N° d'ordre : 1898

THÈSE

PRÉSENTÉE À

L'UNIVERSITÉ BORDEAUX I

ÉCOLE DOCTORALE DE MATHÉMATIQUES ET D'INFORMATIQUE

Par Philippe DUCHON

POUR OBTENIR LE GRADE DE

DOCTEUR

SPÉCIALITÉ : INFORMATIQUE

Q-grammaires: un outil pour l'énumération

Soutenue le : 5 juin 1998

Après avis de : MM. Jean-Marc Fédou, Renzo Pinzani ... Rapporteurs

Devant la Commission d'examen formée de :

MM.	Philippe Flajolet	Directeur de recherche à l'INRIA	Président
	Mireille Bousquet-Mélou	Chargée de recherche au CNRS .	Rapporteur
	Maylis Delest	Professeur	Examinateurs
	Jean-Marc Fédou	Professeur	
	Daniel Krob	Directeur de recherche au CNRS	
	Renzo Pinzani	Professeur	

– 1998 –

Les travaux de Philippe Flajolet ont souvent été pour moi une prćieuse source d'inspiration; c'est un grand honneur qu'il me fait en présidant ce jury, et je l'en remercie vivement.

Maylis Delest a su, dans un emploi du temps toujours chargé, trouver le temps de me prodiguer conseils et encouragements, tout en me laissant la liberté d'avancer à mon propre rythme. Je tiens à l'en remercier, et à lui dire tout le plaisir que j'ai eu à travailler avec elle.

Jean-Marc Fédou et Renzo Pinzani ont lu avec attention mon manuscrit, et m'ont offert conseils et suggestions. Je leur en suis reconnaissant.

Mireille Bousquet-Mélou et Daniel Krob ont également accepté de faire partie de ce jury, et je les en remercie.

Xavier Viennot, par son enthousiasme et la clarté de ses cours, m'a fait découvrir la combinatoire et m'a attiré à Bordeaux; je profite de cette occasion pour lui exprimer toute ma gratitude.

Mes divers camarades de bureau, et tout particulièrement Emmanuel Godard et Augustin Ido, ont supporté avec courage mes moments de découragement tout au long de la rédaction de cette thèse; je leur souhaite de se trouver, le moment venu, dans une aussi bonne ambiance.

Marie-Line sait déjà combien sa présence et son soutien m'ont apporté, mais je le lui redis tout de même.

Table des matières

Introduction

L'objet de la combinatoire énumérative peut être résumé ainsi : déterminer, de manière exacte ou approchée, le *nombre* d'objets vérifiant des propriétés données. Le plus souvent, on s'intéresse à une classe infinie $\mathcal{A}$ d'objets (figures planes, mots d'un langage, cartes, arbres...), sur lesquels on définit un certain nombre de *paramètres* : nombre de sommets, hauteur, largeur, nombre de sous-arbres vérifiant telle ou telle propriété ... Les problèmes peuvent provenir de l'informatique théorique (analyse en moyenne ou dans le cas le pire d'algorithmes et de structures de données), des mathématiques (repésentations du groupe symétrique, bases de fonctions symétriques ...), ou de la physique statistique (modèles discrets de percolation, fonctions de partitions ...). Bien que d'origines et souvent de natures très différentes, ces problèmes ont souvent la caractéristique commune de pouvoir être modélisés par des objets relativement simples, qui se prêtent bien à l'énumération par des techniques combinatoires.

Lorsqu'un paramètre p est suffisamment discriminant pour que, pour chaque valeur possible n de ce paramètre, l'ensemble $\mathcal{A}_n$ des objets de $\mathcal{A}$ pour lesquels le paramètre prend cette valeur soit *fini*, on peut alors se poser la question de savoir *combien* il y a de tels objets. Cette énumération peut être soit exacte, soit approchée. Dans ce dernier cas, on donne un développement asymptotique ou un simple équivalent de la suite $(|\mathcal{A}_n|)_{n\geq 0}$.

L'outil le plus fréquemment utilisé dans les problèmes d'énumération est la *série génératrice* : à chaque objet w d'une classe $\mathcal{A}$, on associe un *poids*, ou *valuation*, $v(w)$, pris dans un anneau de polynômes $\mathbb{K}[x_1, \ldots, x_k]$; la série génératrice de la classe $\mathcal{A}$ suivant la valuation v est alors, sous réserve de convergence, la somme formelle des poids des objets

$$A(x_1, \ldots, x_k) = \sum_{w\in\mathcal{A}} v(w).$$

Généralement, on associe à chacune des variables x_i, un paramètre p_i, défini sur $\mathcal{A}$ et ne prenant que des valeurs positives ou nulles. La valuation v est alors définie par

$$v(w) = \prod_{1\leq i\leq k} x_i^{p_i(w)},$$

et la série génératrice qui en résulte est appelée *série génératrice ordinaire*[1] *de $\mathcal{A}$ suivant les paramètres* $p_1, \dots, p_k$.

Savoir ce qui constitue une solution d'un problème d'énumération n'est pas forcément évident. La réponse dépend fortement de la complexité du problème d'énumération. Ce peut être une formule donnant la série génératrice, ou une autre donnant les coefficients de Taylor de cette série; ou même une simple équation portant sur la série génératrice, ou une relation de récurrence vérifiée par ses coefficients de Taylor. De manière approchée, on pourra se concentrer sur un équivalent asymptotique de la série ou de ses coefficients. Les travaux d'Odlyzko [64], Flajolet et Odlyzko [41], et Flajolet et Sedgewick [45, 46], montrent en quoi le comportement asymptotique des coefficients de Taylor d'une série est lié à son comportement au voisinage de ses singularités dominantes. Pour les problèmes d'origine physique, où les objets combinatoires sont souvent des approximations discrètes de modèles continus, le comportement asymptotique est généralement plus significatif que l'énumération exacte.

Lorsque les séries génératrices sont données par des équations, il est usuel de classifier les problèmes suivant le *type* de ces équations : algébriques, différentielles, fonctionnelles ... Un phénomène fréquemment rencontré pour les séries génératrices à plusieurs variables est qu'une des variables q apparaisse de telle sorte qu'à la limite $q \to 1$, l'équation se transforme en une équation beaucoup plus simple; ces équations sont appelées *q-équations.*

Le travail présenté dans cette thèse se situe dans le cadre de la combinatoire énumérative. Nous formalisons la notion de Q-grammaires, et montrons en quoi elles constituent un moyen d'approche de certains problèmes d'énumération suivant plusieurs paramètres.

Différentes méthodes d'énumération

Une première méthode d'approche possible consiste à calculer par des méthodes appropriées les premières termes de la série génératrice (Redelmeier [72]). Souvent, les objets de "petite taille" sont peu nombreux, et il est ainsi possible d'obtenir plusieurs termes de la suite des coefficients de Taylor. Dans certains cas, une simple comparaison avec un ensemble de suites connues [75, 76] permet d'identifier la suite. Des techniques plus évoluées d'approximation [17] recherchent, à partir des premiers termes, une équation algébrique ou différentielle susceptible d'être satisfaite par la série génératrice. Ces calculs sont automatisés par la bibliothèque Maple *gfun* [43].

1. *Une autre forme fréquemment employée de série génératrice est la série génératrice* exponentielle*, dans laquelle la valuation d'un objet de "taille" n est divisée par $n!$.*

Pour certains problèmes, il est plus facile de chercher à obtenir directement une équation fonctionnelle satisfaite par la série génératrice. Le principe de la combinatoire *bijective* est d'exhiber entre deux classes d'objets combinatoires une bijection qui conserve la valuation; on en déduit alors l'égalité des séries génératrices. Le plus souvent, on s'attache à "décomposer" les objets étudiés de manière à établir une bijection avec une classe d'objets dont l'énumération directe est possible. Lorsqu'une telle bijection est établie entre deux classes d'objets dont les séries génératrices sont déjà connues, elle fournit alors une *preuve bijective* de l'identité des séries, que cette identité ait été ou non préalablement démontrée par le calcul. L'intérêt d'une telle preuve bijective réside généralement dans le fait qu'elle permet en quelque sorte d'"expliquer" certaines propriétés des objets étudiés, en les reliant à des propriétés connues d'autres objets.

Les polyominos verticalement convexes peuvent naturellement se découper en "tranches" successives. En tenant compte de différents paramètres (aire, largeur, périmètre, et hauteur de la dernière tranche), il est possible de dire comment évoluent ces paramètres lorsqu'une nouvelle colonne est ajoutée. On en déduit une équation fonctionnelle vérifiée par la série génératrice ou, de manière équivalente, des relations de récurrence portant sur les coefficients. Cette méthode "à la Temperley" [77, 14] permet ainsi d'étudier différentes classes de polyominos verticalement convexes. La méthode ECO, qui combine ces idées avec l'utilisation d'arbres de génération, a également été appliquée à l'énumération de diverses classes de chemins colorés [6] et d'arbres planaires [7]

Dans la méthodologie DSV, développée initialement par Schützenberger dans [73, 74], on relie l'algébricité de certaines séries génératrices à la théorie des *langages algébriques* non ambigus [1, 5]. Le principe est d'établir une bijection (ou *codage*) entre les objets à étudier et les mots d'un langage *algébrique*, de telle sorte que le paramètre *taille* des objets corresponde à la *longueur* des mots; éventuellement, les différentes lettres de l'alphabet peuvent correspondre à différents paramètres intéressants. Une grammaire non ambiguë engendrant le langage, fournit alors, d'après un théorème de Chomsky et Schützenberger [18], un système d'équations algébriques dont la série génératrice cherchée est une des composantes d'une solution. Ce système peut alors être résolu explicitement, ou, s'il est trop complexe, des informations précises sur les coefficients de la série génératrice peuvent en être extraites. Dans les cas simples, la formule d'inversion de Lagrange [81] ou l'une de ses variantes [53, 50], permet d'obtenir des expressions exactes pour les coefficients; dans d'autres, où aucune formule acceptable n'est accessible, l'analyse des singularités peut donner des renseignements précis sur l'asymptotique des coefficients.

Enfin, les coefficients de Taylor de séries algébriques vérifiant des relations de récurrence polynomiales (Comtet [21]), il est possible de calculer explicitement un grand nombre de coefficients.

L'énumération des polyominos convexes suivant le périmètre, par Delest et Viennot [31], constitue un exemple de résultat nouveau obtenu par l'application de cette méthode. On trouvera d'autres exemples dans [30, 25].

Les *grammaires à opérateurs* (Cori et Richard [24], Cori [23], Chottin [19]) constituent une variante de la méthodologie DSV, et permettent de traiter certains cas où l'on code les objets par les mots d'un langage qui n'est pas forcément algébrique, mais qui est solution d'un système d'équations avec opérateurs en variables non commutatives. Si les opérateurs employés ont une image en variables commutatives, cette équation se traduit directement sur la série génératrice. Cette méthode a notamment permis à Cori et Richard de retrouver certains résultats d'énumération de cartes planaires dûs à Tutte [78].

Il est possible de s'affranchir du passage par les mots en remarquant qu'à certaines opérations sur les objets, correspondent des opérations sur les séries génératrices. Ainsi, l'union disjointe de deux classes d'objets correspond à la somme des séries génératrices, et le produit cartésien, au produit des séries. Ces idées sont utilisées dans la théorie des *structures décomposables* [42, 43, 44, 47], ou, de manière plus visuelle, dans les *grammaires d'objets* [36].

Une même classe d'objets, énumérée suivant des paramètres différents, admet généralement des séries génératrices qui n'ont aucune ressemblance entre elles. Obtenir la série génératrice suivant *plusieurs* paramètres est souvent beaucoup plus compliqué; dès lors que la série génératrice suivant l'un de ces paramètres n'est pas algébrique, la série multivariée n'est pas algébrique, et la méthodologie DSV ne peut s'appliquer directement. Par ailleurs, il arrive que les différentes séries à une variable soient algébriques sans que la série multivariée le soit. Ainsi, les *polyominos verticalement convexes* ont une série génératrice suivant l'aire qui est *rationnelle* (Temperley [77], Klarner [58, 59]), et une série génératrice suivant le périmètre *algébrique* (Delest [25], Feretić [38, 39]), mais la série génératrice bivariée suivant ces deux paramètres est beaucoup plus compliquée, et n'est pas algébrique (Bousquet-Mélou [14]).

La notion de q-grammaire, introduite par Delest et Fédou dans [28], repose sur l'idée que le codage par les mots d'un langage algébrique fournit en fait une *structure* sur les objets codés. En adaptant la notion de *grammaire d'attributs* (Knuth [61]) utilisée en compilation, il est parfois possible d'écrire des équations *non algébriques* vérifiées par la série génératrice suivant un paramètre supplémentaire, compté par une nouvelle variable q. Ces équations sont alors des q-analogues des équations algébriques de départ : une variable x est parfois remplacée par xq dans certaines séries inconnues. En posant $q = 1$ (ce qui revient à "oublier" le paramètre compté par q), on retrouve les équations algébriques. Les séries génératrices solutions de telles équations se présentent donc sous la forme de q-séries [4].

Un exemple d'utilisation de q-grammaires pour l'énumération des polyominos parallélogrammes (suivant l'aire et le périmètre) est donné par Delest et Fédou dans [29]. On trouvera d'autres exemples dans les travaux de Denise et Simion [32] (hauteurs de pyramides et paires extérieures dans les chemins de Dyck), et dans ceux de Delest, Dubernard et Dutour [27] (énumération des polyominos parallélogrammes suivant l'aire, la largeur et le nombre de coins).

La résolution de q-équations dans le cas général est un problème complètement ouvert. Il existe plusieurs q-analogues de la formule d'inversion de Lagrange, dus à Andrews [3], Gessel [49], Garsia [48], Gessel et Stanton [51] ou Krattenthaler [62]. Prellberg et Brak [69] et Bousquet-Mélou [14] ont décrit des méthodes permettant de résoudre des cas particuliers de q-équations. La résolution d'équations q-différentielles a également permis à Bousquet-Mélou et Fédou [16] d'obtenir une expression relativement simple pour la série génératrice des polyominos convexes suivant l'aire et le périmètre.

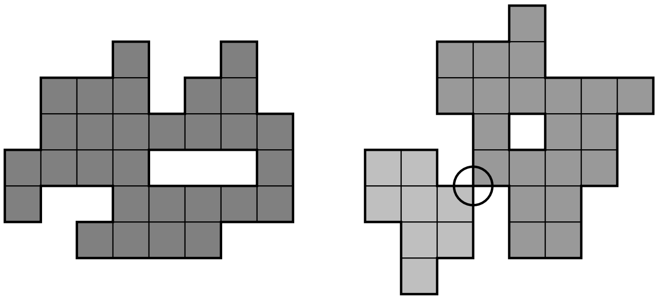
Notre principal objet d'étude, les Q-grammaires, constitue une généralisation de la notion de q-grammaire. Nous nous intéressons à tous les paramètres qu'il est possible d'"attraper" en s'autorisant à multiplier certaines variables d'énumération par d'autres. Ces paramètres, qui dépendent de la grammaire, sont appelés *Q-comptables.*

Polyominos

Nous donnons ici une brève définition des polyominos, car ceux-ci constituent le support de plusieurs exemples étudiés dans ce mémoire.

Les polyominos sont un sujet d'étude fréquent en combinatoire. Les premières recherches ont concerné le pavage de régions du plan au moyen de polyominos donnés (le terme de polyomino est généralement attribué à Golomb [52]). Les problèmes d'énumération ont rapidement été abordés (Temperley [77]; Read [71]; Klarner [58, 59]; Pólya [67]). Des résultats asymptotiques assez précis ont été obtenus pour des sous-classes de polyominos (Bender [9]; Klarner et Rivest [60]).

Une *cellule* est un carré unitaire du plan $\mathbb{R}^2$, dont les sommets ont des coordonnées entières. Un *polyomino* est une union finie de cellules, dont l'intérieur est connexe – voir figure 1.



Un polyomino

Deux polyominos avec un point de contact

FIG. 1: *Exemples de polyominos*

Nous considérerons toujours les polyominos *à translation près*: deux polyominos qui sont images l'un de l'autre par une translation sont pour nous identiques. En revanche, deux polyominos distincts peuvent être images l'un de l'autre par une symétrie ou une rotation.

En physique statistique, où ils sont utilisés comme modèles discrets, les polyominos portent généralement le nom d'*animaux*, et les cellules sont souvent remplacées par leurs centres.

L'énumération des polyominos dans le cas général est un problème extrêmement difficile et complètement ouvert. Le nombre a_n de polyominos d'aire n n'est connu que jusqu'à $n = 24$ (Redelmeier [72]). Le comportement asymptotique de la suite (a_n) n'est connu que de manière partielle : $\lim(a_n)^{1/n} = \mu$, avec $3.72 < \mu < 4.64$.

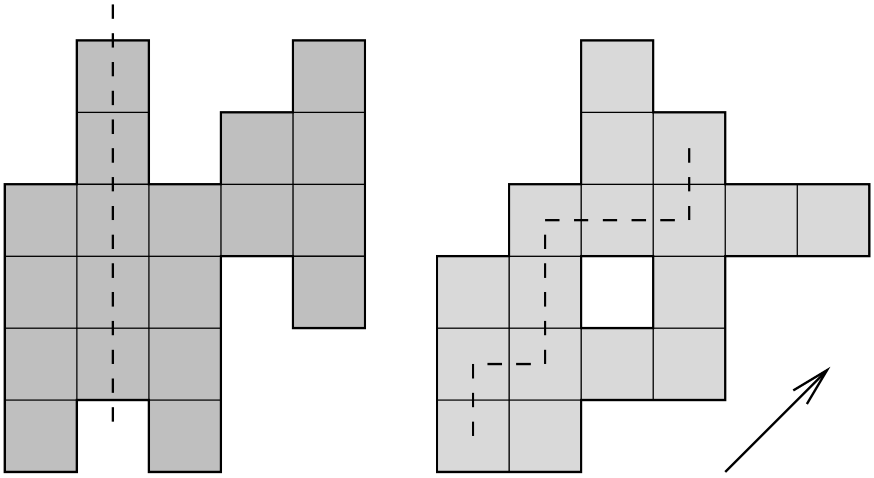
Une manière de rendre abordables les problèmes d'énumération de polyominos est de se restreindre à des sous-classes définies par des propriétés particulières. Les propriétés qui, jusqu'à présent, se sont révélées les plus fécondes (tout au moins du point de vue des résultats d'énumération) sont la *convexité* suivant une direction, et le fait d'être *dirigés* suivant une direction donnée. On trouvera des survols dans [55, 26, 79], et un autre plus récent dans [15], qui contient également une bibliographie détaillée sur le sujet.

Si $\mathcal{D}$ est une direction de droites dans le plan, un polyomino P est *convexe* suivant la direction $\mathcal{D}$ si son intersection avec toute droite de $\mathcal{D}$ qui passe par le centre d'une cellule est un segment. Le plus souvent, la direction $\mathcal{D}$ est soit verticale, soit horizontale[2];

2. Bien évidemment, une symétrie transforme bijectivement les polyominos verticalement convexes en polyominos horizontalement convexes.

les polyominos *diagonalement convexes* ont également été abordés. Un polyomino à la fois horizontalement et verticalement convexe est dit tout simplement *convexe*.

Un polyomino P est *dirigé* suivant la direction Sud-Ouest/Nord-Est s'il existe une cellule $c \subset P$, appelé *cellule source*, telle que chaque cellule de P peut être atteinte à partir de c en n'effectuant que des pas élémentaires Nord ou Est, sans passer par une cellule extérieure à P. Si l'on autorise également les pas Sud, le polyomino est dit *semi-dirigé* suivant la direction Ouest-Est.



verticalement convexe dirigé Sud-Ouest/Nord-Est

FIG. 2: *Exemples de polyominos avec contraintes*

En combinant les conditions de convexité et de directions privilégiées, il est possible de définir la plupart des familles de polyominos étudiées dans la littérature. Ainsi, les polyominos *parallélogrammes* sont exactement ceux qui sont verticalement et diagonalement convexes, et dirigés suivant les directions Sud-Ouest/Nord-Est et Nord-Ouest/Sud-Est; les polyominos *murs*, qui correspondent aux *compositions* d'entiers, sont exactement les polyominos verticalement convexes qui sont dirigés à la fois vers le Nord-Est *et* vers le Nord-Ouest.

Plan de la thèse

Le chapitre 1 introduit la plupart des définitions et notations classiques utilisées. La seule notion qui diffère sensiblement de sa définition usuelle est celle des arbres de dérivation d'une grammaire, qui seront pour nous étiquetés par les règles de réécriture.

Le chapitre 2 introduit les notions de *Q-grammaire* et de *paramètre Q-comptable*.

Le résultat principal est le théorème 2.36, qui établit l'équivalence entre Q-grammaires et certaines formes d'équations vérifiées par les séries génératrices. Les paramètres Q-comptables sont également interprétés en termes d'arbres de dérivation, et leurs ordres de grandeur maximaux sont déterminés.

Le chapitre 3 étudie dans quelles circonstances il est possible de remplacer une grammaire par une autre engendrant le même langage, sans perdre de paramètres Q-comptables. Au moyen de différents lemmes, nous montrons en particulier qu'il est possible de donner des formes normales aux Q-grammaires.

Le chapitre 4 est consacré aux calculs de statistiques sur les paramètres Q-comptables. Par différentiation, les séries de moments sont décrites (théorème 4.14). Les problèmes d'asymptotiques sont également abordés.

Le chapitre 5 donne, à titre d'exemples d'applications, des Q-grammaires pour différentes familles de polyominos verticalement convexes. Dans chaque cas, nous étudions un certain nombre de paramètres et déterminons ceux qui sont Q-comptables pour les grammaires données.

Chapitre 1

Définitions

Dans ce chapitre, nous nous contentons de définir les objets de base utilisés dans ce travail : langages, arbres, séries génératrices, grammaires, et chemins du plan discret. Seuls les arbres de dérivation diffèrent légèrement de leur définition la plus classique.

1.1 Mots et langages

Définition 1.1. Pour tout ensemble X, X^* désigne l'ensemble des suites finies (éventuellement vides) d'éléments de X. Une telle suite w est également appelée un *mot* de X^*, et sa longueur sera notée $|w|$. Chaque terme de la suite est appelé une *lettre.*

L'opération de concaténation fait de X^* un monoïde (monoïde libre engendré par X).

