10^j}$ et on a alors l'encadrement suivant pour $\mathrm{scp}_L(N)$:

$$\begin{aligned} \tfrac{1}{N}\textstyle\sum_{j=0}^{\lfloor \log_{10}(N)\rfloor+1}(\tfrac{N}{10^j} - 1) \quad &\leq \mathrm{scp}_L(N) \leq \quad \tfrac{1}{N}\textstyle\sum_{j=0}^{\lfloor \log_{10}(N)\rfloor+1} \tfrac{N}{10^j} \\ (\textstyle\sum_{j=0}^{\lfloor \log_{10}(N)\rfloor+1} \tfrac{1}{10^j}) - \tfrac{\lfloor \log_{10}(N)\rfloor+1}{N} \quad &\leq \mathrm{scp}_L(N) \leq \quad \textstyle\sum_{j=0}^{\lfloor \log_{10}(N)\rfloor+1} \tfrac{1}{10^j}. \end{aligned}$$

Quand $N \to \infty$, on a $\lfloor \log_{10}(N)\rfloor + 1 \to \infty$ et $\frac{\lfloor \log_{10}(N)\rfloor+1}{N} \to 0$. Les deux bornes de l'encadrement tendent donc vers $\frac{10}{9}$. Dès lors, $\mathrm{CP}_L = \lim_{N\to\infty} \mathrm{scp}_L(N)$ existe et vaut $\frac{10}{9}$, comme prévu.

Dans cet exemple, nous avons procédé en changeant les groupements dans le calcul de $\mathrm{scp}_L(N)$, que nous avons calculé en examinant la contribution de chaque position à la somme, plutôt que celle de chaque nombre. Dans la section suivante, nous introduisons l'arbre du langage. L'objectif de cette définition est de grouper les contributions à $\mathrm{scp}_L(N)$ par longueur, car il est possible de réexprimer plus facilement la somme des propagations de retenue des premiers mots d'une longueur donnée. Par exemple, la somme des propagations de retenue de 10 à un nombre entre 10 et 99 en base 10, ou de 21 à un nombre entre 21 et 34 en base Fibonacci, peut être exprimée en termes de l'arbre du langage.

3.2 L'arbre du langage

Définition 3.2.1. Soit L un langage clos par préfixe. L'arbre du langage L est un arbre associé au langage L, noté $\mathcal{T}_L$, dont les sommets sont les mots de L (ou de manière équivalente, les naturels) et où une arête relie les mots w et wa pour tous $w, wa \in L, a \in A$.

Cet arbre est enraciné en ε. Chaque noeud a autant de fils que le nombre de façons d'étendre le mot correspondant à droite. Si le langage est extensible à droite, l'arbre n'a donc pas de feuilles. On peut ordonner les fils d'un noeud en s'aidant de l'ordre de A. Dans ce cas, un parcours en largeur de $\mathcal{T}_L$, "étage par étage puis de gauche à droite", revient à énumérer les mots de L par ordre radiciel.

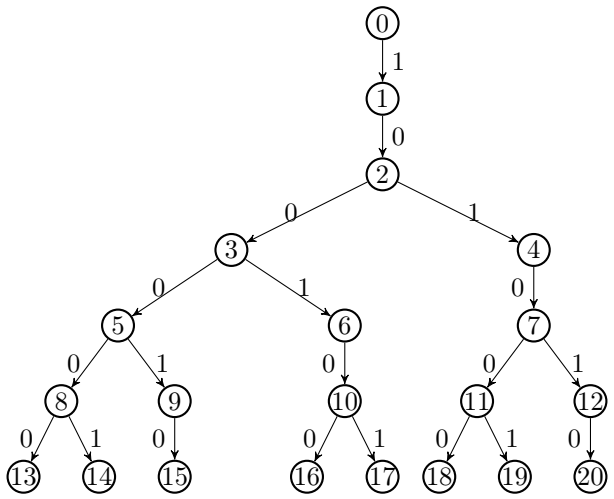

FIGURE 3 – Les premiers niveaux de l'arbre du langage de la numération de Fibonacci.

Exemple 3.2.2. La figure 3 représente les premiers niveaux de l'arbre de la numération de Fibonacci. Sur cet arbre, on voit par exemple que la représentation de 6 est 1001, et qu'elle ne peut être étendue à droite que par 0 si l'on veut garder une représentation du langage (qui sera la représentation de 10).

On introduit maintenant les notions de rives gauche et droite d'un mot dans l'arbre. La proposition 3.2.7 montrera l'utilité de ces notions.

Si $\mathcal{T}_L$ est l'arbre du langage L, soit w un mot de L de longueur l. On note $\mathcal{T}_L^{(l)}$ la partie de $\mathcal{T}_L$ formée des mots de longueur inférieure ou égale à l. Si $\mathrm{Pre}(w)$, l'ensemble des préfixes de w, est vu comme une rivière coulant de w à ε, la définition de la rive gauche et de la rive droite sont naturelles.

Définition 3.2.3.

— La rive gauche de w, notée $\mathrm{LB}_L(w)$ (pour *left bank*), est

$$\mathrm{LB}_L(w) = \{u \in L : |u| \leq |w| \text{ et } u \preccurlyeq w\} \setminus \mathrm{Pre}(w).$$

— La rive droite de w, notée $\mathrm{RB}_L(w)$ (pour *right bank*), est

$$\mathrm{RB}_L(w) = \{u \in L : |u| \leq |w| \text{ et } u \succ w\}.$$

Exemple 3.2.4. la figure 4 représente $\mathcal{T}_L^{(6)}$ et montre en vert les mots de la rive gauche de 100101 et en rouge ceux de la rive droite. Pour la clarté du dessin, chaque mot a été identifié au noeud de l'entier qu'il représente. Par exemple, le mot 10100 est identifié au noeud 11 de l'arbre. On remarque qu'en général $\mathrm{LB}_L(w), \mathrm{Pre}(w)$ et $\mathrm{RB}_L(w)$ partitionnent $\mathcal{T}_L^{(l)}$.

Introduisons les dernières notations avant d'atteindre la proposition 3.2.7. Celles-ci concernent le nombre de mots de chaque longueur dans le langage de la numération.

Définition 3.2.5. On note $\mathbf{u}_L(l)$ le nombre de mots de L de longueur l et $\mathbf{v}_L(l)$ le nombre de mots de L de longueur au plus l,

$$\mathbf{u}_L(l) = \left|L \cap A^l\right| \text{ et } \mathbf{v}_L(l) = \left|L \cap A^{\leq l}\right|.$$

On note $\mathrm{Maxlg}(L)$ l'ensemble des mots radiciellement maximaux de L pour chaque longueur,

$$\mathrm{Maxlg}(L) = \{w \in L : u \in L, |u| = |w| \Rightarrow u \sqsubseteq w\}.$$

Exemple 3.2.6. Pour la numération de Fibonacci, la suite $\mathbf{u}_L$ commence par $1, 1, 1, 2, 3, 5, 8, \ldots$ et la suite $\mathbf{v}_L$ par $1, 2, 3, 5, 8, 13, 21, \ldots$. L'ensemble $\mathrm{Maxlg}(L)$ est l'ensemble $\{0, 1, 2, 4, 7, 12, 20, \ldots\}$.

Comme les mots de L sont énumérés par ordre radiciel, le premier mot de longueur l est la représentation du nombre de mots de longueur inférieure à l. On a donc

$$\mathrm{Maxlg}(L) = \{\langle \mathbf{v}_L(l) \rangle_L : l \in \mathbb{N}\}$$

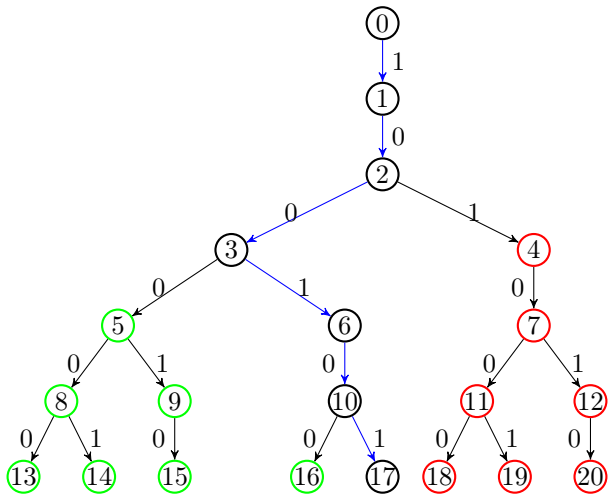

FIGURE 4 – La rive gauche (vert) et la rive droite (rouge) du mot 100101 dans la numération de Fibonacci.

et

$$L \cap A^l = \{u \in L : \mathbf{v}_L(l-1) \leq \pi_L(u) < \mathbf{v}_L(l).\}$$

Remarquons que les suites $\mathbf{u}_L$ et $\mathbf{v}_L$ sont de la bonne forme pour pouvoir utiliser la proposition B.2, puisque

$$\mathbf{v}_L(l) = \sum_{i=0}^{l} \mathbf{u}_L(i).$$

Nous sommes maintenant prêts pour la première proposition majeure, qui permet de calculer la somme des premières propagations de retenue au sein des mots d'une même longueur, en s'aidant de l'arbre du langage. La figure 5 nous montre graphiquement ce que nous allons démontrer. On y remarque que

$$\mathrm{cp}_L(13) = \Delta(100000, 100001) = 1, \mathrm{cp}_L(14) = 2, \mathrm{cp}_L(15) = 3, \mathrm{cp}_L(16) = 1.$$

Dès lors,

$$\sum_{i=13}^{16} \mathrm{cp}_L(i) = 1 + 2 + 3 + 1 = 7 = |\mathrm{LB}_L(17)|\,.$$

Ce résultat reste vrai en général comme nous le montrons maintenant.

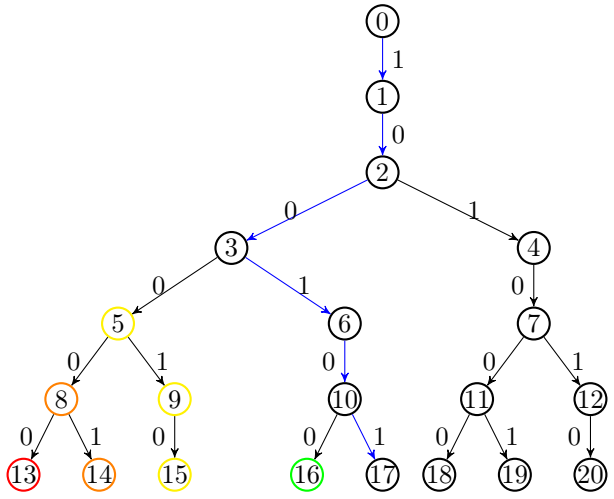

FIGURE 5 – Une partition de la rive gauche du mot 100101 dans la numération de Fibonacci.

Théorème 3.2.7. *Si L est un langage PCE, si u est la représentation de N et est de longueur l, alors on a*

$$\sum_{i=\mathbf{v}_L(l-1)}^{N} \mathrm{cp}_L(i) = \begin{cases} |\mathrm{LB}_L(\langle N+1\rangle_L)| & \text{si } u \notin \mathrm{Maxlg} \\ \mathbf{v}_L(l) & \text{si } u \in \mathrm{Maxlg} \end{cases}$$

Démonstration. Notons $v = \langle N+1\rangle_L$. L'idée clé est la suivante : si $u \notin \mathrm{Maxlg}$, alors

$$\mathrm{LB}_L(v) \setminus \mathrm{LB}_L(u) = \mathrm{Pre}(u) \setminus \mathrm{Pre}(v).$$

L'inclusion $\supseteq$ est claire (si $w \in \mathrm{Pre}(u)$, $w \preccurlyeq u \prec v$). Pour l'autre sens, si $w \in \mathrm{LB}_L(v) \setminus \mathrm{LB}_L(u)$, alors $w \notin \mathrm{Pre}(v)$, $|w| \leq |v|$ et $w \prec v$. Si $w \notin \mathrm{Pre}(u)$, on aurait $u \prec w$. Comme L est extensible à droite, il doit exister un mot de longueur l qui a w comme préfixe. Un tel mot x vérifie $u \prec w \prec x$ et $x \prec v$, donc $u \sqsubset x \sqsubset v$, ce qui contredit que v est le successeur immédiat de u dans l'ordre radiciel parmi les mots de L.

Une fois ceci acquis, on procède par récurrence sur N. Commençons par l'initialisation $N = \mathbf{v}_L(l-1)$. Si $u \in \mathrm{Maxlg}$, cela signifie que L ne contient qu'un mot de chaque longueur inférieure ou égale à l. Dans ce cas, on a $\mathrm{cp}_L(N) = l+1$ et $\mathbf{v}_L(l) = l+1$, comme voulu. Sinon, on a

$$\begin{aligned} \mathrm{cp}_L(N) &= \Delta(u,v) \\ &= |\mathrm{Pre}(u) \setminus \mathrm{Pre}(v)| \\ &= |\mathrm{LB}_L(v) \setminus \mathrm{LB}_L(u)| \\ &= |\mathrm{LB}_L(v)|\,. \end{aligned}$$

Dans ce cas aussi, le résultat est vérifié.

Si maintenant $\mathbf{v}_L(l-1) < N < \mathbf{v}_L(l) - 1$, on a

$$\begin{aligned} \sum_{i=\mathbf{v}_L(l-1)}^{N} \mathrm{cp}_L(i) &= \sum_{i=\mathbf{v}_L(l-1)}^{N-1} \mathrm{cp}_L(i) + \mathrm{cp}_L(N) \\ &= |\mathrm{LB}_L(u)| + |\mathrm{Pre}(u) \setminus \mathrm{Pre}(v)| \\ &= |\mathrm{LB}_L(u)| + |\mathrm{LB}_L(v) \setminus \mathrm{LB}_L(u)| \\ &= |\mathrm{LB}_L(v)|\,, \end{aligned}$$

à nouveau ce qui était désiré.

Enfin, si $N = \mathbf{v}_L(l) - 1$, on a $u \in \mathrm{Maxlg}$ et $\mathrm{LB}_L(u)$ et $\mathrm{Pre}(u)$ partitionnent $\mathcal{T}_L^{(l)}$, et on a

$$\begin{aligned} \sum_{i=\mathbf{v}_L(l-1)}^{N} \mathrm{cp}_L(i) &= \sum_{i=\mathbf{v}_L(l-1)}^{N-1} \mathrm{cp}_L(i) + \mathrm{cp}_L(N) \\ &= |\mathrm{LB}_L(u)| + l + 1 \\ &= |\mathrm{LB}_L(u)| + |\mathrm{Pre}(u)| \\ &= \mathbf{v}_L(l). \end{aligned}$$

Le résultat est donc démontré. □

Remarque 3.2.8. L'hypothèse que le langage est clos par préfixe et extensible à droite est cruciale pour ce résultat, comme on le voit sur la figure 6. A gauche, le mot 0 ne fait pas partie du langage et ce langage n'est donc pas clos par préfixe. On a alors

$$\mathrm{cp}_L(2) = \Delta(00,11) = 2 > 1 = |\mathrm{LB}_L(11)|\,.$$

A droite, le mot 1 ne peut pas être étendu à droite en restant dans le langage. On a alors

$$\mathrm{cp}_L(4) = \Delta(00,22) = 2 < 3 = |\mathrm{LB}_L(22)|\,.$$

Dans les deux cas, le théorème 3.2.7 cesse de s'appliquer.

On en déduit des formules pour calculer la somme des premières valeurs de la propagation de retenue.

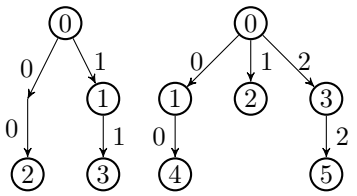

FIGURE 6 – Des arbres d'un langage non-clos par préfixe (gauche) et non-extensible à droite (droite).

Corollaire 3.2.9. *Pour tout entier l,*

$$\sum_{\substack{w\in L\\ |w|=l}} \mathrm{cp}_L(w) = \mathbf{v}_L(l),$$

$$\sum_{i=0}^{\mathbf{v}_L(l)-1} \mathrm{cp}_L(i) = \sum_{i=0}^{l} \mathbf{v}_L(i).$$

De plus, pour tout mot $u \in L$ de longueur l, si $N = \pi_L(u)$,

$$\mathrm{scp}_L(N) = \sum_{i=0}^{l-1} \mathbf{v}_L(i) + |\mathrm{LB}_L(u)|\,.$$

Exemple 3.2.10. Pour la numération de Fibonacci, on a

$$\sum_{i=13}^{20} \mathrm{cp}_L(i) = \mathbf{v}_L(6) = 21 \text{ et } \mathrm{scp}_L(17) = \sum_{l=0}^{5} \mathbf{v}_L(l) + |\mathrm{LB}_L(100101)| = (1+2+3+5+8+13)+7 = 39.$$

Au vu de la forme particulière lorsque N correspond au dernier mot d'une longueur donnée, on peut extraire de la suite des $(\frac{\mathrm{scp}_L(N)}{N})_N$ la sous-suite correspondante :

Définition 3.2.11. On appelle *propagation de retenue filtrée*, et on note FCP_L, la quantité

$$\mathrm{FCP}_L = \lim_{l\to\infty} \frac{\mathrm{scp}_L(\mathbf{v}_L(l))}{\mathbf{v}_L(l)} \text{ si cette limite existe.}$$

On a également

$$\mathrm{FCP}_L = \lim_{l\to\infty} \frac{\sum_{i=0}^{l} \mathbf{v}_L(i)}{\mathbf{v}_L(l)}. \tag{3}$$

Bien sûr, comme nous avons procédé par extraction d'une sous-suite, l'existence de CP_L implique celle de FCP_L, et ces deux quantités sont alors égales, mais l'existence de FCP_L n'implique pas celle de CP_L comme nous le verrons dans l'exemple 3.3.10.

Nous pouvons maintenant revenir à l'exemple de la base entière pour résoudre à nouveau le problème, en pensant cette fois en terme d'arbres.

Exemple 3.2.12. Soient p l'entier de base et N un entier représenté par le mot u, de longueur l. On a $\mathbf{v}_L(l-1) \leq N < \mathbf{v}_L(l)$ et

$$\mathrm{scp}_L(N) = \sum_{i=0}^{l-1} \mathbf{v}_L(i) + |\mathrm{LB}_L(u)|\,.$$

On a $\mathbf{v}_L(l) = p^l$ et $\frac{\mathbf{v}_L(l+1)}{\mathbf{v}_L(l)} = p$. De là, le lemme B.2 donne

$$\lim_{l\to\infty} \frac{\sum_{i=0}^{l-1} \mathbf{v}_L(i)}{\mathbf{v}_L(l-1)} = \frac{p}{p-1}.$$

c'est-à-dire

$$\frac{\sum_{i=0}^{l-1} \mathbf{v}_L(i)}{\mathbf{v}_L(l-1)} - \frac{p}{p-1} = \varepsilon(l) \text{ où } \varepsilon(l) \to 0 \text{ si } l \to \infty.$$

ou encore

$$\sum_{i=0}^{l-1} \mathbf{v}_L(i) = (\mathbf{v}_L(l-1)) \left(\frac{p}{p-1} + \varepsilon(l) \right) \text{ où } \varepsilon(l) \to 0 \text{ si } l \to \infty.$$

Maintenant, évaluons $|\mathrm{LB}_L(u)|$. Le cas où $u \in \mathrm{Maxlg}$ est recouvert par le cas précédent, dans lequel on a

$$\mathrm{scp}_L(N) = \sum_{i=0}^{l} \mathbf{v}_L(i).$$

Posons $N = \mathbf{v}_L(l-1) + M$ avec M entre 1 inclus et $\mathbf{u}_L(l)$ exclus. On a $\left|\mathrm{LB}_L(u) \cap A^l\right| = M$. Comme chaque noeud interne de $\mathcal{T}_L$ a p fils à la hauteur suivante, on trouve

$$\left|\mathrm{LB}_L(u) \cap A^{l-1}\right| = \left\lfloor \frac{M}{p} \right\rfloor$$

puis

$$\left|\mathrm{LB}_L(u) \cap A^{l-k}\right| = \left\lfloor \frac{M}{p^k} \right\rfloor$$

en procédant par induction sur $k \in \{1, \ldots, l\}$. On utilise à nouveau l'encadrement $x - 1 < \lfloor x \rfloor \leq x$ pour obtenir

$$\sum_{k=0}^{l} \left\lfloor \frac{M}{p^k} \right\rfloor \geq \sum_{k=0}^{l} \frac{M}{p^k} - l = M \frac{p}{p-1} - \frac{M}{p^l(p-1)} - l$$

et de même $\sum_{k=0}^{l} \left\lfloor \frac{M}{p^k} \right\rfloor \leq M \frac{p}{p-1} - \frac{M}{p^l(p-1)}$. Comme $M < p^l$, on a que $-\frac{M}{p^l(p-1)} \geq -1$ et on a finalement l'encadrement

$$M \frac{p}{p-1} - l - 1 \leq |\mathrm{LB}_L(u)| \leq M \frac{p}{p-1}.$$

Ainsi, on a

$$M \frac{p}{p-1} - l - 1 + (\mathbf{v}_L(l-1)) \left(\frac{p}{p-1} + \varepsilon(l) \right) \leq \mathrm{scp}_L(N) \leq M \frac{p}{p-1} + (\mathbf{v}_L(l-1)) \left(\frac{p}{p-1} + \varepsilon(l) \right).$$

Comme $N = M + \mathbf{v}_L(l-1)$, en divisant par N et en passant à la limite pour $N \to \infty$, les deux bornes tendent vers $\frac{p}{p-1}$, et on retrouve le résultat déjà obtenu à l'exemple 3.1.5.

Nous verrons à la section 3.4 comment étendre le raisonnement ci-dessus au cas de la p/q-numération. Là s'arrêtent les résultats d'existence que l'on peut obtenir avec des arguments élémentaires (ne requérant pas de connaissances théoriques supplémentaires). Dans la section 4, nous réutiliserons le théorème 3.2.7 dans le cadre plus restreint des langages rationnels, pour lesquels nous pourrons mieux évaluer le comportement de $\mathrm{LB}_L(u)$ en général.

3.3 Mesures de croissance et propagation de retenue

Nous avons vu en 3.2.11 que la propagation de retenue filtrée était liée aux suites $\mathbf{u}_L$ et $\mathbf{v}_L$. Nous pouvons alors étudier FCP_L, donc CP_L, en fonction du simple nombre de mots dans le langage plutôt que du langage complet. Dans cette section, nous introduisons deux mesures de croissance d'un langage, qui nous donnent de l'information sur ces suites $\mathbf{u}_L$ et $\mathbf{v}_L$. On obtient alors des résultats généraux, qui donnent la valeur de CP_L quand elle existe (corollaire 3.3.9), et permettent parfois de conclure quand elle n'existe pas. Néanmoins, ces résultats ramenant un langage à la quantité de mots qu'il contient perdent trop d'information pour pouvoir certifier l'existence de la propagation de retenue d'un langage (exemple 3.3.10).

Nous commençons par introduire les mesures de croissance du nombre de mots d'un langage que nous allons utiliser. On pourra consulter [24] pour d'autres utilisations de ces mesures.

Définition 3.3.1. Le *taux de croissance global* d'un langage L est la quantité

$$\eta_L = \limsup_{l \to \infty} \sqrt[l]{\mathbf{u}_L(l)}.$$

Le *taux de croissance local* de L est la quantité

$$\gamma_L = \lim_{l \to \infty} \frac{\mathbf{u}_L(l+1)}{\mathbf{u}_L(l)} \text{ si cette limite existe.}$$

Remarquons que la quantité η_L existe toujours puisqu'elle est définie par une limite supérieure. Elle est plus générale que γ, et en est une extension.

Proposition 3.3.2. *Si γ_L existe, alors $\eta_L = \gamma_L$.*

Démonstration. Soit $\varepsilon > 0$. Si γ_L existe, il existe N naturel tel que pour tout $n > N$, on a

$$(\gamma_L - \varepsilon)\mathbf{u}_L(n) < \mathbf{u}_L(n+1) < (\gamma_L + \varepsilon)\mathbf{u}_L(n)$$

et donc

$$\mathbf{u}_L(N)(\gamma_L - \varepsilon)^{n-N} < \mathbf{u}_L(n) < \mathbf{u}_L(N)(\gamma_L + \varepsilon)^{n-N}.$$

Dès lors,

$$\sqrt[n]{\frac{\mathbf{u}_L(N)}{(\gamma_L-\varepsilon)^N}}(\gamma_L - \varepsilon) < \sqrt[n]{\mathbf{u}_L(n)} < \sqrt[n]{\frac{\mathbf{u}_L(N)}{(\gamma_L+\varepsilon)^N}}(\gamma_L + \varepsilon).$$

Lorsque $n \to \infty$, les deux bornes tendent vers $\gamma_L - \varepsilon$ et $\gamma_L + \varepsilon$ respectivement. On a donc

$$\gamma_L - \varepsilon < \eta_L < \gamma_L + \varepsilon$$

pour tout ε, donc $\eta_L = \gamma_L$. □

Définition 3.3.3. Lorsque $\eta_L > 1$, on parle de *langage à croissance exponentielle*, et de *langage à croissance polynomiale* s'il existe un polynôme P tel que $\mathbf{u}_L(l) \leq P(l)$ pour tout l suffisamment grand.

Proposition 3.3.4. *Il existe un langage qui n'est ni à croissance exponentielle ni à croissance polynomiale.*
Tout langage rationnel est à croissance exponentielle ou à croissance polynomiale.

Démonstration. Pour le premier point, on peut construire un langage tel que $\left|\mathbf{u}_L(l) - l^{\log l}\right| \leq 1$. En effet, on a

$$\lim_{l\to\infty} \sqrt[l]{l^{\log l}} = \lim_{l\to\infty} e^{\frac{(\log l)^2}{l}} = 1,$$

et donc le nombre de mots de longueur l de $\{0,1\}^\star$ croît bien plus vite que $l^{\log l}$. Un tel langage n'est alors pas à croissance exponentielle car on a

$$\lim_{l\to\infty} \sqrt[l]{\mathbf{u}_L(l)} = \lim_{l\to\infty} \sqrt[l]{l^{\log l}} = 1.$$

Il n'est pas à croissance polynomiale non plus puisque $l^{\log l}$ croît plus vite que n'importe quel polynôme.

Pour la deuxième partie de l'énoncé, il nous faut prendre de l'avance et recourir à la proposition 4.2.8. On a que si L est un langage rationnel, alors

$$\mathbf{u}_L(l) = \sum_{j=1}^{k} \lambda_j^l P_j(l)$$

où $\lambda_1, \ldots, \lambda_k$ sont des constantes complexes de la forme $re^{i\theta}$ où $e^{i\theta}$ est une racine de l'unité et P_j est un polynôme.

Dès lors, soit $\lambda = \max_{j=1}^{k} |\lambda_j|$. Si $\lambda > 1$, le langage est à croissance exponentielle et $\eta_L = \lambda$. Si $\lambda = 1$, le langage est à croissance polynomiale. Le cas $\lambda < 1$ est impossible car la suite $\mathbf{u}_L(l)$ est croissante en fonction de l. □

Ces deux familles de langages partitionnent donc les langages rationnels. Dans chaque cas, on peut déduire une information sur les valeurs potentielles de FCP_L.

Proposition 3.3.5. *Si L est un langage PCE tel que $\mathbf{u}_L(l) = P(l)$ où P est un polynôme de degré d, alors CP_L n'existe pas.*

Démonstration. Le coeur de ce résultat est la formule de Faulhaber, démontrée par Jacques Bernoulli dans [3] et dont une version moderne est donnée par Foata [9] :

$$\sum_{i=1}^{n} i^d = \frac{n^{d+1}}{d+1} + \frac{n^d}{2} + \frac{1}{d+1}\sum_{j=2}^{d} \binom{d+1}{j} B_j n^{d+1-j},$$

où les B_j sont des constantes rationnelles appelées nombres de Bernoulli. L'important dans ce résultat est que $\sum_{i=1}^{n} i^d$ s'exprime comme un polynôme P_{d+1} de degré $d+1$ en n.

Dès lors, si $\mathbf{u}_L(l) = P(l)$ avec P un polynôme de degré d de coefficients $a_d, \dots, a_0$, on a

$$\begin{aligned}\mathbf{v}_L(l) &= \sum_{i=1}^{l} \mathbf{u}_L(i) \\ &= \sum_{i=1}^{l} \sum_{j=0}^{d} a_j i^j \\ &= \sum_{j=0}^{d} a_j P_{j+1}(l)\end{aligned}$$

et $\mathbf{v}_L(l)$ est donc également un polynôme de degré $d+1$ en l. De même, on déduit que $\sum_{i=0}^{l} \mathbf{v}_L(i)$ est un polynôme de degré $d+2$ en l. De là, la formule (3) nous donne

$$\mathrm{FCP}_L = \lim_{l\to\infty} \frac{\sum_{i=0}^{l} \mathbf{v}_L(i)}{\mathbf{v}_L(l)} = +\infty.$$

Dès lors, FCP_L n'existe pas, et CP_L non plus. □

Remarque 3.3.6. Nous avons donné l'énoncé et la preuve dans le cas où $\mathbf{u}_L(l)$ est exactement un polynôme en l, mais il est possible de l'étendre au cas de n'importe quel langage à croissance polynomiale. En l'absence d'une formule exacte, il faut comparer le comportement asymptotique de $\mathbf{u}_L(l)$ à ceux de deux polynômes de degrés consécutifs. On obtient la même conclusion.

Exemple 3.3.7. Le système de numération abstrait basé sur le langage $a^\star b^\star$ est à croissance polynomiale, puisque $\mathbf{u}_L(l) = l + 1$. Ce système n'admet donc pas de propagation de retenue.

Proposition 3.3.8. *Si L est un langage PCE à croissance exponentielle, FCP_L existe si et seulement si γ_L existe, auquel cas $\mathrm{FCP}_L = \frac{\gamma_L}{\gamma_L - 1}$.*

Démonstration. Si γ_L existe et que le langage est à croissance exponentielle, on a $\gamma_L > 1$. Le lemme B.2 appliqué à $(\mathbf{u}_L(n))_n$ et $(\sum_{i=0}^{n} \mathbf{u}_L(i))_n = \mathbf{v}_L(n)$ donne

$$\frac{\mathbf{u}_L(n+1)}{\mathbf{u}_L(n)} \overset{n\to\infty}{\to} \gamma_L \Rightarrow \frac{\mathbf{v}_L(n+1)}{\mathbf{v}_L(n)} \overset{n\to\infty}{\to} \gamma_L$$

puis le même lemme appliqué à $(\mathbf{v}_L(n))_n$ et $(\sum_{i=0}^{n} \mathbf{v}_L(i))_n$ donne

$$\frac{\mathbf{v}_L(n+1)}{\mathbf{v}_L(n)} \overset{n\to\infty}{\to} \gamma_L \Rightarrow \frac{\sum_{i=0}^{n} \mathbf{v}_L(i)}{\mathbf{v}_L(n)} \overset{n\to\infty}{\to} \frac{\gamma_L}{\gamma_L - 1}$$

c'est-à-dire que $\mathrm{FCP}_L = \frac{\gamma_L}{\gamma_L - 1}$. L'autre implication s'obtient en utilisant le même lemme dans l'autre sens. □

Corollaire 3.3.9. *Si le langage L est PCE et à croissance exponentielle ou polynomiale, et si CP_L existe, alors γ_L existe et $\mathrm{CP}_L = \frac{\gamma_L}{\gamma_L - 1}$.*

Remarquons que les conditions de l'énoncé ci-dessus s'appliquent à tous les langages PCE rationnels.

Il est temps d'illustrer par un exemple que l'existence de FCP_L, et donc celle de γ_L, n'implique pas celle de CP_L.

Exemple 3.3.10. Le langage L PCE que nous allons construire est tel que son arbre est "déséquilibré", les noeuds de gauche ayant plus de fils que les noeuds de droite. Soit $A = \{a, b, c\}$, ordonné par ordre alphabétique. On désire avoir $\mathbf{u}_L(l) = 2^l$, et on construit $L \cap A^l$ par récurrence pour obtenir cela. On pose d'abord $L \cap A = \{a, c\}$. Si $L \cap A^l$ est défini, on note G_l l'ensemble des 2^{l-1} premiers mots de longueur l dans L, et H_l l'ensemble des 2^{l-1} derniers tels mots, de sorte que $L \cap A^l = G_l \cup H_l$, puis on pose

$$A^{l+1} \cap L = G_l\{a, b, c\} \cup H_l b.$$

qui contient bien 2^{l+1} mots.

L'arbre de ce langage est représenté sur la figure 7. On a $\mathbf{u}_L(l) = 2^l$ et $\mathbf{v}_L(l) = 2^{l+1} - 1$, et le taux de croissance du langage est 2 ainsi que sa propagation de retenue filtrée.

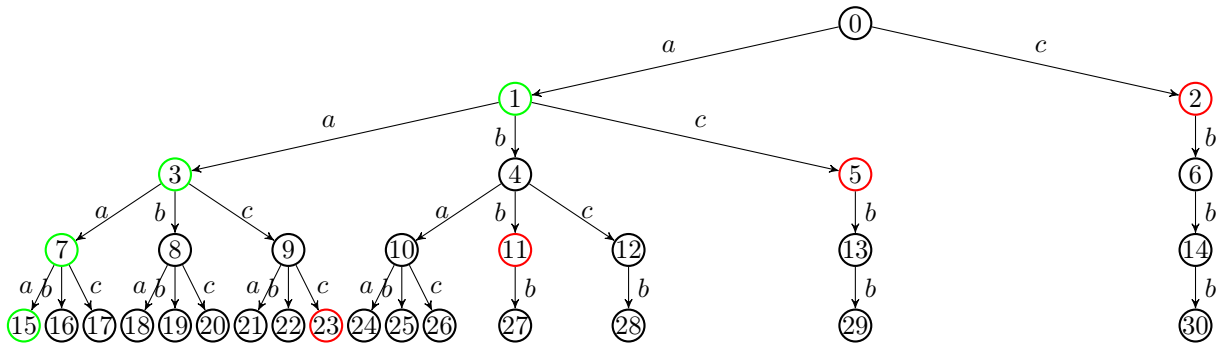

FIGURE 7 – Un langage ayant un arbre déséquilibré. En vert et en rouge, deux suites $k(N)$ telles que $\frac{\mathrm{scp}(k(N))}{k(N)}$ converge vers deux limites différentes le long de ces sous-suites.

Pour montrer que la propagation de retenue n'existe pas, on exhibe une autre sous-suite $k(N)$ telle que $\frac{\mathrm{scp}_L(k(N))}{k(N)}$ ne converge pas vers 2. Nous prenons $k(N) = 2^{N+1} - 1 + 2^N$ et calculons

$$\lim_{N\to\infty} \frac{\mathrm{scp}_L(k(N))}{k(N)}.$$

On rappelle que

$$\mathrm{scp}_L(N) = \sum_{i=0}^{l-1} \mathbf{v}_L(i) + |\mathrm{LB}_L(u)|$$

où l vaut ici $N+1$ et u est la représentation de $k(N)$. On a

$$\sum_{i=0}^{N} \mathbf{v}_L(i) = 2^{N+2} - N - 3,$$

évaluons maintenant $|\mathrm{LB}_L(u)|$.

On remarque que si un noeud est le premier de sa longueur, il est le fils du premier de la hauteur précédente, et que les autres noeuds qui ont trois fils, ce qui est le cas de tous ceux à gauche de u, sont eux-mêmes les fils d'un noeud ayant trois fils. De là, on imite le raisonnement de l'exemple 3.2.12. On a $\left|\mathrm{LB}_L(u) \cap A^{N+1}\right| = 2^N$, et tous les éléments à gauche de u dans l'arbre sont des fils d'un noeud en ayant 3, d'où $\left|\mathrm{LB}_L(u) \cap A^{N}\right| = \left\lfloor \frac{2^N}{3} \right\rfloor$. En répétant ce raisonnement, on trouve

$$\left|\mathrm{LB}_L(u) \cap A^{N+1-j}\right| = \left\lfloor \frac{2^N}{3^j} \right\rfloor, \quad \forall j \in \{0, \dots, N+1\},$$

le membre de droite valant bien sûr 0 pour des j suffisamment grands. De là, on déduit

$$|\mathrm{LB}_L(u)| \leq \sum_{j=0}^{N+1} \frac{2^N}{3^j} \leq (2^N)\frac{3}{2}.$$

En combinant les différentes équations, on trouve

$$\begin{aligned}
\frac{\mathrm{scp}_L(k(N))}{k(N)} &\leq \frac{1}{3.2^N - 1}(2^{N+2} - N - 3 + (2^N)\frac{3}{2}) \\
&\leq \frac{1}{3.2^N - 1}((4 + \frac{3}{2})2^N - N - 3) \\
&\overset{N\to\infty}{\to} \frac{11}{6} < 2.
\end{aligned}$$

et on a donc bien montré que la propagation de retenue du langage ne pouvait pas exister.

Remarque 3.3.11. Remarquons que ce langage L n'est pas rationnel, ce qui peut se montrer en recourant au lemme de la pompe. Nous donnons dans la suite un exemple de langage rationnel ayant un taux de croissance mais pas de propagation de retenue (4.1.2).

Montrons que L n'est pas rationnel. Pour cela, nous allons montrer que $L^c \cap a^\star b^\star a$ n'est pas rationnel en utilisant le lemme de la pompe (lemme A.14), et nous devons donc comprendre quand $a^n b^m$ est dans G_{n+m} (on a alors $a^n b^m a \in L$) et quand $a^n b^m$ est dans H_{n+m} (on a alors $a^n b^m a \notin L$). Remarquons que si w est le x-ième mot de longueur l de L par ordre radiciel, alors wb est le $3x-1$-ième mot de longueur $l+1$ de L si $w \in G_l$, et le $x+2^l$-ième si $w \in H_l$.

Pour tout n naturel, posons $f(n)$ le plus petit m tel que

$$\frac{3^m+1}{2} \geq 2^{m+n-1},$$

i.e. tel que

$$\left(\frac{3}{2}\right)^m + \frac{1}{2^m} \geq 2^n$$

et remarquons que $f(n)$ croît vers l'infini quand n augmente.

Le mot a^n est le premier par ordre radiciel parmi les mots de longueur n dans L. Dès lors, tant que $m < f(n)$, on a $a^n b^m \in G_{n+m}$, et on montre par récurrence que dans ce cas $a^n b^m$ est le $\frac{3^m+1}{2}$-ième mot de $L \cap A^{m+n}$ par ordre radiciel. D'autre part, $a^n b^{f(n)}$ est le $\frac{3^{f(n)}+1}{2}$-ième mot de $L \cap A^{f(n)+n}$ par ordre radiciel vu la récurrence ci-dessus, mais cela implique maintenant qu'il est dans $H_{n+f(n)}$. On peut alors montrer par récurrence que $a^n b^m$ est dans H_{m+n} pour tout m supérieur ou égal à $f(n)$.

Dès lors,

$$L^c \cap a^\star b^\star a = \{a^n b^m a : a^n b^m \in H_{n+m}\} = \{a^n b^m a : m \geq f(n)\}.$$

Supposons que ce langage soit rationnel et nommons N le naturel dont le lemme de la pompe assure l'existence. On sait qu'alors le mot $a^N b^{f(N)} a$ se factorise en xyz, de telle sorte que $y = a^j$ pour un naturel non nul j puisque les N premières lettres de $a^N b^{f(N)} a$ sont toutes a. Cela signifie qu'il existe des naturels arbitrairement grands M tels que $a^M b^{f(N)} a \in L^c \cap a^\star b^\star a$. Ceci est une contradiction, car on devrait avoir $f(N) \geq f(M)$, mais, pour M suffisamment grand, on a $f(N) < f(M)$, le membre de gauche étant fixé. Ainsi, le lemme de la pompe assure que $L^c \cap a^\star b^\star a$ n'est pas rationnel, donc L ne l'est pas non plus.

3.4 Langages à signature périodique

Dans cette sous-section, nous revenons sur la notion de signature d'un langage évoquée à la section 2.4 et étendons le raisonnement fait pour les bases entières (exemple 3.2.12) à tous les systèmes ayant une signature ultimement périodique.

Vu la preuve de la proposition 3.2.7, il est clair que la seule chose importante pour l'existence de la propagation de retenue du langage est la forme de l'arbre, pas son étiquetage (le nombre de lettres qui peuvent étendre à droite le n-ième mot du langage, pas quelles sont précisément ces lettres). Nous définissons la signature d'un arbre, qui correspond à la signature d'un langage définie plus haut.

Définition 3.4.1. Si $\mathcal{T}_L$ est un arbre enraciné en ε, on note $\mathcal{I}_L$ le graphe obtenu en ajoutant un arc de ε sur lui-même étiqueté par une lettre inférieure à toutes les lettres de A (on étend l'alphabet pour qu'il en existe une). $\mathcal{I}_L$ est appelé l'*i-arbre* associé à L, et dans la suite on confondra les objets $\mathcal{T}$ et $\mathcal{I}$. La signature d'un langage L est alors la suite des degrés (sortants) de chaque noeud de $\mathcal{I}_L$ (pour rappel, les sommets de l'arbre sont associés aux nombres naturels).

Proposition 3.4.2. *La signature d'un arbre associé à un langage PCE appartient à* $\{2,\dots,C\}\{1,\dots,C\}^{\mathbb{N}}$ *pour un* $C \in \mathbb{N}_{\geq 2}$. *Réciproquement, toute signature de cette forme est la signature d'un arbre associé à un langage PCE.*

Définition 3.4.3. Si $p > q \geq 1$ sont deux entiers, on appelle rythme de paramètres (q,p) un $q-$uple de naturels dont la somme fait p.

Une signature est ultimement périodique si elle s'écrit st^ω où s est une suite finie d'éléments de $\mathbb{N}$ et t est un rythme.

Remarque 3.4.4. Penchons-nous sur la condition $p > q$. Il est clair que $p \geq q$ est une condition nécessaire pour avoir la signature d'un langage PCE, puisque tous les termes de la signature doivent être au moins 1. Si $p = q$, nous

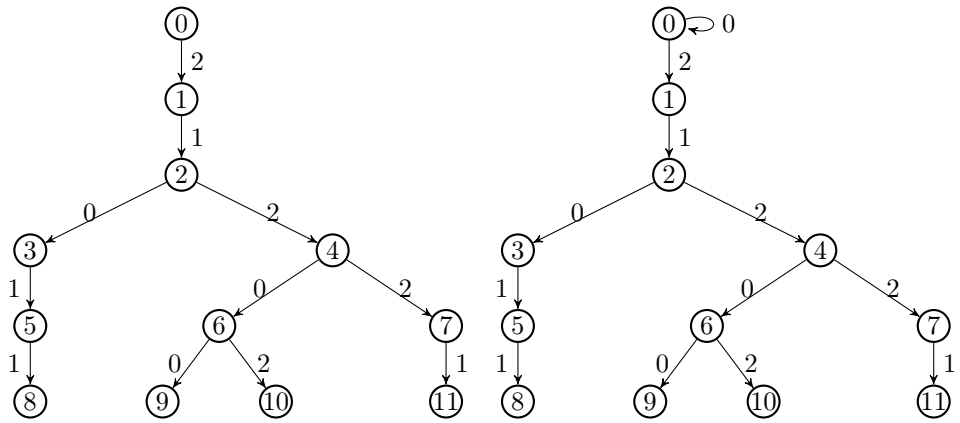

FIGURE 8 – Un arbre de signature 11212121 . . . et l'i-arbre associé, de signature 21212121 . . ..

montrons que $\gamma_L = 1$. Une fois que cela est fait, la propagation de retenue ne peut pas exister. Si elle existait, on aurait $\mathrm{CP}_L = \frac{\gamma_L}{\gamma_L - 1}$, ce qui est une contradiction.

Si $p = q$, dans la portion périodique de la signature, chaque noeud doit avoir exactement un fils si le langage associé est PCE, et on a alors $\mathbf{u}_L(l+1) = \mathbf{u}_L(l)$ dans la "partie périodique" de la signature, d'où la conclusion.

On rappelle que les $p/q-$numérations sont les seuls langages qui possèdent une signature strictement périodique, où $s = \varepsilon$ dans la définition ci-dessus (remarque 2.4.10 et fin de la section 2.4). Nous arrivons maintenant à l'extension annoncée.

Théorème 3.4.5. *Si un langage PCE a une signature périodique et que (q, p) est le paramètre de la période, alors* CP_L *existe et*

$$\mathrm{CP}_L = \frac{p}{p-q}.$$

Le résultat obtenu à l'exemple 3.2.12 n'est autre que le cas où $q = 1$ et où la signature est strictement périodique.

Démonstration. La démonstration consiste en le calcul de

$$\frac{\mathrm{scp}_L(N)}{N} = \frac{\sum_{i=0}^{l-1} \mathbf{v}_L(i) + |\mathrm{LB}_L(u)|}{N}$$

où N est représenté par le mot u, de longueur l. Nous utiliserons ensuite la périodicité de la signature pour contrôler le nombre d'éléments présents dans $\mathrm{LB}_L(u)$. Pour cela, il est nécessaire de prendre N suffisamment grand pour être dans la partie périodique de cette signature.

Soit st^ω la signature de L, soit l_0 tel que la somme des chiffres de s, notée P, soit strictement inférieure à $\mathbf{v}_L(l_0)$. Dans ce cas, les noeuds à partir de la hauteur l_0 dans l'arbre se trouvent "dans la partie périodique de la signature". Ce sont ces noeuds que nous considérons dans la suite.

A tout niveau l, vu la périodicité de la signature, chaque bloc de q noeuds consécutifs a p fils à la hauteur suivante. Par exemple, sur la figure 8, chaque bloc de deux noeuds a trois fils à la hauteur suivante si l'on est suffisamment bas dans l'arbre. On a alors l'encadrement

$$p\left\lfloor \frac{\mathbf{u}_L(l)}{q} \right\rfloor \leq \mathbf{u}_L(l+1) \leq p\left(\left\lfloor \frac{\mathbf{u}_L(l)}{q} \right\rfloor + 1 \right)$$

et même

$$p\left(\frac{\mathbf{u}_L(l)}{q} - 1 \right) \leq \mathbf{u}_L(l+1) \leq p\left(\frac{\mathbf{u}_L(l)}{q} + 1 \right).$$

Remarquons que vu la condition $p > q$, au moins un des termes de la période est supérieur ou égal à 2. A chaque fois qu'on rencontre un de ces termes, on a $\mathbf{u}_L(l) < \mathbf{u}_L(l+1)$, donc $\mathbf{u}_L(l) \overset{l\to\infty}{\to} +\infty$. Cette dernière information donne $\gamma_L = \frac{p}{q}$ en divisant par $\mathbf{u}_L(l)$ dans l'encadrement ci-dessus.

De manière similaire à la démonstration dans l'exemple 3.2.12, soit N un entier représenté par u de longueur l. On a $N = \mathbf{v}_L(l-1) + M$ pour un M entre 0 et $\mathbf{u}_L(l) - 1$ et

$$\mathrm{scp}_L(N) = \sum_{i=0}^{l-1} \mathbf{v}_L(i) + |\mathrm{LB}_L(u)|.$$

Comme $\gamma_L = \frac{p}{q}$, on a

$$\lim_{l\to\infty} \frac{\sum_{i=0}^{l-1} \mathbf{v}_L(i)}{\mathbf{v}_L(l-1)} = \frac{p}{p-q}$$

par le lemme B.2. On réécrit cela

$$\sum_{i=0}^{l-1} \mathbf{v}_L(i) = \mathbf{v}_L(l-1)(\frac{p}{p-q} + \varepsilon(l)) \text{ avec } \lim_{l\to\infty} \varepsilon(l) = 0.$$

Pour évaluer $|\mathrm{LB}_L(u)|$, on remarque que tout bloc de p noeuds est issu d'un bloc de q parents à la hauteur précédente de l'arbre. De là, on obtient

$$\left|\mathrm{LB}_L(l) \cap A^l\right| = M$$

puis l'encadrement

$$\frac{q}{p}M - q < q\left\lfloor \frac{M}{p} \right\rfloor \leq \left|\mathrm{LB}_L(l) \cap A^{l-1}\right| \leq q\left\lfloor \frac{M}{p} \right\rfloor + q - 1 < \frac{q}{p}M + q.$$

En itérant le même raisonnement, on obtient

$$\left(\frac{q}{p}\right)^k M - q\sum_{i=0}^{k-1}\left(\frac{q}{p}\right)^i \leq \left|\mathrm{LB}_L(l) \cap A^{l-k}\right| \leq \left(\frac{q}{p}\right)^k M + q\sum_{i=0}^{k-1}\left(\frac{q}{p}\right)^i$$

pour tous les k tels que $l - k \geq l_0$. Remarquons que

$$q\sum_{i=0}^{k-1}\left(\frac{q}{p}\right)^i = \frac{pq(1-(\frac{q}{p})^k)}{p-q} < \frac{pq}{p-q}.$$

Soit $h = l - l_0 + 1$ et $B = \cup_{j=l_0}^{l} A^j$. On obtient

$$\frac{p}{p-q}(M(1-(\frac{p}{q})^h) - qh) \leq |\mathrm{LB}_L(l) \cap B| \leq \sum_{j=l_0}^{l}((\frac{q}{p})^{l-j}M + \frac{pq}{p-q}) < \frac{p}{p-q}(M + qh)$$

en sommant les égalités ci-dessus pour k allant de 0 à $h-1$. D'autre part, on a

$$0 \leq \left|\mathrm{LB}_L(l) \cap A^{\leq l_0}\right| \leq \mathbf{v}_L(l_0).$$

En combinant les différentes égalités, on obtient

$$\mathbf{v}_L(l-1)\varepsilon(l) + N\frac{p}{p-q} - M\frac{p}{p-q}(\frac{q}{p})^h - \frac{qhp}{p-q} \leq \mathrm{scp}_L(N) \leq \mathbf{v}_L(l-1)\varepsilon(l) + N\frac{p}{p-q} + \frac{phq}{p-q} + \mathbf{v}_L(l_0).$$

Il ne reste plus qu'à constater que les deux bornes tendent vers $\frac{p}{p-q}$ après division par N. □

Nous sommes au bout de cette première approche du problème. La reformulation de la question obtenue à la proposition 3.2.7 permet d'obtenir des résultats généraux n'utilisant que les mesures de croissance du langage étudié (corollaire 3.3.9). Pour la famille des langages à signature périodique, un examen précis de la structure de l'arbre du langage permet d'obtenir des encadrements sur le cardinal de $\mathrm{LB}_L(u)$ pour tout mot u suffisamment long, et on obtient la valeur de la propagation de retenue du langage dans ce cas (théorème 3.4.5).