La suite (unique) de longueur nulle, aussi appelée *mot vide*, sera notée ϵ.

Définition 1.2. On appelle *langage* sur X une partie L de X^*; X est alors appelé l'*alphabet* de L.

Nous ne nous intéresserons qu'à des langages sur des alphabets finis. Il existe une classification extrêmement détaillée pour ces langages; voir à ce sujet [11]. Dans ce travail, nous nous intéresserons plus particulièrement aux langages *algébriques*, définis à la section 1.4.

Définition 1.3. Un mot w' est un *facteur* d'un mot w, s'il existe des mots w_1 et w_2 tels que $w = w_1 w' w_2$.

Le mot w' est un *facteur gauche* si $w_1 = \epsilon$, et un *facteur droit* si $w_2 = \epsilon$. Un facteur de w qui n'est ni w, ni le mot vide, est appelé *facteur propre.*

De manière plus générale, on peut parler de *factorisation* :

Définition 1.4. Une *factorisation* d'un mot w est une suite finie de mots $(w_1, \ldots, w_k)$, telle que $w = w_1 w_2 \ldots w_k$. L'entier k est la *longueur* de la factorisation.

Lorsqu'aucun des mots w_i n'est vide, la factorisation est dite *propre*.

Ainsi, un facteur propre de w est un facteur qui peut apparaître dans une factorisation propre non triviale (dont la longueur n'est pas 1).

Définition 1.5. Un *sous-mot* d'un mot $w = x_1 \ldots x_n$ $(x_i \in X)$ est une sous-suite $w' = x_{i_1} \ldots x_{i_k}$, avec $i_1 < i_2 < \ldots < i_k$.

Il est clair qu'un facteur d'un mot w, n'est rien d'autre qu'un sous-mot pour lesquels les indices $i_1, \ldots, i_k$ sont consécutifs.

Définition 1.6.

- Le *nombre d'occurrences* d'un *facteur* w' de w, est le nombre de factorisations distinctes $F = (w_1, w', w_2)$ de w.
- Le *nombre d'occurrences* d'un *sous-mot* w' de $w = x_1 x_2 \ldots x_n$, est le nombre de sous-suites $i_1 < \ldots < i_k$ de $1, \ldots, n$, telles que $w' = x_{i_1} \ldots x_{i_k}$.

Exemple 1.7. Dans le mot $w = aababbab$, il y a 3 occurrences du facteur ab, mais 12 occurrences du sous-mot ab.

Un exemple classique de nombre d'occurrences de sous-mots est celui du nombre d'inversions :

Définition 1.8. Supposons l'alphabet X totalement ordonné. Une *inversion* d'un mot $w = x_1 \ldots x_n$, est un couple (i, j) tel que $i < j$ et $x_i > x_j$.

Le nombre d'inversions d'un mot w est noté $\mathrm{inv}(w)$.

Le nombre d'inversions d'un mot est la somme des nombres d'occurrences de tous les sous-mots possibles de longueur 2 dont les lettres sont décroissantes. Ainsi, dans le cas d'un alphabet à deux lettres $\{a, b\}$ (avec $a < b$), il s'agit tout simplement du nombre d'occurrences du sous-mot ba.

Notation 1.9. Soient $w \in X^*$, $x \in X$, et $A \subset X$. Le nombre d'occurrences de x dans w est noté $|w|_x$, et également appelé *longueur de w en x*.

De même, le nombre de lettres de w qui appartiennent à A est appelé *longueur de w en A*, et noté $|w|_A$.

1.2 Arbres

Il existe différentes catégories d'arbres en informatique; dans cette thèse, le terme d'arbre désigne exclusivement un arbre planaire, généralement étiqueté.

1.2.1 Arbres planaires

Définition 1.10. Soit S un ensemble fini non vide. Un *arbre d'ensemble de sommets S* est défini de la manière suivante :

- Si S est un singleton, $\mathcal{A} = (s)$ est le seul arbre d'ensemble de sommets S.
- Sinon, $\mathcal{A} = (s, \mathcal{A}_1, \dots, \mathcal{A}_k)$, où $s \in S$, et pour chaque $i \leq k$, $\mathcal{A}_i$ est un arbre d'ensemble de sommets S_i, avec la condition que $\{S_1, \dots, S_k\}$ forme une partition de $S \backslash \{s\}$.

Dans les deux cas, s est appelé la *racine* de $\mathcal{A}$.

Avec les notations de la définition 1.10, les éléments de S sont appelés *sommets* ou *nœuds* de $\mathcal{A}$. La *taille* de $\mathcal{A}$ est son nombre de sommets.

Lorsque $\mathcal{A} = (s, \mathcal{A}_1, \dots, \mathcal{A}_k)$ est un arbre, les racines respectives $s_1, \dots, s_k$ des arbres $\mathcal{A}_1, \dots, \mathcal{A}_k$ sont appelés *fils* de s, et s est leur *père* commun. $\mathcal{A}_i$ est appelé le *i-ème sous-arbre* de $\mathcal{A}$, ou encore le *sous-arbre issu de s_i*; de manière générale, un *sous-arbre de $\mathcal{A}$* est le sous-arbre issu d'un sommet quelconque de $\mathcal{A}$.

Traditionnellement, les arbres sont représentés par des graphes, chaque sommet étant relié à ses fils ordonnés de gauche à droite et placés en dessous de lui, de sorte que la racine est placée en haut de la figure (voir figure 1.1).

Un sommet d'un arbre qui n'a aucun fils est appelé une *feuille*. Les sommets qui ne sont pas des feuilles sont parfois appelés sommets ou nœuds *internes*.

Une *branche* d'un arbre est une suite $(s_1, \dots, s_k)$ de sommets telle que, pour $1 \leq i < k$, s_{i+1} soit un fils de s_i, s_1 étant la racine de l'arbre. L'entier k est la *longueur* de la branche. Pour chacun des sommets s d'un arbre, il existe une unique branche qui se termine en s. La *hauteur* de l'arbre $\mathcal{A}$ est la plus grande longueur de ses branches.

Si $(s_1, \dots, s_k)$ est une branche d'un arbre, k est la *profondeur* du sommet s_k dans cet arbre.

Une branche est *maximale* (pour l'inclusion) si son dernier sommet est une feuille de l'arbre.

Les *descendants* d'un sommet s sont tous les sommets du sous-arbre issu de s; les *ancêtres* de s sont tous les sommets qui composent la branche dont il est le dernier sommet. En ce sens, un sommet est son propre ancêtre et son propre descendant.

Si $\mathcal{A}$ et $\mathcal{B}$ sont deux arbres, et si s est un sommet de $\mathcal{A}$, on notera $\mathcal{A}(s,\mathcal{B})$ l'arbre obtenu en remplaçant, dans $\mathcal{A}$, le sous-arbre de racine s par $\mathcal{B}$ (voir figure 1.1).

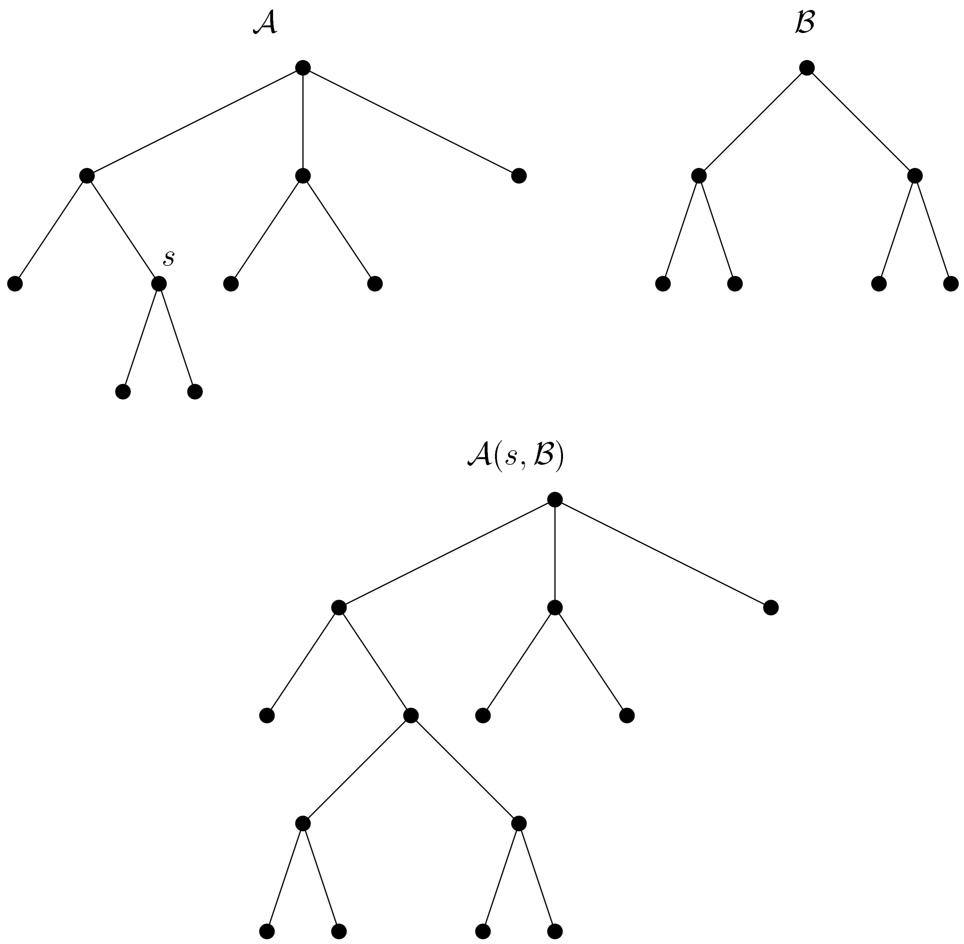


FIG. 1.1: *Substitution d'arbres*

Dans le cadre de cette thèse, les arbres manipulés sont exclusivement des *arbres de dérivation* dans une grammaire; voir section 1.4.

1.3 Séries formelles et séries génératrices

Soient $X = \{x_1, \ldots, x_k\}$ un alphabet fini, et A un anneau unitaire; $A \ll X \gg$ désigne l'algèbre des *séries formelles à variables non commutatives dans X et à coefficients dans*

A. L'algèbre des *séries formelles à variables commutatives dans X et à coefficients dans A* sera notée $A[[X]]$. Dans la pratique, A sera $\mathbb{Z}$, $\mathbb{Q}$, ou $\mathbb{C}$.

Pour chaque k-uplet $\mathbf{n} = (n_1, \ldots, n_k) \in \mathbb{N}^k$, nous notons $\mathbf{x}^\mathbf{n}$ le monôme $x_1^{n_1} \ldots x_k^{n_k}$.

A toute série formelle à variables non commutatives correspond naturellement une série à variables commutatives, obtenue par le morphisme χ_0 qui "fait commuter" les lettres.

Un langage L peut être identifié à la série formelle en variables non commutatives $\sum_{w \in L} w$ (les coefficients sont 1 pour les mots de L et 0 pour les autres). La série $\chi_0(L)$ est alors appelée *série génératrice* du langage L; le coefficient de $\mathbf{x}^\mathbf{n}$ est le nombre de mots de L qui ont, pour $1 \leq i \leq k$, exactement n_i occurrences de la lettre x_i. Nous utiliserons fréquemment la même lettre pour désigner un langage et sa série génératrice.

De manière plus générale, soit L un ensemble dénombrable d'"objets" combinatoires, et soient $p_1, \ldots, p_k$ des *paramètres* positifs ou nuls définis sur L (chaque p_i est une application de L dans $\mathbb{N}$). La *série génératrice de L suivant les paramètres* $(p_i)_{1 \leq i \leq k}$ sera alors la série formelle à variables commutatives

$$L(x_1, \ldots, x_k) = \sum_{w \in L} x_1^{p_1(w)} \cdots x_k^{p_k(w)}$$

Pour qu'une telle série soit définie, il faut que, pour chaque k-uplet $(v_1, \ldots, v_k)$, l'ensemble des objets pour lesquels chaque paramètre p_i vaut v_i soit un ensemble fini.

Définition 1.11. Un paramètre λ, défini sur un ensemble d'objets A, est une *taille* si, pour chaque valeur de n, l'ensemble

$$A_n = \{a \in A, \lambda(a) = n\}$$

est fini.

L'existence d'une taille parmi les paramètres $p_1, \ldots, p_k$, est une condition suffisante pour que la série génératrice suivant ces paramètres soit définie. Le paramètre "longueur totale" est toujours une taille pour les mots d'un langage, les alphabets étant toujours finis.

La série génératrice d'un langage suivant les paramètres "nombres d'occurrences des différentes lettres" est la série $\chi_0(L)$, que nous avons appelée plus haut sa série génératrice. Il est possible qu'aucun paramètre "nombre d'occurrences de la lettre x_i" ne soit une taille : il peut parfaitement y avoir, dans le langage considéré, une infinité de mots ne comportant pas la lettre x_i, et ce, pour chaque x_i. Cependant, la série génératrice est toujours bien définie, le nombre de mots du langage ayant une composition fixée étant fini.

Il arrivera fréquemment, dans le cours de cette thèse, que nous manipulions des combinaisons linéaires formelles de mots. Nous travaillons alors implicitement dans l'algèbre $\mathbb{Q} < X >$ des polynômes à variables non commutatives dans X et à coefficients dans $\mathbb{Q}$. Le plus souvent, une application $\varphi : X \to \mathbb{Q} < Y >$ sera définie, et implicitement étendue à $\mathbb{Q} < X >$ comme morphisme d'algèbres : d'abord à X^* par concaténation, puis à $\mathbb{Q} < X >$ par combinaisons linéaires.

Exemple 1.12. Soient $X = \{a, b\}$ et $Y = \{a, a', b\}$. Si φ est définie par

$$\begin{cases} \varphi(a) & = & a + a' \\ \varphi(b) & = & \epsilon + b, \end{cases}$$

alors on a, par exemple,

$$\begin{array}{rcl} \varphi(aba + aa) & = & \varphi(aba) + \varphi(aa) \\ & = & \varphi(a)\varphi(b)\varphi(a) + \varphi(a)\varphi(a) \\ & = & (a + a')(\epsilon + b)(a + a') + (a + a')(a + a') \\ & = & 2aa + 2aa' + 2a'a + 2a'a' + aba + aba' + a'ba + a'ba'. \end{array}$$

1.4 Grammaires

Définition 1.13. Une *grammaire hors-contexte (context-free)* est un quadruplet $G = (X, N, \mathcal{R}, S)$, où :

- X est un ensemble fini appelé *alphabet*. Les éléments de X, que dans la mesure du possible nous noterons par des minuscules, sont appelés *lettres* (ou *symboles terminaux*).
- N est un ensemble fini disjoint de X, appelé *alphabet des symboles*; les éléments de N (normalement notés par des majuscules) sont appelés *symboles non terminaux*, ou *non terminaux*, ou *symboles*.
- $\mathcal{R}$ est une partie finie de $N \times (X \cup N)^*$, dont chaque élément est appelé *règle de dérivation* ou *transition*. Pour plus de clarté, les règles de dérivation seront présentées sous la forme $U \to W$, avec $U \in N$ et $W \in (X \cup N)^*$.
- S est un symbole non terminal appelé *axiome*.

Il pourra arriver que nous définissions une grammaire sans préciser son axiome, ou que nous changions cet axiome.

Pour chaque règle de dérivation $R = (U \to W)$, U est appelé *membre gauche* de R, et noté $g(R)$. Le *membre droit* de R, noté $d(R)$, est W. Le nombre de symboles non terminaux de W est appelé l'*arité* de la règle, et noté $\alpha(R)$. Une règle d'arité 0 est dite *terminale*. Enfin, $d(R,i)$ désigne le i-ème symbole du membre droit de R.

Nous écrirons fréquemment nos règles sous la forme $R : U \to w_0 U_1 w_1 \dots U_k w_k$; une telle notation suppose implicitement $w_i \in X^*$ et $U_i \in N$, de telle sorte que $d(R,i) = U_i$. Les symboles du membre droit seront systématiquement numérotés de gauche à droite.

Pour chaque symbole U, les règles de dérivation ayant U comme membre gauche sont appelées *U-dérivations*. L'ensemble des U-dérivations est noté $\mathcal{R}_U$.

Dans une grammaire, chaque règle de dérivation $R = (U \to V)$ définit sur $(X \cup N)^*$ une relation binaire $\stackrel{R}{\to}$ de la manière suivante : pour tous mots W_1 et W_2 de $(X \cup N)^*$, $W_1 U W_2 \stackrel{R}{\to} W_1 V W_2$; en d'autres termes, $W \stackrel{R}{\to} W'$ si l'on peut obtenir W' en remplaçant, dans W, une occurrence de U (le "membre gauche" de la règle de dérivation) par V (le "membre droit"). On dit alors que R *réécrit* ou *dérive* W en W'.

La réunion de toutes les relations $\stackrel{R}{\to}$ pour $R \in \mathcal{R}$ est notée $\stackrel{G}{\to}$, et la clôture transitive de $\stackrel{G}{\to}$ est notée $\stackrel{*}{\to}$. On a $W \stackrel{G}{\to} W'$ si et seulement si il existe une règle R qui réécrit W en W', et $W \stackrel{*}{\to} W'$ si et seulement si il existe une suite finie de règles qui, appliquées successivement, réécrivent W en W'.

Définition 1.14. Soit $G = (X, N, \mathcal{R}, S_0)$ une grammaire.

Le *langage engendré par G pour le symbole S* est l'ensemble $L_G(S)$ des mots $w \in X^*$ tels que $S \stackrel{*}{\to} w$.

Le *langage non terminal* engendré par G pour le symbole S est l'ensemble $\tilde{L}_G(S)$ des mots $w \in (X \cup V)^*$ tels que $S \stackrel{*}{\to} w$.

Lorsque le symbole n'est pas précisé, le langage engendré par une grammaire est celui qui est engendré pour l'axiome de la grammaire.

La notion de grammaire non ambiguë, ou grammaire algébrique, est fondamentale lorsqu'il s'agit d'obtenir des résultats d'énumération. Intuitivement, une grammaire est non ambiguë si elle n'engendre chaque mot que d'une seule manière, à certaines permutations inévitables près.

Définition 1.15. Une grammaire $G = (X, N, \mathcal{R}, S)$ est dite *non ambiguë* si, pour chaque symbole $U \in N$ et chaque mot w tel que $U \stackrel{G}{\to} w$, il existe une *unique* suite finie $(W_i)_{0 \le i \le n}$ de mots de $(X \cup N)^*$ telle que $U = W_0$, $W_n = w$, et, pour chaque $i \le n-1$, $W_i \stackrel{G}{\to} W_{i+1}$, avec la condition que W_{i+1} est obtenu en réécrivant dans W_i le *premier* symbole non terminal.

La non-ambiguïté d'une grammaire s'exprime beaucoup plus simplement par le fait que chaque mot d'un langage engendré par une grammaire non ambiguë possède un unique arbre de dérivation; cette propriété est valable aussi bien avec la définition “classique” des arbres de dérivation qu'avec la convention que nous adoptons dans cette thèse (voir le paragraphe 1.4.1 plus loin). Cette unicité doit être comprise dans le sens suivant : si un même mot appartient à plus d'un langage engendré par la grammaire, il possède un unique arbre de dérivation pour chaque langage auquel il appartient.

Définition 1.16. Un langage L est *algébrique* s'il existe une grammaire non ambiguë G qui engendre L.

Il existe des langages engendrés par des grammaires hors-contexte, mais qui ne peuvent l'être par une grammaire non ambiguë (Parikh [66]). De tels langages sont dits ambigus. L'ambiguïté d'un langage hors-contexte est un problème indécidable; on trouvera un éventail de méthodes permettant dans certains cas de démontrer cette ambiguïté dans Flajolet [40].

Définition 1.17. Soit G une grammaire, et soient S_1 et S_2 deux symboles non terminaux de G. On dira que S_2 est *accessible* à partir de S_1 s'il existe deux mots $W_1, W_2 \in (X \cup N)^*$ tels que $S_1 \overset{*}{\to} W_1 S_2 W_2$, avec $W_1 W_2 \neq \epsilon$.

On dira que S_1 et S_2 sont *simultanément accessibles* à partir de S s'il existe trois mots $W_1, W_2, W_3 \in (X \cup N)^*$ tels que $S \overset{*}{\to} W_1 S_1 W_2 S_2 W_3$ ou $S \overset{*}{\to} W_1 S_2 W_2 S_1 W_3$. Dans le cas où $S_1 = S_2$, il est indispensable que le symbole S_1 apparaisse deux fois.

La relation “accessible à partir de” est transitive. Les grammaires que nous manipulerons seront *propres*, ce qui signifie que tout symbole autre que l'axiome doit être accessible à partir de celui-ci (voir [1]). En effet, on ne change pas le langage engendré par une grammaire en éliminant les symboles non accessibles à partir de l'axiome, ainsi que toutes les règles de dérivation dont le membre gauche ou droit comporte l'un des symboles éliminés.

Etant donnée une grammaire, si l'on change l'axiome pour le remplacer par un autre des symboles non terminaux, on change généralement le langage engendré. Les langages $L_G(U)$, où U est un symbole autre que l'axiome, sont appelés *langages auxiliaires.*

Exemple 1.18 (Langage de Dyck). Un exemple fondamental de langage algébrique est le langage de Dyck, qui reviendra fréquemment dans nos exemples. Le langage de Dyck est celui des systèmes correctement formés de parenthèses (Comtet [22]). En remplaçant par la lettre a la parenthèse ouvrante, et par b la parenthèse fermante, un mot $w \in \{a,b\}^*$ est un *mot de Dyck* s'il vérifie les conditions suivantes :

- $|w|_a = |w|_b$;
- pour tout facteur gauche w' de w, $|w'|_a \geq |w'|_b$.