Malheureusement, là s'arrêtent les propriétés qui peuvent être obtenues sans recours à un bagage théorique plus conséquent. Dans la section suivante, nous nous restreignons au cas des langages rationnels. On peut alors utiliser la théorie des séries formelles ainsi que des représentations matricielles pour obtenir des résultats plus précis.

4 Approche algébrique du problème

Dans cette section, nous nous intéressons plus particulièrement au cas de la propagation de retenue dans les langages rationnels. Dans ce contexte, nous pouvons faire intervenir la théorie concernant les séries rationnelles à coefficients positifs ou nuls, ce qui nous permet de caractériser les langages rationnels où la propagation de retenue existe.

Nous supposons que le lecteur est familier avec la théorie des séries formelles. Dans cette section, nous ferons le lien entre cette théorie et l'étude des langages et nous expliquerons les résultats que l'on peut tirer du caractère rationnel des langages, avant d'utiliser ces résultats pour donner une condition nécessaire et une condition suffisante pour l'existence de la propagation de retenue d'un langage rationnel. Le lecteur qui n'est pas familier avec les séries formelles trouvera une introduction à l'annexe B. Il pourra aussi consulter [4].

4.1 Introduction et exemples

Rappelons que l'ensemble des langages rationnels est défini par, de manière équivalente :

— Le plus petit ensemble contenant les langages finis et stable pour les opérations d'union, de concaténation et d'étoile de Kleene des langages.

— L'ensemble des langages reconnaissables par un automate fini (l'utilisation d'automates non-déterministes ou déterministes uniquement ne change pas cette définition).

— L'ensemble des langages L tels que les quotients

$$w^{-1}L = \{x : wx \in L\}$$

sont en nombre fini.

Nous commençons par quelques exemples pour illustrer différents comportements qui peuvent survenir dans les langages rationnels.

Il convient de remarquer que la rationalité d'un langage n'est pas une condition suffisante pour obtenir l'existence du taux de croissance. Dans un tel cas, la propagation de retenue ne saurait exister, au vu du corollaire 3.3.9.

Exemple 4.1.1. Considérons le langage $K_1 = (a\{a,b,c,d\})^\star\{a,\varepsilon\}$. Ce langage est rationnel et accepté par l'automate de la figure 9. On obtient alors

$$\mathbf{u}_{K_1}(2l) = \mathbf{u}_{K_1}(2l+1) \text{ et } \mathbf{u}_{K_1}(2l+2) = 4\mathbf{u}_{K_1}(2l+1).$$

Dès lors,

$$\lim_{l\to\infty} \frac{\mathbf{u}_{K_1}(2l+1)}{\mathbf{u}_{K_1}(2l)} = 1 \neq 4 = \lim_{l\to\infty} \frac{\mathbf{u}_{K_1}(2l+2)}{\mathbf{u}_{K_1}(2l+1)}.$$

Donc le taux de croissance local γ_{K_1} n'existe pas, et CP_{K_1} non plus.

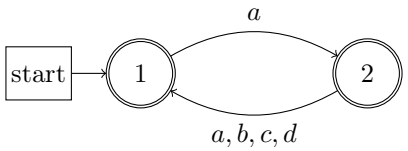

FIGURE 9 – Un automate acceptant le langage K_1.

Même si le taux de croissance existe, la propagation de retenue n'existe pas toujours, comme en témoigne l'exemple suivant.

Exemple 4.1.2. Considérons les langages

$$K_1 = (a\{a,b,c,d\})^\star\{a,\varepsilon\},\ K_1' = (\{a,b,c,d\}a)^\star\{a,b,c,d,\varepsilon\}$$

et

$$K_4 = \{a,b\}K_1 \cup cK_1' \cup \{\varepsilon\}.$$

Le langage K_4 est accepté par l'automate de la figure 10. Etudions son taux de croissance local et sa propagation de retenue.

On a

$$\mathbf{u}_{K_1}(l) = \begin{cases} 2^l & \text{si } l \text{ est pair} \\ 2^{l-1} & \text{si } l \text{ est impair} \end{cases} \text{ et } \mathbf{u}_{K_1'}(l) = \begin{cases} 2^l & \text{si } l \text{ est pair} \\ 2^{l+1} & \text{si } l \text{ est impair} \end{cases}.$$

Dès lors, si $l > 0$,

$$\mathbf{u}_{K_4}(l) = 2\mathbf{u}_{K_1}(l-1) + \mathbf{u}_{K_1'}(l-1) = \begin{cases} 2 * 2^{l-2} + 2^l = 3 * 2^{l-1} & \text{si } l \text{ est pair} \\ 2 * 2^{l-1} + 2^{l-1} = 3 * 2^{l-1} & \text{si } l \text{ est impair} \end{cases}.$$

On a donc, si $l > 0$,

$$\begin{aligned} \mathbf{v}_{K_4}(l) &= 1 + \sum_{i=1}^{l} 3 * 2^{l-1} \\ &= 1 + 3 * (2^l - 1) \\ &= 3 * 2^l - 2 \end{aligned}$$

et cette égalité est valable également pour $l = 0$. De plus, on a

$$\begin{aligned} \sum_{i=0}^{l} \mathbf{v}_{K_4} &= \sum_{i=0}^{l} (3 * 2^i - 2) \\ &= 3 * (2^{l+1} - 1) - (2l + 2) \\ &= 3 * 2^{l+1} - 2l - 5. \end{aligned}$$

Le langage K_4 a un taux de croissance local, on a $\gamma_{K_4} = 2$. Nous allons montrer qu'il n'a pas de propagation de retenue, en exhibant une sous-suite $N(l)$ le long de laquelle

$$\lim_{l \to \infty} \frac{\mathrm{scp}_{K_4}(N(l))}{N(l)}$$

n'existe pas. Nous choisissons la sous-suite

$$N(l) = \begin{cases} 3 * 2^{l-1} - 2 + 2 * 2^{l-1} & \text{si } l \text{ est impair} \\ 3 * 2^{l-1} - 2 + 2 * 2^{l-2} & \text{si } l \text{ est pair} \end{cases}.$$

En regardant la figure 11, la raison pour ce choix de $N(l)$ est que la représentation de $N(l)$ est le premier mot de longueur l de K_4 qui commence par la lettre c. La rive gauche de ce mot dans l'arbre est alors composée de deux copies de l'arbre de K_1, qui a un comportement différent selon la parité de l. Ce comportement oscillant ne se voit pas dans $\mathbf{u}_{K_4}$ car il est contrebalancé par celui de K_1', mais il aura un impact ici.

Effectuons les calculs. Il nous faut calculer

$$\frac{\mathrm{scp}_{K_4}(N(l))}{N(l)} = \frac{\sum_{i=0}^{l-1} \mathbf{v}_{K_4}(i) + |\mathrm{LB}_{K_4}(\langle N(l) \rangle_{K_4})|}{N(l)}.$$

On a vu plus haut que

$$\sum_{i=0}^{l-1} \mathbf{v}_{K_4}(i) = 3 * 2^l - 2l - 3.$$

D'autre part,

$$|\mathrm{LB}_{K_4}(\langle N(l) \rangle_{K_4})| = 2 * \mathbf{v}_{K_1}(l-1)$$

et on a

$$\mathbf{v}_{K_1}(l-1) = \begin{cases} (1+4+\ldots+4^{\frac{l-1}{2}}) + (1+4+\ldots+4^{\frac{l-3}{2}}) & \text{si } l \text{ est impair} \\ (1+4+\ldots+4^{\frac{l-2}{2}}) + (1+4+\ldots+4^{\frac{l-2}{2}}) & \text{si } l \text{ est pair} \end{cases}$$
$$= \begin{cases} \frac{4^{\frac{l+1}{2}}-1}{3} + \frac{4^{\frac{l-1}{2}}-1}{3} & \text{si } l \text{ est impair} \\ \frac{4^{\frac{l}{2}}-1}{3} + \frac{4^{\frac{l}{2}}-1}{3} & \text{si } l \text{ est pair} \end{cases}$$
$$= \begin{cases} \frac{1}{3}(5*2^{l-1}) - \frac{2}{3} & \text{si } l \text{ est impair} \\ \frac{1}{3}(4*2^{l-1}) - \frac{2}{3} & \text{si } l \text{ est pair.} \end{cases}$$

Dès lors, si l est pair on a

$$\frac{\sum_{i=0}^{l-1} \mathbf{v}_{K_4}(i) + |\mathrm{LB}_{K_4}(\langle N(l)\rangle_{K_4})|}{N(l)} = \frac{3*2^l - 2l - 3 + 2\left(\frac{1}{3}(4*2^{l-1}) - \frac{2}{3}\right)}{3*2^{l-1} - 2 + 2*2^{l-2}}$$
$$= \frac{6*2^{l-1} - 2l - 3 + \frac{8}{3}2^{l-1} - \frac{4}{3}}{4*2^{l-1} - 2}$$
$$\to \frac{6 + \frac{8}{3}}{4} = \frac{13}{6}$$

quand l tend vers l'infini. Si l est impair, on a

$$\frac{\sum_{i=0}^{l-1} \mathbf{v}_{K_4}(i) + |\mathrm{LB}_{K_4}(\langle N(l)\rangle_{K_4})|}{N(l)} = \frac{3*2^l - 2l - 3 + 2\left(\frac{1}{3}(5*2^{l-1}) - \frac{2}{3}\right)}{3*2^{l-1} - 2 + 2*2^{l-1}}$$
$$= \frac{6*2^{l-1} - 2l - 3 + \frac{10}{3}2^{l-1} - \frac{4}{3}}{5*2^{l-1} - 2}$$
$$\to \frac{6 + \frac{10}{3}}{5} = \frac{28}{15}$$

quand l tend vers l'infini. Comme ces deux limites sont distinctes, CP_{K_4} n'existe pas.

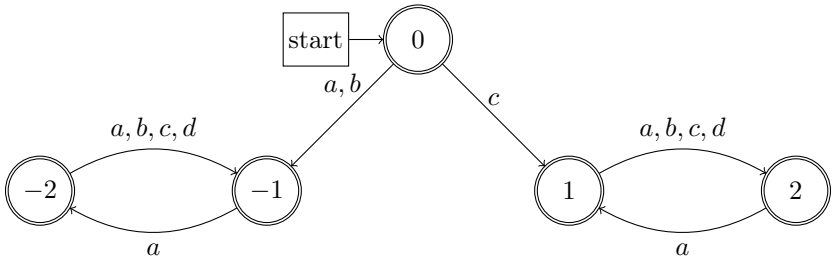

FIGURE 10 – Un automate acceptant le langage K_4.

Enfin, introduisons deux autres cas auxquels nous reviendrons dans la sous-section suivante.

Exemple 4.1.3. Le langage $K_2 = (\{a,b\}\{c,d\})^\star\{a,b,\varepsilon\}$ est accepté par l'automate de la figure 12. On a $\mathbf{u}_{K_2}(l) = 2^l$.

Exemple 4.1.4. Le langage $K_3 = (\{a,b\}\{c,d\})^\star\left(\{a,b,\varepsilon\} \cup c\{a,b\}^\star\right)$ est accepté par l'automate de la figure 13.

4.2 Fonctions génératrices et séries rationnelles

Dans cette section, nous lions les résultats théoriques obtenus dans l'annexe C au contexte des langages formels et présentons les différentes propriétés qui en découlent.

Nous supposerons que le lecteur est familier avec la théorie des séries formelles, ou qu'il a lu l'annexe C.

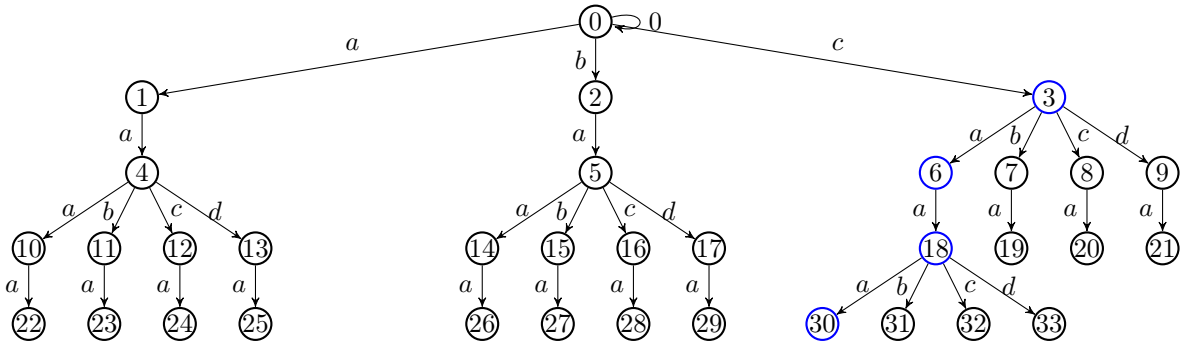

FIGURE 11 – Le début de l'arbre du langage K_4. En bleu, une sous-suite le long de laquelle la propagation de retenue moyenne n'existe pas.

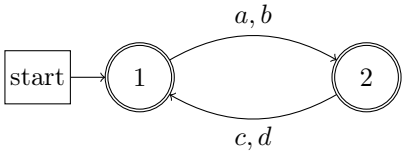

FIGURE 12 – Un automate acceptant le langage K_2.

4.2.1 Fonction génératrice d'un langage, rationalité et conséquences

Définition 4.2.1. Soit L un langage. Nous notons $g_L(z)$ la série formelle à une variable z associée à la suite $(\mathbf{u}_L(n))_{n\in\mathbb{N}}$, à savoir la série dont les coefficients sont définis par

$$(g_L(z), n) = \mathbf{u}_L(n).$$

Cette série est appelée *fonction génératrice* de L.

La série génératrice de L est un élément de $\mathbb{C}[[z]]$, et même de $\mathbb{N}[[z]]$ (le symbole sur lequel s'effectue la sommation formelle est muet, on peut donc remplacer le X utilisé dans l'annexe C par z ici).

Proposition 4.2.2. *Si L est un langage rationnel, la série $g_L(z)$ est $\mathbb{N}$-rationnelle.*

Démonstration. Soit $\mathcal{A}$ un automate acceptant le langage L. Quitte à renommer les états de $\mathcal{A}$, on peut supposer que l'ensemble d'états de cet automate est $\{1, \dots, m\}$. Soit M la matrice de transition associée à cet automate, définie par

$$M_{ij} = |\{a \in A : \delta(i, a) = j\}|.$$

Autrement dit, M_{ij} est le nombre de transitions allant de l'état i à l'état j. La théorie des graphes, ou, à défaut, un raisonnement par récurrence, nous apprend alors que $(M^n)_{ij}$ est le nombre de chemins de longueur n reliant l'état i à l'état j dans l'automate vu comme un graphe. Si maintenant I est le vecteur (ligne) caractéristique de l'état initial ($I_i = 1$ si l'état i est l'état initial, 0 sinon) et si T est le vecteur (colonne) caractéristique de l'ensemble des états finaux ($T_j = 1$ si l'état j est un état final, 0 sinon), alors la quantité

$$\sum_{i,j=1}^{m} I_i (M^n)_{ij} T_j$$

représente le nombre de chemins de longueur n allant d'un état initial à un état final. Chaque chemin de longueur n dans l'automate étant associé à exactement un mot de longueur n, on a en fait montré

$$(g_L(z), n) = \mathbf{u}_L(n) = \sum_{i,j=1}^{m} I_i (M^n)_{ij} T_j = I M^n T$$

et la fonction génératrice de L est donc une série $\mathbb{N}$-reconnaissable. Le théorème de Schützenberger assure alors qu'elle est $\mathbb{N}$-rationnelle. □

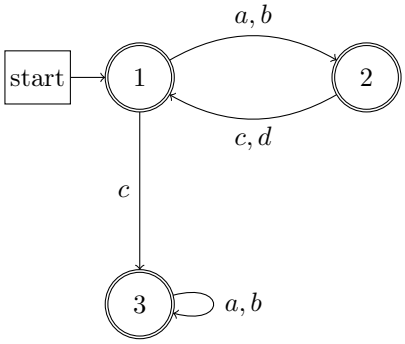

FIGURE 13 – Un automate acceptant le langage K_3.

Si de plus L est PCE, cette série génératrice est à coefficients strictement positifs. En effet, ε appartient à L car celui-ci est clos par préfixe, et L doit donc contenir au moins un mot de chaque longueur puisqu'il est extensible à droite.

Les résultats de l'annexe C nous permettent alors d'introduire un certain nombre de notions associées à un langage rationnel, et nous donnent des propriétés de ces notions.

Le corollaire C.34 assure que la suite $(\mathbf{u}_L(l))_l$ suit une relation de récurrence linéaire. On sait donc que la suite $(\mathbf{u}_L(l))_l$ suit une relation de récurrence linéaire minimale. On peut utiliser le vocabulaire de la définition C.35 :

Définition 4.2.3. Le polynôme associé à la relation de récurrence linéaire la plus courte vérifiée par $(\mathbf{u}_L(l))_l$ est appelé *polynôme minimum* de cette suite, et de L. Il est noté P_L. Les *valeurs propres* de L sont celles de $(\mathbf{u}_L(l))_l$, leurs *multiplicités* également. Ces valeurs propres sont les zéros de P_L. Le *module* de L est le module maximal d'une de ses valeurs propres.

Notons que le module de L est au moins 1, puisque la suite $(\mathbf{u}_L(l))_l$ est croissante et au vu de la proposition C.43.

Exemple 4.2.4. Rappelons que le langage K_1 est celui accepté par l'automate de la figure 14. On a

$$\mathbf{u}_{K_1}(l) = \begin{cases} 2^{l-1} & \text{si } l \text{ est impair} \\ 2^l & \text{si } l \text{ est pair.} \end{cases}$$

En procédant comme décrit ci-dessus, on trouve I, M et T tels que $\mathbf{u}_{K_1}(l) = IM^lT$. Les vecteurs et matrices cherchées sont

$$I = \begin{pmatrix} 1 & 0 \end{pmatrix}, M = \begin{pmatrix} 0 & 1 \\ 4 & 0 \end{pmatrix}, T = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Pour trouver un polynôme associé à une relation de récurrence vérifiée par K_1, on applique le théorème de Cayley-Hamilton à M. Le polynôme caractéristique de M est

$$\det(M - \lambda I) = \det \begin{vmatrix} -\lambda & 1 \\ 4 & -\lambda \end{vmatrix} = \lambda^2 - 4 = (\lambda - 2)(\lambda + 2).$$

Le polynôme $\lambda^2 - 4$ est bien le polynôme minimum de K_1. Remarquons que nous avons en fait obtenu que $\mathbf{u}_{K_1}(l+2) = 4\mathbf{u}_{K_1}(l)$ pour tout naturel l, ce qu'on aurait pu voir directement. Les valeurs propres de K_1 sont donc 2 et -2, de multiplicité 1 chacune, et le module de K_1 est 2.

Exemple 4.2.5. Rappelons que l'automate acceptant le langage K_2 est reproduit à la figure 12.

En procédant comme ci-dessus, on trouve

$$I = \begin{pmatrix} 1 & 0 \end{pmatrix}, M = \begin{pmatrix} 0 & 2 \\ 2 & 0 \end{pmatrix}, T = \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Le polynôme caractéristique de M est à nouveau $\lambda^2 - 4$, et c'est également le polynôme minimum de cette matrice. De fait, la suite $(\mathbf{u}_{K_2}(l))_l$ respecte la relation de récurrence $\mathbf{u}_{K_2}(l+2) = 4\mathbf{u}_{K_2}(l)$ pour tout naturel l.

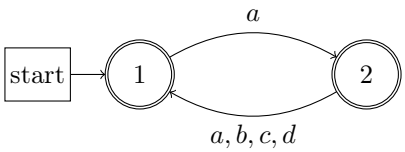

FIGURE 14 – Un automate acceptant le langage K_1.

Cependant, cette relation n'est pas la plus courte vérifiée par $(\mathbf{u}_{K_2}(l))_l$. En effet, cette suite vérifie aussi la relation $\mathbf{u}_{K_2}(l+1) = 2\mathbf{u}_{K_2}(l)$. Le polynôme minimum de K_2 est donc $\lambda - 2$ et non $\lambda^2 - 4$. La seule valeur propre de K_2 est 2.

Cet exemple illustre que le polynôme minimum d'un langage n'est pas toujours le polynôme minimum de la matrice de transition d'un automate acceptant ce langage, même si cet automate est minimal. La raison intuitive de ce fait est que le polynôme minimum d'un langage ne tient compte que du nombre de mots de chaque longueur, alors que les automates acceptant ce langage doivent en plus tenir compte de la nature exacte des mots du langage, et contiennent donc plus d'informations.

Proposition 4.2.6. *Au vu des propositions C.39 et C.40, on peut écrire*

$$g_L(z) = T(z) + \frac{R(z)}{S(z)}$$

où T, R et S sont des polynômes de $\mathbb{Q}[X]$, avec $\deg R < \deg S$, $S(0) \neq 0$ et on sait de plus que S est le polynôme réciproque du polynôme minimum de L.

Définition 4.2.7. Les zéros de S, qui sont les inverses des valeurs propres non-nulles de L, avec les mêmes multiplicités, sont appelés *pôles* de L.

Répétons que modifier les coefficients initiaux de g_L ajoute 0 comme valeur propre, mais ne change ni la réciproque du polynôme minimum, ni les pôles de L.

Proposition 4.2.8. *Au vu de la propriété C.43, on a la formule*

$$\mathbf{u}_L(l) = \sum_{j=1}^{k} \lambda_j^l P_j(l) \text{ si } l \text{ est suffisamment grand,} \tag{4}$$

où $\lambda_1, \ldots, \lambda_k$ sont les valeurs propres de L, et où P_j est un polynôme de degré égal à $\mu_j - 1$, si μ_j est la multiplicité de λ_j comme valeur propre de L.

La condition "l suffisamment grand" vient de ce que les premières valeurs de la série peuvent être modifiées sans changer les valeurs propres autres que 0 et ne respectent alors plus cette formule.

Exemple 4.2.9. Reprenons l'exemple de K_1. On sait que ses valeurs propres sont 2 et -2 de multiplicité 1 chacune. On a donc l'existence de constantes a et b (des polynômes de degré 0) telles que

$$\mathbf{u}_{K_1}(l) = a2^l + b(-2)^l \, \forall l \in \mathbb{N}\,.$$

Nous savons que $\mathbf{u}_{K_1}(0) = 1 = \mathbf{u}_{K_1}(1)$. Cela donne le système d'équations

$$\begin{cases} a + b & = 1 \\ a * 2 + b * (-2) & = 1 \end{cases}$$

dont la solution est $a = \frac{3}{4}$, $b = \frac{1}{4}$. Ainsi,

$$\mathbf{u}_{K_1}(l) = \frac{3}{4}2^l + \frac{1}{4}(-2)^l \, \forall l \in \mathbb{N}\,.$$

Exemple 4.2.10. Continuons l'exemple du langage K_2. Supposons que nous n'ayons pas vu à l'exemple 4.2.5 que $\lambda - 2$ est dans l'idéal syntaxique de $g_{K_2}(z)$. Nous savons tout de même que $\lambda^2 - 4$ est un polynôme associé à une relation de récurrence vérifiée par K_2. Comme ci-dessus, on a donc l'existence de constantes a et b telles que

$$\mathbf{u}_{K_2}(l) = a2^l + b(-2)^l \, \forall l \in \mathbb{N}\,.$$

Le système obtenu en considérant les conditions initiales est

$$\begin{cases} a+b & = 1 \\ a*2+b*(-2) & = 2 \end{cases}$$

dont la solution est $a = 1, b = 0$. Nous revoyons apparaître le fait que -2 n'est pas vraiment une valeur propre de K_2, et on a $\mathbf{u}_{K_2}(l) = 2^l$ comme prévu.

Comme $g_L(z)$ est $\mathbb{N}$-rationnel, en utilisant les propositions C.49 et C.50, et en se rappelant que les valeurs propres non nulles de L sont les inverses de ses pôles, on obtient la proposition suivante :

Proposition 4.2.11. *Si L est un langage rationnel, et si λ est le maximum des modules de ses valeurs propres, alors :*

a) λ est une valeur propre de L.
b) Chaque autre valeur propre de L de module égal à λ est de la forme $\lambda e^{i\theta}$ où $e^{i\theta}$ est une racine de l'unité.
c) La multiplicité de chaque valeur propre de module λ est inférieure ou égale à la multiplicité de λ.

A la sous-section suivante, nous tirons parti de cette propriété pour définir les langages DEV et ADEV, pour lesquels l'équation (4) prend une forme encore plus intéressante.

4.2.2 Langages à valeurs propres dominantes et presque dominantes

Au vu de la proposition 4.2.11, on définit deux groupes de langages :

Définition 4.2.12. Un langage est DEV (pour *dominating eigenvalue*) si λ est l'unique valeur propre de module λ ; dans ce cas, λ est appelé valeur propre dominante.

Un langage est ADEV (pour *almost dominating eigenvalue*) si sa fonction génératrice n'est pas un polynôme et si la multiplicité de toute valeur propre de module λ autre que λ est strictement inférieure à celle de λ.

Exemple 4.2.13. Parmi les langages ci-dessus, K_1 n'est ni DEV ni ADEV et K_2 est DEV. Le langage K_3 a pour valeurs propres 2 de multiplicité 2 et -2 de multiplicité 1, il est donc ADEV sans être DEV.

Dans la suite de cette section, nous utiliserons la notation de Landau pour décrire des comportements asymptotiques. Nous rappelons que la notation $f \in o(g)$ signifie que

$$\lim_{l\to\infty} \frac{|f(l)|}{|g(l)|} = 0$$

si $g(l)$ est non-nul pour l suffisamment grand [2]. Cette notation signifie que f est négligeable devant g pour des valeurs suffisamment grandes de l'argument. La notation $o(g)$ désigne une fonction dont on connaît seulement l'appartenance à $o(g)$. En pratique, le comportement asymptotique de l'expression $g + o(g)$ est donc celui de g.

Si nous revenons à l'expression

$$\mathbf{u}_L(l) = \sum_{j=1}^{t} \lambda_j^l P_j(l),$$

nous pouvons maintenant trier les λ_j par module. Si $|\mu| < |\lambda|$, on a

$$\mu^n n^d \in o(\lambda^n n^{d'}) \text{ quand } n \to \infty,$$

c'est-à-dire

$$\lim_{n\to\infty} \frac{\left|\mu^n n^d\right|}{\lambda^n n^{d'}} = 0.$$

2. La notation o ne sera utilisée dans ce mémoire que pour des études de comportement à l'infini, nous omettrons donc de préciser "quand $l \to \infty$".

Dans la suite, nous étudierons uniquement le comportement asymptotique des différentes quantités impliquées, et nous pouvons réécrire l'équation ci-dessus comme

$$\mathbf{u}_L(l) = \lambda^l l^d \left(\delta_1 + \sum_{j=2}^{k} \delta_j e^{il\theta_j} \right) + o(\lambda^l l^d) \text{ quand } l \to \infty, \tag{5}$$

où les valeurs propres de même module que λ sont $\lambda, \lambda e^{i\theta_2}, \ldots, \lambda e^{i\theta_k}$, la multiplicité de λ est $d+1$, et δ_j est le degré du coefficient de degré d dans le polynôme associé à $\lambda e^{i\theta_j}$, ou 0 si ce polynôme est de degré strictement inférieur. Nous savons au vu de la proposition 4.2.11 qu'aucun de ces polynômes ne peut être de degré strictement supérieur à d. Nous posons $\theta_1 = 0$ pour simplifier les énumérations.

Si un langage est DEV, il est ADEV, et si un langage est ADEV, on a $\delta_2 = \ldots = \delta_k = 0$ dans l'expression ci-dessus, qui se réécrit alors

$$\mathbf{u}_L(l) = \lambda^l l^d \delta_1 + o(\lambda^l l^d) \text{ quand } l \to \infty. \tag{6}$$

Tous les éléments sont maintenant en place et nous pouvons utiliser ce nouveau bagage théorique pour donner des conditions nécessaires et suffisantes pour qu'un langage rationnel possède une propagation de retenue. Avant cela, nous examinons l'exemple de K_3 pour illustrer ces nouvelles notions.

Exemple 4.2.14. Nous reproduisons à la figure 15 l'automate acceptant K_3. La matrice de transition de cet automate est

$$M = \begin{pmatrix} 0 & 2 & 1 \\ 2 & 0 & 0 \\ 0 & 0 & 2 \end{pmatrix}$$

dont le polynôme caractéristique est $(2-\lambda)^2(-2-\lambda)$. En utilisant l'équation (4) et les conditions initiales

$$\mathbf{u}_{K_3}(0) = 1, \mathbf{u}_{K_3}(1) = 3, \mathbf{u}_{K_3}(2) = 6,$$

on obtient

$$\mathbf{u}_{K_3}(l) = (\frac{1}{4}l + \frac{9}{8})2^l + \frac{-1}{8}(-2)^l.$$

Qu'aucun de ces coefficients ne soit 0 assure que nous avons bien trouvé le polynôme minimum de K_3. Asymptotiquement, cette expression se réécrit

$$\mathbf{u}_{K_3}(l) = \frac{1}{4} l\, 2^l + o(l\, 2^l).$$

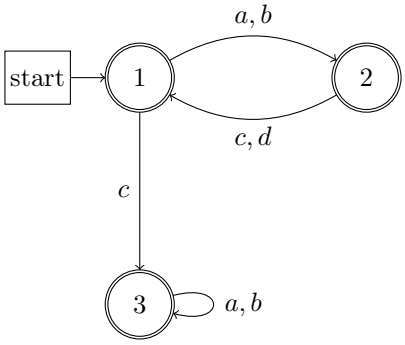

FIGURE 15 – Un automate acceptant le langage K_3.

4.3 Résolution du problème

Jusqu'ici, nous avons importé des résultats depuis la théorie des séries formelles, et nous avons obtenu des résultats dans le cadre des langages, principalement les expressions (5) et (6). A présent, nous récoltons les fruits de notre travail.

Nous procédons en deux étapes. Le théorème 4.3.1 donne une condition nécessaire pour que la propagation de retenue d'un langage rationnel existe, en termes de ses valeurs propres. Le théorème 4.3.2 donne une condition suffisante pour que la propagation de retenue d'un langage rationnel existe, en termes non seulement de ses valeurs propres mais aussi de celles de ses quotients.

Théorème 4.3.1. *Un langage rationnel L est ADEV si et seulement si son taux de croissance local γ_L existe, auquel cas le module de L est égal à γ_L.*

Démonstration. Si L est ADEV, on peut utiliser l'équation (6) et on a

$$\mathbf{u}_L(l) = \lambda^l l^d(\delta_1 + o(1)) \text{ quand } l \to \infty.$$

Remarquons que si f et g sont dans $o(1)$, alors $\frac{c+f}{c+g} - 1$ également pour toute constante réelle c puisque

$$\lim_{x\to\infty} \frac{c + f(x)}{c + g(x)} - 1 = \frac{c + \lim_{x\to\infty} f(x)}{c + \lim_{x\to\infty} g(x)} - 1 = \frac{c}{c} - 1 = 0.$$

La notation $\frac{c+o(1)}{c+o(1)} = 1 + o(1)$ et l'interchangeabilité de ces deux membres dans des équations traduisent cette propriété. Dès lors, on a

$$\frac{\mathbf{u}_L(l+1)}{\mathbf{u}_L(l)} = \frac{\lambda^{l+1}(l+1)^d(\delta_1 + o(1))}{\lambda^l l^d(\delta_1 + o(1))} = \lambda \left(\frac{l+1}{l}\right)^d (1 + o(1)) \text{ quand } l \to \infty$$

et donc

$$\lim_{l\to\infty} \frac{\mathbf{u}_L(l+1)}{\mathbf{u}_L(l)} = \lambda,$$

ce qui exprime l'existence du taux de croissance local et son égalité à λ.

Dans l'autre sens, supposons que le taux de croissance γ existe. Rappelons l'équation (5) :

$$\mathbf{u}_L(l) = \lambda^l l^d(w(l) + o(1)) \text{ où } w(l) = \delta_1 + \sum_{j=2}^{k} \delta_j e^{il\theta_j}$$

et posons $\theta_1 = 0$. Notre objectif est de montrer $\delta_2 = \ldots = \delta_k = 0$, ce qui montrera que toutes les multiplicités des autres valeurs propres de L de module λ sont strictement inférieures à $d+1$. On aura ainsi bien que L est ADEV. On sait que tous les $\theta_j, j \in \{1, \ldots k\}$ sont des racines de l'unité, et s'écrivent donc $\frac{2m_j\pi}{n_j}$ pour des naturels m_j et n_j. Si r désigne le plus petit commun multiple des n_j, alors on a $w(j + r) = w(j)$ pour tout naturel j : la suite $(w_j)_j$ est périodique de période r.

Pour tout s compris entre 0 et $r - 1$ inclus, on a

$$\begin{aligned}
\lim_{l\to\infty} \frac{\mathbf{u}_L(lr+s+1)}{\mathbf{u}_L(lr+s)} &= \lim_{l\to\infty} \frac{\lambda^{lr+s+1}(lr+s+1)^d(w(lr+s+1) + o(1))}{\lambda^{lr+s}(lr+s)^d(w(lr+s) + o(1))} \\
&= \lambda \left(\frac{lr+s+1}{lr+s}\right)^d \frac{w(s+1)}{w(s)}(1 + o(1)) \\
&= \lambda \frac{w(s+1)}{w(s)}.
\end{aligned}$$

D'autre part, on a

$$\lim_{l\to\infty} \frac{\mathbf{u}_L(lr+s+1)}{\mathbf{u}_L(lr+s)} = \gamma$$

par définition de γ. Dès lors, $\frac{w(s+1)}{w(s)} = \frac{\gamma}{\lambda}$ pour tout s entre 0 et $r - 1$. D'autre part, on a

$$1 = \frac{w(1)}{w(0)} \frac{w(2)}{w(1)} \cdots \frac{w(0)}{w(r-1)} = \left(\frac{\gamma}{\lambda}\right)^r$$

et donc $\gamma = \lambda$ puisque ce sont deux réels positifs. Dès lors, $w(0) = w(1) = \ldots = w(r-1) = \delta_1$. On en conclut que $\sum_{j=2}^{k} \delta_j e^{is\theta_j} = 0$ pour tout s entre 0 et $r - 1$. Exprimé matriciellement, on a donc

$$\begin{pmatrix} 1 & 1 & \cdots & 1 \\ e^{i\theta_1} & e^{i\theta_2} & \cdots & e^{i\theta_k} \\ \vdots & \vdots & & \vdots \\ e^{i(k-1)\theta_1} & e^{i(k-1)\theta_2} & \cdots & e^{i(k-1)\theta_k} \end{pmatrix} \begin{pmatrix} 0 \\ \delta_2 \\ \vdots \\ \delta_k \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$$

mais cette matrice est inversible, en utilisant la formule du déterminant de Vandermonde. On trouve donc $\delta_2 = \ldots = \delta_k = 0$, comme voulu, et L est bien ADEV.

□

Ainsi, il est nécessaire d'être ADEV pour avoir une propagation de retenue. Nous pouvons maintenant être plus précis.

Théorème 4.3.2. *Soit L un langage rationnel PCE ADEV de module λ. Si chaque quotient de L qui est de module λ est également ADEV, alors L a une propagation de retenue et* $\mathrm{CP}_L = \frac{\lambda}{\lambda-1}$.

Démonstration. La preuve consiste en une évaluation de

$$\lim_{N\to\infty} \frac{\mathrm{scp}(N)}{N}.$$

Pour cela, nous avons besoin de maîtriser le comportement de $|\mathrm{LB}_L(u)|$ pour des mots u quelconques. Nous introduisons une nouvelle façon de décomposer la rive gauche d'un mot, en éléments dont la taille asymptotique sera connue grâce aux formules de la section précédente appliquées aux quotients du langage L.

Pour cette preuve, nous allons devoir introduire de nouvelles notations. Nous en profitons pour simplifier les anciennes : nous supprimons les références à L quand c'est possible, en utilisant par exemple $\mathbf{u}(l)$ au lieu de $\mathbf{u}_L(l)$, P au lieu de P_L, $g(z)$ au lieu de $g_L(z)$,...

Soit $\mathcal{A} = (Q, q_0, F, A, \delta)$ un automate acceptant L. Pour tout $q \in Q$, on note $\mathcal{A}_q$ l'automate (Q, q, F, A, δ), L_q le langage que cet automate accepte et $\mathbf{u}_q(l), \mathbf{v}_q(l)$ plutôt que $\mathbf{u}_{L_q}(l)$ et $\mathbf{v}_{L_q}(l)$. Remarquons que le langage accepté par $\mathcal{A}_{\delta(q_0,w)}$ est exactement le langage $w^{-1}L$ des mots u tels que $wu \in L$. En supposant de plus que l'automate $\mathcal{A}$ est accessible, pour tout état q il existe un mot w tel que $\delta(q_0, w) = q$. On a alors

$$\mathbf{u}(l + |w|) \geq \mathbf{u}_q(l). \tag{7}$$

Nous commençons par donner une nouvelle description de $\mathrm{LB}(u)$. Si un mot w s'écrit $a_1 \cdots a_{l+1}$, on écrit $w_{[j]} = a_1 \cdots a_j$ son préfixe de longueur j. On a donc $w_{[0]} = \varepsilon$ et $w_{[l+1]} = w$. Rappelons que $\mathrm{LB}(w)$ est défini comme

$$\{u \in L : |u| \leq |w|, u \preccurlyeq w, u \notin \mathrm{Pre}(w)\}.$$

On peut voir la rive gauche de w comme une union disjointe de branches de l'arbre du langage, en considérant les différents points de divergence possibles entre un mot $u \in \mathrm{LB}(w)$ et le mot w. Si $u \in \mathrm{LB}(w)$, u ne peut pas être un préfixe de w, donc il existe un plus petit j, compris entre 1 et $l+1$, tel que $u_j \neq w_j$. On doit alors forcément avoir $u_j < w_j$ dans A pour avoir $u \preccurlyeq w$, et, pour chaque tel choix de j et de u_j, on peut concaténer $u_{[j]}$ à droite par n'importe quel mot de longueur au plus $l + 1 - j$ tel que la concaténation soit dans L pour obtenir un mot de $\mathrm{LB}(w)$, c'est-à-dire un mot qui soit dans $L_{\delta(q_0,u_{[j]})}$. De plus, tous les choix de j et de u_j produisent des ensembles distincts de mots. Au final, on a la formule

$$\mathrm{LB}(w) = \bigsqcup_{j=1}^{l+1} \bigsqcup_{a<w_j} w_{[j-1]}a \left(L_{\delta(q_0,w_{[j-1]}a)} \cap A^{\leq l+1-j}\right). \tag{8}$$

La figure 16 illustre cette décomposition.

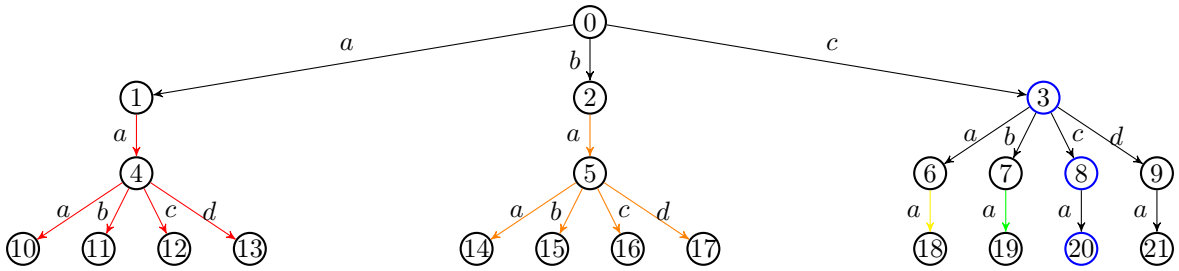

FIGURE 16 – Décomposition de la rive gauche de 18.

Pour montrer que CP_L existe et vaut $\frac{\lambda}{\lambda-1}$, on va montrer que

$$\lim_{N\to\infty} \left(\frac{\mathrm{scp}(N)}{N} - \frac{\lambda}{\lambda-1}\right) = 0.$$

Remarquons qu'il suffirait en fait de montrer que cette limite existe. Au vu de la proposition 3.3.9, si CP_L existe alors γ_L existe et $\mathrm{CP}_L = \frac{\gamma_L}{\gamma_L-1}$, et si γ_L existe alors il est égal à λ par la proposition 4.3.1 ; la limite ci-dessus ne peut donc valoir que 0 lorsqu'elle existe.

Si N est un naturel, représenté par un mot w de longueur $l+1$, exprimons les quantités N et $\mathrm{scp}_L(N)$ à l'aide de l'équation (8). On sait que $\mathrm{scp}_L(N) = \sum_{j=0}^{l} \mathbf{v}(j) + |\mathrm{LB}(w)|$, donc

$$\mathrm{scp}_L(N) = \sum_{j=0}^{l} \mathbf{v}(j) + \sum_{j=1}^{l+1} \sum_{a<w_j} \mathbf{v}_{\delta(q_0, w_{[j-1]}a)}(l+1-j).$$

D'autre part, N est égal au nombre de mots de L dont la décomposition est inférieure à w par ordre radiciel. En séparant ces mots en ceux de longueur inférieure ou égale à l, qui sont au nombre de $\mathbf{v}(l)$, et ceux de longueur $l+1$, qui sont l'intersection de $\mathrm{LB}(w)$ avec A^{l+1}, on trouve

$$N = \mathbf{v}(l) + \sum_{j=1}^{l+1} \sum_{a<w_j} \mathbf{u}_{\delta(q_0, w_{[j-1]}a)}(l+1-j).$$

De ces deux expressions, on obtient

$$\begin{aligned}\frac{1}{N}\left(\mathrm{scp}_L(N) - \frac{\lambda}{\lambda-1}N\right) &= \frac{1}{N}\left(\sum_{j=0}^{l} \mathbf{v}(j) - \frac{\lambda}{\lambda-1}\mathbf{v}(l)\right) \\ &\quad + \frac{1}{N}\left(\sum_{j=1}^{l+1}\sum_{a<w_j} \mathbf{v}_{\delta(q_0, w_{[j-1]}a)}(l+1-j) - \frac{\lambda}{\lambda-1}\mathbf{u}_{\delta(q_0, w_{[j-1]}a)}(l+1-j)\right). \qquad (9)\end{aligned}$$

Remarquons que si N tend vers l'infini, alors l également. Rappelons que

$$\lim_{l\to\infty} \frac{\mathbf{u}(l+1)}{\mathbf{u}(l)} = \gamma \Rightarrow \lim_{l\to\infty} \frac{\mathbf{v}(l+1)}{\mathbf{v}(l)} = \gamma \Rightarrow \lim_{l\to\infty} \frac{\sum_{j=0}^{l} \mathbf{v}(j)}{\mathbf{v}(l)} = \frac{\gamma}{\gamma-1}$$

par le lemme B.2, et que $\gamma = \lambda$ comme le langage L est ADEV.

Pour le premier terme du membre de droite de (9), on trouve, vu $\mathbf{v}(l) \leq N$,

$$\left|\frac{1}{N}\left(\sum_{j=0}^{l} \mathbf{v}(j) - \frac{\lambda}{\lambda-1}\mathbf{v}(l)\right)\right| \leq \left|\frac{\sum_{j=0}^{l} \mathbf{v}(j)}{\mathbf{v}(l)} - \frac{\lambda}{\lambda-1}\right|$$

et cette dernière quantité tend vers 0 si N tend vers l'infini.

Il reste à montrer que le deuxième terme tend également vers 0. Posons, pour tout q dans Q et tout l dans $\mathbb{N}$,

$$\mathbf{z}_q(l) = \mathbf{v}_q(l) - \frac{\lambda}{\lambda-1}\mathbf{u}_q(l),$$

de sorte que le terme à évaluer soit

$$\lim_{N\to\infty} \frac{1}{N}\left(\sum_{j=1}^{l+1}\sum_{a<w_j} \mathbf{z}_{\delta(q_0, w_{[j-1]}a)}(l+1-j)\right).$$

Le terme de la somme est borné par $\sum_{q\in Q} |\mathbf{z}_q(j)|$, qui ne dépend pas de a. La somme intérieure contient au plus $|A|$ termes. En changeant l'indice dans la somme extérieure et en majorant $\frac{1}{N}$ par $\frac{1}{\mathbf{v}(l)}$, la quantité étudiée est inférieure en module à

$$\frac{|A|}{\mathbf{v}(l)} \sum_{k=0}^{l}\sum_{q\in Q} |\mathbf{z}_q(k)|.$$

C'est ce terme dont nous devons montrer qu'il tend vers 0 lorsque N tend vers l'infini, et, pour ce faire, il suffit de montrer que c'est le cas de

$$\frac{\sum_{j=0}^{l} |\mathbf{z}_q(j)|}{\mathbf{v}(l)} \qquad (10)$$

pour chaque q dans Q. Pour cela, nous allons étudier le comportement asymptotique de L_q.

Remarquons que L_q est accepté par l'automate $\mathcal{A}_q$, qui a la même matrice de transition que l'automate $\mathcal{A}$, donc que L_q vérifie au moins une relation de récurrence linéaire qui est également vérifiée par L. Cependant, on ne sait pas si le polynôme associé à $\mathcal{A}$ est le polynôme minimum de L, on ne sait donc pas forcément mettre en relation le polynôme P avec P_q, le polynôme minimum de L_q. On peut toutefois appliquer le même raisonnement de la section précédente au langage L_q, qui est lui aussi rationnel. Ce langage a donc un module μ_q, des valeurs propres toutes de module inférieur ou égal à μ_q, parmi lesquelles les seules de module μ_q sont les $\mu_q e^{i\theta_{q,j}}$, avec j entre 1 et k_q et $\theta_{q,1} = 0$, et on peut exprimer asymptotiquement la quantité $\mathbf{u}_q(l)$, de manière similaire à l'équation (5) :

$$\mathbf{u}_q(l) = \mu_q^l l^{d_q} \left(\delta_{q,1} + \sum_{j=2}^{k_q} \delta_{q,j} e^{il\theta_{q,j}} \right) + o(\mu_q^l l^{d_q}) \text{ quand } l \to \infty,$$

avec $k_q = 1$ quand L_q est ADEV. Remarquons qu'on ne peut avoir $\mu_q > \lambda$, car cela compromettrait l'équation (7) vue plus haut dans cette preuve. Remarquons aussi que si l'on pose $w_q(j) = \delta_{q,1} + \sum_{j=2}^{k_q} \delta_{q,j} e^{il\theta_{q,j}}$, alors cette quantité est périodique en fonction de j et on ne peut avoir $w_q(j) \leq 0$ pour aucuns q et j, car cela est incompatible avec la croissance stricte de la suite $(u_q(l))_l$.

Pour obtenir la convergence cherchée, on sépare en deux cas :

Cas 1 : $\mu_q < \lambda$. Si $\mu_q = 1$, alors la suite $(\mathbf{u}_q(l))_l$ est à croissance polynomiale, donc $(\mathbf{v}_q(l))_l$ et $\sum_{j=0}^{l} |\mathbf{z}_q(j)|$ aussi. Comme la suite $(\mathbf{v}(l))_l$ est à croissance exponentielle, on a la convergence cherchée. On peut donc considérer que $1 < \mu_q$.

Comme la quantité $w_q(j)$ est périodique en fonction de j et toujours positive, on peut trouver des constantes strictement positives α_q et β_q qui encadrent $w_q(j)$ indépendamment de j.