Le langage de Dyck est engendré par la grammaire $G = (\{a,b\},\{D\},\mathcal{R},D)$, où $\mathcal{R}$ contient les deux règles de dérivation

$$\begin{cases} R_1: & D & \rightarrow & \epsilon, \\ R_2: & D & \rightarrow & aDbD. \end{cases}$$

Le nombre de mots de Dyck de longueur $2n$ est le n-ième *nombre de Catalan* $C_n = \frac{1}{n+1}\binom{2n}{n}$. Ce résultat bien connu peut être montré de multiple façons : grâce à la formule d'inversion de Lagrange [81, 54] en utilisant l'équation sur la série génératrice $D(x)$ fournie par la grammaire ($D(x) = 1 + xD(x)^2$); en extrayant directement les coefficients de Taylor de la série génératrice obtenue explicitement en résolvant cette équation. Il en existe également de nombreuses preuves combinatoires; voir André [2], Bertrand [12] pour les plus anciennes. Parmi les méthodes permettant d'obtenir de telles preuves, le *principe de réflexion* d'André et le *principe de Raney* (Raney [70]), basé sur la conjugaison de mots, sont les plus universels.

1.4.1 Arbres de dérivation

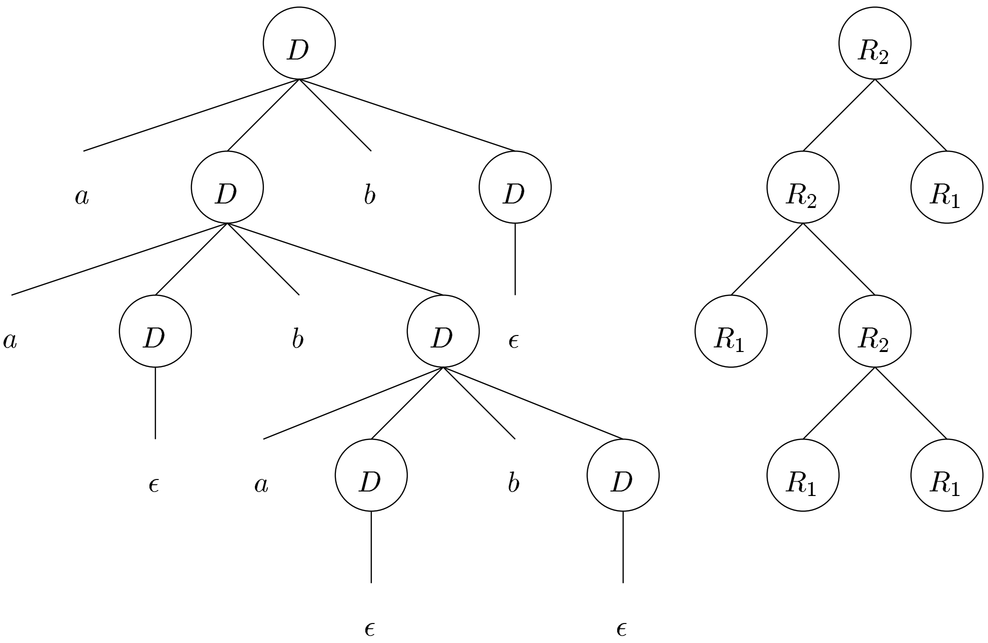
Nous utiliserons fréquemment la notion d'*arbre de dérivation* d'un mot dans une grammaire. L'arbre de dérivation d'un mot est la représentation arborescente de la suite de transitions qui permettent de passer d'un symbole non terminal à un mot du langage engendré par ce symbole.

Dans le sens usuel, l'arbre de dérivation d'un mot comporte un nœud interne par transition. Les nœuds internes sont alors étiquetés par des symboles non terminaux, et les feuilles par des mots (éventuellement vides) formés de lettres terminales. Le mot représenté par un arbre de dérivation donné est alors obtenu en lisant les étiquettes des feuilles dans l'ordre symétrique.

Dans notre travail, nous avons besoin de repérer quelle règle de dérivation a été utilisée à chaque étape de la formation du mot. C'est pourquoi les nœuds de nos arbres de dérivation sont étiquetés par les “noms” des règles de dérivation utilisées (plutôt que par les symboles non terminaux qui en forment les membres gauches). La connaissance des règles de dérivation rendant alors superflues les feuilles et leurs étiquettes, nous les omettons systématiquement. La taille d'un arbre de dérivation est alors exactement la longueur de la suite de transitions qui fait passer du symbole non terminal au mot produit.

La figure 1.2 montre les deux formes possibles d'arbres de dérivation pour le mot $w = aababb$ dans la grammaire classique engendrant le langage de Dyck :

$$\begin{array}{llll} R_1 : & D & \rightarrow & \epsilon \\ R_2 : & D & \rightarrow & aDbD. \end{array}$$



(a) Arbre de dérivation classique (b) Arbre de dérivation condensé

FIG. 1.2: *Exemples d'arbres de dérivation*

Formellement, nos arbres de dérivation sont donc définis récursivement de la manière suivante :

Définition 1.19. Soit U un symbole d'une grammaire G, et $w \in L_G(U)$ un mot. Les arbres de dérivation de w dans le langage $L_G(U)$ sont définis ainsi :

- Si $R : U \rightarrow w$ est une des règles de dérivation de la grammaire, l'arbre réduit à une racine étiquetée R est un arbre de dérivation de w;
- si $R : U \rightarrow u_0U_1u_1 \ldots U_ku_k$ est une des règles de dérivation de la grammaire, si $w = u_0w_1u_1 \ldots w_ku_k$, avec pour $1 \leq i \leq k$, $w_i \in L_G(U_i)$, et si pour $1 \leq i \leq k$, T_i est

un arbre de dérivation de w_i dans le langage $L_G(U_i)$, alors l'arbre $T = (R; T_1, \ldots, T_k)$ est un arbre de dérivation de w dans le langage $L_G(U)$.

La définition donnée plus haut d'une grammaire non ambiguë est équivalente à la suivante :

Définition 1.20. Une grammaire G est non ambiguë si et seulement si, pour chaque symbole U et chaque mot $w \in L_G(U)$, w a un unique arbre de dérivation dans le langage $L_G(U)$.

En termes d'arbres de dérivation, un symbole U_2 est accessible à partir d'un symbole U_1 s'il existe une arbre de dérivation de la grammaire, dont la racine est étiquetée par une U_1-dérivation, et qui contient un sommet étiqueté par une U_1-dérivation (avec la condition supplémentaire que ce sommet doit être distinct de la racine, si $U_2 = U_1$). De même, U_2 et U_1 sont simultanément accessibles à partir de U, s'il existe un arbre de dérivation dont la racine est étiquetée par une U-dérivation, et qui contient deux sommets, étiquetés respectivement par une U_1-dérivation et une U_2-dérivation, qui ne sont pas sur la même branche (l'un ne doit pas être un descendant de l'autre).

Lorsque $\mathcal{A}_1$ et $\mathcal{A}_2$ sont deux arbres de dérivation d'une même grammaire, et que s est un sommet de $\mathcal{A}_1$, l'arbre $\mathcal{A}_1(s, \mathcal{A}_2)$ est un arbre de dérivation à condition que les étiquettes de s et de la racine de $\mathcal{A}_2$ aient le même membre gauche; nous nous interdirons d'effectuer des substitutions dans les arbres de dérivation lorsque cette condition ne sera pas remplie.

1.4.2 Grammaires attribuées

Les grammaires attribuées sont classiquement utilisées pour la construction de compilateurs. Nous ne nous intéressons ici qu'à leur utilisation dans le domaine de la combinatoire, et nous nous limitons à des attributs *synthétisés* tels qu'ils sont définis dans l'article de Knuth [61].

Définition 1.21. Soit $G = (X, N, \mathcal{R}, S)$ une grammaire. Une *famille d'attributs* définie sur G est la donnée, pour chaque symbole $U \in N$, d'un ensemble fini T_U d'attributs ayant les caractéristiques suivantes :

- chaque attribut $\tau \in T_U$ possède un *domaine de valeurs* D_τ, qui est un ensemble (fini ou non); le produit cartésien $\prod_{\tau \in T_U} D_\tau$ est noté $\mathcal{D}_U$;

- pour chaque attribut $\tau \in T_U$ et chaque U-dérivation $R : U \to w_0 U_1 \dots U_k w_k$, on donne une *règle de calcul* $f_{\tau,R}$, qui est une fonction définie sur $\mathcal{D}_{U_1} \times \dots \times \mathcal{D}_{U_k}$ et à valeurs dans D_τ. Si la règle R est d'arité 0, $f_{\tau,R}$ est simplement un élément de D_τ.

Le couple $\big(G, (T_U)_{U \in N}\big)$ est alors appelé *grammaire attribuée.*

Une grammaire attribuée permet de calculer récursivement, pour chaque mot w de chaque langage $L_G(U)$ engendré par G, une valeur $\tau(w) \in D_\tau$, et ce, pour chaque attribut défini sur U. Ce calcul est plus simple à lire sur un arbre de dérivation $\mathcal{A}$: à chaque sommet s de l'arbre, étiqueté par une U-dérivation, on donne une étiquette supplémentaire $\tau(s)$ pour chaque $\tau \in T_U$, cette étiquette étant calculée en fonction des différentes étiquettes des fils de s : si le sommet s est étiqueté par la règle R, $\tau(s)$ est calculé en appliquant la "règle de calcul" $f_{\tau,R}$ à l'ensemble des valeurs calculées pour les fils du sommet s. De proche en proche, on obtient ainsi les attributs de la racine $(\tau(s_0))_{\tau \in T_U}$, qui sont les valeurs des attributs pour le *mot* w dont $\mathcal{A}$ est l'arbre de dérivation.

Exemple 1.22. Considérons la grammaire $G = (\{a,b\}, \{D\}, \mathcal{R}, D)$, dont les règles de dérivation sont :

$$\left\{ \begin{array}{rccl} R: & D & \to & \epsilon, \\ R': & D & \to & aDbD. \end{array} \right.$$

Les arbres de dérivation de G sont exactement les *arbres binaires complets* (les sommets internes, étiquetés R', ont 2 fils, et les feuilles sont étiquetées R).

Le *nombre de Strahler* $St(\mathcal{A})$ d'un arbre binaire complet $\mathcal{A}$ est défini récursivement de la manière suivante :

- $St(\mathcal{A}) = 0$ si $\mathcal{A}$ se réduit à sa racine;

- si $\mathcal{A}$ a pour sous-arbres gauche et droit $\mathcal{A}_g$ et $\mathcal{A}_d$ respectivement, $St(\mathcal{A})$ est le *maximum* de $St(\mathcal{A}_g)$ et $St(\mathcal{A}_d)$, auquel on ajoute 1 si $St(\mathcal{A}_g) = St(\mathcal{A}_d)$.

Le nombre de Strahler peut par conséquent être défini par un attribut St, avec les règles de calcul :

$$\left\{ \begin{array}{rcl} f_{St,R} & = & 0 \\ f_{St,R'}(n,m) & = & \left\{ \begin{array}{l} n+1 \text{ si } n = m \\ \max(n,m) \text{ si } n \neq m \end{array} \right. \end{array} \right.$$

Le calcul du nombre de Strahler est illustré figure 1.3.

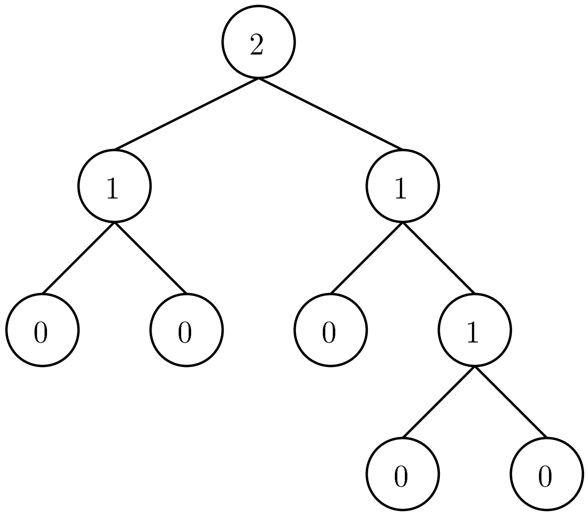


FIG. 1.3: *Exemple de calcul du nombre de Strahler*

Les q-grammaires, introduites par Fédou [37], sont un cas particulier de grammaires attribuées, où un attribut unique est défini sur chaque symbole, les règles de calcul étant astreintes à avoir une forme particulière. Nous en proposons une version plus générale, appelée Q-grammaires, au chapitre 2, et qui constituent l'objet principal d'étude de ce travail.

1.5 Chemins discrets

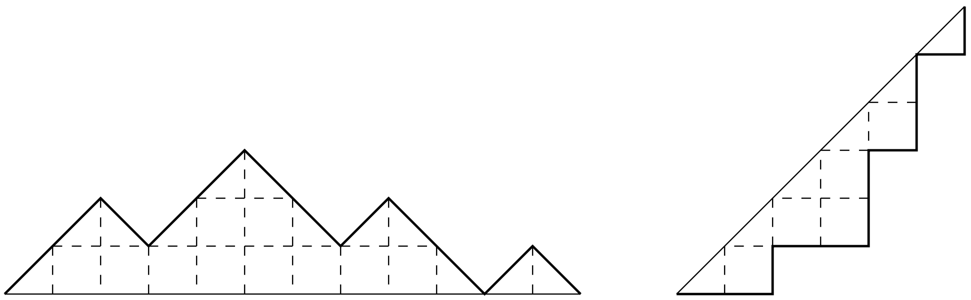
Dans notre travail, un *chemin* est une suite $(s_0, \ldots, s_n)$ de points du plan discret $\mathbb{Z} \times \mathbb{Z}$. Nous ne considérons les chemins qu'à translation près, aussi supposerons-nous le plus souvent que le premier sommet s_0 est l'origine $(0, 0)$.

Un *pas* d'un chemin est le vecteur reliant deux sommets consécutifs, de coordonnées $(x_{i+1} - x_i, y_{i+1} - y_i)$. Il est fréquent d'utiliser les points cardinaux pour désigner certains types de pas : un pas $(0, 1)$ est ainsi un pas Nord, un pas $(1, -1)$ est un pas Sud-Est, etc.

Il est commode de représenter certains mots par des chemins, le plus simple étant alors d'associer à chaque lettre de l'alphabet un type de pas, ceux-ci se succédant dans le même ordre que dans le mot.

Les représentations les plus courantes des mots de Dyck sont celles que nous appellerons représentations diagonale et horizontale.

La représentation *diagonale* consiste à traduire la lettre a par un pas Est et la lettre b par un pas Nord. Le chemin se termine alors sur la droite d'équation $y = x$, sans avoir de sommet au-dessus de cette droite. Dans la représentation *horizontale*, la lettre a se traduit



Représentation horizontale Représentation diagonale

FIG. 1.4: *Chemins de Dyck associés à* $w = aabaabbabbab$

par un pas Nord-Est, et la lettre b se traduit par un pas Sud-Est. Le chemin se termine alors sur la droite d'équation $y = 0$, sans avoir de sommet en-dessous de cette droite.

Dans les deux cas, le chemin obtenu est appelé chemin de Dyck. Ces deux représentations sont illustrées figure 1.4.

Le passage de l'une à l'autre de ces représentations se fait très simplement par un changement d'échelle suivi d'une symétrie.

1.6 Deux exemples classiques

Deux exemples classiques de grammaires et de paramètres reviendront fréquemment au cours de cette thèse, tous deux basés sur le langage des mots de Dyck. La terminologie d'origine géométrique que nous emploierons, est basée sur la représentation horizontale des mots de Dyck par des chemins discrets allant du point $(0, 0)$ à un point $(2n, 0)$, où $2n$ est la longueur du mot.

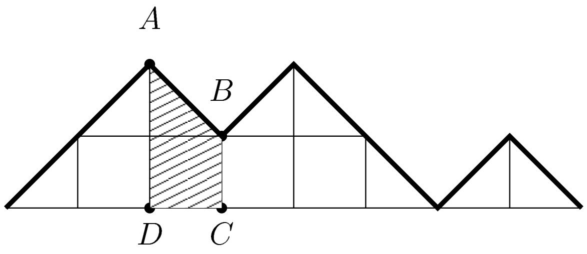
1.6.1 Aire des chemins de Dyck

Par définition, l'*aire* d'un chemin de Dyck est l'aire comprise entre ce chemin et l'axe d'équation $y = 0$. Par extension, nous parlerons également de l'aire d'un mot de Dyck. En découpant cette surface en tranches verticales délimitées par les pas du chemin, cette aire apparaît comme la somme d'aires de trapèzes : chaque pas AB du chemin, contribue à l'aire pour $(y_A + y_B)/2$.

Lorsque le pas AB est un pas Nord-Est, $y_B = y_A + 1$, et par conséquent, si $h = y_B$ est l'ordonnée du point B (également appelée *hauteur finale* du pas AB), la contribution à l'aire du pas AB est $h - 1/2$.

Inversement, si le pas AB est un pas Sud-Est, $y_B = y_A + 1$, et la contribution du pas AB est $h + 1/2$.

Lors de la sommation, les termes $+1/2$ et $-1/2$ associés aux pas Nord-Est et Sud-Est se compensent, puisqu'un chemin de Dyck comporte autant de pas des deux types; par conséquent *l'aire d'un chemin de Dyck* est aussi égale à *la somme des hauteurs finales des pas qui le constituent.*



$$\begin{array}{rcl}
1/2 + 3/2 + 3/2 + 3/2 + 3/2 + 1/2 + 1/2 + 1/2 & = & 8 \\
1 + 2 + 1 + 2 + 1 + 0 + 1 + 0 & = & 8 \\
2(\quad 1 \quad + \quad 1 + 0 \quad + \quad 0 \quad) + 4 & = & 8
\end{array}$$

FIG. 1.5: *Différents modes de calcul de l'aire d'un chemin de Dyck*

Par ailleurs, les pas Nord-Est et Sud-Est peuvent être naturellement appariés de manière à ce qu'à chaque pas Nord-Est de hauteur finale h, corresponde un pas Sud-Est de hauteur finale $h - 1$ (ce qui, en interprétant le mot de Dyck comme un mot de parenthèses ouvrantes et fermantes, correspond aux paires de parenthèses). Par conséquent, l'aire d'un chemin de Dyck est également obtenue en faisant la somme de sa *demi-longueur* et de *deux fois la somme des hauteurs finales de ses pas Sud-Est.*

Tous ces moyens de calcul de l'aire d'un chemin de Dyck sont illustrés figure 1.5.

Tous ces paramètres se "lisent" parfaitement sur les mots de Dyck correspondant aux chemins. En effet, si $w = w_1 w_2$ est un mot de Dyck, la hauteur finale du chemin correspondant au facteur gauche w_1 est $h = |w_1|_a - |w_1|_b$; il est donc simple de transformer les relations donnant l'aire d'un chemin de Dyck, en expressions portant sur les mots de Dyck associés.

Le paramètre *aire* des mots de Dyck est lié à un autre paramètre classique : le *nombre d'inversions* [32]. Nous considérerons l'ordre $a < b$, et par conséquent une inversion dans un mot de Dyck sera une occurrence du sous-mot ba. Le nombre d'inversions d'un tel mot

w peut donc être décrit de plusieurs façons :

– la somme, pour chaque occurrence de a dans w, du nombre d'occurrences de b qui se trouvent avant elle dans w;

– la somme, pour chaque occurrence de b dans w, du nombre d'occurrences de a qui se trouvent après elle dans w.

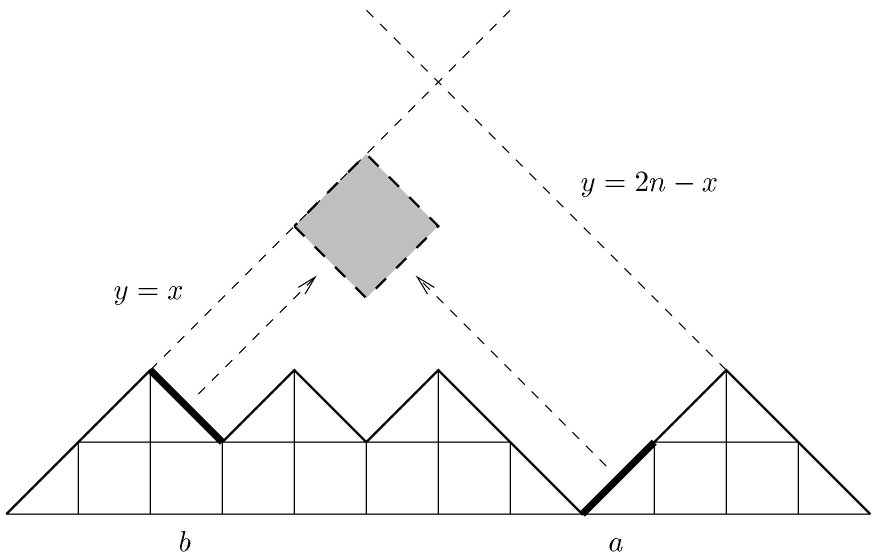


FIG. 1.6: *Interprétation géométrique du nombre d'inversions*

La figure 1.6 montre comment le nombre d'inversions d'un mot de Dyck peut être interprété géométriquement comme la moitié de l'aire comprise *au-dessus* du chemin de Dyck correspondant, et *en-dessous* des droites diagonales d'équations $y = x$ et $y = 2n - x$ (où $2n$ est la longueur du mot) : à chaque inversion du mot, correspond exactement un carré d'aire 2 situé dans cette zone.

Une autre définition, très proche, de l'aire associée à un mot de Dyck, est celle qui correspond aux q-analogues des nombres de Catalan étudiés par Carlitz. Dans ce cas, on considère des chemins de Dyck sous-diagonaux, et l'aire d'un mot est celle de la zone située entre le chemin sous-diagonal et la diagonale d'équation $y = x$. Cette aire n'étant pas entière pour les chemins de longueur $2n$ lorsque n est impair, on ne considère que la différence avec l'aire associée au mot $(ab)^n$, soit $n/2$.

Le passage d'une représentation géométrique à l'autre se fait par une symétrie suivie d'une homothétie de rapport $\sqrt{2}$, par conséquent, si l'aire d'un mot de Dyck w de longueur

$2n$ est notée $A(w)$, et l'aire de Carlitz, $A_c(w)$, on a

$$A(w) = 2A_c(w) + n.$$

L'aire de Carlitz est plus directement liée au nombre d'inversions; on a, pour un mot de longueur $2n$,

$$A_c(w) + \mathrm{inv}(w) = \binom{n}{2}.$$

Les deux définitions de l'aire sont illustrées figure 1.7.

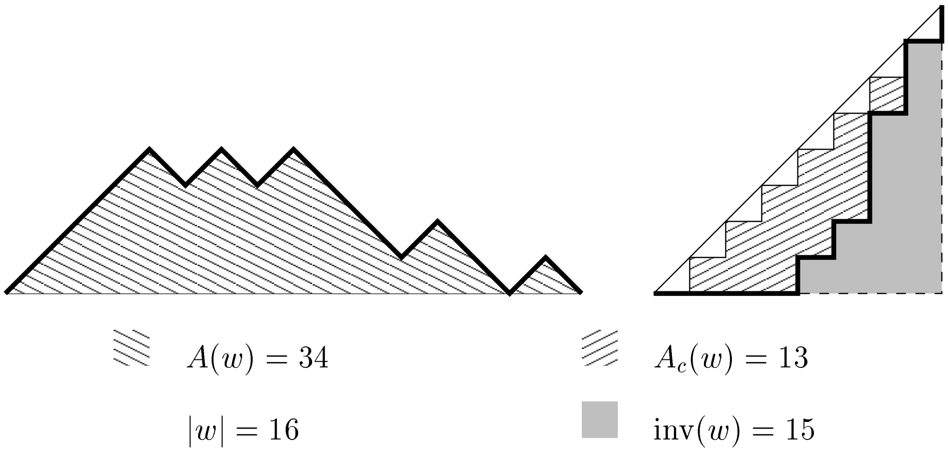


FIG. 1.7: *Les deux définitions de l'aire pour* $w = aaaabababbbabbab$

Les q-analogues des nombres de Catalan définis par Carlitz énumèrent les mots de Dyck suivant le nombre d'inversions ou l'aire de Carlitz :

$$\begin{aligned} C_n(q) &= \sum_{w \in D_n} q^{\mathrm{inv}(w)}, \\ \tilde{C}_n(q) &= \sum_{w \in D_n} q^{A_c(w)}. \end{aligned}$$

Ces polynômes vérifient les récurrences suivantes, qui sont deux q-analogues de la récurrence classique des nombres de Catalan :

$$\begin{aligned} \tilde{C}_0 &= 1, \\ \tilde{C}_{n+1} &= \sum_{0 \le k \le n} q^k \tilde{C}_k \tilde{C}_{n-k}; \\ C_0 &= 1, \\ C_{n+1} &= \sum_{0 \le k \le n} q^{(k+1)(n-k)} C_k C_{n-k}. \end{aligned}$$

Nous mentionnons pour mémoire deux résutats d'énumération concernant l'aire des chemins de Dyck :

- Soit, pour chaque mot de Dyck de longueur $2n$, $A'(w) = A(w) + 2n + 1$; $A'(w)$ s'interprète comme le nombre de points du plan discrets situés (au sens large) entre le chemin de Dyck et l'axe horizontal. Alors
$$\sum_w A'(w) = 4^n,$$
où la sommation porte sur tous les mots de Dyck de longueur $2n$. On trouvera une preuve bijective de cette formule dans [20].
- La somme des aires de Carlitz des chemins de Dyck de longueur $2n$,
$$\sum_w A_c(w) = \frac{1}{2}\left(4^n - (3n+1)C_n\right),$$
est également le nombre de cartes planaires pointées sans isthmes à 2 sommets et n arêtes (alors que le nombre de cartes planaires pointées à 1 sommet et n arêtes est le nombre de Catalan C_n). La suite de nombres $(1, 7, 37, 176\dots)$ apparaît dans [80]; un codage de ces cartes par des mots de Dyck marqués est donné dans [63].

1.6.2 Somme des hauteurs de pics de chemins de Dyck

Dans un chemin de Dyck, un *pic* est un sommet immédiatement précédé d'un pas Nord-Est, et immédiatement suivi d'un pas Sud-Est.

Inversement, un *creux* d'un chemin de Dyck est un sommet immédiatement précédé d'un pas Sud-Est, et immédiatement suivi d'un pas Nord-Est.

Dans le mot de Dyck correspondant, un pic correspond à un facteur ab, et un creux, à un facteur ba. La *hauteur* d'un pic ou d'un creux est l'ordonnée du sommet correspondant. Comme pour le calcul de l'aire, la hauteur d'un pic se calcule simplement sur le mot de Dyck correspondant. La hauteur du pic situé entre les facteurs w_1a et bw_2 dans $w = w_1abw_2$, est $h = 1 + |w_1|_a - |w_1|_b$.

Un mot de Dyck peut parfaitement être défini par la suite de ses hauteurs de pics et de creux; cette remarque est à la base d'un codage des polyominos parallélogrammes par les mots de Dyck [25], dans lequel la hauteur de chaque pic devient la hauteur d'une colonne du polyomino, et, par conséquent, la somme des hauteurs de pics devient l'aire du polyomino codé. La longueur du mot de Dyck devient, à un décalage près, le périmètre du polyomino codé.

Fédou a montré dans [37], que la somme des hauteurs des pics de mots de Dyck a la même distribution que le paramètre *somme des nombres de feuilles des sous-arbres gauches* défini sur les arbres binaires complets; nous verrons au chapitre 2 que ce genre d'interprétation peut être automatisé.

$w = aabaaabbababbbab$

Longueur 16

5 pics de hauteurs 2, 4, 3, 3, 1

Périmètre 18

5 colonnes

de hauteurs 2, 4, 3, 3, 1.

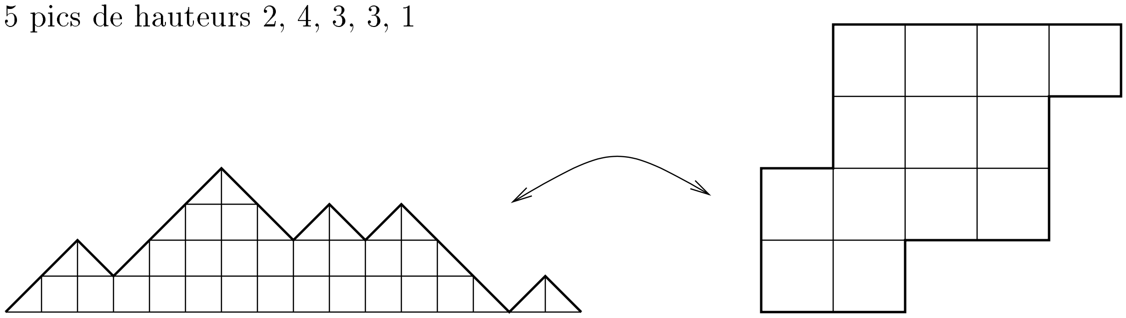


FIG. 1.8: *Somme des hauteurs de pics et polyominos parallélogrammes*

La figure (1.8) montre un exemple de mot de Dyck de somme des hauteurs de pics 13, et le polyomino parallélogramme correspondant.

Chapitre 2

Q-grammaires

Dans ce chapitre, nous définissons les objets principalement étudiés dans cette thèse : les Q-grammaires, qui sont des grammaires attribuées dont les attributs ont une forme particulière.

2.1 Notations

Dans ce chapitre, chaque fois qu'une grammaire $G = (X, N, \mathcal{R}, S)$ aura été définie et qu'il n'y aura pas d'ambiguïté possible, nous identifierons un symbole et le langage qu'il engendre, c'est-à-dire que nous écrirons U pour $L_G(U)$.

Une règle de dérivation générique d'une grammaire sera notée sous la forme

$$R : U \to w_0 U_1 w_1 \dots w_{n-1} U_n w_n,$$

avec $w_i \in X^*$ et $U_i \in N$.

Dans cette notation, R est le nom de la règle de dérivation, utilisé pour étiqueter les arbres de dérivation; n est l'arité de la règle R.

Un mot $u \in L_G(U)$ est obtenu par l'utilisation de la règle R lorsque la racine de son arbre de dérivation est étiquetée par R. Cela revient à dire qu'il existe des mots $u_i \in U_i$ pour $1 \leq i \leq n$, tels que

$$u = w_0 u_1 w_1 \dots w_{n-1} u_n w_n.$$

Les grammaires considérées étant non ambiguës, pour chaque symbole U et chaque mot $u \in U$, il n'y a qu'une seule règle dont est issu u; nous noterons alors sans plus de précision u_i le i-ème mot apparaissant dans la décomposition ci-dessus. Ce mot u_i est celui qu'engendre le i-ème sous-arbre de l'arbre de dérivation de u.

2.2 Paramètres Q-comptables

2.2.1 Termes de croissance d'un paramètre

Soit $G = (X, N, \mathcal{R}, S)$ une grammaire non ambiguë engendrant un langage L, et soit p un paramètre défini sur la réunion disjointe[1] des langages engendrés par G (même si p n'est défini a priori que sur le langage L, nous supposerons qu'un prolongement à cette réunion disjointe a été choisi).

Définition 2.1. Soit u un mot d'un langage engendré par G.

Avec les notations définies précédemment, on appelle *terme de croissance de p pour u*, la quantité

$$\theta_p(u) = p(u) - \sum_{i=1}^{n} p(u_i).$$

Lorsque le mot u se trouve être le membre droit d'une U-dérivation d'arité 0, le terme de croissance de p pour u (dans le langage U) est simplement $\theta_p(u) = p(u)$.

En termes d'arbres de dérivation, et si l'on considère p comme un attribut, $\theta_p(u)$ est la différence entre l'attribut calculé sur l'arbre tout entier, et la somme des valeurs attribuées aux sous-arbres issus des fils de la racine; c'est pourquoi nous l'appelons terme de croissance. Il est parfaitement possible, pour un attribut quelconque, qu'il n'y ait aucun lien entre $p(u)$ et la somme des $p(u_i)$. Dans ce cas, le terme de croissance n'a aucun sens particulier.

Exemple 2.2. Considérons le langage des mots de Dyck, engendré par la grammaire classique G_1 correspondant aux deux règles de dérivation

$$\begin{cases} R_1 : & D & \to & \epsilon \\ R_2 : & D & \to & aDbD. \end{cases}$$

Notons, pour tout mot $w \in \{a, b\}^*$, $h(w) = |w|_a - |w|_b$, et soit

$$p(w) = \max_{w = w_1 w_2} h(w_1).$$

Le paramètre p représente l'ordonnée maximale atteinte par le chemin de Dyck associé au mot w; il est aussi appelé hauteur maximale de w. Il est clair que le paramètre p peut

1. Dans le cas où un même mot appartient à plus d'un des langages engendrés par une grammaire, nous considérerons qu'un même paramètre peut prendre, pour ce mot, une valeur différente dans chaque langage.

être calculé comme attribut synthétisé sur la grammaire G_1 : $p(\epsilon) = 0$, et, si $w = aw_1bw_2$ est obtenu par la règle R_2, on a

$$p(w) = \begin{cases} 1 + p(w_1) & \text{si } p(w_1) \geq p(w_2), \\ p(w_2) & \text{si } p(w_2) > p(w_1). \end{cases}$$

Dans ce cas, la somme $p(w_1) + p(w_2)$ n'apparaît pas dans le calcul de $p(w)$: le terme de croissance n'a donc ici pas de sens naturel.

Un exemple similaire est celui du *nombre de Strahler* vu au chapitre 1. De manière générale, on peut s'attendre à ce que le terme de croissance d'un paramètre défini comme un *maximum* n'ait pas de sens "naturel".

Notre travail a pour cadre le cas inverse, où chaque paramètre étudié peut être "simplement" défini au moyen de ses termes de croissance.

Exemple 2.3. Considérons de nouveau le langage de Dyck, engendré par les deux grammaires G_1 (voir exemple 2.2) et $G_2 = (\{a, b\}, \{D, E\}, \{R'_1, R'_2, R'_3, R'_4, R'_5, R'_6\}, D)$:

$$\begin{cases} R'_1: & D & \to & \epsilon \\ R'_2: & D & \to & E \\ R'_3: & E & \to & ab \\ R'_4: & E & \to & abE \\ R'_5: & E & \to & aEb \\ R'_6: & E & \to & aEbE. \end{cases}$$

Il est facile de voir que, dans la grammaire G_2, seules les règles R'_3 et R'_4 font apparaître des facteurs ab (les pics des chemins de Dyck correspondants); par conséquent, dans G_2, le terme de croissance du paramètre "nombre de pics" p' est 1 pour les règles R'_3 et R'_4, et 0 pour les autres règles.

En revanche, dans G_1, la situation n'est pas aussi simple. Le terme de croissance du même paramètre p' pour la règle R_2 est toujours positif ou nul; si le mot $w_1 \in D$ a k_1 facteurs ab et le mot $w_2 \in D$ en a k_2, le mot $aw_1bw_2 \in D$ en a au moins $k_1 + k_2$. En fait, on a $p'(aw_1bw_2) = p'(w_1) + p'(w_2)$ si $w_1 \neq \epsilon$, et $p'(aw_1bw_2) = p'(w_1) + p'(w_2) + 1 = p'(w_2) + 1$ si $w_1 = \epsilon$.

Nous parlerons également de terme de croissance d'un paramètre pour un arbre de dérivation, ou pour un nœud d'un arbre de dérivation.

Ainsi, pour un arbre de dérivation $\mathcal{A}$, le terme de croissance du paramètre p pour $\mathcal{A}$ sera par définition $\theta_p(\mathcal{A}) = \theta_p(u)$, où u est le mot dont $\mathcal{A}$ est l'arbre de dérivation.

De même, si s est un nœud d'un arbre de dérivation $\mathcal{A}$, le terme de croissance de p pour s est $\theta_p(s) = \theta_p(\mathcal{B})$, où $\mathcal{B}$ est le sous-arbre de $\mathcal{A}$ dont la racine est s.

2.2.2 Paramètres Q-comptables et Q-grammaires

Le principe de la formation d'une Q-grammaire consiste à définir tous les paramètres auxquels l'on s'intéresse par le moyen de leurs termes de croissance. Chaque terme de croissance ne doit dépendre que de la règle de dérivation utilisée et d'autres paramètres préalablement définis. Chaque paramètre est ainsi défini par un ensemble de “règles de calcul”, une règle étant associée à chaque règle de dérivation de la grammaire. Les “bons” paramètres, que nous nommerons Q-comptables, sont ceux dont les termes de croissance sont eux-mêmes définis par d'autres paramètres Q-comptables :

Définition 2.4 (paramètre Q-comptable). Un paramètre p est dit :

- Q-comptable de rang 1, si, pour chaque règle de dérivation R de G, il existe une constante entière $c_R \geq 0$ telle que, pour tout mot u obtenu par l'application de la règle R, $\theta_p(u) = c_R$;
- Q-comptable de rang $k+1$, si, pour chaque règle de dérivation R de G, il existe des paramètres Q-comptables $p_1, \ldots p_n$, de rangs inférieurs ou égaux à k, et une constante $c_R \geq 0$, tels que, pour tout mot u obtenu par l'application de la règle R, on ait $\theta_p(u) = c_R + p_1(u_1) + \cdots + p_n(u_n)$.

Afin que notre définition soit consistante, nous exigeons que, pour un paramètre de rang 1, l'une au moins des constantes c_R soit non nulle, et que, pour un paramètre de rang $k+1$, l'un au moins des paramètres utilisés soit exactement de rang k.

Un “paramètre” constamment égal à un nombre entier positif est considéré comme Q-comptable de rang 0.

Un paramètre Q-comptable est donc un attribut synthétisé sur la grammaire G, avec des restrictions sur la forme des règles de calcul de cet attribut, qui doivent ne faire intervenir que des combinaisons affines d'autres paramètres Q-comptables.

Par la suite, nous ne nous intéresserons qu'aux paramètres Q-comptables d'une grammaire.

La définition ci-dessous généralise celle des q-grammaires définie dans [37, 28] :

Définition 2.5. On appelle Q-grammaire, une grammaire attribuée $(G; p_1, \ldots, p_n)$, où les attributs $p_1, \ldots, p_n$ sont des paramètres Q-comptables sur G et tels que, pour $1 \leq i \leq n$, les termes de croissance de p_i ne fassent intervenir que les paramètres $p_1, \ldots, p_{i-1}$.

La notion de paramètre Q-comptable est liée à la grammaire G plus qu'au langage qu'elle engendre (dans l'exemple 2.3, le paramètre "nombre de pics" est clairement Q-comptable dans la grammaire G_2, alors que nous verrons qu'il ne l'est pas dans la grammaire G_1). Certains paramètres sont Q-comptables pour toutes les grammaires.

Lorsqu'il y aura risque d'ambiguïté sur la grammaire, on précisera celle-ci en disant qu'un paramètre est G-Q-comptable.

Exemple 2.6. Dans toute grammaire, le paramètre "longueur" $|w|$ et, plus généralement, tout paramètre "longueur en X'" où $X' \subset X$, est Q-comptable de rang 1; en effet, le terme de croissance de chaque règle $R : U \to w_0 U_1 w_1 \dots U_n w_n$ est la constante $\theta_p = c_R = |w_0 w_1 \dots w_n|_{X'}$.

Exemple 2.7. Reprenons la grammaire G_1 de l'exemple 2.2 (grammaire classique engendrant le langage de Dyck).

Notons $A(w)$ l'aire d'un mot de Dyck quelconque w. Comme nous l'avons vu précédemment, $A(w)$ est aussi la somme des hauteurs de tous les sommets du chemin de Dyck correspondant à w.

L'aire des chemins de Dyck est Q-comptable de rang 2 pour cette grammaire, avec les termes de croissance ainsi définis :

- Pour la règle $D \to \epsilon$: $A(\epsilon) = 0$ (l'aire du chemin vide est nulle);
- Pour la règle $D \to aDbD$: $A(ad_1bd_2) = A(d_1) + A(d_2) + 1 + |d_1|$, donc le terme de croissance est $1 + |d_1|$. La figure 2.1 illustre cette règle de calcul.

Puisque la longueur est elle-même Q-comptable de rang 1, l'aire est alors Q-comptable de rang 2.

De même, l'aire de Carlitz est également Q-comptable de rang 2 dans la grammaire G_1. En effet, $A_c(ad_1bd_2) = A_c(d_1) + A_c(d_2) + |d_1|/2$, donc le terme de croissance est $\theta_{A_c}(ad_1bd_2) = |d_1|/2 = |d_1|_a$ qui est lui aussi un paramètre Q-comptable de rang 1.

Enfin, nous pouvons aller un cran plus loin et définir pour un mot de Dyck son *moment d'inertie* de la manière suivante : $M(w)$ est la somme des ordonnées des points à coordonnées entières positives situés (au sens large) sous le chemin codé par w. Ainsi, la contribution à $M(w)$ des points situés sous un sommet de hauteur h, est $1 + \cdots + h = h(h+1)/2$.

Le moment d'inertie M est un paramètre Q-comptable de rang 3, et sa règle de calcul pour la règle $D \to aDbD$ peut être obtenue en examinant de nouveau la figure 2.1. Dans le chemin codé par $d = ad_1bd_2$, chacun des points à coordonnées entières situés sous le chemin

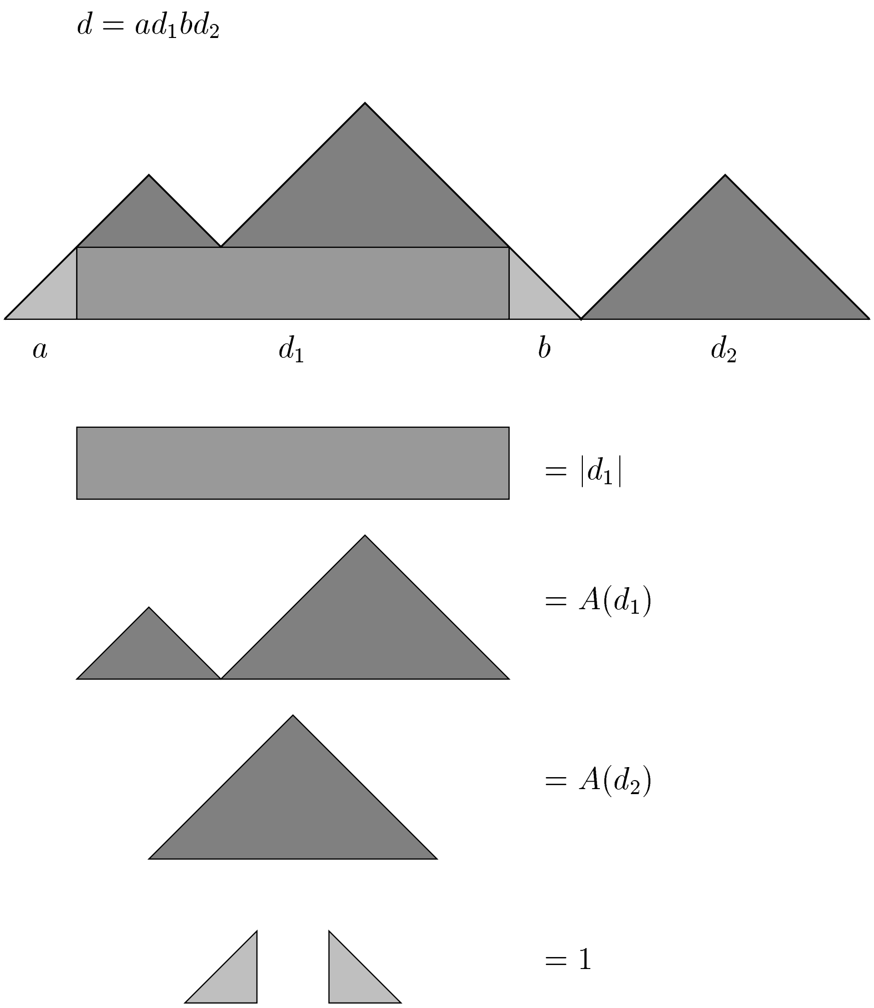


FIG. 2.1: *Règle de calcul de l'aire d'un chemin de Dyck*

codé par d_1 voit son ordonnée augmentée de 1; ces points sont au nombre de $A(d_1)+|d_1|+1$. Par conséquent, le moment d'inertie $M(d)$ est donné par

$$M(d) = (M(d_1) + A(d_1) + |d_1| + 1) + M(d_2).$$

Les règles de calcul du paramètre M sont donc :

- Pour la règle $D \to \epsilon$: $M(\epsilon) = 0$;

- Pour la règle $D \to aDbD$: $M(ad_1bd_2) = M(d_1) + M(d_2) + A(d_1) + |d_1| + 1$; le terme de croissance est $A(d_1) + |d_1| + 1$: l'aire étant un paramètre de rang 2 et la longueur de rang 1, le moment d'inertie est bien de rang 3.