On a alors

$$\mu_q^l l^{d_q}(\alpha_q + o(1)) \leq \mathbf{u}_q(l) \leq \mu_q^l l^{d_q}(\beta_q + o(1)) \quad \forall l \in \mathbb{N}\,.$$

Nous pouvons alors utiliser un résultat prouvé par D.Foata dans [9], qui généralise la formule de Faulhaber. Ce résultat indique que, si $t \neq 1$,

$$\sum_{i=1}^{n} t^i i^d = \frac{t}{t-1} t^n n^d + o(t^n n^d).$$

Dans notre cas, on obtient, en sommant les l premières valeurs de $\mathbf{u}_L(i)$ et en remarquant que les termes apparaissant dans le $o(1)$ sont des exponentielles-polynômes,

$$\mu_q^l l^{d_q}(\frac{\mu_q}{\mu_q - 1}\alpha_q + o(1)) \leq \mathbf{v}_q(l) \leq \mu_q^l l^{d_q}(\frac{\mu_q}{\mu_q - 1}\beta_q + o(1)) \quad \forall l \in \mathbb{N}\,.$$

Dès lors, en posant

$$\alpha'_q = \frac{\mu_q}{\mu_q - 1}\alpha_q - \frac{\lambda}{\lambda - 1}\beta_q \text{ et } \beta'_q = \frac{\mu_q}{\mu_q - 1}\beta_q - \frac{\lambda}{\lambda - 1}\alpha_q,$$

on trouve

$$\mu_q^l l^{d_q}(\alpha'_q + o(1)) \leq \mathbf{z}_q(l) \leq \mu_q^l l^{d_q}(\beta'_q + o(1)) \quad \forall l \in \mathbb{N}\,.$$

Dès lors, dans l'équation (10), le numérateur est de l'ordre de μ_q^l et le dénominateur est de l'ordre de λ^l, donc on a bien la convergence vers 0 annoncée.

Cas 2 : $\mu_q = \lambda$. Dans ce cas, L est ADEV par hypothèse. On a donc, comme ci-dessus, $\delta_{q,2} = \ldots = \delta_{q,k_q}$ et

$$\mathbf{u}_q(l) = \lambda^l l^{d_q}(\delta_{q,1} + o(1)) \text{ quand } l \to \infty.$$

On peut effectuer le même raisonnement que ci-dessus. On obtient $\alpha'_q = \beta'_q = 0$. Dès lors, $\mathbf{z}_q(l)$ est un $o(\lambda^l l^{d_q})$, donc le numérateur de (10) l'est également. Comme on a $d_q \leq d$ et que le dénominateur de (10) est de l'ordre de $\lambda^l l^d$, on a la convergence cherchée, ce qui conclut le raisonnement. □

Remarque 4.3.3. Nous n'avons ici qu'une condition suffisante. Le théorème ci-dessus n'interdit pas l'existence d'un langage L rationnel PCE ADEV ayant une propagation de retenue malgré qu'il existe un quotient de L de même module que L et non-ADEV. Cependant, nous n'avons trouvé ni une preuve que cette condition est en fait nécessaire et suffisante, ni un exemple montrant que la condition n'est pas nécessaire.

Annexe : résultats utiles venant d'autres branches des mathématiques

A Rappels sur la théorie des langages formels

Ce mémoire traitera des systèmes de numération et il est donc essentiel de comprendre les définitions sous-jacentes. Dans cette annexe, nous les rappelons à cette fin. Nous ne donnerons pas les preuves ici ; le lecteur qui n'est pas familier du domaine pourra se référer à [20] ou à ([21], volume 2).

Les systèmes de numération associent à des nombres des suites finies de chiffres. Nous avons donc besoin d'un objet mathématique qui représente ces suites de chiffres. La théorie des langages formels étudie ces objets sous le nom de mots.

Définition A.1.
- Un *alphabet* est un ensemble fini de symboles. Nous noterons A_r l'alphabet $\{0, 1, \dots, r-1\}$.
- Un *mot* (fini) sur un alphabet A est une suite finie d'éléments de A, à savoir une fonction de $\{0, \dots, l-1\}$ dans A. Le nombre l est appelé *longueur* du mot w et noté $\mid w \mid$, et on note

$$w = w_{l-1} \cdots w_0$$

  un tel mot, w_i désignant la i-ème lettre du mot.
- Il existe un unique mot de longueur 0, ce mot est appelé *mot vide* et noté ε.
- L'ensemble de tous les mots finis est noté $A^\star$, celui des mots finis non vides est noté A^+.
- Un *langage* est un ensemble de mots.

Remarque A.2. le choix d'indicer les lettres d'un mot de longueur l de 0 à $l-1$ ou bien de 1 à l est variable selon la source choisie. L'indice 1 est utilisé comme point de départ dans [20], l'indice 0 dans [6] et [21]. Une convention sera choisie plutôt que l'autre simplement pour éviter le plus possible le recours à des indices $n-1$ ou $n+1$. Dans notre cadre, nous choisissons de commencer à 0 pour simplifier des formules telles que l'évaluation en base entière (section 2, exemple 2.1.2).

Une autre convention est le choix d'indicer les chiffres en commençant par le plus significatif ($w = w_0 \cdots w_{l-1}$) ou en commençant par le moins significatif ($w = w_{l-1} \cdots w_0$). A nouveau, les deux conventions existent. La première est principalement utilisée par la théorie des langages formels, la seconde par la théorie des systèmes de numération.

On peut définir naturellement une opération sur les mots consistant à les "coller ensemble". Cette opération s'appelle la concaténation, et elle permet de décomposer chaque mot comme une concaténation de lettres.

Définition A.3. La concaténation de deux mots $u = u_1 \cdots u_m$ et $v = v_1 \cdots v_n$ est le mot uv défini par

$$\mid uv \mid = \mid u \mid + \mid v \mid \quad \text{et} \quad \begin{cases} (uv)_i = u_i \text{ si } i \in \{1, \dots, m\} \\ (uv)_i = v_{i-m} \text{ si } i \in \{m+1, \dots, m+n\} \end{cases}$$

c'est-à-dire plus visuellement

$$(u_1 \cdots u_m)(v_1 \cdots v_n) = u_1 \cdots u_m v_1 \cdots v_n.$$

Il s'agit d'une opération associative, et l'élément ε est neutre. Cette opération munit donc $A^\star$ d'une structure de monoïde, qu'on nomme le *monoïde libre* sur A. Vu l'associativité de l'opération, on peut définir la concaténation de n mots, $u^{(1)} \cdots u^{(n)}$, et on peut poser $u^n = u \cdots u$ avec n répétitions du facteur u. Les mots peuvent alors tous s'écrire comme concaténations de lettres, et la notation est $w = w_0 \cdots w_{l-1}$ est consistante avec celle présentée ici.

Parfois, il est utile de considérer des séquences de symboles infinies plutôt que finies. La notion de mot infini répond à ce besoin.

Définition A.4.
- Un *mot infini* sur un alphabet A est une suite d'éléments de A.
- On note $A^{\mathbb{N}}$ l'ensemble des mots infinis sur A.
- On note u^ω le mot infini formé par la concaténation de u avec lui-même une infinité de fois. Ses lettres vérifient $(u^\omega)_n = u_m$ si $m \equiv n \mod |u|$.
- La concaténation d'un mot fini u et d'un mot infini v peut être définie par

$$\begin{cases} (uv)_i = u_i \text{ si } i \in \{1, \dots, m\} \\ (uv)_i = v_{i-m} \text{ si } i > m \end{cases}.$$

— Un mot est dit *ultimement périodique* s'il s'écrit comme uv^ω pour des mots finis u et v.

Les mots infinis permettent d'associer à chaque naturel un symbole, qui peut être vu comme une étiquette qualifiant une propriété de ce naturel, et les mots de la forme u^ω représentent alors des comportements périodiques. Par exemple, la suite des puissances de 2 peut être représentée par le mot

$$011010001000000010\ldots,$$

où la i-ième lettre vaut 1 si et seulement si le naturel i est une puissance de 2. La suite des multiples de 5, qui est périodique, peut être représentée par le mot périodique

$$100001000010000010\ldots = (10000)^\omega.$$

Dans la théorie, il est souvent utile (par exemple dans la définition A.16 de l'ordre lexicographique) d'avoir une notation décrivant quand un mot "commence par un autre" ou "se termine par un autre".

Définition A.5.

— Si $w = uv$, on dit que u est un *préfixe* de w et v est un *suffixe* de w.
— On note $\mathrm{Pre}(w)$ l'ensemble des préfixes de w et $\mathrm{Suff}(w)$ l'ensemble de ses suffixes.
— Un préfixe (resp. suffixe) propre de w est un préfixe (suffixe) qui n'est ni ε ni w.
— L'ensemble des préfixes d'un langage est le langage

$$\mathrm{Pre}(L) = \cup_{w\in L}\mathrm{Pre}(w).$$

L'ensemble des suffixes de L est défini de même.
— Enfin, un facteur de w est un élément de $\mathrm{Pre}(\mathrm{Suff}(w)) = \mathrm{Suff}(\mathrm{Pre}(w))$; on note $\mathrm{Fac}(w)$ l'ensemble des facteurs de w.

Par exemple, si $w = abba$, on a

$$\mathrm{Pre}(w) = \{\varepsilon, a, ab, abb, abba\},\ \mathrm{Suff}(w) = \{\varepsilon, a, ba, bba, abba\} \text{ et } \mathrm{Fac}(w) = \{\varepsilon, a, ab, b, abb, bb, abba, bba, ba\}$$

Les langages qui nous intéresseront particulièrement dans le cadre des systèmes de numération sont ceux qui sont *clos par préfixe et extensibles à droite (PCE)*, car ils permettront d'utiliser la propriété 3.2.7.

Définition A.6.

Un langage L est *clos par préfixe* si $u \in L$, $v \in \mathrm{Pre}(u) \Rightarrow v \in L$.
Un langage L est *extensible à droite* si $u \in L \Rightarrow \exists a \in A : ua \in L$.

Un langage L sur A classifie les mots de $A^\star$ en ceux qui font partie de L, et sont "valables", et ceux qui n'en font pas partie et ne le sont pas. Considérons le cas du langage de programmation Python. Dans ce langage, "i++" n'est pas une instruction valable, malgré que 'i' et '+' soient des symboles utilisables dans un programme Python. On peut dire que le mot "i++" ne fait pas partie du langage des instructions Python. De même, "parce que je le bien", vu comme un mot sur l'alphabet $\{a, \ldots, z, ' \ '\}$, n'est pas un mot de la langue française (il ne contient pas de verbe, et n'est donc pas grammaticalement correct).

Tous les langages ne sont pas des "filtres" de qualité équivalente. Un langage de programmation informatique doit être suffisamment simple pour qu'un ordinateur puisse détecter facilement si un mot donné fait partie du langage et le décomposer en ses éléments significatifs pour pouvoir exécuter les instructions correspondantes (cette dernière partie, sémantique plutôt que syntaxique, n'est pas du ressort de la théorie des langages formels). A ce titre, un langage obtenu en tirant au hasard pour chaque mot l'appartenance ou non de celui-ci au langage n'est pas un bon langage de programmation, puisqu'il est virtuellement impossible de déterminer si un programme est correct sans stocker en mémoire la liste complète (infinie) de tous les programmes corrects. Ces réflexions amènent à la définition de différentes familles de langages, selon la facilité avec laquelle certains problèmes, tels que l'appartenance ou non d'un mot au langage, peuvent être résolus : langages réguliers, $LL(1)$, algébriques, décidables,... Nous renvoyons le lecteur à [1] pour une explication détaillée d'une des applications de la théorie des langages formels à l'informatique.

Dans la suite, la famille de langages qui nous intéressera le plus est celle des langages rationnels. En plus d'être utiles en pratique, ils possèdent de bonnes propriétés théoriques de stabilité qui les rendent particulièrement faciles à étudier. Pour définir ces langages, nous rappelons les définitions des opérations de Kleene :

Définition A.7.

— L'union de deux langages est définie par l'union ensembliste usuelle.

— La concaténation de deux langages L et M est définie par

$$LM = \{uv \mid u \in L, v \in M\}.$$

La concaténation de langages est également une opération associative, et on pose $L^n = L \cdots L$ avec n répétitions du facteur L.

— Enfin, l'étoile de Kleene d'un langage L est définie comme

$$L^\star = \bigcup_{n=0}^{\infty} L^n.$$

— La famille $\mathcal{R}$ des *langages rationnels*, aussi appelés *langages réguliers*, est alors définie comme la plus petite famille contenant les langages finis et stable pour les trois opérations ci-dessus, à savoir vérifiant

$$L, M \in \mathcal{R} \Rightarrow L \cup M, LM, L^\star \in \mathcal{R}.$$

On se permet souvent d'abuser des notations et d'écrire w plutôt que $\{w\}$, ce qui permet d'obtenir des notations comme $\{0,1\}^\star 11 \{0,1\}^\star$ pour le langage des mots qui contiennent le facteur 11.

Les langages rationnels ont plusieurs attraits. Cette famille possède de bonnes propriétés de stabilité, par l'union, la concaténation et l'étoile de Kleene bien sûr, mais également par d'autres opérations que nous rappelons ci-dessous. Il s'agit également d'une famille de langages "simples", au sens où ils sont reconnaissables rapidement par des machines de faible puissance. Rappelons les résultats nécessaires.

Définition A.8.

— Un *automate fini* (déterministe) est un quintuple $(Q, q_0, F, \Sigma, \delta)$, où
  — Q est appelé l'ensemble d'états de l'automate,
  — $q_0 \in Q$ est l'état initial,
  — $F \subset Q$ est l'ensemble d'états finaux,
  — Σ est l'alphabet,
  — $\delta : Q \times \Sigma \to Q$ est la fonction de transition.

— La fonction de transition peut être étendue à $Q \times \Sigma^\star \to Q$ via

$$\delta(q, \varepsilon) = q \text{ et } \delta(q, wa) = \delta(\delta(q, w), a).$$

— Un mot w est *accepté* par l'automate $\mathcal{A}$ si $\delta(q_0, w) \in F$.

— le langage accepté par l'automate est l'ensemble des mots acceptés par l'automate, il est noté $L(\mathcal{A})$

Remarque A.9. On peut accepter que la fonction de transition δ soit partielle, c'est-à-dire que $\delta(q, a)$ puisse ne pas être défini. Dans ce cas, les mots acceptés par l'automate sont les mots w tels que $\delta(q_0, w)$ est défini et appartient à F.

Pour se ramener à la définition ci-dessus depuis un automate ayant une fonction de transition partielle, on peut rajouter un état q, appelé *puits*, qui n'est pas final, tel que $\delta(q, a) = q$ pour toute lettre a et que toute transition non définie arrive en cet état.

Cette notion s'accompagne souvent de celle d'automate non-déterministe, dans laquelle la fonction de transition est remplacée par une relation de transition, $\Delta \subset Q \times A^\star \times Q$. On peut alors montrer que la famille des langages rationnels est exactement celle des langages L tels qu'il existe un automate fini $\mathcal{A}$ vérifiant $L = L(\mathcal{A})$. Le lecteur peut se référer à [20] pour plus de détails.

Les automates sont souvent représentés par des graphes. Un sommet correspond à un état, et l'égalité $\delta(p, a) = q$ se traduit par un arc du sommet p vers le sommet q étiqueté par la lettre a.

Exemple A.10. Considérons l'automate de la figure 17. Les états finaux sont les états $1, 2$ et 3, l'état initial est l'état 1. Pour déterminer si le mot 2120 est accepté par le langage, on calcule successivement

$$\begin{aligned} \delta(1, 2) &= 3, \\ \delta(1, 21) &= \delta(3, 1) = 3 \\ \delta(1, 212) &= \delta(3, 2) = 4 \\ \delta(1, 2120) &= \delta(4, 0) = 4. \end{aligned}$$

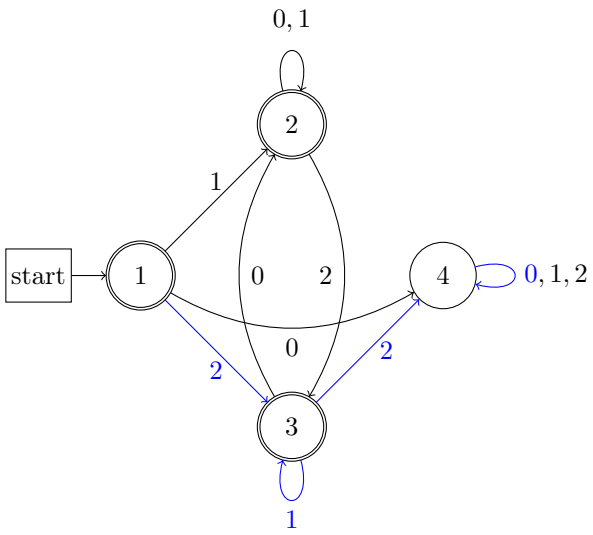

FIGURE 17 – Un automate. En bleu, le chemin emprunté par la lecture du mot 2120.

Puisque l'état 4 n'est pas final, le mot 2120 n'est pas accepté par l'automate. En examinant cet automate, on peut voir que les mots refusés sont ceux qui commencent par 0 et ceux qui contiennent un facteur 21^n2 pour un naturel n, le langage accepté est donc

$$L(\mathcal{A}) = \{0,1,2\}^\star \setminus (0\{0,1,2\}^\star \cup \{0,1,2\}^\star 21^\star 2\{0,1,2\}^\star)\,.$$

L'intérêt des automates (déterministes ou non) est double. D'une part, ils permettent de faire intervenir des résultats de théorie des graphes, notamment sur la matrice d'adjacence, pour étudier les propriétés d'un langage. Ceci fournira notre angle d'approche à la section 4. D'autre part, ils permettent de démontrer des propriétés telles que la suivante, grâce auxquelles on peut manipuler plus facilement les langages rationnels, avec la certitude que les résultats de diverses opérations restent des langages rationnels.

Proposition A.11. *Si L et M sont deux langages rationnels, alors L^c et $L \cap M$ sont également rationnels.*

La preuve, détaillée dans [20], se base sur la construction d'automates appropriés pour chacun des deux langages cibles.

les quotients de langages, définis ci-après, apparaissent dans plusieurs résultats théoriques, et nous seront utiles pour le théorème 4.3.2.

Définition A.12. Si $L \subset A^\star$ est un langage et w est un mot sur A, on pose

$$w^{-1}L = \{v \in A^\star \mid wv \in L\}.$$

Le langage $w^{-1}L$ est appelé un *quotient* de L.

Proposition A.13. *Un langage est rationnel seulement s'il possède un nombre fini de quotients différents.*

Montrer qu'un langage n'est pas rationnel est compliqué en général. Une arme particulièrement efficace dans ce cadre est le lemme de la pompe.

Proposition A.14 (Lemme de la pompe)**.** *Si L est un langage rationnel, il existe un naturel N vérifiant que tout mot w de $L \cap A^{\geq N}$ se factorise comme $w = xyz$ avec $y \neq \varepsilon$, $0 < |xy| < N$ et $xy^nz \in L$ pour tout $n \geq 0$.*

L'idée de la démonstration, détaillée dans [20], est de prendre pour N le nombre d'états d'un automate fini acceptant L. Alors, à tout mot de L est associé un chemin dans l'automate. Si le mot est suffisamment long, ce chemin contient une boucle. On peut alors prendre pour y l'étiquette de cette boucle, puisque répéter y revient à parcourir plusieurs fois la boucle, ce qui ne change pas l'état de l'automate.

Exemple A.15. Pour le langage de l'automate de la figure 17, la constante N cherchée est 3 : tout mot de longueur au moins 3 accepté par l'automate passe par deux fois par un des états $1, 2$ ou 3. Le mot 2021, par exemple, passe deux fois (et même trois) par l'état 3, et on peut en conclure que le mot $2(02)^n1$ est accepté quelque soit n, car le chemin parcouru par ce mot dans l'automate est le même que celui parcouru par 2021, au nombre de passages dans une boucle près.

Le dernier élément que nous devons rappeler est la façon d'ordonner les mots de $A^\star$. Cela nous sera utile à la section 2.3 pour définir les systèmes de numération abstraits, entre autres.

Définition A.16. Soit A un alphabet, ordonné par l'ordre total $\leq$.

— L'ordre lexicographique sur $A^\star$ est défini par

$$u \preccurlyeq v \text{ si } u \in \mathrm{Pre}(v) \text{ ou si } u = war, v = wbs \text{ avec } w, r, s \in A^\star \text{ et } a \leq b.$$

L'ordre strict associé est noté $\prec$.

— L'ordre radiciel sur $A^\star$ est défini par

$$u \sqsubseteq v \text{ si } |u| < |v| \text{ ou si } |u| = |v| \text{ et } u \preccurlyeq v.$$

L'ordre strict associé est noté $\sqsubset$.

L'ordre radiciel sera le plus important pour nous, car il a une propriété que l'ordre lexicographique n'a pas : c'est un bon ordre.

Définition A.17. Un ordre $\leq$ sur un ensemble E est un *bon ordre* si pour tout sous-ensemble L de E il existe un élément w vérifiant $w \leq u$ pour tout u dans L. Un tel élément est appelé *élément minimum* de L.

Proposition A.18. *L'ordre $\preccurlyeq$ n'est pas un bon ordre sur $A^\star$.*
L'ordre $\sqsubseteq$ est un bon ordre sur $A^\star$.

Démonstration. Considérons l'alphabet $\{0, 1\}$ et l'ensemble

$$\{1, 01, 001, 0001, \ldots\} = \{0^n1 \mid n \in \mathbb{N}\}.$$

Cet ensemble n'a pas d'élément minimum pour $\preccurlyeq$, car $0^{n+1}1 \prec 0^n1$ pour tout naturel n.

Prouvons maintenant que l'ordre radiciel est un bon ordre. Etant donné un langage $L \subset A^\star$, comme l'ordre usuel sur $\mathbb{N}$ est un bon ordre, $\{|w| : w \in L\}$ possède un élément minimum, notons-le l_0. Alors, $L \cap A^{l_0}$ est un ensemble fini non vide qui est totalement ordonné par $\sqsubseteq$. Il possède donc un élément minimum, qui est aussi minimum de L. □

L'intérêt de ce résultat est que l'on peut alors énumérer les éléments d'un langage L dans l'ordre radiciel. Ce constat est fondamental, et sera la base de la section 2.3.

B Un lemme d'analyse

Le résultat suivant est particulièrement utile pour passer d'informations sur le taux de croissance d'une suite à des informations sur le taux de croissance des sommes partielles de cette suite, ce qui sera utile dans les sections 3 et 4 pour manipuler les suites $\mathbf{u}_L$ et $\mathbf{v}_L$ associées à un langage L.

Lemme B.1 (Théorème de Stolz-Cesàro)**.** *Si $(u_n)_{n\in\mathbb{N}}$ et $(v_n)_{n\in\mathbb{N}}$ sont deux suites de nombres réels strictement positifs et si la série $\sum_{n=0}^{\infty} v_n$ diverge, alors*

$$\lim_{n\to\infty} \frac{u_n}{v_n} = l \Rightarrow \lim_{n\to\infty} \frac{\sum_{j=0}^{n} u_j}{\sum_{j=0}^{n} v_j} = l.$$

Remarquons que, puisque la série $\sum_{n=0}^{\infty} v_n$ est à termes strictement positifs, elle diverge si et seulement si $\sum_{n=0}^{\infty} v_n = +\infty$.

Démonstration. Si le membre de gauche de l'implication est vrai, alors pour tout ε strictement positif il existe un naturel N tel que

$$n \geq N \Rightarrow l - \varepsilon < \frac{u_n}{v_n} < l + \varepsilon.$$

ou encore

$$n \geq N \Rightarrow (l - \varepsilon)v_n < u_n < (l + \varepsilon)v_n.$$

Pour tout n supérieur à N, on trouve, en sommant ces égalités de N à n,

$$(l - \varepsilon) \sum_{k=N}^{n} v_k < \sum_{k=N}^{n} u_k < (l + \varepsilon) \sum_{k=N}^{n} v_k.$$

En divisant par $\sum_{k=1}^{n} v_k$, il vient

$$(l - \varepsilon)\left(1 - \frac{\sum_{k=1}^{N-1} v_k}{\sum_{k=1}^{n} v_k}\right) < \frac{\sum_{k=1}^{n} u_k}{\sum_{k=1}^{n} v_k} - \frac{\sum_{k=1}^{N-1} u_k}{\sum_{k=1}^{n} v_k} < (l + \varepsilon)\left(1 - \frac{\sum_{k=1}^{N-1} v_k}{\sum_{k=1}^{n} v_k}\right)$$

et comme la série $\sum_{k=1}^{\infty} v_k$ est divergente, pour n suffisamment grand on obtient

$$l - \varepsilon < \frac{\sum_{k=1}^{n} u_k}{\sum_{k=1}^{n} v_k} < l + \varepsilon$$

et la convergence voulue est acquise. □

Proposition B.2. *Soit $(x_n)_{n\in\mathbb{N}}$ une suite strictement croissante de nombres réels strictement positifs et $y_n = \sum_{i=0}^{n} x_i$. Les conditions suivantes sont équivalentes :*

i) $\lim_{n\to\infty} \frac{x_{n+1}}{x_n}$ *existe et vaut* $\gamma > 1$.

ii) $\lim_{n\to\infty} \frac{y_{n+1}}{y_n}$ *existe et vaut* $\gamma > 1$.

iii) $\lim_{n\to\infty} \frac{y_n}{x_n}$ *existe et vaut* $\frac{\gamma}{\gamma-1}$.