2.2.3 Paramètres élémentaires

Certaines propriétés des paramètres Q-comptables découlent immédiatement de leur définition par règles de croissance :

Proposition 2.8. *Toute combinaison linéaire à coefficients entiers positifs de paramètres Q-comptables, est un paramètre Q-comptable dont le rang est le rang maximum des paramètres concernés.*

Preuve. Soient $p_1, \ldots, p_n$, n paramètres Q-comptables, et soit $p = \lambda_1 p_1 + \cdots + \lambda_n p_n$ (avec $\lambda_i \in \mathbb{N}$) une combinaison linéaire de ces paramètres. Notons, pour chaque paramètre p_i et pour chaque règle de dérivation R, $\theta_{R,i}$ le terme de croissance de p_i pour cette règle : il est clair que le terme de croissance de p pour R est $\lambda_1 \theta_{R,1} + \cdots + \lambda_n \theta_{R,n}$; par conséquent, si chaque p_i est Q-comptable et si le maximum des rangs est k, p est bien Q-comptable de rang k. $\square$

Dès lors, il est naturel de rechercher une description atomique des paramètres Q-comptables.

Définition 2.9. Si p est un paramètre Q-comptable de rang 1, p est *élémentaire* s'il existe une règle $R_0 \in \mathcal{R}$, telle que le terme de croissance de p soit 0 pour toute autre règle que R_0, et 1 pour R_0.

Récursivement, si un paramètre p est Q-comptable de rang $k+1$, p est *élémentaire* s'il existe une règle $R_0 \in \mathcal{R}$, un entier $i \leq \alpha(R_0)$, et un paramètre p', élémentaire de rang k, tels que le terme de croissance de p soit 0 pour toute autre règle que R_0, et $p'(u_i)$ pour la règle R_0.

Exemple 2.10. Reprenons l'exemple de l'aire de Carlitz telle que nous l'avons définie comme paramètre Q-comptable de rang 2 dans la grammaire G_1. Soit $p_1(w) = |w|_a$, et $p_2(w) = A_c(w)$. Les termes de croissance des paramètres p_1 et p_2 pour les règles de dérivation R_1 et R_2 sont, d'après les exemples 2.6 et 2.7 :

$$\begin{array}{lll} R_1 : & \theta_{p_1}(\epsilon) = 0; & \theta_{p_2}(\epsilon) = 0; \\ R_2 : & \theta_{p_1}(ad_1bd_2) = 1; & \theta_{p_2}(ad_1bd_2) = p_1(d_1). \end{array}$$

Par conséquent, p_1 est un paramètre élémentaire de rang 1, et p_2, un paramètre élémentaire de rang 2.

Afin de décrire tous les paramètres élémentaires existant dans une grammaire donnée, il convient de considérer l'ensemble $\mathcal{R}$ des règles de dérivation comme un nouvel alphabet, et de définir l'alphabet $\mathcal{R}_p$ des “dérivations pointées” :

Définition 2.11. Une dérivation pointée est une règle de dérivation R dont on a distingué l'un des symboles non terminaux du membre droit. Si l'on a distingué le k-ème symbole, on la notera $R^{(k)}$.

Exemple 2.12. Dans la grammaire G_1 de l'exemple 2.2, la règle R_1 (règle terminale) ne donne aucune dérivation pointée, et la règle R_2 (d'arité 2) en donne deux. On a donc $\mathcal{R}_p = \left\{R_2^{(1)}, R_2^{(2)}\right\}$.

Les alphabets $\mathcal{R}$ et $\mathcal{R}_p$ nous permettent de décrire tous les paramètres élémentaires en leur donnant des “noms”. Les noms de paramètres élémentaires sont des mots de $\mathcal{R}_p^*\mathcal{R}$, c'est-à-dire des suites de noms de règles de dérivation, dont toutes sauf la dernière sont des dérivations pointées.

Soit en effet p un paramètre élémentaire. Le nom de p est défini récursivement de la manière suivante :

- Si p est de rang 1, le nom de p est R, où R est l'unique règle de dérivation telle que $c_R = 1$;
- si p est de rang $k+1$, il existe un paramètre élémentaire p', de rang k, une règle $R \in \mathcal{R}$, et un entier $i \leq \alpha(R)$, tel que l'unique terme de croissance non identiquement nul de p soit, pour la règle R, $p'(u_i)$. Alors, si le nom de p' est W', le nom de p est $W = R^{(i)}W'$.

Notons que la longueur du nom d'un paramètre élémentaire, est également le rang de ce paramètre.

Notation 2.13. Pour tout mot $W \in \mathcal{R}_p^*\mathcal{R}$, le paramètre élémentaire dont le nom est W est noté p_W.

Par extension, si $W = \lambda_1 W_1 + \cdots + \lambda_k W_k$ est une combinaison linéaire à coefficients entiers positifs de mots de $\mathcal{R}_p^*\mathcal{R}$, on notera p_W le paramètre Q-comptable

$$p_W = \sum_{i=1}^{k} \lambda_i p_{W_i}.$$

De telles combinaison linéaires de noms de paramètres élémentaires, seront appelées *noms de paramètres Q-comptables.*

Exemple 2.14. Dans la grammaire G_1 de l'exemple 2.2, l'aire de Carlitz est le paramètre $A_c = p_{R_2^{(1)}R_2}$. L'aire géométrique est $A = 2A_c + p_{R_2} = 2p_{R_2^{(1)}R_2} + p_{R_2}$. Le moment d'inertie (voir exemple 2.7) est $M = 2p_{R_2^{(1)}R_2^{(1)}R_2} + 3p_{R_2^{(1)}R_2} + p_{R_2}$.

La proposition suivante montre clairement que les paramètres élémentaires suffisent à décrire tous les paramètres Q-comptables :

Proposition 2.15. *Tout paramètre Q-comptable de rang k peut s'écrire comme combinaison linéaire à coefficients entiers positifs de paramètres élémentaires de rangs inférieurs ou égaux à k, l'un au moins étant de rang exactement k.*

Preuve. La preuve, par récurrence sur k, est sans difficulté. Si $k = 1$, le paramètre p est entièrement défini par les constantes $(c_R)_{R\in\mathcal{R}}$. Il est alors immédiat que l'on a la décomposition

$$p = \sum_{R\in\mathcal{R}} c_R p_R.$$

Supposons maintenant la propriété vraie pour k, et soit p un paramètre Q-comptable de rang $k+1$. Pour chaque règle $R \in \mathcal{R}$, le terme de croissance de p est de la forme

$$\theta_p(w_0 u_1 \dots u_{\alpha(R)} w_{\alpha(R)}) = c_R + \sum_{i=1}^{\alpha(R)} p_{R,i}(u_i),$$

où chaque $p_{R,i}$ est un paramètre Q-comptable de rang au plus k (éventuellement, identiquement nul).

D'après l'hypothèse de récurrence, chaque paramètre $p_{R,i}$ peut s'écrire $p_{R,i} = p_{W_{R,i}}$, où chaque $W_{R,i}$ est une combinaison linéaire de mots de $\mathcal{R}_p^*\mathcal{R}$. Il est alors immédiat que l'on peut décomposer p de la manière suivante :

$$p = \sum_{R\in\mathcal{R}} \left(c_R p_R + \sum_{i=1}^{\alpha(R)} p_{R^{(i)}W_{R,i}} \right).$$

Ceci termine la récurrence, et la preuve de la proposition. □

Nous verrons plus tard que cette association entre paramètres et noms de paramètres n'est pas forcément bijective : deux paramètres p_W et $p_{W'}$ (avec $W \neq W'$) peuvent être identiques sur tous les mots engendrés par la grammaire; toutefois, lorsqu'un paramètre est donné par des règles de croissance, ces règles ne l'associent qu'à une seule combinaison de mots de $\mathcal{R}_p^*\mathcal{R}$.

2.2.4 Interprétation des paramètres Q-comptables

Les paramètres naturellement étudiés sur les mots sont fréquemment décrits comme le “nombre d’occurrences de tel ou tel événement” : nombre d’inversions, nombre d’occurrences d’une lettre, d’un facteur ... Nous allons donner une interprétation formelle de tout paramètre élémentaire et donc, par suite, de tout paramètre Q-comptable.

Il est relativement aisé d’interpréter un paramètre élémentaire de rang 1 : puisque le terme de croissance vaut 1 à chaque utilisation d’une règle de dérivation particulière, la valeur du paramètre sur un mot donné sera le nombre d’utilisations de cette règle de dérivation qui sont nécessaires pour obtenir ce mot à partir de l’axiome. En d’autres termes, un paramètre élémentaire de rang 1 “compte” une certaine règle, ou, ce qui est équivalent, les nœuds de l’arbre de dérivation qui sont étiquetés par cette règle.

Nous avons vu précédemment qu’un paramètre “nombre d’occurrences de la lettre x” est toujours Q-comptable de rang 1. Réciproquement, pour tout paramètre p, Q-comptable de rang 1, il est possible d’ajouter au langage une lettre qui sert de “marqueur” du paramètre p, de telle sorte que le nombre d’occurrences de cette nouvelle lettre soit toujours la valeur du paramètre :

Proposition 2.16. *Soit L' un langage algébrique engendré par une grammaire $G' = (X, N, \mathcal{R}', S)$, et soit p un paramètre Q-comptable de rang 1 sur G'. Soit également m une lettre n’appartenant pas à l’alphabet X.*

Il existe un langage L, engendré par une grammaire $G = (X \cup \{m\}, N, \mathcal{R}, S)$, tel que

- *la projection $\varphi : (X \cup \{m\})^* \to X^*$ établit une bijection de L sur L';*
- *pour tout mot $w \in L$, $|w|_m = p(\varphi(w))$;*
- *pour tout mot $w \in L$, les arbres de dérivation de w (dans G) et de $\varphi(w)$ (dans G') ont même forme.*

Preuve. L’ensemble des règles de dérivation $\mathcal{R}$ se forme très simplement à partir de $\mathcal{R}'$: pour chaque règle $R \in \mathcal{R}'$, si R apparaît c_R fois dans la décomposition en paramètres élémentaires de p, la règle correspondante de $\mathcal{R}$ s’obtient en ajoutant c_R occurrences de m dans le membre droit de R. Ces occurrences de m peuvent être ajoutées en n’importe quelles positions; le simple fait que la grammaire G' soit non ambiguë suffit à assurer que G le sera. $\square$

Pour interpréter les paramètres de rang supérieur à 1, nous avons besoin de la notion de *chaîne* dans un arbre :

Définition 2.17. Soit $\mathcal{A}$ un arbre de dérivation de la grammaire G. Une *chaîne de longueur* k de $\mathcal{A}$ est un k-uplet $(s_1, \dots, s_k)$ de nœuds de $\mathcal{A}$ tel que, pour tout $i < k$, s_{i+1} soit un descendant de s_i, distinct de s_i. Le *type* d'une chaîne de longueur k est le mot $R_1^{(d_1)} \dots R_{k-1}^{(d_{k-1})} R_k \in \mathcal{R}_p^{k-1}\mathcal{R}$, où R_i est l'étiquette de s_i et où s_{i+1} appartient à l'arbre dont la racine est le d_i-ème fils de s_i.

Remarque. Les différents nœuds qui composent une chaîne doivent se trouver sur une même branche, mais peuvent parfaitement ne pas être consécutifs sur cette branche.

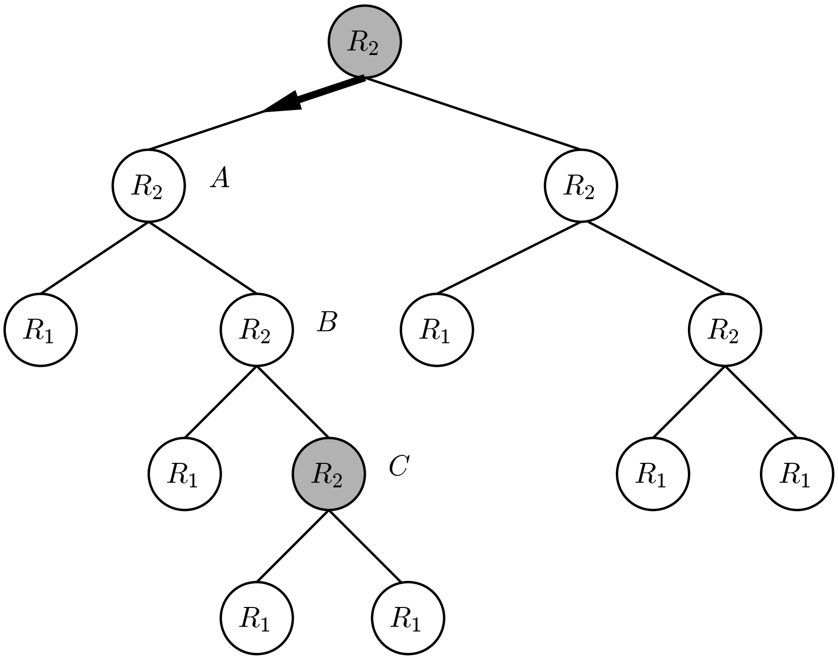


FIG. 2.2: *Une chaîne de type* $R_2^{(1)} R_2$

Exemple 2.18. Nous avons vu précédemment que, dans la grammaire G_1 engendrant les mots de Dyck, le paramètre “aire de Carlitz” est le paramètre élémentaire $p_{R_2^{(1)} R_2}$. Autrement dit, son terme de croissance, pour chaque sommet s étiqueté R_2 d'un arbre de dérivation, est égal au nombre de sommets étiquetés R_2 dans le sous-arbre gauche de s. Ceci revient à dire que le paramètre $p_{R_2^{(1)} R_2}$ compte le nombre de chaînes de type $R_2^{(1)} R_2$ dans les arbres de dérivation.

Ainsi, l'arbre de la figure 2.2 possède 3 chaînes de type $R_2^{(1)} R_2$ (chacune ayant la racine comme premier sommet, et l'un des 3 sommets internes du sous-arbre gauche, A, B, et C, comme second sommet). Cet arbre est, dans la grammaire G_1, l'arbre de dérivation du mot $w = aabababbabab$. Le chemin de Dyck correspondant, représenté figure 2.3, a bien pour aire de Carlitz 3.

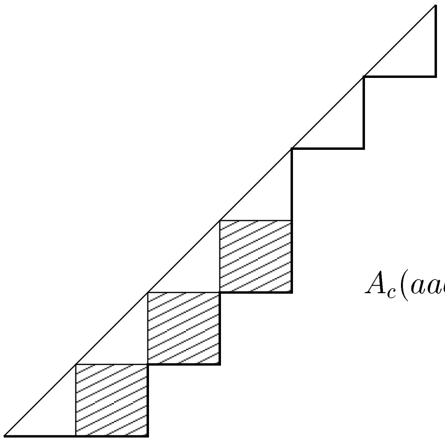


FIG. 2.3: *Chemin de Dyck associé à l'arbre de la figure 2.2*

Nous allons voir que ce type d'interprétation est généralisable à tous les paramètres élémentaires, quel que soit leur rang.

Lemme 2.19. *Soit w un mot engendré par la grammaire G, et p un paramètre Q-comptable.*

Soit $\mathcal{A}$ l'arbre de dérivation de w. Pour chaque nœud s de $\mathcal{A}$, soit $p'(s)$ le terme de croissance de p correspondant au nœud s.

Alors

$$p(w) = \sum_{s \in \mathcal{A}} p'(s) \tag{1}$$

Preuve. Ce lemme se prouve par récurrence sur la taille des arbres de dérivation. Il est en effet évident pour un arbre n'ayant qu'un seul nœud; supposons-le vrai pour les arbres ayant au plus n nœuds, et soit w un mot dont l'arbre de dérivation comporte $n+1$ nœuds.

Soit r la racine de cet arbre, $s_1, \dots, s_k$ ses fils, et $w_1, \dots, w_k$ les mots dont les arbres de dérivation sont les sous-arbres $\mathcal{A}_1, \dots, \mathcal{A}_k$ de $\mathcal{A}$ de racines respectives $s_1, \dots, s_k$. Chacun des sous-arbres de racine s_i a au plus n nœuds, donc le lemme s'applique à chaque w_i. Or, la différence entre $p(w)$ et la somme $\sum_{i=1}^{k} p(w_i)$ est, par définition, le terme de croissance

$p'(r)$. On a donc

$$\begin{aligned} p(w) &= p'(r) + \sum_{i=1}^{k} p(w_i) \\ &= p'(r) + \sum_{i=1}^{k} \sum_{s \in \mathcal{A}_i} p'(s) \\ &= \sum_{s \in \mathcal{A}} p'(s) \end{aligned}$$

ce qui prouve le lemme pour $n+1$ et permet de conclure par récurrence. □

Ce lemme exprime simplement le fait qu'un paramètre est la somme de tous ses termes de croissance; il nous permet de donner rapidement une interprétation des paramètres élémentaires.

Proposition 2.20. *Soit w un mot engendré par la grammaire G, et soit p_W un paramètre élémentaire ($W \in \mathcal{R}_p^*\mathcal{R}$). Alors $p_W(w)$ est le nombre de chaînes de type W de l'arbre de dérivation de w.*

Preuve. Dans le cas où le paramètre est de rang 1 (en ce cas, $W = R \in \mathcal{R}$), il s'agit d'une application immédiate du lemme 2.19. En effet, dans l'arbre de dérivation, le terme de croissance d'un paramètre élémentaire de rang 1 vaut 1 pour un nœud étiqueté par la règle R, et 0 dans les autres cas. Par conséquent, la somme de ces contributions sera le nombre de nœuds étiquetés R, qui forment les "chaînes de type R".

Supposons maintenant que p_W soit de rang $k > 1$; W est alors de longueur k, et $W = R^{(i)}W'$ où $R \in \mathcal{R}$ et $|W'| = k-1$. Toute chaîne de type W est composée d'un premier nœud s_1, étiqueté R, et d'une chaîne de type W' du sous-arbre dont la racine est le i-ème fils de s_1. Le nombre de chaînes de type W dont le premier nœud est s_1, est donc le terme de croissance $p'(s_1)$. Le lemme 2.19 permet alors d'affirmer que $p_W(w)$ est le nombre de chaînes de type W de l'arbre de dérivation de w. □

La proposition 2.20 permet, connaissant une décomposition d'un paramètre Q-comptable comme combinaison linéaire de paramètres élémentaires, d'en donner une interprétation comme comptant certains types de chaînes dans les arbres de dérivations. Ce comptage doit être pondéré par les coefficients de la décomposition. Par exemple, si un paramètre élémentaire est présent avec un coefficient 2, chaque chaîne correspondante doit être comptée 2 fois.

Dans [37], Fédou considère sur les arbres binaires complets une valuation qui fait intervenir un paramètre “somme des nombres de feuilles des sous-arbres gauches”, noté $\mathcal{G}(\mathcal{A})$. Les arbres binaires complets sont équivalents aux arbres de dérivation de la grammaire classique engendrant le langage de Dyck (grammaire G_1 de l'exemple 2.3), puisque dans ces arbres de dérivation les feuilles sont toujours étiquetées R_1, et les nœuds internes, R_2. Dans ces conditions, compter la somme des nombres de feuilles des sous-arbres gauches revient exactement à compter les chaînes de type $R_2^{(1)}R_1$.

Notons à ce propos le lien que l'on peut établir entre ce paramètre $\mathcal{G}$ et l'aire de Carlitz : tout arbre binaire complet ayant une feuille de plus qu'il n'a de sommets internes, un arbre de dérivation a toujours autant de chaînes de type $R_2^{(1)}R_1$ que de chaînes de type $R_2^{(1)}R_2$ ou R_2. En d'autres termes, $p_{R_2^{(1)}R_1} = p_{R_2} + p_{R_2^{(1)}R_2}$, ce qui signifie que le paramètre $\mathcal{G}$ est identique au paramètre “aire de Carlitz plus demi-longueur”.

2.2.5 Ordre de grandeur maximal de paramètres

La proposition 2.20 permet également de donner a priori une borne maximale à la valeur d'un paramètre Q-comptable sur un mot dont l'arbre de dérivation est de taille fixée ou, ce qui revient à peu près au même, sur un mot de longueur fixée.

En effet, si un arbre comporte n nœuds, il est évident qu'il ne peut avoir plus de n^k chaînes – tous types confondus – de longueur k. Par conséquent, un paramètre élémentaire de rang k est forcément inférieur ou égal à n^k sur un mot dont l'arbre de dérivation comporte n nœuds.

Dans une grammaire non ambiguë, il y a une relation linéaire entre la longueur d'un mot et la taille de son arbre de dérivation :

Proposition 2.21. *Soit G une grammaire non ambiguë, engendrant un langage L. Pour chaque mot non vide $w \in L$, notons $||w||$ la taille de l'arbre de dérivation de w.*

Alors il existe deux constantes strictement positives C_1 et C_2 telles que, pour tout mot non vide $w \in L$,

$$C_1||w|| \leq |w| \leq C_2||w||. \tag{2}$$

Preuve. La majoration de $|w|$ est facile, et ne dépend pas de la non-ambiguïté de la grammaire : on peut prendre pour C_2, le nombre maximum de lettres (avec multiplicités) écrites par une règle de dérivation, ce qui assurera $|w| \leq C_2||w||$ pour tout w engendré par la grammaire.

Pour la minoration, il est nécessaire que la grammaire soit non ambiguë.

Soit, pour chaque $U \in N$, $\mathcal{A}_U$ un arbre de dérivation dont l'étiquette de la racine soit une U-dérivation, et qui représente un mot dont la longueur soit minimale parmi les mots du langage $L_G(U)$. Soit $C' = \max_{U\in N} ||\mathcal{A}_U||$.

Soit maintenant k un entier tel que tout arbre de dérivation de profondeur k, vérifie $||\mathcal{A}|| > C'$ ($k = C' + 1$ convient parfaitement), et soit C la plus grande taille d'un arbre de dérivation de profondeur inférieure ou égale à k.