Démonstration. i)⇒ii) La série $\sum_{i=0}^{\infty} x_i$ est clairement divergente. Considérons $u_n = x_{n+1}$ et $v_n = x_n$. Le théorème de Stolz-Cesàro s'applique et on a

$$\lim_{n\to\infty} \frac{x_{n+1}}{x_n} = \gamma \Rightarrow \lim_{n\to\infty} \frac{\sum_{i=0}^{n} x_{i+1}}{\sum_{i=0}^{n} x_i} = \gamma$$

c'est-à-dire

$$\lim_{n\to\infty} \frac{y_{n+1} - x_0}{y_n} = \gamma.$$

Comme x_0 est constant et y_n tend vers $+\infty$ quand n croît, le point ii) en résulte.

ii)⇒i) Le point ii) donne

$$\lim_{n\to\infty} \frac{y_n + x_{n+1}}{y_n} = \gamma$$

donc $\frac{x_{n+1}}{y_n} \to \gamma - 1$ si n tend vers ∞. Dès lors,

$$\frac{x_{n+1}}{x_n} = \frac{x_{n+1}}{y_n}\frac{y_n}{y_{n-1}}\frac{y_{n-1}}{x_n} \overset{n\to\infty}{\to} (\gamma - 1)\gamma\frac{1}{\gamma - 1} = \gamma$$

comme annoncé.

ii)⇒iii) En reprenant le début du point ci-dessus, on a

$$\frac{y_n}{x_n} = \frac{y_n}{y_{n-1}}\frac{y_{n-1}}{x_n} \overset{n\to\infty}{\to} \gamma\frac{1}{\gamma - 1} = \frac{\gamma}{\gamma - 1}$$

comme prévu.

iii)⇒ii) On déduit

$$\lim_{n\to\infty} \frac{y_{n-1}}{x_n} = \lim_{n\to\infty} \frac{y_n - x_n}{x_n} = \frac{\gamma}{\gamma - 1} - 1 = \frac{1}{\gamma - 1}$$

puis

$$\lim_{n\to\infty} \frac{y_{n+1}}{y_n} = \lim_{n\to\infty} \frac{y_{n+1}}{x_{n+1}}\frac{x_{n+1}}{y_n} = \frac{\gamma}{\gamma - 1}(\gamma - 1)$$

et on a une fois encore le résultat annoncé.

□

C Séries formelles

Dans cette section, nous donnons les développements et les démonstrations nécessaires concernant les séries formelles et rationnelles, utilisées dans la section 4. Notre approche suit celle de [4], qui est un ouvrage de référence classique.

C.1 Bases

Dans cette sous-section, nous introduisons la notion de série formelle, une structure qui étend celle des polynômes. Imposons d'abord une structure aux coefficients de ces séries.

Définition C.1. Un *semi-anneau* est un ensemble $\mathbb{K}$ muni d'une opération d'addition $+$ et d'une opération de multiplication $\cdot$ telles que

- L'addition est associative et commutative et possède un élément neutre noté 0 ; autrement dit, $(\mathbb{K}, +)$ est un monoïde commutatif.
- La multiplication est associative et possède un neutre noté 1 ; autrement dit, $(\mathbb{K}, \cdot)$ est un monoïde.
- La multiplication distribue l'addition.
- L'élément 0 est absorbant pour la multiplication : pour tout a dans $\mathbb{K}$, $a \cdot 0 = 0 = 0 \cdot a$.

Si la multiplication est commutative, on parle de *semi-anneau commutatif*. Si tous les éléments de $\mathbb{K}$ admettent un opposé pour l'addition, on parle d'*anneau*.

Remarque C.2. Intuitivement, un semi-anneau est un "anneau sans soustraction".

Le quatrième point de la définition est une conséquence des trois autres dans le cas où $\mathbb{K}$ est un anneau, mais il ne l'est pas en général. On peut citer l'exemple de $\mathbb{N}$ muni de l'opération max en tant qu'addition et de l'addition usuelle en tant que multiplication : 0 est neutre pour max et $+$, qui sont associatives, et $+$ distribue max car $a + \max(b, c) = \max(a + b, a + c)$ et $\max(b, c) + a = \max(b + a, c + a)$. Pourtant, 0 n'est pas absorbant pour $+$, et $(\mathbb{N}, \max, +)$ n'est donc pas un semi-anneau.

De même qu'on définit les polynômes en munissant d'opérations l'ensemble des suites finies à coefficients dans $\mathbb{K}$, on peut munir d'opérations l'ensemble des suites infinies à coefficients dans $\mathbb{K}$. Ceci donne l'ensemble des séries formelles.

Définition C.3. Soit $(a_n)_{n\in\mathbb{N}}$ une suite à coefficients dans $\mathbb{K}$. La *série formelle* associée à cette suite est la série $\sum_{n=0}^{\infty} a_n X^n$. Le *coefficient d'indice* n de la série $\sum_{n=0}^{\infty} a_n X^n$ est a_n, il est noté (S, n).

L'ensemble des séries formelles est l'ensemble

$$\mathbb{K}[[X]] = \left\{ \sum_{n=0}^{\infty} a_n X^n : (a_n)_n \in \mathbb{K}^{\mathbb{N}} \right\}.$$

Cet ensemble est muni de l'opération d'addition définie par

$$\sum_{n=0}^{\infty} a_n X^n + \sum_{n=0}^{\infty} b_n X^n = \sum_{n=0}^{\infty} (a_n + b_n) X^n,$$

qui peut aussi être définie par

$$(S + T, n) = (S, n) + (T, n).$$

Le rôle de multiplication est tenu par le *produit de Cauchy*, défini par

$$\left(\sum_{n=0}^{\infty} a_n X^n \sum_{n=0}^{\infty} a_n X^n \right) \left(\sum_{n=0}^{\infty} b_n X^n \right) = \sum_{n=0}^{\infty} \left(\sum_{i=0}^{n} a_i b_{n-i} \right) X^n,$$

ou encore par

$$(ST, n) = \sum_{i=0}^{n} (S, i)(T, n - i). \tag{11}$$

Remarque C.4.

- Cette théorie est adaptable au cas beaucoup plus général où les coefficients sont indicés par des mots plutôt que des naturels. C'est cette version générale qui est présentée dans [4]. Nous n'utiliserons cependant que le cas où les indices sont naturels, c'est pourquoi nous donnons cette définition plus restrictive.

— La définition du produit de Cauchy paraît peu naturelle, mais il s'agit en fait d'une généralisation du produit des polynômes ; si l'on convient qu'on peut grouper arbitrairement les termes de la somme et que $X^n X^m = X^{n+m}$, alors les termes "de degré n" dans le produit sont bien tous les produits d'un terme "de degré j" dans $\sum_{i=0}^{\infty} a_i X^i$ et d'un terme "de degré $n-j$" dans $\sum_{i=0}^{\infty} b_i X^i$.
— Le statut du symbole X est, pour le moment, celui d'un pur instrument formel. On aurait pu définir les opérations directement sur les suites, par exemple en définissant le produit de Cauchy par

$$(a_n)_{n\in\mathbb{N}}(b_n)_{n\in\mathbb{N}} = (\sum_{j=0}^{n} a_j b_{n-j})_{n\in\mathbb{N}}.$$

Néanmoins, ce symbole permet de comprendre plus intuitivement les définitions (par exemple celle du produit de Cauchy, comme vu ci-dessus), et nous verrons aux propriétés C.47 et C.48 qu'avec un peu de prudence on peut justifier des résultats sur la série formelle $\sum_{i=0}^{\infty} a_i X^i$ de $\mathbb{C}[[X]]$ en s'aidant de la série (au sens usuel) $\sum_{i=0}^{\infty} a_i x^i$, à défaut de pouvoir toujours évaluer cette série.

Proposition C.5. *L'ensemble $\mathbb{K}[[X]]$ muni de l'addition et du produit de Cauchy est un semi-anneau. Si $\mathbb{K}$ est commutatif, alors $\mathbb{K}[[X]$ est commutatif. Si $\mathbb{K}$ est un anneau, alors $\mathbb{K}[[X]]$ est un anneau.*

Démonstration. Ce sont de simples vérifications reposant sur les propriétés de $\mathbb{K}$.
— Le neutre pour $+$ est la série O définie par $(O, n) = 0$.
— Le neutre pour $\cdot$ est la série U définie par $(U, 0) = 1$ et $(U, n) = 0$ si $n > 0$.
— La valeur commune de $(ST)U$ et de $S(TU)$ est $\sum_{n=0}^{\infty} \left(\sum_{i+j+k=n}(S, i)(T, j)(U, k)\right) X^n$.
— Si $\mathbb{K}$ est un anneau, l'opposé de la série $\sum_{n=0}^{\infty} a_n X^n$ est la série $\sum_{n=0}^{\infty}(-a_n)X^n$.
— Si $\mathbb{K}$ est commutatif, on a

$$(ST, n) = \sum_{i=0}^{n}(S, i)(T, n-i) = \sum_{j=0}^{n}(S, n-j)(T, j) = \sum_{j=0}^{n}(T, j)(S, n-j) = (TS, n)$$

en réindiçant la somme puis par commutativité de $\mathbb{K}$.

□

Exemple C.6. Considérons la série S_1 définie par $(S_1, n) = 1$ pour tout naturel n. Cette série est aussi écrite $\sum_{i=0}^{\infty} X^i$, ou même $1 + X + X^2 + \ldots$. D'autre part, considérons la série $1 - X$, définie par $(S, 0) = 1, (S, 1) = -1$ et $(S, n) = 0$ si $n \geq 2$. Leur somme est la série

$$(1 + X + X^2 + X^3 + \ldots) + (1 - X) = 2 + X^2 + X^3 + X^4 + \ldots.$$

Leur produit est la série

$$(1 + X + X^2 + X^3 + \ldots)(1 - X) = (1 + X + X^2 + X^3 + \ldots) - (X + X^2 + X^3 + \ldots) = 1,$$

ce qu'on pouvait également voir en calculant les coefficients par la formule (11). On dit que les séries $\sum_{i=0}^{\infty} X^i$ et $1 - X$ sont inverses l'une de l'autre.

Remarquons que nous avons écrit 1 plutôt que U ci-dessus. La raison en est la propriété suivante.

Proposition C.7. *Si $\mathbb{K}$ est un semi-anneau (resp. un anneau, un anneau commutatif), la fonction*

$$\phi : \mathbb{K} \to \mathbb{K}[[X]] : k \mapsto S_k \text{ où } (S_k, 0) = k \text{ et } (S_k, n) = 0 \text{ si } n > 0$$

est un homomorphisme de semi-anneaux (resp. d'anneaux, d'anneaux commutatifs) entre $\mathbb{K}$ et $\mathbb{K}[[X]]$.

Démonstration. Il s'agit de simples vérifications : ϕ est injective, $\phi(0) = O$, $\phi(1) = U$ $S_k + S_{k'} = S_{k+k'}$, $S_k \cdot S_{k'} = S_{kk'}$. □

L'intérêt de cette propriété est que nous pouvons traiter les éléments de $\mathbb{K}$ comme des séries. Nous pouvons donc abandonner les notations O et U et simplement écrire 0 et 1.

Continuons notre tour d'horizon des propriétés de base de $\mathbb{K}[[X]]$, en définissant l'inverse et l'étoile d'une série, quand cela est possible. Notre objectif est d'aboutir à une notion de série rationnelle similaire à celle de langage rationnel (définition A.7).

Proposition C.8. *Si $\mathbb{K}$ est un anneau, un élément S de $\mathbb{K}[[X]]$ est inversible si et seulement si $(S,0)$ l'est.*

Démonstration. Soit $S \in \mathbb{K}[[X]]$. On procède par analyse-synthèse pour déterminer un inverse éventuel de S. Si T est un inverse de S, on a $TS = 1$, où ce 1 représente bien la série neutre pour la multiplication, et donc

$$(T,0)(S,0) = 1 \text{ et pour tout } n \in \mathbb{N}_0, \sum_{i=0}^{n}(T,i)(S,n-i) = 0.$$

Pour satisfaire la première égalité, il est nécessaire que $(S,0)$ soit inversible dans $\mathbb{K}$. Si cette condition est vérifiée, on peut déterminer les coefficients de T par récurrence sur l'indice du coefficient. Le coefficient d'indice 0 doit être l'inverse de $(S,0)$ et, si tous les coefficients d'indice strictement inférieur à n sont déterminés, le coefficient d'indice n de T doit être

$$\left(-\sum_{i=0}^{n-1}(T,i)(S,n-i)\right)(S,0)^{-1}.$$

C'est ici qu'intervient l'hypothèse que $\mathbb{K}$ est un anneau.

Si $(S,0)$ est inversible et qu'on définit $(T,0) = (S,0)^{-1}$ puis que les coefficients de T sont déterminés par induction en utilisant la formule ci-dessus, il est clair que T est l'inverse de S.

□

Exemple C.9. La série $1-X$ est inversible, la série X ne l'est pas. On vérifie également qu'un élément de $\mathbb{K}$ est simultanément inversible comme élément de $\mathbb{K}$ et de $\mathbb{K}[[X]]$, et que les deux inverses sont images par l'homomorphisme ϕ.

Définition C.10. L'ensemble $\mathbb{K}[[X]]$ peut être muni d'une notion de convergence. Si $(S_m)_{m\in\mathbb{N}}$ est une suite de séries formelles, on dit que la suite $(S_m)_m$ converge vers la série $S \in \mathbb{K}[[X]]$, ce qu'on note $S_m \to S$, si

$$\forall l \in \mathbb{N}\, \exists M \in \mathbb{N} : \forall m \geq M, \forall k \leq l, (S_m,k) = (S,k).$$

Autrement dit, $(S_m)_m$ converge vers S si, quelle que soit la longueur l fixée, les l premiers coefficients de S_m deviennent égaux à ceux de S lorsque m est suffisamment grand.

Exemple C.11. La suite de séries formelles définie par

$$S_m = \sum_{i=0}^{m} X^i$$

converge vers la série $S = \sum_{i=0}^{\infty} X^i$.

Remarque C.12. Cette définition de la convergence est bien naturelle. On peut munir $\mathbb{K}[[X]]$ de la topologie produit associée à la topologie discrète de $\mathbb{K}$, la plus petite qui rend tous les ensembles

$$\{S \in K[[X]] : (S,m) = k\}$$

ouverts quels que soient $m \in \mathbb{N}$ et $k \in \mathbb{K}$. Cette définition d'une topologie sur un espace de suites est classique, elle est par exemple appliquée à $A^{\mathbb{N}}$ dans ([21], volume 1). La convergence donnée ci-dessus correspond alors à la notion de convergence de cette topologie, et l'addition et le produit de Cauchy sont continus. Cela résulte de ce que si les séries S_1 et S_1' ont leurs l premiers coefficients égaux, et les séries S_2 et S_2' aussi, alors c'est également le cas de $S_1 + S_2$ et $S_1' + S_2'$, ainsi que de $S_1 \cdot S_2$ et $S_1' \cdot S_2'$. On pourra consulter [25] pour une référence générale sur la topologie.

De même, on peut définir une distance sur $\mathbb{K}[[X]]$ par

$$d(S,T) = \begin{cases} 0 & \text{si } S = T \\ 2^{-\min\{n:(S,n)\neq(T,n)\}} & \text{si } S \neq T \end{cases}.$$

Cette distance donne la même topologie que celle décrite ci-dessus, et donc la même notion de convergence.

Définition C.13. Une série $S \in \mathbb{K}[[X]]$ est dite *propre* si $(S,0) = 0$.

Proposition C.14. *Si S est une série propre, la suite*

$$\left(\sum_{m=0}^{n} S^m\right)_{n\in\mathbb{N}}$$

converge dans $K[[X]]$. La limite est notée $S^\star$ et appelée l'étoile de Kleene *de S et elle vérifie $1 + SS^\star = S^\star$. Si K est un anneau, $S^\star$ est l'inverse de $1 - S$.*

Démonstration. Commençons par remarquer que si S est propre, alors pour tout $n \in \mathbb{N}$ on a $(S^n, 0) = \ldots = (S^n, n-1) = 0$. Nous montrons ceci par récurrence. Si $n = 0$ ou $n = 1$, cela est clairement vrai. Supposons que cela soit vrai pour n et montrons-le pour $n + 1$. On a

$$(S^{n+1}, m) = (S^n S, m) = \sum_{i=0}^{m} (S^n, i)(S, m-i).$$

Si $i < n$, le premier facteur est nul par hypothèse de récurrence. De cela, on déduit que (S^{n+1}, m) est nul si $m < n$, et que $(S^{n+1}, n) = (S^n, n)(S, 0) = 0$ également. Ceci conclut notre remarque.

Autrement dit, $(S^m, i) = 0$ si $m > i$. Dès lors, $(\sum_{m=0}^{n} S^m, i) = (\sum_{m=0}^{i} S^m, i)$ pour tout $n > i$, puisque les termes autres que les $i + 1$ premiers dans la somme n'influent pas sur le coefficient d'indice i. De là, considérons la série T définie par $(T, i) = (\sum_{m=0}^{i} S^m, i)$. Vu les considérations ci-dessus, la série $\sum_{m=0}^{n} S^m$ est égale à T en tous les coefficients d'indice inférieur ou égal à n. On a donc bien la convergence de la suite de l'énoncé vers T, vu la définition de la convergence.

Définissons donc $S^\star = \lim_{n\to\infty} \sum_{m=0}^{n} S^m$. Pour les mêmes raisons, on peut définir de manière similaire $S^+ = \lim_{n\to\infty} \sum_{m=1}^{n} S^m$. On a $S^\star = 1 + S^+$. On a également

$$SS^\star = S\left(\lim_{n\to\infty} \sum_{m=0}^{n} S^m\right) = \lim_{n\to\infty} \sum_{m=0}^{n} SS^m = \lim_{n\to\infty} \sum_{m=0}^{n} S^{m+1} = S^+,$$

en se servant de ce que la multiplication est continue par rapport à la convergence donnée ci-dessus.

De ces deux égalités, on déduit $S^\star = 1 + SS^\star$. Dès lors, si $\mathbb{K}$ est un anneau, on a $S^\star(1 - S) = 1$ et finalement $S^\star = (1 - S)^{-1}$. □

Remarque C.15. Le deuxième paragraphe peut aussi s'exprimer en constatant que la suite de l'énoncé est de Cauchy. L'espace $\mathbb{K}[[x]]$ étant métrique compact, il est complet, donc la suite étudiée converge.

D'autre part, remarquons la similarité entre cette étoile de Kleene et celle définie pour les langages formels. Les conséquences de ceci n'apparaîtront pas clairement dans notre contexte car nous nous sommes restreints au cas des séries formelles indicées par des naturels plutôt que des mots, mais le lecteur intéressé pourra consulter ([4], chapitre III) pour une utilisation plus poussée de cette similarité.

Exemple C.16. La série X est propre. Vu l'exemple C.11, on a

$$X^\star = \sum_{i=0}^{\infty} X^i.$$

Remarquons que le membre de droite est l'inverse de $1 - X$, comme prévu.

Définition C.17. Une série $S \in \mathbb{K}[[X]]$ est un *polynôme* s'il existe $N \in \mathbb{N}$ tel que $n \geq N \Rightarrow (S, n) = 0$, autrement dit si elle n'a qu'un nombre fini de coefficients non nuls. L'ensemble des polynômes est noté $\mathbb{K}[X]$.

Le *degré* d'un polynôme P est l'indice le plus élevé d'un de ses coefficients non nuls. Il est noté $\deg P$.

Proposition C.18. *L'ensemble $\mathbb{K}[X]$ est stable pour l'addition et le produit de Cauchy. Si $\mathbb{K}$ est un champ et si S et T sont deux polynômes, on a*

$$\deg(S + T) \leq \deg S + \deg T \text{ et } \deg(ST) = \deg S + \deg T.$$

Démonstration. Il s'agit de simples vérifications. □

Toutes les notions sont maintenant en place pour pouvoir définir l'ensemble des séries rationnelles. Celui-ci contient les séries "simples" d'un point de vue théorique (celles qui peuvent être construites à partir d'opérations sur les polynômes), qui bénéficient de certaines propriétés théoriques que nous détaillerons dans la suite.

Définition C.19. L'ensemble $\mathcal{R}$ des séries rationnelles est le plus petit ensemble vérifiant les propriétés suivantes :
- $\mathcal{R}$ contient les polynômes : $\mathbb{K}[X] \subset \mathcal{R}$.
- $\mathcal{R}$ est stable pour l'addition et le produit de Cauchy : $S, T \in \mathcal{R} \Rightarrow S + T, ST \in \mathcal{R}$.
- $\mathcal{R}$ est stable pour l'étoile de Kleene des séries propres : $S \in \mathcal{R}, (S, 0) = 0 \Rightarrow S^\star \in \mathcal{R}$.

Remarque C.20. Dans la définition ci-dessus, "le plus petit" est à comprendre comme "l'intersection de tous les sous-ensembles de $\mathbb{K}[[X]]$ vérifiant cette propriété". On constate en effet que l'intersection d'une famille d'ensembles vérifiant ces trois propriétés de stabilité continue de les vérifier. Cette construction est similaire à celle du sous-groupe engendré en algèbre, par exemple.

Exemple C.21. La série $\sum_{i=0}^{\infty} X^i$ est rationnelle car c'est l'étoile de Kleene du polynôme X. La série $\sum_{i=0}^{\infty}(i+1)X^i$ est rationnelle car c'est le produit de Cauchy de la série $\sum_{i=0}^{\infty} X^i$ avec elle-même.

Proposition C.22. *Si $\mathbb{K}$ est un anneau, l'ensemble $\mathcal{R}$ des séries rationnelles est également le plus petit sous-ensemble de $\mathbb{K}[[X]]$ qui contient les polynômes, est stable par addition et produit de Cauchy et qui est également stable pour l'inversion des séries inversibles, i.e. tel que $S \in \mathcal{R}, S$ est inversible $\Rightarrow S^{-1} \in \mathcal{R}$.*

Démonstration. Soit $\mathcal{T}$ le plus petit ensemble contenant les polynômes, stable pour l'addition et le produit de Cauchy et pour l'inverse des séries inversibles. Si l'on montre que $\mathcal{T}$ est stable pour l'étoile de Kleene des séries propres, on aura $\mathcal{R} \subset \mathcal{T}$, comme $\mathcal{R}$ est le plus petit ensemble vérifiant de telles propriétés de régularité. De même, si l'on montre que $\mathcal{R}$ est stable pour l'inverse des séries inversibles, on aura $\mathcal{T} \subset \mathcal{R}$ et la preuve sera terminée.

Soit S une série propre de $\mathcal{T}$. Comme $\mathbb{K}$ est un anneau, on sait que $S^\star = (1 - S)^{-1}$. Mais $1 - S_i n \mathcal{T}$ car 1 est un polynôme et $-S$ n'est autre que le produit de Cauchy de S par le polynôme -1 (pour rappel, tout élément de $\mathbb{K}$ peut être vu comme un polynôme de degré 0 sur $\mathbb{K}[[X]]$). De plus, le coefficient d'indice 0 de $1 - S$ est 1, qui est inversible dans K. Les propriétés de $\mathcal{T}$ permettent donc bien de conclure que $S^\star = (1 - S)^{-1} \in \mathcal{T}$.

Soit maintenant S une série inversible de $\mathcal{R}$. Le coefficient $(S, 0)$ est inversible dans $\mathbb{K}$; soit b son inverse. La série Sb se calcule terme à terme en multipliant tous les coefficients de S par b. Elle est également dans $\mathcal{R}$, et son coefficient d'indice 0 est 1. Dès lors, $1 - Sb$ est une série propre. On a donc $(1 - Sb)^\star \in \mathcal{R}$, mais $(1 - Sb)^\star = (1 - (1 - Sb))^{-1} = (Sb)^{-1} = b^{-1}S^{-1}$. En multipliant à nouveau par b à gauche, on obtient $S^{-1} \in \mathcal{R}$, ce qui conclut la stabilité de $\mathcal{R}$ par l'inverse des séries inversibles. □

Remarque C.23. C'est de cette propriété que vient l'appellation "série rationnelle", puisqu'ainsi toutes les séries de la forme $P(Q^{-1})$ sont rationnelles si P et Q sont des polynômes. Conjuguée à certaines propriétés ci-dessous faisant le lien entre séries et langages, elle explique également l'appellation "langage rationnel" qui coexiste avec celle de "langage régulier", utilisée à l'origine par Kleene dans [13].