Nous allons montrer par récurrence sur n que, pour tout arbre de dérivation $\mathcal{A}$, si $|\mathcal{A}| \leq n$, alors $||\mathcal{A}|| \leq Cn + C'$.

La propriété est clairement vraie pour $n = 0$, par définition de C'. Supposons-la vraie pour n, et soit $\mathcal{A}$ un arbre de dérivation tel que $|\mathcal{A}| = n + 1$.

Si $\mathcal{A}$ est de profondeur inférieure à k, alors nécessairement $||\mathcal{A}|| \leq C \leq C(n+1) + C'$. Supposons donc que $\mathcal{A}$ soit de profondeur supérieure ou égale à k, et choisissons dans $\mathcal{A}$ un sommet s tel que le sous-arbre issu de s soit, lui, de profondeur k; soit U le membre gauche de l'étiquette de s, et soit $\mathcal{B} = \mathcal{A}(s, \mathcal{A}_U)$.

Nous avons

$$|\mathcal{B}| = |\mathcal{A}| - |s| + |\mathcal{A}_U|, \tag{3}$$

$$||\mathcal{B}|| = ||\mathcal{A}|| - ||s|| + ||\mathcal{A}_U||. \tag{4}$$

Nous avons forcément $|s| > C' \geq |\mathcal{A}_U|$, donc $|\mathcal{B}| \leq n$ et l'hypothèse de récurrence s'applique à $\mathcal{B}$: $||\mathcal{B}|| \leq Cn + C'$. En reprenant (4), nous obtenons alors

$$\begin{aligned} ||\mathcal{A}|| &= ||\mathcal{B}|| + ||s|| - ||\mathcal{A}_U|| \\ &\leq ||\mathcal{B}|| + ||s|| \\ &\leq Cn + C' + C, \end{aligned}$$

qui prouve bien que la propriété est vraie pour $n + 1$. □

L'intérêt de cette proposition est essentiellement de nous garantir que, pour des mots très longs, la taille d'un arbre de dérivation ou la longueur du mot correspondant ont toujours le même ordre de grandeur. Par conséquent, évaluer un ordre de grandeur en fonction de la taille des mots, ou en fonction de la taille des arbres de dérivation, est équivalent.

Nous commençons par donner une majoration élémentaire de l'ordre de grandeur d'un paramètre Q-comptable de rang connu.

Proposition 2.22. *Soit p un paramètre Q-comptable de rang k. Il existe une constante K telle que, pour tout mot non vide engendré par la grammaire,*

$$p(w) \leq K|w|^k. \tag{5}$$

Preuve. Dans le cas d'un paramètre élémentaire de rang k, cela découle immédiatement du fait que le nombre de chaînes est forcément inférieur au nombre de k-uplets de sommets de l'arbre; la décomposition en combinaison linéaire de paramètres élémentaires permet d'étendre la proposition à tout paramètre Q-comptable. □

Essentiellement, nous venons de montrer que l'ordre de grandeur maximal d'un paramètre Q-comptable par rapport à la taille des mots sur lesquels on le calcule, est limité par son rang. Nous allons maintenant donner une *minoration* de cet ordre de grandeur maximal qui permettra, dans certains cas, d'affirmer que cette limite est bien atteinte.

Définition 2.23. On appelle *rang minimal* d'un paramètre Q-comptable, la plus petite constante positive k telle qu'il existe une constante K vérifiant, pour tout mot w non vide engendré par la grammaire,

$$p(w) \leq K|w|^k. \tag{6}$$

Il est clair, d'après la proposition 2.22, que le rang minimal d'un paramètre Q-comptable existe[2] et est inférieur ou égal à son rang. Notons ici que ce rang minimal ne dépend pas de la grammaire, mais seulement des valeurs du paramètre, contrairement au rang. Il est moins évident que, comme nous allons le montrer, il s'agit toujours d'un entier.

Le cas des paramètres de rang 1 est un peu à part, et nous le traitons en premier.

Proposition 2.24. *Soit $G = (X, N, \mathcal{R}, S_0)$ une grammaire non ambiguë.*

Soit $p = p_R$ un paramètre élémentaire de rang 1, et soit S le symbole du membre gauche de R. Les conditions suivantes sont équivalentes :

1. *p est de rang minimal 1.*
2. *p n'est pas borné sur $L_G(S_0)$.*
3. *il existe un symbole S' tel que p ne soit pas borné sur $L_G(S')$.*
4. *S est accessible à partir d'un symbole figurant dans le membre droit de R, ou il existe un symbole S' tel que S et S' soient simultanément accessibles à partir de S'.*

2. Tel qu'il est défini ci-dessus, le rang minimal n'est qu'une borne inférieure et non un minimum; toutefois, dans toutes les démonstrations qui suivent pour les paramètres Q-comptables, il est aisé de vérifier qu'il s'agit effectivement d'un minimum.

La condition (1) est celle qui nous intéresse; les conditions (2) et (3) sont en apparence plus faibles, et la condition (4) est celle qui, techniquement, est la plus simple à tester sur une grammaire donnée lorsque le paramètre est seulement connu par les chaînes qu'il énumère.

En particulier, les conditions (1) et (2) impliquent qu'un paramètre élémentaire de rang 1, est forcément de rang minimal 0 ou 1; il n'est pas possible qu'un tel paramètre ait un ordre de grandeur maximal qui suive une loi de puissance non entière, ou logarithmique.

Preuve. (1) $\Rightarrow$ (2) $\Rightarrow$ (3) est évident. (3) $\Rightarrow$ (4) est plus simple à prouver par l'absurde : montrons donc que, si la condition (4) n'est pas vérifiée, le paramètre p est borné sur chaque langage $L_G(S')$, ce qui revient à dire qu'un arbre de dérivation de G ne peut contenir qu'un nombre borné de sommets étiquetés R.

Tout d'abord, dire que S n'est pas accessible à partir des symboles du membre droit de S, revient à dire qu'aucune branche d'un arbre de dérivation de G ne peut contenir plus d'un sommet étiqueté R. De plus, si une branche d'un tel arbre de dérivation contient deux sommets s_1 et s_2 ayant la même étiquette, aucun sommet compris entre s_1 et s_2 sur cette branche ne peut avoir R pour étiquette.

Supposons de plus que, pour tout symbole S', S et S' ne sont pas simultanément accessibles à partir de S'. Alors, si deux sommets s_1 et s_2 d'un arbre de dérivation ont même étiquette et sont sur une même branche (s_1 étant un ancêtre de s_2), tout sommet étiqueté R dans le sous-arbre issu de s_1, doit en fait se trouver dans le sous-arbre issu de s_2. Par conséquent, on peut, dans l'arbre de dérivation, remplacer le sous-arbre issu de s_1 par celui issu de s_2, sans changer le nombre de sommets étiquetés R. En procédant ainsi tant qu'il reste dans l'arbre des branches contenant des sommets de même étiquette, on se ramène à un arbre de profondeur bornée (par le nombre de règles de dérivation), sans modifier la valeur de p. Par conséquent, p, qui prend toutes ses valeurs sur un ensemble fini d'arbres de dérivation, est borné.

Supposons maintenant (4) vraie, et prouvons (1). Nous avons deux cas possibles : soit S est accessible à partir des symboles du membre droit de R, soit il existe un symbole S' tel que S et S' soient simultanément accessibles à partir de S'. Nous exhibons, dans chaque cas, une famille d'arbres de dérivation $(\mathcal{A}_n)_{n\geq 1}$, tels que $|\mathcal{A}| \leq Kn$ et $p(\mathcal{A}_n) \geq n$.

Considérons d'abord le cas où S est accessible à partir d'un symbole du membre droit de R : cela revient à dire que, dans un arbre de dérivation dont la racine est étiquetée R, il peut y avoir un autre sommet étiqueté R. Soit $\mathcal{A}$ un tel arbre, et soit s un sommet étiqueté R de $\mathcal{A}$. Soit alors $(\mathcal{A}_n)_{n\geq 1}$ la suite d'arbres définie récursivement par :

- $\mathcal{A}_1 = \mathcal{A}$;

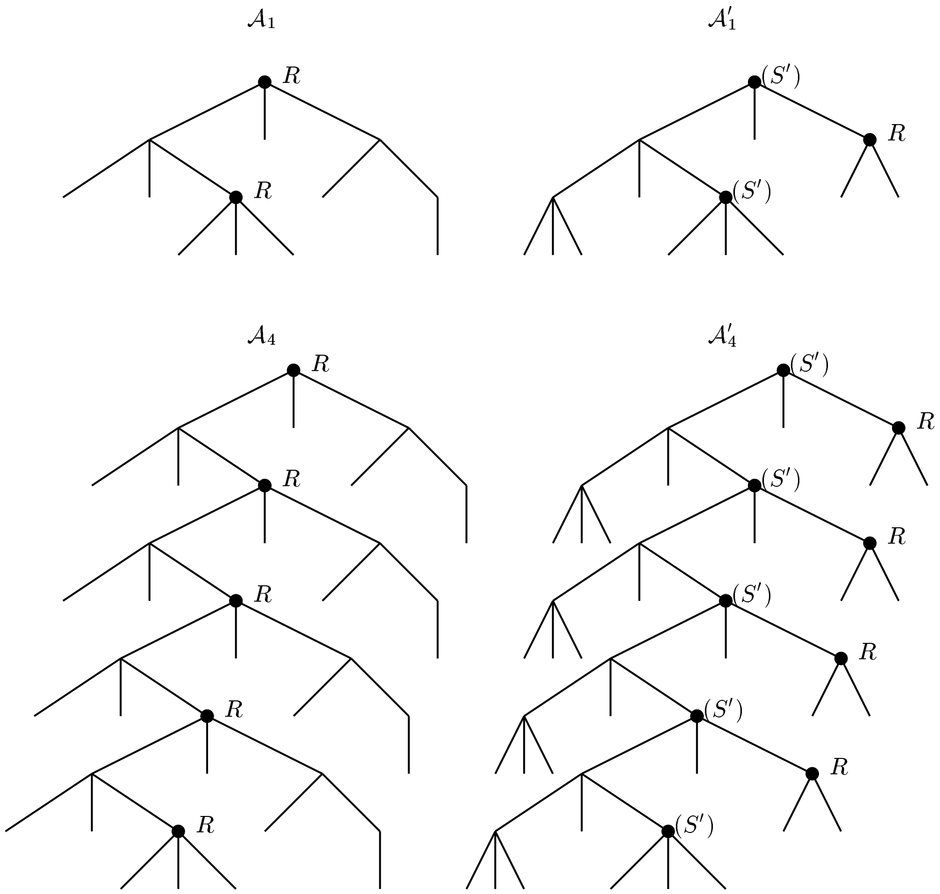


FIG. 2.4: *Construction des arbres $\mathcal{A}_k$ et $\mathcal{A}'_k$*

– $\mathcal{A}_{k+1} = \mathcal{A}(s, \mathcal{A}_k)$.

Un exemple de construction des arbres $\mathcal{A}_k$ est présenté figure 2.4.

Il est clair que $|\mathcal{A}_n| \leq n|\mathcal{A}|$ et que $p(\mathcal{A}_n) \geq n+1$.

Lorsque S et S' sont simultanément accessibles à partir de S', il suffit de prendre comme arbre $\mathcal{A}'$ un arbre dont la racine est étiquetée par une S'-dérivation, et comportant des sommets s et s' respectivement étiquetés par cette même S'-dérivation et par R, sans que ces deux sommets soient sur la même branche. La famille $(\mathcal{A}'_n)_{n\geq 1}$ est alors définie de la même manière : on a encore $|\mathcal{A}'_n| \leq n|\mathcal{A}'|$ et $p(\mathcal{A}'_n) \geq n+1$.

La construction des arbres $\mathcal{A}'_k$ est également présentée figure 2.4. $\square$

Exemple 2.25. Reprenons la grammaire G_2 de l'exemple 2.3, qui engendre le langage de

Dyck. Les règles de cette grammaire d'axiome D sont :

$$\begin{array}{llll} R_1: & D & \to & \epsilon \\ R_2: & D & \to & E \\ R_3: & E & \to & ab \\ R_4: & E & \to & abE \\ R_5: & E & \to & aEb \\ R_6: & E & \to & aEbE. \end{array}$$

Dans cette grammaire, D n'est accessible à partir d'aucun symbole (il n'apparaît dans le membre droit d'aucune règle), tandis que E est accessible à partir de D comme de E. Pour finir, E et E sont simultanément accessibles à partir de D ou de E (règle R_6).

Les règles R_1 et R_2 étant des D-dérivations, les paramètres p_{R_1} et p_{R_2} sont de rang minimal 0 : ils sont tous deux bornés. Plus précisément, p_{R_1} vaut 1 pour le mot vide et 0 pour tout autre mot de Dyck, et p_{R_2} vaut 0 pour le mot vide et 1 pour tout autre mot de Dyck.

En revanche, les règles R_3 à R_6 étant des E-dérivations, les paramètres p_{R_3}, p_{R_4}, p_{R_5} et p_{R_6} sont de rang minimal 1. Le paramètre p_{R_3} représente, sur tous les arbres de dérivation de mots de Dyck non vides, le nombre de feuilles (la règle R_3 est la seule règle terminale apparaissant dans leurs arbres de dérivation); p_{R_6} compte, pour sa part, les sommets internes de degré 2 (R_6 est la seule règle d'arité 2), et p_{R_4} et p_{R_5} comptent chacun un type différent de sommets de degré 1.

A titre d'exemples de suites de mots illustrant le fait que ces paramètres sont bien de rang minimal 1, citons :

- pour p_{R_4} : $w_n = (ab)^n$ vérifie $p_{R_4}(w_n) = n - 1$, et $|w_n| = 2n$;
- pour p_{R_5} : $w_n = a^n b^n$ vérifie $p_{R_5}(w_n) = n - 1$, et $|w_n| = 2n$;
- pour p_{R_3} et p_{R_6} : $w_n = (aabb)^n ab$ vérifie $p_{R_3}(w_n) = n + 1$ et $p_{R_6}(w_n) = n$, et $|w_n| = 4n + 2$.

Le rang minimal d'un paramètre Q-comptable quelconque peut être déterminé assez aisément par la proposition suivante :

Proposition 2.26. *Soit $p = p_W$ un paramètre élémentaire de rang k, non identiquement nul, avec $W = R_1^{(d_1)} \dots R_{k-1}^{(d_{k-1})} R_k$. Notons, pour chaque $i \leq k$, S_i le symbole non terminal du membre gauche de R_i, et, pour $i < k$, S'_i le d_i-ème symbole du membre droit de R_i. S_0 désigne l'axiome de la grammaire.*

Le rang minimal de p est obtenu en comptant, parmi les indices $i < k$, ceux pour lesquels S_i est accessible depuis S'_i; à ce total, il convient d'ajouter 1 si et seulement si p_{R_k} est de rang minimal 1.

Preuve. Tout d'abord, remarquons que p_W n'est pas identiquement nul, si et seulement si, pour chaque $i < k$, S_{i+1} est accessible à partir de S'_i ou égal à S'_i (il faut que le symbole S_{i+1} apparaisse dans au moins un arbre de dérivation dont la racine est une S'_i-dérivation).

Notons k' le rang minimal donné par l'énoncé. Nous commençons par prouver que le paramètre p est au moins de rang minimal k', en exhibant une famille $(\mathcal{A}_n)_{n\geq 0}$ d'arbres de dérivation tels que $|\mathcal{A}_n| \leq Kn$ et $p(\mathcal{A}_n) \geq n$.

Pour chaque $i < k$, soit $\mathcal{B}_i$ un arbre de dérivation vérifiant les conditions suivantes :

- la racine de $\mathcal{B}_i$ est étiquetée par R_i;
- le d_i-ème sous-arbre de $\mathcal{B}_i$ comporte un sommet s_{i+1}, étiqueté par R_{i+1};
- si S_i est accessible à partir de S'_i (c'est-à-dire, si l'indice i fait partie de ceux qui contribuent au rang minimal annoncé), le d_i-ème sous-arbre de $\mathcal{B}_i$ comporte un sommet s'_i étiqueté R_i.

Pour chaque indice i tel que S_i soit accessible à partir de S'_i, soit $(\mathcal{B}_{i,n})_{n\geq 1}$ la suite d'arbres de dérivation définie par :

- $\mathcal{B}_{i,1} = \mathcal{B}_i$;
- $\mathcal{B}_{i,k+1} = \mathcal{B}_i(s'_i, \mathcal{B}_{i,k})$.

Cette définition assure que, dans chaque arbre $\mathcal{B}_{i,n}$, il existe une branche contenant n sommets étiquetés R_i, chacun étant un descendant du d_i-ème fils du précédent. Autrement dit, l'arbre $\mathcal{B}_{i,n}$ possède au moins une chaîne de type $\left(R_i^{(d_i)}\right)^{n-1} R_i$. La taille de cet arbre vérifie également $|\mathcal{B}_{i,n}| \leq n|\mathcal{B}_i|$.

Posons maintenant

$$\mathcal{B}'_i = \begin{cases} \mathcal{B}_{i,n} & \text{si } S_i \text{ est accessible à partir de } S'_i, \\ \mathcal{B}_i & \text{sinon.} \end{cases}$$

Pour $\mathcal{B}'_k$, nous prenons l'arbre $\mathcal{A}_n$ construit dans la preuve de la proposition 2.24 si p_{R_k} est de rang minimal 1 (c'est-à-dire si k contribue au rang minimal), et $\mathcal{B}_k$ sinon.

Soit maintenant $\mathcal{A}_n = \mathcal{B}'_1(s_2, \mathcal{B}'_2(s_3, \ldots \mathcal{B}'_{k-1}(s_k, \mathcal{B}'_k)\cdots))$. Il est clair que la taille de $\mathcal{A}_n$ est $|\mathcal{A}_n| \leq n(|\mathcal{B}_1| + \cdots + |\mathcal{B}_k|$.

Par ailleurs, l'arbre $\mathcal{A}_n$ vérifie $p(\mathcal{A}_n) \geq n^{k'}$, ce qui permet d'affirmer que p est au moins de rang minimal k'.

Montrons maintenant, par récurrence sur k, que k' est bien le rang minimal de p; pour cela, nous devons montrer que tout arbre $\mathcal{A}$ tel que $p(\mathcal{A}) \geq Kn^{k'}$ vérifie $|\mathcal{A}| \geq K'n$. La propriété est vraie pour $k = 1$ (c'est la proposition 2.24), supposons donc qu'elle l'est pour k et choisissons un paramètre élémentaire $p = p_W$ de rang $k + 1$. Posons $W = R_1^{(d_1)} W'$.

Deux cas se présentent : ou bien S_1 est accessible à partir de S'_1, auquel cas $p_{W'}$ est de rang minimal $k' - 1$; ou bien il ne l'est pas, auquel cas $p_{W'}$ est de rang minimal k'.

Le premier cas ($p_{W'}$ est de rang minimal $k' - 1$) est facile : en effet, un arbre de taille n a au plus $Kn^{k'-1}$ chaînes de type W', et au plus n sommets étiquetés R_1; par conséquent, il ne peut avoir plus de $n.K.n^{k'-1} = Kn^{k'}$ chaînes de type W.

Le deuxième cas ($p_{W'}$ est déjà de rang minimal k') est légèrement plus complexe. Chaque chaîne de type W' ne peut être complétée en chaîne de type W qu'au plus une fois, puisque la branche reliant la racine de l'arbre au sommet s_1 ne peut comporter qu'au plus une dérivation pointée de type $R_1^{d_1}$. Par conséquent $p_W(\mathcal{A}) \leq p_{W'}(\mathcal{A}) \leq Kn^{k'}$.

Dans tous les cas, le paramètre p_W est bien de rang minimal k'. □

Un cas particulier intéressant est le suivant :

Corollaire 2.27. *Si chaque symbole de la grammaire est accessible à partir de chaque symbole, tout paramètre Q-comptable a un rang minimal égal à son rang.*

Remarque. Si la grammaire ne comporte qu'un seul symbole, et engendre un langage infini, l'unique symbole est forcément accessible à partir de lui-même, et par conséquent chaque paramètre a un rang minimal égal à son rang.

Il est intéressant de noter que, dans le cas général, calculer le rang minimal d'un paramètre est assez simple, puisque tout se ramène à déterminer quels symboles sont accessibles à partir desquels, et quels couples de symboles sont accessibles à partir desquels. Dutour donne dans [36] un algorithme permettant de décider si un symbole est ou non accessible à partir d'un autre; quant à l'accessibilité simultanée, elle peut être calculée en adaptant aux couples de symboles l'algorithme donné dans [36].

La proposition 2.26 peut être utile dans la pratique lorsqu'il s'agit de déterminer si un paramètre donné p, défini indépendamment de toute grammaire, est Q-comptable dans une grammaire donnée. Si le rang minimal de p est connu, les paramètres élémentaires pouvant intervenir dans une décomposition de p sont en nombre fini. En calculant, pour un assez grand nombre de mots, les valeurs de chacun de ces paramètres élémentaires $p_1, \ldots, p_N$, il est possible, soit de montrer que p ne peut pas s'écrire comme combinaison linéaire de tels paramètres (parce que, par exemple, pour M mots $w_1, \ldots, w_M$, le vecteur $(p(w_j))_{1 \leq j \leq M}$ ne peut s'écrire comme combinaison linéaire des vecteurs $(p_i(w_j))_{1 \leq j \leq M}$, pour $1 \leq i \leq N$), soit de restreindre les choix possibles de telles combinaisons linéaires jusqu'à ce qu'une preuve directe de la Q-comptabilité de p soit envisageable. Le principal obstacle à l'automatisation de ce genre de calculs réside dans le choix judicieux des mots pour lesquels effectuer les calculs; le nombre de mots d'un langage croît fréquemment de manière exponentielle avec leur longueur. Dans les cas simples, toutefois, il est fréquent que l'examen de quelques mots courts du langage engendré soit suffisant pour montrer qu'un paramètre donné n'est pas Q-comptable; la quasi-totalité de nos preuves de non-Q-comptabilité seront de ce type.

2.3 Séries génératrices

Nous nous intéressons maintenant aux liens entre Q-grammaires et séries génératrices, qui nous fournissent en fait la principale motivation pour l'étude des Q-grammaires et des paramètres Q-comptables.

2.3.1 Substitutions de variable

Nous considérons ici un ensemble ordonné $(x_1, \ldots, x_n)$ de variables formelles, et des séries formelles $F(x_1, \ldots, x_n)$, à coefficients entiers (le plus souvent positifs ou nuls).

Dans de nombreux problèmes d'énumération suivant plus d'un paramètre, on rencontre des équations portant sur des séries formelles et qui font intervenir, au lieu de la série à une variable $F(x)$, une série bivariée $F(x, q)$ et la série $F(xq, q)$, où la variable x a été multipliée par q; voir par exemple [37, 13, 28]. Nous allons considérer une généralisation de ce genre de transformation, au cas où les variables formelles sont plus nombreuses.

Définition 2.28. Soit, pour tout $i \leq n$, $A_i = x_i x_{i+1}^{\alpha_{i,i+1}} \ldots x_n^{\alpha_{i,n}}$ un monôme ne faisant intervenir que les variables formelles x_j telles que $i \leq j$ (et dans lequel x_i n'apparaît qu'au degré 1).

Nous noterons $\sigma_{(x_i \leftarrow A_i)}$ l'opérateur linéaire défini sur l'algèbre de séries formelles $\mathbb{Q}[[x_1, \ldots, x_n]]$ par

$$\sigma_{(x_i \leftarrow A_i)} F(x_1, \ldots, x_n) = F(A_1, \ldots, A_n). \tag{7}$$

Nous appellerons une telle transformation une substitution des variables $x_1, \ldots, x_n$.

Notons que si l'on pose $\alpha_{i,i} = 1$ et $\alpha_{i,j} = 0$ lorsque $i > j$, une substitution de variables σ est entièrement définie par la matrice à coefficients entiers positifs ou nuls $M_\sigma = (\alpha_{i,j})_{1\leq i,j\leq n}$. Nous appellerons cette matrice M_σ, la *matrice de la substitution.* Les matrices de substitutions de variables sont exactement les matrices triangulaires supérieures à coefficients dans $\mathbb{N}$ et dont les coefficients diagonaux sont tous égaux à 1.

Le rapport entre une substitution de variables et sa matrice est donné par le lemme suivant :

Lemme 2.29. *Soient σ et σ' deux substitutions de variables de matrices respectives $M_\sigma = (\alpha_{i,j})_{1\leq i,j\leq n}$ et $M_{\sigma'} = \left(\alpha'_{i,j}\right)_{1\leq i,j\leq n}$. Alors $\sigma' \circ \sigma$ est une substitution de variables, dont la matrice est $M_{\sigma'} M_\sigma$.*

Preuve. Soit, pour $1 \leq i \leq n$, $B_i = A_i A_{i+1}^{\alpha'_{i,i+1}} \dots A_n^{\alpha'_{i,n}}$. En composant σ et σ', il vient naturellement

$$\begin{aligned} \sigma' \circ \sigma F(x_1, \dots, x_n) &= \sigma' F(A_1, \dots, A_n) \\ &= F(B_1, \dots, B_n). \end{aligned}$$

Ceci montre que $\sigma' \circ \sigma$ est une substitution de variables, pourvu que chaque B_i soit bien de la forme requise.

Or, nous avons

$$\begin{aligned} B_i &= A_i A_{i+1}^{\alpha'_{i,i+1}} \dots A_n^{\alpha'_{i,n}} \\ &= \left(x_i x_{i+1}^{\alpha_{i,i+1}} \dots x_n^{\alpha_{i,n}}\right) \left(x_{i+1} x_{i+2}^{\alpha_{i+1,i+2}} \dots x_n^{\alpha_{i+1,n}}\right)^{\alpha'_{i,i+1}} \dots \left(x_{n-1} x_n^{\alpha_{n-1,n}}\right)^{\alpha'_{i,n-1}} x_n^{\alpha'_{i,n}} \\ &= x_i x_{i+1}^{\beta_{i,i+1}} \dots x_n^{\beta_{i,n}}, \end{aligned}$$

où chaque coefficient $\beta_{i,j}$ (pour $i < j \leq n$) est donné par

$$\beta_{i,j} = \sum_{k=1}^{n} \alpha_{k,j} \alpha'_{i,k}. \tag{8}$$

On reconnaît dans $\beta_{i,j}$ le coefficient de $M_{\sigma'} M_\sigma$, ce qui prouve bien que $\sigma' \circ \sigma$ est la substitution de variables dont la matrice est $M_{\sigma'} M_\sigma$. □

Dans la pratique, il arrivera fréquemment que des substitutions de variables ne modifient qu'une partie des variables formelles :

Notation 2.30. Lorsque certains des monômes A_i ne sont pas définis, $A_i = x_i$ est implicitement supposé; ainsi, dans $\mathbb{Q}[[x, y, z]]$, $\sigma_{x\leftarrow xz^2} = \sigma_{x\leftarrow xz^2, y\leftarrow y, z\leftarrow z}$ et l'on a

$$\sigma_{x\leftarrow xz^2} F(x, y, z) = F(xz^2, y, z).$$

Il est clair que $\sigma_{x_i\leftarrow x_iA}$ et $\sigma_{x_i\leftarrow x_iB}$ commutent, et que l'on a $\sigma_{x_i\leftarrow x_iA}\circ\sigma_{x_i\leftarrow x_iB} = \sigma_{x_i\leftarrow x_iAB}$; cette propriété ne s'étend pas aux substitutions dans deux variables différentes.

Exemple 2.31. Considérons des séries formelles à trois variables x, y, z, et posons $\sigma_1 = \sigma_{x\leftarrow xy}$ et $\sigma_2 = \sigma_{y\leftarrow yz}$. Alors

$$\begin{aligned} \sigma_1\circ\sigma_2 F(x,y,z) &= \sigma_1 F(x,yz,z) \\ &= F(xy,yz,z), \\ \sigma_2\circ\sigma_1 F(x,y,z) &= \sigma_2 F(xy,y,z) \\ &= F(xyz,yz,z). \end{aligned}$$

Ou, en termes de matrices :

$$M_{\sigma_1} = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad M_{\sigma_2} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix},$$

$$M_{\sigma_1\circ\sigma_2} = M_{\sigma_2}.M_{\sigma_1} = \begin{pmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}, \quad M_{\sigma_2\circ\sigma_1} = M_{\sigma_1}.M_{\sigma_2} = \begin{pmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}.$$

Ce genre de substitutions de variables apparaît fréquemment dans les travaux sur les q-grammaires, où l'on ne rencontre généralement qu'une substitution $\sigma_{x\leftarrow xq}$. Ainsi que nous allons le voir, notre définition des paramètres Q-comptables correspond en fait à la généralisation à d'autres substitutions.

L'exemple (2.31) montre que, lorsque les substitutions portent sur des variables différentes, l'ordre dans lequel on les compose est important. Or, si l'on pense à σ_1 comme “multiplier x par y” et à σ_2 comme “multiplier y par z”, le résultat attendu de la composition est $\sigma_1\circ\sigma_2$ plutôt que $\sigma_2\circ\sigma_1$. En d'autres termes, il convient de composer de telles substitutions dans l'ordre décroissant des variables substituées; ainsi, en utilisant la notation définie précédemment,

$$\sigma_{(x_i\leftarrow A_i)} = \sigma_{x_1\leftarrow A_1}\circ\cdots\circ\sigma_{x_{n-1}\leftarrow A_{n-1}}. \tag{9}$$

Définition 2.32. Soient $x_1, \ldots, x_n$ n variables formelles, et soit $\sigma = \sigma_{(x_i\leftarrow A_i)}$ une substitution des variables $x_1, \ldots, x_n$, dont la matrice est M_σ. Nous appellerons restriction de σ à $x_1, \ldots, x_k$ $(k < n)$, la substitution de $x_1, \ldots, x_k$ dont la matrice est formée des k premières lignes et colonnes de M_σ.

En termes de variables et de séries formelles, restreindre une substitution de variables aux k premières variables revient à fixer toutes les autres variables formelles à la valeur 1.

2.3.2 Q-analogue d'un système d'équations

Dans la méthodologie “classique” de Schützenberger, une grammaire algébrique se traduit, en termes de séries génératrices, par un système d'équations algébriques dont les inconnues sont les séries génératrices des langages engendrés par la grammaire. Ces séries sont des séries formelles en des variables $x_1, \ldots, x_n$ correspondant aux lettres de l'alphabet, et les coefficients du système sont des polynômes en ces mêmes variables.

La façon la plus simple d'obtenir un tel système algébrique à partir de la grammaire est tout bonnement d'écrire la grammaire comme un système d'équations (en variables non commutatives) sur les langages, et de faire commuter lettres et langages, ces derniers étant remplacés par leurs séries génératrices qui font figure de séries formelles inconnues. La non ambiguïté des grammaires est ici cruciale.

De manière générale, on peut écrire le système d'équations sous la forme

$$U = \sum_{R \in \mathcal{R}(U)} v(R) \prod_{1 \leq j \leq \alpha(R)} d(R,j) \quad (U \in N) \tag{10}$$

où $\mathcal{R}(U)$ désigne l'ensemble des U-dérivations, $v(R)$ le produit commutatif des lettres produites par la règle R, et $d(R,j)$, le j-ème symbole (pris ici comme série inconnue) du membre droit de R.

La notion usuelle de q-analogue d'un tel système correspond à ajouter aux séries formelles une variable q, et à modifier le système de telle sorte que, lorsque $q = 1$, on retrouve le système d'origine; généralement, ces q-systèmes font intervenir des substitutions $\sigma_{x_i \leftarrow x_i q}$.

Exemple 2.33. La grammaire G_2 de l'exemple 2.3 se traduit automatiquement par le système d'équations

$$\left\{ \begin{array}{rcl} D(a,b) & = & 1 + E(a,b) \\ E(a,b) & = & ab + 2abE(a,b) + abE(a,b)^2. \end{array} \right.$$

Un exemple de q-analogue de ce système est le suivant :

$$\left\{ \begin{array}{rcl} D(a,b;q) & = & 1 + E(a,b;q), \\ E(a,b;q) & = & abq + abqE(a,b;q) + abqE(aq,bq;q) + abqE(a,b;q)E(aq,bq;q). \end{array} \right.$$

En posant $x = ab$ (les variables a et b apparaissent toujours avec le même degré), nous obtenons deux autres systèmes. Le second est toujours un q-analogue du premier :

$$\begin{cases} D(x) &= 1 + E(x), \\ E(x) &= x + 2xE(x) + xE(x)^2; \end{cases}$$

$$\begin{cases} D(x;q) &= 1 + E(x;q), \\ E(x;q) &= xq + xqE(x;q) + xqE(xq^2;q) + xqE(x;q)E(xq^2;q). \end{cases}$$

Définition 2.34. Soit (S) un système de m équations algébriques, où chaque équation est de la forme

$$U_i = P_i(x_1, \ldots, x_n, U_1, \ldots, U_m) \tag{11}$$

avec P_i un polynôme à coefficients dans $\mathbb{N}$.

Soit $Q = (q_1, \ldots, q_k)$ un ensemble ordonné de k nouvelles variables formelles. Un Q-analogue de (S) est un système (S') de m équations, dont les inconnues sont des séries formelles des variables $x_1, \ldots, x_n, q_1, \ldots, q_k$, et qui s'écrit sous la forme

$$\tilde{U}_i = \tilde{P}_i\left(x_1, \ldots, x_n, q_1, \ldots, q_k, (\tilde{U}_i)_{1 \leq i \leq m}, \left(\sigma_j(\tilde{U}_i)\right)_{1 \leq j \leq s, 1 \leq i \leq m}\right) \tag{12}$$

où chaque $\tilde{P}_i$ est un polynôme à coefficients dans $\mathbb{N}$ et σ_j (pour $1 \leq j \leq s$) une substitution des variables $x_1, \ldots, x_n, q_1, \ldots, q_k$, avec les conditions suivantes :

- la restriction de σ_j aux variables $x_1, \ldots, x_n$ est l'identité;
- si, dans $\tilde{P}_i$, chaque variable q_ℓ $(1 \leq \ell \leq k)$ est remplacée par 1 et chaque $\sigma_j(\tilde{U}_i)$ est remplacé par U_i, on retrouve le polynôme P_i.

La façon la plus simple de comprendre comment former un Q-analogue d'un système d'équations est de considérer chaque polynôme P_i comme une somme de monômes unitaires, et de remplacer certains facteurs U_i dans ces monômes par $\sigma_j(U_i)$. De plus, chaque terme du polynôme est multiplié par un monôme unitaire ne faisant intervenir que les variables $q_1, \ldots, q_k$.

Examinons sur un exemple le lien entre Q-grammaires et Q-analogue d'un système d'équations.

Exemple 2.35. Reprenons la grammaire G' de l'exemple (2.33) :

$$\left\{\begin{array}{llllllll} R_1: & D & \to & \epsilon & R_2: & D & \to & E \\ R_3: & E & \to & ab & R_4: & E & \to & aEb \\ R_5: & E & \to & abD & R_6: & E & \to & aEbE \end{array}\right.$$

Le système d'équations algébriques correspondant est le suivant :

$$\left\{\begin{array}{lll} D(x) & = & 1+E(x), \\ E(x) & = & x+2xE(x)+xE(x)^2. \end{array}\right.$$

En le comparant à la grammaire G_2, il apparaît que le paramètre p_x compté par x, est égal à $p_{R'_3}+p_{R'_4}+p_{R'_5}+p_{R'_6}$.

En utilisant la substitution de variables $\sigma=\sigma_{x\leftarrow xq^2, r\leftarrow rs}$, nous pouvons former le (q,r,s)-analogue suivant :

$$\left\{\begin{array}{lll} D(x,q,r,s) & = & 1+E(x,q,r,s), \\ E(x,q,r,s) & = & xqrs+xqrsE(x,q,r,s)+xqE(xq^2,q,rs,s) \\ & & +xqE(xq^2,q,rs,s)E(x,q,r,s). \end{array}\right.$$

Chacune des variables compte un paramètre ayant un sens géométrique sur les chemins de Dyck : q compte *l'aire*, r compte les *pics*, et s compte la *somme des hauteurs des pics.*

La restriction aux variables x et q redonne le q-analogue présenté dans l'exemple (2.33).

La croissance du paramètre compté par q se fait de plusieurs manières :

- lors de l'utilisation des règles R'_3, R'_4, R'_5 et R'_6, il y a croissance de 1 (monômes $xqrs$ et xq);
- lors de l'utilisation des règles R'_5 et R'_6, il y a croissance d'un terme égal au double du degré de x dans une série $E(x,q,r,s)$ (substitution σ).

Par conséquent, q compte le paramètre Q-comptable de rang 2

$$p_q=p_{R'_3+R'_4+R'_5+R'_6}+2.p_{(R'^{(1)}_5+R'^{(1)}_6)(R'_3+R'_4+R'_5+R'_6)}.$$

De même, r croît de 1 à chaque utilisation des règles R'_3 et R'_4, donc $p_r=p_{R'_3+R'_4}$, et, en analysant la croissance de p_s, on obtient

$$p_s=p_{R'_3+R'_4}+p_{(R'^{(1)}_5+R'^{(1)}_6)(R'_3+R'_4)}.$$

Ainsi, nous avons décomposé en somme de paramètres élémentaires chacun des quatre paramètres.

Le théorème suivant indique dans toute sa généralité le lien entre Q-grammaires et Q-analogues de systèmes d'équations :

Théorème 2.36. *Soit G une grammaire engendrant k langages $L_1, \ldots, L_k$, et soit (S) le système d'équations algébriques correspondant à G.*

- *Les solutions de tout Q-analogue du système S, sont les séries génératrices d'une Q-grammaire basée sur la grammaire G;*
- *réciproquement, les séries génératrices de toute Q-grammaire basée sur G sont les solutions d'un Q-analogue de (S).*

Preuve. Commençons par montrer de quelle manière, à partir d'une Q-grammaire, on peut former un système d'équations qui est un Q-analogue du système algébrique (S).

Notons $q_1, \ldots, q_N$ les lettres de la grammaire G, et $q_{N+1}, \ldots, q_{N+M}$ les M lettres supplémentaires. Soient $p_1, \ldots, p_{N+M}$ les paramètres suivant lesquels s'effectue l'énumération (les N premiers étant de rang 1), et notons, pour chaque règle R et chaque paramètre p_m, les règles de calcul

$$R: \quad p_m(w) = \sum_{i=1}^{a(R)} p_m(w_i) + C_{R,m} + \sum_{\substack{1 \leq j < m \\ 1 \leq i \leq a(R)}} C_{R,m,i,j} . p_j(w_i).$$

Les N premiers paramètres étant de rang 1, pour $m \leq N$ les coefficients $C_{R,m,i,j}$ sont tous nuls, et $C_{R,m} = |v_0 v_1 \ldots v_{a(R)}|_{q_m}$.

Notons encore, pour $w \in L_G(U)$,

$$v(w) = \prod_{1 \leq m \leq N+M} q_m^{p_m(w)}$$

la valuation donnée au mot w par ces paramètres. Si w appartient à plusieurs langages $L_G(U)$ différents, on distinguera le langage, en notant $v(w, U)$ et $p_m(w, U)$.

Les règles de calcul sommatoire pour les paramètres se transforment en produits; ainsi, si $w = v_0 w_1 \ldots w_k v_k$ pour une certaine règle $R : U \to v_0 U_1 \ldots U_k v_k$, avec $w_i \in L_G(U_i)$, on a

$$\begin{aligned}
v(w,U) &= \prod_{m=1}^{N+M} q_m^{p_m(w,U)} \\
&= \prod_{m=1}^{N+M} q_m^{\left(\sum_{i=1}^{k} p_m(w_i,U_i)+C_{R,m}+\sum_{\substack{1\le j\le m\\ 1\le i\le k}} C_{R,m,i,j}\cdot p_j(w_i,U_i)\right)} \\
(13) \qquad &= \left(\prod_{i=1}^{k}\prod_{m=1}^{N+M} q_m^{p_m(w_i,U_i)}\right)\left(\prod_{m=1}^{N+M} q_m^{C_{R,m}}\right)\left(\prod_{\substack{1\le j<m\le N+M\\ 1\le i\le k}} q_m^{C_{R,m,i,j}\cdot p_j(w_i,U_i)}\right) \\
(14) \qquad &= \left(\prod_{1\le m\le N+M} q_m^{C_{R,m}}\right)\left(\prod_{i=1}^{k} v(w_i,U_i)\prod_{1\le j<m\le N+M} q_m^{C_{R,m,i,j}\cdot|v(w_i,U_i)|_{q_j}}\right).
\end{aligned}$$

Posons, pour chaque i $(1\le i\le k)$,

$$\sigma_{R,i} = \sigma_{\left(q_j\leftarrow q_j\prod_{m=j+1}^{N+M} q_m^{C_{R,m,i,j}}\right)}. \tag{15}$$

Les conditions sur les coefficients $C_{R,m,i,j}$ qui définissent les termes de croissance des paramètres, sont exactement celles qui font de chaque $\sigma_{R,i}$ une substitution de variables. L'équation (14) devient

$$v(w,U) = \left(\prod_{m=1}^{N+M} q_m^{C_{R,m}}\right)\prod_{i=1}^{k}\sigma_{R,i}(v(w_i,U_i)). \tag{16}$$

Pour obtenir la série génératrice $U(q_1,\ldots,q_{N+M})$ d'un langage $L_G(U)$, il suffit alors de sommer suivant toutes les U-dérivations :

$$\begin{aligned}
&U(q_1,\ldots,q_{N+M}) \\
&\quad= \sum_{w\in L_G(U)} v(w,U) \\
&\quad= \sum_{R\in\mathcal{R}(U)}\left(\prod_{m=1}^{N+M} q_m^{C_{R,m}}\right)\sum_{w_i\in L_G(d(R,i))}\prod_{i=1}^{a(R)}\sigma_{R,i}(v(w_i,d(R,i))).
\end{aligned}$$

Nous avons donc,

$$U(q_1,\ldots,q_{N+M}) = \sum_{R\in\mathcal{R}(U)}\left(\prod_{m=1}^{N+M} q_m^{C_{R,m}}\right)\prod_{i=1}^{a(R)}\sigma_{R,i}\left(d(R,i)\right). \tag{17}$$

Le système formé des équations (17) (pour $U\in N$), est un Q-analogue du système donné par (10).

Inversement, étant donné un Q-analogue du système algébrique (10), il est toujours possible de l'écrire sous la forme (17). Notons à ce propos que, à chaque fois qu'une règle R présente dans son membre droit deux symboles identiques (c'est-à-dire $d(R,i) = d(R,j)$ avec $i \neq j$), les substitutions $(\sigma_{R,i})_{R\in\mathcal{R},1\leq i\leq a(R)}$ ne sont pas forcément déterminées de manière unique. Les substitutions $\sigma_{R,i}$ et $\sigma_{R,j}$ peuvent être échangées sans modifier le système d'équations; bien sûr, ceci peut avoir des conséquences sur l'interprétation qui sera donnée des paramètres d'énumération.

Une fois déterminées les substitutions $\sigma_{R,i}$, les coefficients $C_{R,m,i,j}$ sont obtenus en utilisant la formule (15). Ceci permet de reconstituer les règles de calcul des paramètres $p_{N+1}, \ldots, p_{N+M}$, et donc la Q-grammaire dont les séries génératrices vérifient le système d'équations donné. □

Remarque. Le théorème 2.36 donne en quelque sorte une justification *a posteriori* de la définition que nous avons adoptée des paramètres Q-comptables : il s'agit des paramètres qui peuvent être paramètres d'énumération d'un système qui soit un Q-analogue du système algébrique fourni par la grammaire elle-même. Ainsi, il aurait été possible de définir les paramètres Q-comptables en autorisant les combinaisons linéaires à coefficients entiers (et non seulement positifs), mais les équations obtenues auraient impliqué des exposants négatifs, et donc, parfois, la perte des séries génératrices comme séries formelles.

2.4 Grammaires linéaires et croissance polynômiale

2.4.1 Grammaires et langages linéaires

Un cas très particulier de grammaires algébriques est celui des grammaires *linéaires.* Une grammaire est dite linéaire, si le second membre de chaque règle de dérivation fait apparaître au plus un symbole non terminal.