Remarque C.24. Une série $\mathbb{Z}$-rationnelle à coefficients dans $\mathbb{N}$ n'est pas forcément $\mathbb{N}$-rationnelle. Un exemple est la série S donnée par

$$S = \frac{X + 5X^2}{1 + X - 5X^2 - 125X^3}$$

dont les premiers coefficients sont $0, 1, 4, 1, 144, \ldots$. Cette série est clairement $\mathbb{Z}$-rationnelle vu la propriété C.22 ci-dessus, car c'est un quotient de polynômes à coefficients dans $\mathbb{Z}$, qui est un anneau.

Cette série a tous ses coefficients positifs, mais elle n'est pas $\mathbb{N}$-rationnelle. Nous montrerons ces deux faits à l'exemple C.51, mais nous n'avons pas encore tous les outils nécessaires pour cela.

Définition C.25. Une série $S \in \mathbb{K}[[X]]$ est *reconnaissable* s'il existe un naturel m, une matrice $M \in \mathbb{K}_m^m$, un vecteur ligne $I \in \mathbb{K}_m$ et un vecteur colonne $T \in \mathbb{K}^m$ tels que, pour tout naturel n, on ait

$$(S, n) = IM^nT.$$

Exemple C.26. La série $\sum_{i=0}^{\infty} X^i$ est obtenue avec

$$I = M = T = \begin{pmatrix} 1 \end{pmatrix}.$$

La série $\sum_{i=0}^{\infty}(i + 1)X^i$ est reconnaissable, en prenant

$$I = \begin{pmatrix} 1 & 0 & 1 \end{pmatrix}, M = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \text{ et } T = \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}.$$

Ce n'est pas par hasard que nous avons illustré la notion de série reconnaissable et celle de série rationnelle par les mêmes exemples. Le théorème de Schützenberger garantit en effet que ces deux définitions coïncident. Ce résultat est très intéressant car il permet de manipuler ces séries par la forme la plus facile dans le contexte d'utilisation, en plus de nous rassurer sur le bien-fondé de nos définitions.

Proposition C.27 (Schützenberger)**.** *Une série est rationnelle si et seulement si elle est reconnaissable.*

Nous admettons ce théorème, dont la démonstration nécessite l'introduction de nouveaux objets et lemmes, ce qui la fait sortir du cadre de ce mémoire. Le lecteur pourra trouver cette démonstration dans [4].

C.2 Propriétés des séries rationnelles

Dans cette section, nous examinons quelles propriétés sont apportées par l'hypothèse de rationalité d'une série. Nos objectifs sont les propriétés C.34, C.39 et C.43. L'outil principal de cette section est l'association à une série rationnelle de deux idéaux de $\mathbb{K}[X]$: son idéal syntaxique, que nous montrerons être associé à l'idéal des relations de récurrence que cette série vérifie, et l'idéal des polynômes qui peuvent apparaître au dénominateur d'une écriture de cette série comme une fraction rationnelle. Nous ne donnons pas d'exemples ici, mais invitons le lecteur à étudier ceux présentés à la section 4.2. Cette section utilise des résultats de base d'algèbre : division euclidienne, décomposition en fractions simples, réduction à la forme de Jordan,... Le lecteur qui aurait besoin de rappels en trouvera dans [19], ou dans [15].

Proposition C.28. *Si $\mathbb{K}$ est un anneau commutatif, une série $S \in K[[X]]$ est rationnelle si, et seulement si, il existe deux polynômes P et Q de $\mathbb{K}[X]$ avec $(Q,0) = 1$ tels que $S = \frac{P}{Q}$.*

Démonstration. Soit $\mathcal{E}$ l'ensemble des séries telles qu'il existe deux polynômes P et Q de $\mathbb{K}[X]$ avec $(Q,0) = 1$ tels que $S = \frac{P}{Q}$. Clairement, toutes les séries de $\mathcal{E}$ sont rationnelles vu la propriété C.22. De plus, tous les polynômes sont dans $\mathcal{E}$ en prenant pour Q le polynôme 1. Comme on a

$$\frac{P_1}{Q_1} + \frac{P_2}{Q_2} = \frac{P_1Q_2 + P_2Q_1}{Q_1Q_2} \text{ et } \frac{P_1}{Q_1} \cdot \frac{P_2}{Q_2} = \frac{P_1P_2}{Q_1Q_2},$$

$\mathcal{E}$ est stable pour l'addition et le produit de Cauchy.

Il ne reste donc plus qu'à montrer que $\mathcal{E}$ est stable pour l'inverse des séries inversibles. Si S est inversible avec $S = \frac{P}{Q}$, alors on a que $(S,0)$ est inversible dans $\mathbb{K}$, et de plus $(P,0) = (Q,0)(S,0) = (S,0)$. Si l'on pose $\lambda = (P,0)$, on peut alors écrire

$$S^{-1} = \frac{\lambda^{-1}Q}{\lambda^{-1}P}$$

et le coefficient d'indice 0 du dénominateur est 1, donc $S^{-1} \in \mathcal{E}$, ce qui conclut la stabilité cherchée. on a donc bien $\mathcal{E} = \mathcal{R}$. □

Dans la suite, $\mathbb{K}$ est un champ. Dans ce cas, $\mathbb{K}[[X]]$ est un anneau commutatif, intègre (le produit de deux éléments non-nuls est non-nul) et principal (tout idéal est engendré par un élément). Le lecteur qui ne serait pas familier de ce fait pourra consulter ([11], chapitres 7 et 12).

Soit S une série rationnelle. L'ensemble des polynômes Q tels que S peut s'écrire $\frac{P}{Q}$ est un idéal de $\mathbb{K}[X]$. En effet, si $S = \frac{P}{Q}$ et si A est un polynôme, on a $S = \frac{AP}{AQ}$, et de plus si $S = \frac{P_1}{Q_1} = \frac{P_2}{Q_2}$, alors $S = \frac{P_1+P_2}{Q_1+Q_2}$. Comme $\mathbb{K}[X]$ est principal, cet idéal est engendré par un élément de $K[X]$.

Définition C.29. Le *dénominateur minimal* de S est l'unique polynôme de terme indépendant 1 et de degré minimal Q tel que $S = \frac{P}{Q}$ pour un polynôme P.

Les zéros du dénominateur minimal sont appelés les *pôles* de S.

Définition C.30. Si S est une série formelle et n est un naturel, on définit la série

$$m^{-1}S = \sum_{n=0}^{\infty} (S, m+n)X^n.$$

On définit également la fonction

$$\circ : \mathbb{K}[[X]] \times \mathbb{K}[X] \to \mathbb{K}[[X]] : (S,P) \mapsto S \circ P = \sum_{n \in \mathbb{N}} (P,n)n^{-1}S.$$

L'*idéal syntaxique* de la série S est

$$\{P \in \mathbb{K}[X] : S \circ P = 0\}.$$

Proposition C.31. *L'idéal syntaxique est un idéal.*

Démonstration. On a $S \circ (P+Q) = S \circ P + S \circ Q$, d'où la stabilité pour l'addition. Soit maintenant P dans cet ensemble et $Q \in \mathbb{K}[X]$. On montre que $S \circ (PQ) = (S \circ P) \circ Q$, ce qui permettra de conclure que PQ est dans l'ensemble étudié, donc que cet ensemble est un idéal. On a

$$\begin{aligned}
((S \circ P) \circ Q, k) &= \sum_{m=0}^{\infty} (Q, m)(m^{-1}(S \circ P), k) \\
&= \sum_{m=0}^{\infty} (Q, m)(S \circ P, m + k) \\
&= \sum_{m=0}^{\infty} (Q, m) \sum_{n=0}^{\infty} (P, n)(n^{-1}S, m + k) \\
&= \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} (Q, m)(P, n)(S, n + m + k) \\
&= \sum_{l=0}^{\infty} \sum_{m+n=l} (Q, m)(P, n)(S, l + k) \\
&= \sum_{l=0}^{\infty} (PQ, l)(l^{-1}S, k) \\
&= (S \circ (PQ), k)
\end{aligned}$$

quel que soit le naturel k, ce qui termine bien la preuve. □

Proposition C.32. *Une série S est rationnelle si et seulement si son idéal syntaxique n'est pas $\{0\}$.*

Démonstration. Soit S une série rationnelle, et $I \in \mathbb{K}_k, M \in \mathbb{K}_k^k, T \in \mathbb{K}^k$ tels que

$$\forall n \in \mathbb{N}, (S, n) = IM^n T.$$

Considérons l'ensemble

$$J = \{P \in \mathbb{K}[X] : \sum_{n=0}^{\infty} (P, n) IM^n = 0\}.$$

Cet ensemble est non-vide. En effet, les vecteurs $I, IM, \ldots, IM^k$ sont $k+1$ éléments de $\mathbb{K}_k$ et sont donc linéairement dépendants. Il existe donc un polynôme de degré k dans J. Nous montrons que J est inclus dans l'idéal syntaxique de S. En effet, si $P \in J$, on a

$$\begin{aligned}
S \circ P &= \sum_{n=0}^{\infty} (P, n) n^{-1} S \\
&= \sum_{n=0}^{\infty} (P, n) \sum_{m=0}^{\infty} (S, m + n) \\
&= \sum_{n=0}^{\infty} (P, n) \sum_{m=0}^{\infty} IM^{m+n} T \\
&= \sum_{m=0}^{\infty} \left(\sum_{n=0}^{\infty} (P, n) IM^n \right) M^m T \\
&= 0,
\end{aligned}$$

où on peut permuter les sommes car la somme sur n n'a qu'un nombre fini de termes non nuls. On a donc bien prouvé l'existence d'un polynôme non-zéro dans l'idéal syntaxique de S.

Supposons maintenant que l'idéal syntaxique de S soit différent de $\{0\}$, et soit Q le polynôme engendrant cet idéal et $k = \deg Q$. Notons

$$S \circ \mathbb{K}[X] = \{S \circ P : P \in \mathbb{K}[X]\}$$

et constatons que $S \circ \mathbb{K}[X]$ est un sous-espace vectoriel de $\mathbb{K}[[X]]$ de dimension k (puisqu'on peut remplacer le polynôme P par son reste après division euclidienne par Q dans le calcul de $S \circ P$ sans changer le résultat).

L'application

$$\phi : S \circ \mathbb{K}[X] \to S \circ \mathbb{K}[X] : T \mapsto T \circ X$$

est bien stable dans $S \circ \mathbb{K}[X]$ puisque $(S \circ P) \circ X = S \circ (PX)$. Cette application peut donc, dans une base fixée de $S \circ \mathbb{K}[X]$, se représenter par une matrice M^T (attirons l'attention sur le fait que M est la transposée de la matrice qui représente ϕ). Notons que $S = S \circ 1$ et S appartient donc à $S \circ \mathbb{K}[X]$. On peut noter I le vecteur ligne qui est la transposée du vecteur des composantes de S dans la base fixée ci-dessus. Enfin, la fonction

$$\psi : S \circ \mathbb{K}[X] \to \mathbb{K} : T \mapsto (T, 0)$$

peut se représenter par un vecteur dont la transposée est le vecteur colonne U.

Avec ces notations, on a

$$\begin{aligned}(S, n) &= (S \circ X^n, 0) \\ &= \psi(\phi^n(S)) \\ &= U^T (M^T)^n I^T \\ &= (I M^n U)^T \\ &= I M^n U\end{aligned}$$

et nous avons donc montré que S était reconnaissable, donc rationnelle, comme prévu. □

Proposition C.33. *Soit $S = \sum_{n=0}^{\infty} a_n X^n$. Le polynôme $R = X^k - \alpha_1 X^{k-1} - \ldots - \alpha_k \in K[X]$ est dans l'idéal syntaxique de S si et seulement si les coefficients de S vérifient la relation de récurrence linéaire*

$$\forall n \in \mathbb{N}, a_{n+k} = \alpha_1 a_{n+k-1} + \ldots + \alpha_k a_n$$

Démonstration. Remarquons que $\circ$ est linéaire en son second argument et que

$$S \circ X^n = \sum_{m=0}^{\infty} (S, m+n) X^m.$$

On a donc

$$\begin{aligned}R \text{ est dans l'idéal syntaxique de } S &\Leftrightarrow \sum_{m=0}^{\infty} (S, m+k) X^m = \sum_{i=1}^{k} \sum_{m=0}^{\infty} \alpha_i (S, m+k-i) X^m \\ &\Leftrightarrow \forall m \in \mathbb{N},\ a_{m+k} = \sum_{i=1}^{k} \alpha_i a_{m+k-i},\end{aligned}$$

ce qui termine la preuve. □

Des deux propositions précédentes, on tire le corollaire suivant.

Corollaire C.34. *Une série S est rationnelle si et seulement si elle vérifie une relation de récurrence linéaire.*

On a vu que chaque relation de récurrence vérifiée par une série S est associée à un polynôme de l'idéal syntaxique de S. Comme $\mathbb{K}[X]$ est un anneau principal, cet idéal est engendré par un polynôme.

Définition C.35. Le *polynôme minimum* de la série rationnelle S est celui qui engendre son idéal syntaxique. Les *valeurs propres* de S sont les racines de ce polynôme, et les *multiplicités* de ces valeurs propres sont leurs multiplicités en tant que zéros de ce polynôme.

Remarque C.36. Ce n'est pas la seule façon de parvenir au corollaire C.34. Si S est rationnel, on sait qu'il existe des vecteurs I et T et une matrice M tels que

$$(S, n) = I M^n T.$$

Dans ce cas, tout polynôme annulé par M est associé à une relation de récurrence vérifiée par S. En particulier, le polynôme caractéristique de M peut être trouvé par le théorème de Cayley-Hamilton et donne un polynôme (pas

forcément minimal) associé à une relation de récurrence vérifiée par S. Notons également que du fait que le choix de M est libre, le polynôme minimum de M n'est pas forcément celui de S.

Réciproquement, si une série $S = \sum_{i=0}^{\infty} a_i X^i$ vérifie la relation de récurrence

$$\forall n \in \mathbb{N}, a_{n+k} = \alpha_1 a_{n+k-1} + \ldots + \alpha_k a_n,$$

alors on peut prendre

$$I = \begin{pmatrix} 0 & 0 & \cdots & 0 & 1 \end{pmatrix}, M = \begin{pmatrix} \alpha_1 & \alpha_2 & \cdots & \alpha_{k-1} & \alpha_k \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{pmatrix}, T = \begin{pmatrix} a_{k-1} \\ \vdots \\ a_0 \end{pmatrix}$$

pour montrer que S est reconnaissable.

L'avantage de notre approche est qu'elle permet de préparer le terrain pour la proposition C.39 et d'introduire la notion de pôle, qui sera utilisée dans la suite.

En pratique, si l'on veut trouver le dénominateur minimal d'une série, surtout d'une série associée à un langage, il est souvent plus facile d'exhiber un triplet I, M, T permettant de montrer la reconnaissabilité, puis de tester si les diviseurs du polynôme caractéristique de M (obtenu via Cayley-Hamilton) sont associés à une relation de récurrence vérifiée par S.

Définition C.37. Une relation de récurrence $a_{n+k} = \alpha_1 a_{n+k-1} + \ldots + \alpha_k a_n$ est *stricte* si $\alpha_k \neq 0$. Une série rationnelle est stricte si elle vérifie une relation de récurrence stricte.

Cette définition permet "d'isoler 0" comme zéro du polynôme minimum d'une série, qui est souvent un cas problématique. Considérons l'exemple d'une suite $(G_n)_{n\in\mathbb{N}}$ définie par $G_n = F_n$ si $n > 0$, où F_n est le n-ième nombre de Fibonacci, et $G_0 = 42$. Le polynôme minimum de cette suite est $X^3 - X^2 - X$, tandis que celui de la suite de Fibonacci est $X^2 - X - 1$. Le facteur X qui intervient dans notre nouvelle suite traduit le fait que le premier terme de la suite ne respecte pas la relation de récurrence. Ce cas particulier mérite un instant de réflexion à chaque nouvelle proposition pour s'assurer que tout se passe comme attendu.

Proposition C.38. *Soit $S = \sum_{n=0}^{\infty} a_n X^n$ une série rationnelle. Les assertions suivantes sont équivalentes :*

a) S est une série rationnelle stricte.
b) La plus petite relation de récurrence linéaire vérifiée par S est stricte.
c) Il existe un polynôme $P \in K[X]$ satisfaisant $S \circ P = 0$ et $(P, 0) \neq 0$.
d) Le polynôme minimum de S a un terme constant non nul.
e) Les valeurs propres de S sont non nulles.
f) On peut écrire $S = \frac{P}{Q}$ avec des polynômes P, Q tels que $\deg P < \deg Q$.
g) Si $S = \frac{P}{Q}$ pour des polynômes P, Q tels que la fraction est irréductible, alors $\deg P < \deg Q$.

Démonstration. Clairement, si S satisfait une relation stricte, le polynôme associé à cette relation a son terme indépendant non nul, donc $a) \Rightarrow c)$. Ce polynôme n'a pas 0 pour racine. Le polynôme minimum de S en est un diviseur, donc lui non plus n'a pas zéro pour racine ; ainsi, $c) \Rightarrow d)$. Les valeurs propres de S sont les zéros du polynôme minimum, donc $d) \Leftrightarrow e)$. Les implications $d) \Rightarrow b)$ et $b) \Rightarrow a)$ sont évidentes, et on a donc l'équivalence entre les points de $a)$ à $e)$.

Soit S une série rationnelle vérifiant la relation $a_{n+k} - \sum_{i=1}^{k} \alpha_i a_{n+k-i} = 0$ avec $\alpha_k \neq 0$. Considérons l'expression

$$\left(1 - \alpha_1 X - \ldots - \alpha_k X^k\right) S.$$

Si $m \geq k$, le coefficient d'indice m de cette série vaut

$$(S, m) - \alpha_1 (S, m-1) - \ldots - \alpha_k (S, m-k) = a_{m-k+k} - \sum_{i=1}^{k} \alpha_i a_{m-k+k-i}$$

et cette expression vaut 0. Dès lors, $\left(1 - \alpha_1 X - \ldots - \alpha_k X^k\right) S$ est un polynôme de degré au plus $k-1$, que nous notons P. Si l'on pose $Q = 1 - \alpha_1 X - \ldots - \alpha_k X^k$, qui est de degré k puisque $\alpha_k \neq 0$, on a la décomposition cherchée. Donc $a) \Rightarrow f)$. On a aussi $f) \Rightarrow g)$ en simplifiant les facteurs communs, et $g) \Rightarrow f)$ est clair. Supposons maintenant

que $S = \frac{P}{Q}$ avec $\deg P < \deg Q$. On a que $(Q, 0)$ est inversible et, quitte à multiplier numérateur et dénominateur par son inverse, on peut supposer que $(Q, 0) = 1$. Dans ce cas, on peut écrire $Q = 1 - \alpha_1 X - \ldots - \alpha_k X^k$ avec $\alpha_k \neq 0$. On sait que $QS = P$, et ce dernier est un polynôme de degré au plus $k-1$. Dès lors, pour tout $m \geq k$ on a

$$0 = (P, m) = a_m - \alpha_1 a_{m-1} - \ldots - \alpha_k a_{m-k}.$$

Ainsi, S vérifie la relation de récurrence $a_{m+k} - \sum_{i=1}^{k} \alpha_i a_{m+k-i} = 0$, qui est stricte puisque α_k est non-nul. □

Rappelons que le *polynôme réciproque* du polynôme $\sum_{i=0}^{d} a_i X^i$ est le polynôme $\sum_{i=0}^{d} a_{d-i} X^i$. Si P est le polynôme original et Q le polynôme réciproque, on écrit aussi $Q(z) = P(\frac{1}{z}) z^d$. Remarquons que multiplier un polynôme par une puissance de X ne change pas son polynôme réciproque.

Proposition C.39. *Si S est une série rationnelle stricte et $S = \frac{P}{Q}$ avec P et Q deux polynômes tels que la fraction est irréductible, alors Q est le polynôme réciproque du polynôme minimum de S.*

Démonstration. La preuve précédente montre en fait que si $\alpha_k \neq 0$, S vérifie la relation de récurrence associée au polynôme $X^n - \alpha_1 X^{n-1} - \ldots - \alpha_k$ si et seulement si S peut s'écrire comme $\frac{P}{1-\alpha_1 X - \ldots - \alpha_k X^k}$ avec P de degré au plus $k-1$. Ainsi, les polynômes de terme indépendant 1 pouvant être le dénominateur d'une fraction rationnelle propre valant S sont exactement les polynômes réciproques des polynômes moniques de l'idéal syntaxique de S. En passant au polynôme minimum pour chacun de ces ensembles, on obtient bien que le dénominateur minimal de S est le polynôme réciproque de son polynôme minimum. □

Proposition C.40. *Toute série rationnelle S peut s'écrire de manière unique comme $S = T + R$ avec T une série rationnelle stricte et R un polynôme.*

Démonstration. Toute série rationnelle S peut s'écrire $S = \frac{P}{Q}$. On peut alors effectuer la division euclidienne de P par Q pour obtenir $P = RQ + P'$ avec $\deg P' < \deg Q$. On a alors $S = \frac{RQ+P'}{Q} = R + \frac{P'}{Q}$ et on prend $T = \frac{P'}{Q}$, qui est rationnelle stricte vu la condition sur les degrés de P' et Q.

Si maintenant $S = T + R = T' + R'$, en écrivant $T = \frac{P}{Q}$ et $T' = \frac{P'}{Q'}$, on peut supposer $Q = Q'$ quitte à mettre les expressions au même dénominateur, et on a alors $P + RQ = P' + R'Q$, donc $P = P'$ et $R = R'$ vu l'unicité de la division euclidienne. □

Remarque C.41. L'intérêt de cette proposition est à nouveau de séparer la valeur propre 0 des autres. En effet, tout polynôme de degré d vérifie la relation de récurrence $a_{n+d+1} = 0$ et toute série qui vérifie cette relation est un polynôme de degré d. Dès lors, les polynômes sont exactement les séries rationnelles qui n'ont que 0 comme valeur propre. Comme les séries propres sont exactement celles qui n'ont pas 0 comme valeur propre, cela explique la décomposition de la proposition ci-dessus.

Quelle est l'interprétation de ceci en termes des coefficients de la série ? Perturber une série en en modifiant les coefficients de faible indice revient à lui ajouter un polynôme. Cela a aussi pour effet d'invalider la relation de récurrence suivie pour les premières valeurs. Si une série suit la relation

$$\forall n \in \mathbb{N}, a_{n+k} = \alpha_1 a_{n+k-1} + \ldots + \alpha_k a_n,$$

et qu'on en modifie le coefficient a_5, la série résultante suit la relation

$$\forall n > 5, a_{n+k} = \alpha_1 a_{n+k-1} + \ldots + \alpha_k a_n,$$

ou encore

$$\forall n \in \mathbb{N}, a_{n+6+k} = \alpha_1 a_{n+k+5} + \ldots + \alpha_k a_{n+6}.$$

Son polynôme minimum est donc multiplié par X^6. Ceci ne change pas le polynôme réciproque de ce polynôme, donc les dénominateurs possibles dans l'expression de la série comme une fraction rationnelle.

Passons au dernier résultat de cette section, qui donne une formule pour les coefficients d'une série $\mathbb{C}-$rationnelle en termes de ses valeurs propres.

Lemme C.42. *Pour tout naturel $j \geq 1$ et tout complexe λ, on a*

$$\frac{1}{(1-\lambda X)^j} = \sum_{n=0}^{\infty} \binom{n+j-1}{j-1} \lambda^n X^n.$$

Démonstration. On procède par récurrence sur $j \geq 1$. Pour l'initialisation, il faut vérifier que $\frac{1}{1-\lambda X} = \sum_{n=0}^{\infty} \lambda^n X^n$. Cela peut se faire en multipliant le dénominateur par le membre de droite et en vérifiant qu'on obtient 1. De fait,

$$((1-\lambda X)(\sum_{n=0}^{\infty} \lambda^n X^n), m) = \begin{cases} 1 \cdot 1 \text{ si } m = 0 \\ 1 \cdot \lambda^m - \lambda \cdot \lambda^{m-1} \text{ si } m > 0 \end{cases}$$

et les deux séries sont bien inverses l'une de l'autre.

Passons à l'hérédité. Supposons l'énoncé vrai pour tout naturel non nul inférieur ou égal à j et montrons-le pour $j+1$. On a

$$\begin{aligned} \frac{1}{(1-\lambda X)^{j+1}} &= \frac{1}{1-\lambda X} \frac{1}{(1-\lambda X)^j} \\ &= (\sum_{n=0}^{\infty} \lambda^n X^n)(\sum_{n=0}^{\infty} \binom{n+j-1}{j-1} \lambda^n X^n) \text{ par hypothèse de récurrence} \end{aligned}$$

et le coefficient d'indice n de cette série est donc

$$\sum_{m=0}^{n} \lambda^{n-m} \cdot \binom{m+j-1}{j-1} \lambda^m = \lambda^n \sum_{m=0}^{n} \binom{m+j-1}{j-1}.$$

Il n'y a donc qu'à montrer que $\sum_{m=0}^{n} \binom{m+j-1}{j-1} = \binom{n+j+1-1}{j+1-1}$. Mais on a

$$\begin{aligned} \binom{j-1}{j-1} + \binom{j}{j-1} + \binom{j+1}{j-1} + \ldots + \binom{m+j-1}{j-1} &= \binom{j}{j} + \binom{j}{j-1} + \binom{j+1}{j-1} + \ldots + \binom{m+j-1}{j-1} \\ &= \binom{j+1}{j} + \binom{j+1}{j-1} + \ldots + \binom{m+j-1}{j-1} \\ &= \binom{j+2}{j} + \ldots + \binom{m+j-1}{j-1} \\ &= \ldots \\ &= \binom{m+j}{j} \end{aligned}$$

en utilisant répétitivement la formule du triangle de Pascal, ce qui donne le résultat annoncé. □

Proposition C.43. *Si S est une série rationnelle stricte à coefficients complexes, et si $\lambda_1, \ldots, \lambda_k$ sont les valeurs propres de S et $\mu_1, \ldots, \mu_k$ leurs multiplicités, alors on a*

$$\forall n \in \mathbb{N}, (S, n) = \sum_{j=1}^{k} \lambda_j^n P_j(n)$$

où P_j est un polynôme de degré inférieur ou égal à μ_j.

Démonstration. Notons que les λ_i et les μ_i de l'énoncé se rapportent au polynôme minimum de S. Les racines de son polynôme réciproque, qui est le dénominateur minimal de S, sont les $\frac{1}{\lambda_i}$ et ces racines ont la même multiplicité comme zéros de ce polynôme. Remarquons que le caractère strict est important ici, pour que les λ_i soient non-nuls.

On sait que S s'écrit $\frac{P}{Q}$ avec $\deg P < \deg Q$ et $(Q, 0) = 1$. Vu le paragraphe ci-dessus, le polynôme Q peut se factoriser comme

$$Q(z) = C(z - \frac{1}{\lambda_1})^{\mu_1}(z - \frac{1}{\lambda_2})^{\mu_2} \ldots (z - \frac{1}{\lambda_k})^{\mu_k}$$

et vu que $(Q, 0) = 1$, la valeur de constante peut être déterminée et on peut réécrire

$$Q(z) = (1 - \lambda_1 z)^{\mu_1}(1 - \lambda_2 z)^{\mu_2} \ldots (1 - \lambda_k z)^{\mu_k}.$$

On peut alors décomposer $\frac{P}{Q}$ en fractions simples. On obtient que $\frac{P}{Q}$ s'écrit sous la forme

$$\frac{P}{Q} = \sum_{j=1}^{k} \sum_{i=1}^{\mu_j} \frac{A_{ij}}{(1 - \lambda_j z)^i}.$$

On a donc

$$\begin{aligned}(S,n) &= \sum_{j=1}^{k}\sum_{i=1}^{\mu_j}(\frac{A_{ij}}{(1-\lambda_j z)^i},n)\\ &= \sum_{j=1}^{k}\sum_{i=1}^{\mu_j} A_{ij}\binom{n+i-1}{i-1}\lambda_j^n\end{aligned}$$

par le lemme précédent. Il suffit alors de remarquer que $\binom{n+i-1}{i-1}$ est un polynôme de degré $i-1$ en n. Dès lors,

$$\sum_{i=1}^{\mu_j} A_{ij}\binom{n+i-1}{i-1}$$

est un polynôme en n de degré $\mu_j - 1$. C'est le P_j cherché dans l'énoncé. □

Remarque C.44. On peut en fait montrer que les P_j doivent tous être de degré exactement égal à $\mu_j - 1$.

L'ensemble des séries s'écrivant $\frac{P}{Q}$ où $Q = (1-\lambda_1 z)^{\mu_1}(1-\lambda_2 z)^{\mu_2}\dots(1-\lambda_k z)^{\mu_k}$ et où $\deg P < \deg Q$ est un sous-espace vectoriel de dimension $\mu_1 + \ldots + \mu_k$ de $\mathbb{K}[[X]]$ puisqu'il est isomorphe à $K^{<\deg Q}[[X]]$.

D'autre part, l'ensemble des séries vérifiant

$$\forall n \in \mathbb{N}, (S,n) = \sum_{j=1}^{k}\lambda_j^n P_j(n)$$

est lui aussi un sous-espace vectoriel de dimension $\mu_1 + \ldots + \mu_k$ de $\mathbb{K}[[X]]$ puisqu'il est isomorphe à $K^{<\mu_1}[X] \times K^{<\mu_2}[X] \times \ldots \times K^{<\mu_k}[X]$. Cela implique que la correspondance associant à une série S les polynômes $P_1, \ldots, P_k$, qui est clairement injective, est également surjective.

Si l'un de ces polynômes n'a pas le degré maximal possible, on peut alors voir que la série S lui correspondant a sa valeur propre correspondante de multiplicité strictement inférieure à celle annoncée.

Remarque C.45. A nouveau, il est possible de retrouver ce résultat en se servant de ce que S est reconnaissable, plutôt que de ce que S est rationnel. A partir de la remarque C.36, on peut appliquer le théorème de décomposition de Jordan, puis calculer la forme générale de

$$\left(\begin{pmatrix}\lambda & 1 & 0 & \cdots & 0\\ 0 & \lambda & 1 & \cdots & 0\\ 0 & 0 & \lambda & \cdots & 0\\ \vdots & \vdots & \vdots & & \vdots\\ 0 & 0 & 0 & \cdots & 1\\ 0 & 0 & 0 & \cdots & \lambda\end{pmatrix}\right)^n.$$

De là, on peut retrouver le résultat ci-dessus.

C.3 Séries à coefficients positifs

Dans cette section, nous considérons des séries à coefficients complexes. Notre objectif est la proposition C.50, qui nous donne des informations plus précises sur les pôles (et donc les valeurs propres) d'une série à coefficients positifs.

Rappelons d'abord la notion de rayon de convergence d'une série. Il s'agit d'une notion classique, qu'on trouve déjà dans [14].

Définition C.46. Considérons une série de la forme $\sum_{n=0}^{\infty} a_n z^n$. Ici, il s'agit bien d'une série complexe ; z est un nombre complexe et non une indéterminée. On pose

$$R = \sup\{r \geq 0 : \sum_{n=0}^{\infty} |a_n|\, r^n < \infty\}$$

Alors, la série $\sum_{n=0}^{\infty} a_n z^n$ converge uniformément sur tout compact inclus dans $\{y \in \mathbb{C} : |y| < R\}$. Le réel R est appelé *rayon de convergence* de la série.

Les coefficients d'une série formelle de $\mathbb{C}[[X]]$ sont aussi les coefficients d'une série complexe. Ces deux objets sont très liés, pour peu que la série converge. En effet, on a, comme démontré dans [14],

Proposition C.47. *Si deux séries (complexes) $\sum_{n=0}^{\infty} a_n$ et $\sum_{n=0}^{\infty} b_n$ sont absolument convergentes, alors leur somme, qui est la série de terme général $c_n = a_n + b_n$, et leur produit de Cauchy, qui est la série de terme général $d_n = \sum_{j=0}^{n} a_j b_{n-j}$, sont également absolument convergentes, et on a*

$$\sum_{n=0}^{\infty} c_n = \sum_{n=0}^{\infty} a_n + \sum_{n=0}^{\infty} b_n \text{ et } \sum_{n=0}^{\infty} d_n = \left(\sum_{n=0}^{\infty} a_n\right)\left(\sum_{n=0}^{\infty} b_n\right).$$

Etant donnée une série formelle, on peut lui associer une fonction, qui évalue la série en son entrée. La proposition ci-dessus garantit alors que sous condition de convergence absolue, les opérations d'addition et de produit dans les séries se traduiront par l'addition et le produit point à point usuel des fonctions de $\mathbb{C}$ dans $\mathbb{C}$. Ceci permet d'utiliser des résultats d'analyse pour étudier des séries formelles. Avant cela, il nous faut tout de même vérifier où la convergence absolue a effectivement lieu. C'est le rôle de la proposition suivante.

Proposition C.48. *Le minimum des modules des pôles de $S \in \mathbb{C}[[X]]$ est égal au rayon de convergence de la fonction*

$$x \in \mathbb{C} \mapsto \sum_{n=0}^{\infty} (S, n) x^n \in \mathbb{C}.$$

vue comme une série complexe.

Démonstration. Un polynôme n'a pas de pôles et le rayon de convergence de la série associée est $+\infty$, le résultat est donc vérifié dans ce cas.

La série $(\frac{1}{1-\lambda X})^j$ s'écrit aussi $\sum_{n=0}^{\infty} \binom{n+j-1}{j-1} \lambda^n X^n$. On peut utiliser le critère du quotient sur la série associée : on a

$$\frac{\binom{n+1+j-1}{j-1} \lambda^{n+1} x^{n+1}}{\binom{n+j-1}{j-1} \lambda^n x^n} = \frac{n-j}{n+1} \lambda x \to \lambda x \text{ si } n \to \infty.$$

Ainsi, la série associée est absolument convergente si $|\lambda x| < 1$, et le rayon de convergence de cette série est $\frac{1}{|\lambda|}$, qui est bien le module de l'unique pôle de cette série. Le résultat tient également dans ce cas.

Enfin, une série quelconque s'écrit $T + R$ où R est un polynôme et T est une série rationnelle stricte. On a vu plus haut qu'on pouvait, en décomposant T en fractions simples, l'exprimer comme une somme de séries de la forme $(\frac{1}{1-\lambda_j X})^n$. Le rayon de convergence de ces termes est $\frac{1}{|\lambda_j|}$ par le cas ci-dessus. On sait qu'alors la somme de ces termes a un rayon de convergence supérieur ou égal au minimum des rayons de convergence de chacun des termes, et il est égal ici vu la forme des termes en jeu, qui sont linéairement indépendants. Le rayon de convergence est donc $\min_j \frac{1}{|\lambda_j|}$, ce qui est bien le minimum des modules des pôles de la série. □

Nous pouvons maintenant aborder les deux propositions qui sont notre objectif, et qui nous renseignent sur le comportement des pôles d'une série dont tous les coefficients sont positifs. Les résultats obtenus sont à rapprocher des théorèmes de Perron et Perron-Frobenius, résultats classiques de la théorie des matrices à coefficients positifs que le lecteur intéressé pourra trouver dans [22].

Proposition C.49. *Soit f une fraction rationnelle, qui n'est pas un polynôme et dont le développement en série n'a que des coefficients positifs ou nuls. Soit également ρ le minimum des modules des pôles de f. Alors, ρ est un pôle de f, et tout autre pôle de f de module égal à ρ a une multiplicité moindre que celle de ρ.*

Démonstration. Soit z un complexe de module inférieur à ρ. Par la proposition ci-dessus, la série associée à f converge absolument autour de z, et on a

$$|f(z)| = \left|\sum_{n=0}^{\infty} a_n z^n\right| \leq \sum_{n=0}^{\infty} a_n |z_n| = f(|z|)$$

puisque les coefficients a_n de la série sont tous positifs, donc égaux à leur module. Soit maintenant z_0 un pôle de module ρ et de multiplicité π de f et supposons que la multiplicité de ρ soit strictement inférieure à π. Si nous montrons une absurdité, l'énoncé sera prouvé : il existe un pôle de module ρ, donc ρ lui-même est un pôle car il est de multiplicité plus grande, et la deuxième partie de l'énoncé est claire.

Si la multiplicité de ρ est strictement inférieure à π, alors la fonction g définie par $g(z) = f(z)(\rho - z)^\pi$ peut se prolonger par continuité en ρ par 0, et on a donc

$$\lim_{\substack{r\to 1\\ r<1}} (\rho - \rho r)^\pi f(\rho r) = 0.$$

D'autre part, la fonction $h(z) = (z_0 - z)^\pi f(z)$ est prolongeable par continuité en z_0 par un nombre non nul. On a donc

$$\lim_{\substack{z\to z_0\\ |z|<\rho}} |(z_0 - z)^\pi f(z)| > 0.$$

En particulier, en considérant la convergence le long de la demi-droite allant de l'origine à z_0, on trouve

$$\lim_{\substack{r\to 1\\ r<1}} |z_0^\pi (1 - r)^\pi f(r z_0)| > 0$$

et vu que $|f(z)| \leq f(|z|)$, on trouve

$$0 < \lim_{\substack{r\to 1\\ r<1}} \rho^\pi (1 - r)^\pi \, |f(r z_0)| \leq \rho^\pi (1 - r)^\pi f(r\rho) = 0$$

ce qui est l'absurdité cherchée. Ceci conclut la preuve. □

Proposition C.50. *Soit f une fraction rationnelle, qui n'est pas un polynôme et dont le développement en série est $\mathbb{R}_+$-rationnel. Soit également ρ le minimum des modules des pôles de f. Alors ρ est un pôle de f, et tout autre pôle de f de module égal à ρ est de la forme $\rho\theta$ où θ est une racine de l'unité.*

Démonstration. Notons $\mathcal{E}$ l'ensemble contenant d'une part tous les polynômes à coefficients réels positifs et d'autre part toutes les séries $\mathbb{R}_+$-rationnelles associées à une fonction f telle que ρ est un pôle de f et les autres pôles de même module sont de la forme $\rho\theta$ avec θ une racine de l'unité. Notre objectif est de montrer que $\mathcal{E}$ contient les séries $\mathbb{R}_+$-rationnelles. Comme $\mathcal{E}$ contient déjà les polynômes à coefficients dans $\mathbb{R}_+$, il suffit de montrer qu'il est stable pour l'addition, le produit de Cauchy et l'étoile.

Soient f et g deux fonctions associées à une série de $\mathcal{E}$. Les formules

$$\frac{P}{Q} + \frac{P'}{Q'} = \frac{PQ' + P'Q}{P'Q'} \text{ et } \frac{P}{Q} \cdot \frac{P'}{Q'} = \frac{PQ}{P'Q'}$$

impliquent que l'ensemble des pôles de fg, ainsi que celui de $f+g$, sont l'union des ensembles de pôles de f et g. On prend la convention que l'ensemble des pôles d'un polynôme est le vide et que $\min \emptyset = +\infty$. Comme les coefficients des séries associées sont positifs, il n'est pas possible que des simplifications aient lieu entre les pôles de f et de g dans le cas de l'addition.

Dès lors, si ρ_f et ρ_g sont les minima des modules des pôles de f et g respectivement, et en supposant $\rho_f \leq \rho_g$, alors le minimum des pôles de $f + g$ et fg est ρ_f, qui est bien un pôle de $f + g$ et fg, et les autres pôles de même module sont obtenus en multipliant ρ_f par une racine de l'unité, comme voulu.

Soit maintenant f la fonction associée à la série $\sum_{n=0}^\infty a_n X^n$ avec $a_0 = 0$. On rappelle que $f^\star = (1 - f)^{-1}$. Les pôles de f sont donc les zéros de $1 - f$. On doit avoir $\sum_{n=0}^\infty a_n \rho_f^n) = +\infty$ puisque ρ_f est un pôle de f. Cela signifie que la fonction

$$[0, \rho_f[\to [0, +\infty[: r \mapsto \sum_{n=0}^\infty a_n r^n$$

est continue, strictement croissante, vaut 0 en 0 et tend vers l'infini quand r tend vers r_n. Dès lors, il existe un unique $r_f \in [0, \rho_f[$ tel que $\sum_{n=0}^\infty a_n r_f^n = 1$.

Soit z un zéro de $1 - f$. Si l'on montre que z est de la forme $r_f\theta$ avec θ une racine de l'unité, on aura bien que $f^\star \in \mathcal{E}$ et la preuve sera terminée. On a les relations

$$1 = \sum_{n=0}^\infty a_n z^n = \Re(\sum_{n=0}^\infty a_n z^n) = \sum_{n=0}^\infty a_n \Re(z^n) \leq \sum_{n=0}^\infty a_n r_f^n = 1.$$

On a donc les égalités partout, et on doit avoir $\Re(z^n) = r_f^n$ si a_n est non-nul. Si tous les a_n sont nuls, il est évident que $f^\star = 1$ est dans $\mathcal{E}$. Sinon, en prenant un a_n non-nul, on doit avoir $z^n = r_f^n$, donc z est bien obtenu en multipliant r_f par une racine n-ième de l'unité, comme voulu. □

Pour terminer cette introduction aux séries formelles, nous pouvons revenir sur la remarque C.24.

Exemple C.51. Soit S la série

$$\frac{X+5X^2}{1+X-5X^2-125X^3}.$$

Clairement, cette série est $\mathbb{Z}$-rationnelle par la proposition C.22.

Pour trouver les coefficients de cette série, nous commençons par la décomposer en fractions simples. Le polynôme au dénominateur admet $\frac{1}{5}$ comme racine. De là, on peut le factoriser par la formule de Horner puis en trouvant ses deux autres racines, qui sont $\frac{-1}{3+4i}$ et $\frac{-1}{3-4i}$. On sait donc qu'il existe trois constantes complexes a, b et c telles que

$$\frac{X+5X^2}{1+X-5X^2-125X^3} = \frac{a}{X-\frac{1}{5}} + \frac{b}{X-\frac{-1}{3+4i}} + \frac{c}{X-\frac{-1}{3-4i}},$$

ou encore qu'il existe trois nouvelles constantes a, b, c telles que

$$\frac{X+5X^2}{1+X-5X^2-125X^3} = \frac{a}{5X-1} + \frac{b}{(3+4i)X+1} + \frac{c}{(3-4i)X+1}.$$

En mettant les expressions au même dénominateur et en égalant les termes de même degré, on trouve

$$\frac{X+5X^2}{1+X-5X^2-125X^3} = \frac{-1/8}{5X-1} + \frac{-1/16}{(3+4i)X+1} + \frac{-1/16}{(3-4i)X+1}.$$

Le lemme C.42 permet alors d'écrire

$$\begin{aligned}\frac{X+5X^2}{1+X-5X^2-125X^3} &= \frac{1}{8}\sum_{j=0}^{\infty} 5^j X^j + \frac{-1}{16}\sum_{j=0}^{\infty}(-3-4i)^j X^j + \frac{-1}{16}\sum_{j=0}^{\infty}(-3+4i)^j X^j \\ &= \frac{1}{16}\sum_{j=0}^{\infty}\left(2\cdot 5^j - (-3-4i)^j - (-3+4i)^j\right) X^j.\end{aligned}$$

On remarque alors que chaque coefficient de cette série est positif, puisque

$$5^j \geq \Re((-3-4i)^j) \text{ et } 5^j \geq \Re((-3+4i)^j).$$

Cette série est donc $\mathbb{Z}$-rationnelle, à coefficients positifs. Nous montrons cependant qu'elle n'est pas $\mathbb{N}$-rationnelle. Si elle l'était, on pourrait appliquer la proposition C.50, car S n'est clairement pas un polynôme. Les pôles de S sont, on l'a vu, $\frac{1}{5}$, $\frac{-1}{3+4i}$ et $\frac{-1}{3-4i}$, qui sont tous de module $\frac{1}{5}$. Pour montrer que S n'est pas $\mathbb{N}$-rationnel, il nous suffit alors de montrer que $\frac{-5}{3+4i}$ n'est pas une racine de l'unité, car cela contredira la proposition C.50. L'ensemble des racines de l'unité étant stable par passage à l'inverse, à l'opposé et au conjugué, nous montrons maintenant que $\frac{3+4i}{5}$ n'est pas une racine de l'unité.

Pour cela nous remarquons que

$$(3+4i)^2 = -7+24i \text{ et } (3+4i)^3 = -117+44i.$$

Ceci nous pousse à montrer, en s'inspirant de l'arithmétique modulaire,

$$\forall j \in \mathbb{N}\, \exists a_j, b_j \in \mathbb{Z} : (3+4i)^j = (3+4i) + 5(a+bi).$$

Cette propriété se montre sans mal par récurrence sur j. Comme 4 n'est pas multiple de 5, il est impossible qu'il existe j tel que $(3+4i)^j = 5^j + 0i$, et il n'existe donc pas de j tel que $\left(\frac{3+4i}{5}\right)^j = 1$. Ceci conclut le raisonnement et l'exemple.

Références

[1] Alfred V. Aho, Monica S. Lam, Ravi Sethi, et Jeffrey D. Ullman, *Compilers : principles, techniques, and tools*, Addison-Wesley series in computer science, Addison-Wesley Publishing Company, Reading, 1986.

[2] Shigeki Akiyama, Christiane Frougny, et Jacques Sakarovitch, *Powers of rationals modulo 1 and rational base number systems*, Israel Journal of Mathematics **168** (2008), n° 1, 53–91.

[3] Jacques Bernoulli, *Ars conjectandi, opus posthumum. accedit tractatus de seriebus infinitis, et epistola gallicé scripta de ludo pilae reticularis*, Impensis Thurnisorium fratrum, Basel, 1713.

[4] Jean Berstel et Christophe Reutenauer, *Noncommutative rational series with applications*, Encyclopedia of Mathematics and its Applications, vol. 137, Cambridge University Press, 2010.

[5] Valérie Berthé, Christiane Frougny, Michel Rigo, et Jacques Sakarovitch, *The carry propagation of the successor function*, Advances in Applied Mathematics **120** (2020), 102062.

[6] Valérie Berthé et Michel Rigo (éds), *Combinatorics, automata, and number theory*, Encyclopedia of mathematics and its applications, vol. 135, Cambridge University Press, Cambridge, 2010.

[7] Alan Cobham, *On the base-dependence of sets of numbers recognizable by finite automata*, Mathematical Systems Theory **3** (1969), n° 2, 186–192.

[8] ———, *Uniform tag sequences*, Mathematical Systems Theory **6** (1972), n° 1-2, 164–192.

[9] Dominique Foata, *Eulerian polynomials: From Euler's time to the present*, The legacy of Alladi Ramakrishnan in the mathematical sciences, Springer New York, New York, NY, 2010, p. 253–273.

[10] Aviezri S. Fraenkel, *Systems of numeration*, The American Mathematical Monthly **92** (1985), n° 2, 105–114.

[11] Georges Hansoul, *Algèbre II*, Notes de cours, Université de Liège, 2016.

[12] Michaël Hollander, *Greedy numeration systems and regularity*, Theory Comput. Systems **31** (1998), n° 2, 111–133.

[13] Stephen C Kleene, *Representation of events in nerve nets and finite automata*, Rand. Corporation, 1951.

[14] Konrad Knopp, *Theory and application of infinite series*, 2e éd., Blackie, London, 1951.

[15] Serge Lang, *Linear algebra*, 2e éd., Addison-Wesley world student series, Addison-Wesley, Reading, 1971.

[16] Pierre Lecomte et Michel Rigo, *Numeration systems on a regular language*, Theory Comput. Systems **34** (2001), 27–44.

[17] Victor Marsault, *Enumeration and numeration*, Thèse de doctorat, Télecom-Paristech, 2016.

[18] Victor Marsault et Jacques Sakarovitch, *Trees and languages with periodic signature*, Indagationes mathematicae **28** (2017), n° 1, 221–246.

[19] Michel Rigo, *Algèbre linéaire*, Notes de cours, Université de Liège, 2009-2010.

[20] ———, *Théorie des automates et langages formels*, Notes de cours, Université de Liège, 2009-2010.

[21] ———, *Formal languages, automata and numeration systems*, Networks and telecommunications series, ISTE, London, 2014.

[22] Eugene Seneta, *Non-negative matrices and Markov chains*, Springer New York, New York, 1981.

[23] J Shallit, *Numeration systems, linear recurrences, and regular sets*, Information and computation **113** (1994), n° 2, 331–347.

[24] Arseny M Shur, *Combinatorial complexity of regular languages*, Computer science - theory and applications, Lecture Notes in Computer Science, vol. 5010, Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, p. 289–301.

[25] Stephen Willard, *General topology*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1970.

[26] Edouard Zeckendorf, *Représentation des nombres naturels par une somme de nombres de Fibonacci ou de nombres de Lucas*, Bull. Soc. Roy. Sci. Liège **41** (1972), 179–182.