Les langages engendrés par des grammaires linéaires sont appelés langages linéaires. Un exemple est $L = \{a^n b^n, n \geq 0\}$, engendré par la grammaire

$$\begin{cases} L & \rightarrow & \epsilon \\ L & \rightarrow & aLb. \end{cases}$$

Les langages linéaires, bien que ne se limitant pas aux langages rationnels comme le montre l'exemple ci-dessus, ont des séries génératrices rationnelles.

2.4.2 Paramètres à croissance polynômiale

Reprenons la définition d'un terme de croissance de paramètre telle qu'elle est donnée en 2.2.1. Pour définir les paramètres Q-comptables, nous avons exigé que tous les termes de croissance s'expriment de manière linéaire en fonction d'autres paramètres.

Définition 2.37. Soit G une grammaire d'alphabet $X = \{x_1, \dots, x_k\}$.

Un paramètre p, défini sur G, est dit *à croissance polynômiale*, si, pour chaque règle de dérivation $R \in \mathcal{R}$, de la forme $R : U \to w_1 U' w_2$, il existe un polynôme P de k variables, à coefficients entiers positifs, tel que le terme de croissance de p pour le mot $w = w_1 w' w_2$, soit

$$\theta_p(w) = P\left(|w'|_{x_1}, \dots, |w'|_{x_k}\right).$$

En d'autres termes, un paramètre à croissance polynômiale est un paramètre dont le terme de croissance, au lieu d'être défini par une combinaison linéaire de paramètres Q-comptables, est défini par un polynôme des nombres d'occurrences des différentes lettres. Les polynômes doivent avoir des coefficients positifs afin que les termes de croissance soient assurés d'être positifs. Cette condition, un peu restrictive, est également une condition suffisante pour que la preuve du lemme suivant ne fasse pas intervenir de coefficients négatifs.

Lemme 2.38. *Soit G une grammaire linéaire d'alphabet $X = \{x_1, \dots, x_k\}$, et soit, pour chaque $i \le k$, p_k le paramètre "nombre d'occurrences de la lettre x_k". Soit également $p = p_1^{\alpha_1} \dots p_k^{\alpha_k}$ un monôme en les variables $p_1, \dots, p_k$, de degré total $\alpha_1 + \dots + \alpha_k$.*

Alors, p est un paramètre à croissance polynômiale, et les polynômes qui définissent ses termes de croissance sont tous de degré total strictement inférieur à $\alpha_1 + \dots + \alpha_k$.

Preuve. Lors de l'application de la règle $R : U \to w_1 U' w_2$, la croissance de chaque paramètre p_i est une constante : $p_i(w) - p_i(w') = |w_1 w_2|_{x_i} = a_i$. Le terme de croissance de p pour R est donc

$$\begin{aligned} \theta_p &= p(w) - p(w') \\ &= (p_1(w') + a_1)^{\alpha_1} \dots (p_k(w') - a_k)^{\alpha_k} - (p_1(w'))^{\alpha_1} \dots (p_k(w'))^{\alpha_k} \end{aligned}$$

En développant cette dernière expression, on trouve bien un polynôme des variables $p_1(w'), \dots, p_k(w')$, à coefficients entiers positifs; le seul terme de degré total $\alpha_1 + \dots + \alpha_k$ ayant été annulé, ce polynôme est de degré total strictement inférieur. □

Une conséquence de ce lemme est la proposition suivante :

Proposition 2.39. *Tout paramètre à croissance polynômiale dans une grammaire linéaire, est un paramètre Q-comptable de cette grammaire; le rang de ce paramètre Q-comptable est au plus supérieur de 1 au degré total maximal des polynômes qui définissent ses termes de croissance.*

Preuve. Nous procédons par récurrence sur le degré total maximal des polynômes P qui définissent les termes de croissance.

Si tous ces polynômes sont de degré total au plus 1, alors le paramètre p est déjà défini comme un paramètre Q-comptable de rang 2 (à moins que tous les polynômes ne soient constants, auquel cas p est de rang 1).

Supposons maintenant que la propriété soit vraie lorsque tous les polynômes qui définissent les termes de croissance sont de degré total inférieur ou égal à n. Alors, d'après le lemme précédent, tous les polynômes qui définissent les termes de croissance du paramètre p, peuvent eux-mêmes se définir par des termes de croissance qui sont des polynômes de degré total strictement inférieur à n. D'après l'hypothèse de récurrence, chacun des monômes de degré inférieur à n peut s'écrire comme paramètre Q-comptable de rang au plus n dans G; par conséquent, le terme de croissance de p s'écrit bien comme combinaison linéaire de paramètres Q-comptables de rang au plus n, et p est lui-même Q-comptable de rang au plus $n + 1$. □

La proposition 2.39 n'a pas d'équivalent pour les grammaires non linéaires. Pour s'en convaincre, il suffit d'examiner l'exemple suivant :

Exemple 2.40. Reprenons la grammaire classique des mots de Dyck :

$$\left\{ \begin{array}{lccl} R' : & D & \rightarrow & \epsilon, \\ R : & D & \rightarrow & aDbD. \end{array} \right.$$

Les arbres de dérivation de cette grammaire sont équivalents aux arbres binaires complets non étiquetés (toutes les feuilles sont étiquetées R', et tous les nœuds internes sont étiquetés R); comme, dans tout arbre binaire complet, le nombre de feuilles est supérieur de 1 au nombre de nœuds internes, on a pour tout mot de Dyck w,

$$p_R(w) = 1 + p_{R'}(w).$$

Par conséquent, tout paramètre Q-comptable dans G peut s'exprimer en n'utilisant que les paramètres élémentaires ne faisant intervenir que la règle R: ainsi, $p_{R^{(1)}R^{(2)}R'} = p_{R^{(1)}R^{(2)}R} + p_{R^{(1)}R}$.

Notons $l(w) = p_R(w)$ (le paramètre l est la demi-longueur du mot w), et soit p le paramètre défini par les règles de croissances polynômiales suivantes :

$$
\begin{aligned}
p(\epsilon) &= 0 \\
p(aw_1bw_2) &= p(w_1) + p(w_2) + l(w_1).l(w_2)
\end{aligned}
$$

Le paramètre p est la somme, sur tous les nœuds internes de l'arbre de dérivation, du produit des tailles (nombres de nœuds internes) des sous-arbres gauche et droit. Afin de prouver qu'un tel paramètre n'est pas Q-comptable, nous nous contentons d'établir un tableau des premières valeurs des différents paramètres :

w	R	$R^{(1)}R$	$R^{(2)}R$	$R^{(1)}R^{(1)}R$	$R^{(1)}R^{(2)}R$	$R^{(2)}R^{(1)}R$	$R^{(2)}R^{(2)}R$	$p(w)$
ab	1	0	0	0	0	0	0	0
$aabb$	2	1	0	0	0	0	0	0
$abab$	2	0	1	0	0	0	0	0
$aaabbb$	3	3	0	1	0	0	0	0
$aababb$	3	2	1	0	1	0	0	0
$abaabb$	3	1	2	0	0	1	0	0
$ababab$	3	0	3	0	0	0	1	0
$aabbab$	3	1	1	0	0	0	0	1

Il apparaît, au vu de ce tableau, que la dernière colonne ne saurait être combinaison linéaire des précédentes. Ajouter d'autres colonnes, correspondant à des paramètres Q-comptables de rangs plus élevés, n'y changerait rien : les lignes correspondant aux mots déjà inscrits seraient toutes nulles, puisque des paramètres de rangs supérieurs à 3 compteraient des chaînes de longueur supérieure à 3 (et l'arbre de dérivation de $aabbab$ n'est que de profondeur 3). Tout simplement, le paramètre p n'est pas Q-comptable dans la grammaire G.

2.5 Résolution de Q-équations

Il n'existe aucune méthode générale pour résoudre les équations données par une Q-grammaire dans le cas général. Des méthodes *ad hoc* pour résoudre une équation faisant intervenir la substitution $\sigma_{x\leftarrow xq}$ existent cependant dans la littérature; nous en donnons ici un exemple, que nous adaptons à des substitutions légèrement plus générales.

2.5.1 La méthode de Prellberg et Brak

Dans [69], Prellberg et Brak donnent une méthode pour résoudre, sous certaines conditions, une équation de la forme

$$F.\sigma(F) + aF + b\sigma(F) + c = 0, \tag{18}$$

où a, b et c appartiennent à $\mathbb{K}[[x,q]]$. En posant $G = F + b$, (18) devient

$$G.\sigma(G) + a'G + c' = 0, \tag{19}$$

avec $a' = a - \sigma(b)$ et $c' = c - a.b$. L'équation (19) peut à son tour être linéarisée en posant

$$G = \alpha\frac{\sigma(H)}{H}, \tag{20}$$

α (qui ne dépend plus de x) devant être choisi de manière à assurer la compatibilité pour $x = 0$ et $q = 0$, c'est-à-dire, α doit être solution de $\alpha^2 + \alpha.a'(0,0) + c'(0,0) = 0$. L'équation (19) devient alors,

$$\alpha^2\sigma^2(H) + \alpha a'\sigma(H) + c'H = 0. \tag{21}$$

Lorsque la substitution σ est $\sigma_{x\leftarrow xq}$, Prellberg et Brak obtiennent un développement de H suivant les puissances de x dans le cas où c' ne dépend pas de x, et où a' est de degré 1 en x. Dutour [36] donne une version plus générale, où c' peut également être de degré 1 en x.

L'idée est de réécrire (21) sous la forme plus générale

$$\sum_{k=0}^{N} (\alpha_k + x.\beta_k)\,\sigma^k(H) = 0, \tag{22}$$

où les coefficients α_k et β_k ne dépendent plus de x.

En développant la série inconnue H comme série en x : $H(x) = \sum_n x^n h_n(q)$, et en extrayant le coefficient de x^n dans l'équation (22), nous obtenons alors la récurrence suivante :

$$h_n.\sum_{k=0}^{N}\alpha_k q^{kn} + h_{n-1}.\sum_{k=0}^{N}\beta_k q^{k(n-1)} = 0. \tag{23}$$

Une fois posé

$$\begin{cases} \Lambda_\alpha(t) &= \sum_{k=0}^{N}\alpha_k t^k, \\ \Lambda_\beta(t) &= \sum_{k=0}^{N}\beta_k t^k, \end{cases}$$

l'équation (23) a pour solution (en prenant $h_0 = 1$)

$$h_n = (-1)^n \frac{\prod_{i=0}^{n-1} \Lambda_\beta(q^i)}{\prod_{i=0}^{n-1} \Lambda_\alpha(q^{i+1})}. \tag{24}$$

Notons ici que la condition de compatibilité sur α se traduit par $\Lambda_\alpha(1) = 0$, condition retrouvée en extrayant le terme constant (en x) de (22).

Par ailleurs, notons qu'il existe un résultat plus général permettant de résoudre des équations linéarisées similaires à (21), mais faisant également intervenir l'image de la série inconnue H par la spécialisation $x = 1$; voir à ce sujet [14] et une variante dans [7].

2.5.2 Une extension de la méthode

Il serait agréable de pouvoir étendre la méthode ci-dessus à des équations sur des séries à plus de deux variables et faisant intervenir des substitutions de variables quelconques, voire plusieurs substitutions différentes.

Attaquer le problème des substitutions quelconques semble largement hors de portée; toutefois, il est possible d'étendre la résolution de l'équation (22) au cas où la série H est une série à plusieurs variables $H \in \mathbb{K}[[x_1, \dots x_k; q_1, \dots q_\ell]]$, à condition que la substitution σ ne corresponde qu'au calcul de paramètres de rang 2, c'est-à-dire qu'elle soit de la forme

$$\sigma = \sigma_{(x_i \leftarrow x_i q_1^{a_{i,1}} \dots q_\ell^{a_{i,\ell}})}.$$

Cette substitution est entièrement déterminée par la matrice $\mathbf{A} = (a_{i,j})_{1 \leq i \leq k, 1 \leq j \leq \ell}$ (qui n'est qu'une sous-matrice particulière de la matrice M_σ définie au paragraphe 2.3.1).

Nous traitons complètement le cas $k = \ell = 2$; le cas le plus général n'est pas plus difficile, mais demande des notations plus lourdes.

L'équation à résoudre se présente alors sous la forme

$$\sum_{i=0}^{N} (\alpha_i + x_1 \beta_{1,i} + x_2 \beta_{2,i}) \sigma^i(H) = 0. \tag{25}$$

Nous notons $\mathbf{x}$ le vecteur-ligne (x_1, x_2), et, par convention, $\mathbf{x}^\mathbf{n} = x_1^{n_1} x_2^{n_2}$. Lorsque $\mathbf{x}$ est un tel vecteur-ligne, $\mathbf{x}'$ désigne le vecteur-colonne transposé.

En posant

$$\left\{ \begin{array}{rcl} H(\mathbf{x}, \mathbf{q}) & = & \sum_{n_1, n_2 \geq 0} \mathbf{x}^\mathbf{n} h_\mathbf{n}(\mathbf{q}), \\ \Lambda_\alpha(t) & = & \sum_{0 \leq i \leq N} \alpha_i t^i, \\ \Lambda_{\beta_j}(t) & = & \sum_{0 \leq i \leq N} \beta_{j,i} t^i, \end{array} \right.$$

nous avons

$$\sigma^i(H) = \sum_{n_1,n_2\geq 0} \mathbf{x}^{\mathbf{n}}\mathbf{q}^{i\mathbf{A}.\mathbf{n}'}h_{\mathbf{n}}(\mathbf{q}). \tag{26}$$

Ainsi, en extrayant le coefficient de $\mathbf{x}^{\mathbf{n}}$ de (25), nous obtenons la relation de récurrence

$$\sum_{i=0}^{N}\left(\alpha_i\mathbf{q}^{i\mathbf{A}.\mathbf{n}'}h_{\mathbf{n}} + \beta_{1,i}\mathbf{q}^{i\mathbf{A}.(\mathbf{n}-(1,0))'}h_{\mathbf{n}-(1,0)} + \beta_{2,i}\mathbf{q}^{i\mathbf{A}.(\mathbf{n}-(0,1))'}h_{\mathbf{n}-(0,1)}\right) = 0. \tag{27}$$

Dans la relation ci-dessus, nous avons par convention $h_{n_1,-1} = h_{-1,n_2} = 0$.

Nous pouvons donc écrire h_{n_1,n_2} en fonction de h_{n_1-1,n_2} et de h_{n_1,n_2-1} :

$$h_{n_1,n_2} = -\frac{\Lambda_{\beta_1}\left(\mathbf{q}^{\mathbf{A}.(n_1-1,n_2)'}\right)h_{n_1-1,n_2} + \Lambda_{\beta_2}\left(\mathbf{q}^{\mathbf{A}.(n_1,n_2-1)'}\right)h_{n_1,n_2-1}}{\Lambda_\alpha\left(\mathbf{q}^{\mathbf{A}.\mathbf{n}'}\right)h_{\mathbf{n}}}. \tag{28}$$

Nous pouvons ainsi interpréter les coefficients $h_{\mathbf{n}}$ en termes de *chemins dirigés*:

Définition 2.41. Soient A et B deux points du plan discret $\mathbb{N}\times\mathbb{N}$. Un *chemin dirigé* de A à B est un chemin discret $S = (s_i)_{0\leq i\leq n}$ du plan $\mathbb{N}^2$, avec $s_0 = A$, $s_n = B$, et ne faisant que des pas Nord ($s_{i+1} - s_i = (0,1)$) et Est ($s_{i+1} - s_i = (1,0)$).

Nous notons $P_{A,B}$ l'ensemble des chemins dirigés de A à B.

Pour chaque chemin dirigé, nous définissons une valuation en utilisant les notations introduites précédemment :

Définition 2.42. Soit $S = (s_0,\ldots,s_n)$ un chemin dirigé de $A = s_0$ à $B = s_n$. Notons, pour $0\leq i\leq n$, (n_i, m_i) les coordonnées de s_i.

La *valuation* du pas s_is_{i+1} est la série

$$v(s_i, s_{i+1}) = -\frac{\Lambda_{\beta_j}\left(\mathbf{q}^{\mathbf{A}.(n_i,m_i)'}\right)}{\Lambda_{\alpha\left(\mathbf{q}^{\mathbf{A}.(n_{i+1},m_{i+1})'}\right)}}, \tag{29}$$

où $j = 1$ si le pas s_is_{i+1} est un pas Est, et $j = 2$ si s_is_{i+1} est un pas Nord.

La *valuation* du *chemin* S est

$$v(S) = \prod_{i=0}^{n-1} v(s_i, s_{i+1}). \tag{30}$$

La récurrence (28) se traduit alors immédiatement de la manière suivante :

Proposition 2.43. *Le coefficient h_{n_1,n_2} de la série H est*

$$h_{n_1,n_2} = \sum_{S\in P_{O,(n_1,n_2)}} v(S). \tag{31}$$

Remarque. En définissant une seconde valuation $v'(S) = v(S)\mathbf{x}^{\mathbf{n}}$, où n_1 et n_2 sont les coordonnées du dernier point de S, c'est la série H toute entière qui s'exprime comme somme de valuations de chemins :

$$H(\mathbf{x}, \mathbf{q}) = \sum_{S \in P_O} v'(S), \tag{32}$$

où P_O désigne l'ensemble de tous les chemins dirigés d'origine O.

Il semble difficile d'étendre ce genre de méthodes à des équations faisant intervenir des paramètres de rang supérieur à 2. Ainsi, dans le cas des chemins de Dyck énumérés suivant les paramètres longueur (comptée par x), aire géométrique (comptée par q) et moment d'inertie (compté par r; voir exemple 2.7), l'équation obtenue est :

$$D(x, q, r) = 1 + x^2 qr D(x, q, r) D(xqr, qr, r).$$

La linéarisation de l'équation est toujours possible en posant

$$D(x, q, r) = F(xqr, qr, r)/F(x, q, r),$$

et l'équation obtenue est alors :

$$x^2 qr F(xq^2r^3, qr^2, r) - F(xqr, qr, r) + F(x, q, r) = 0.$$

Toutefois, bien que l'équation soit en quelque sorte linéaire, développer F suivant les puissances de x ne fait pas disparaître complètement les substitutions de variables. Si nous posons

$$F(x, q, r) = \sum_{n=0}^{\infty} x^{2n} f_n(q, r),$$

l'équation se traduit, pour $n \geq 1$, par une relation de récurrence faisant intervenir la substitution de variables $\sigma_{q \leftarrow qr}$:

$$q^{4n-3} r^{6n-5} f_{n-1}(qr^2, r) - q^{2n} r^{2n} f_n(qr, r) + f_n(q, r) = 0.$$

La condition initiale $f_0(q, r) = 1$ permet de résoudre pour $n = 1$, et l'on obtient alors

$$f_1(q, r) = -\sum_{k=0}^{\infty} q^{2k+1} r^{(k+1)^2};$$

toutefois, pousser le calcul ne serait-ce qu'un cran plus loin amène à développer suivant les puissances de q, et la récurrence double obtenue ne semble pas pouvoir se résoudre de manière générale.

Chapitre 3

Changements de grammaires

Dans ce chapitre, nous étudions dans quelle mesure les paramètres Q-comptables associés à un langage dépendent de la grammaire utilisée pour l'engendrer. Nous montrons également qu'il est possible de faire subir un certain nombre de transformations classiques à une grammaire, sans perte de paramètres Q-comptables.

Nous avons vu que certains paramètres, comme la longueur des mots et ses variantes (nombre d'occurrences d'une lettre donnée ou de certaines des lettres de l'alphabet) sont toujours des paramètres Q-comptables, indépendamment de la grammaire.

Nous montrons ici que cette situation n'est pas générale. Ainsi, un paramètre peut être Q-comptable dans une grammaire, mais pas dans une autre qui engendre le même langage. Nous commençons par un exemple de deux paramètres qui sont Q-comptables dans deux grammaires différentes, sans qu'il existe de grammaire dans laquelle ils soient tous deux Q-comptables.

3.1 Un exemple: nombre de passages au niveau final ou initial

Considérons le langage $\{a, b\}^*$, formé de tous les mots sur les deux lettres a et b.

Définition 3.1. Un mot $w \in \{a, b\}^*$ est dit *équilibré* si $|w|_a = |w|_b$.

Les mots équilibrés sont aussi appelés mots de Dyck bilatères. Soient p et p' les paramètres définis sur $\{a, b\}^*$ par : $p(w)$ est le nombre de facteurs gauches équilibrés de w, et $p'(w)$ est le nombre de ses facteurs droits équilibrés.

Par symétrie, il est clair que p et p' ont la même distribution; par ailleurs, si p est Q-comptable dans une grammaire G, p' est forcément Q-comptable dans la grammaire G'

obtenue en remplaçant par leur image miroir tous les membres droits de toutes les règles de G.

A chaque mot, associons un chemin du plan discret en transformant chaque a en un pas Nord-Est et chaque b en un pas Sud-Est (représentation horizontale classique d'un mot de Dyck par un chemin de Dyck). Le paramètre p devient le nombre de sommets du chemin dont l'ordonnée est nulle (passages au niveau initial), et p', le nombre de sommets dont l'ordonnée est celle du dernier sommet (passages au niveau final) – voir figure 3.1. Il apparaît intuitivement qu'il est possible de "marquer" les passages au niveau initial en lisant le mot "de gauche à droite", ou les passages au niveau final en le lisant "de droite à gauche"; nous allons montrer de manière précise qu'il est impossible de marquer les deux types de passages.

$w = aabaabbbbaabbbaaaa$
$p(w) = 5$
$p'(w) = 4$

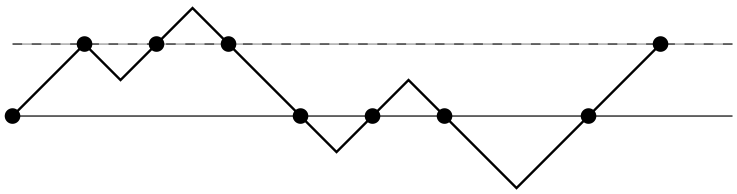


FIG. 3.1: *Passages aux niveaux initial et final*

Nous allons montrer que p peut être Q-comptable, mais que p et p' ne peuvent pas être Q-comptables dans la même grammaire. Puisque la somme de deux paramètres Q-comptable est Q-comptable, il nous suffit de montrer que $p + p'$ n'est Q-comptable dans aucune grammaire.

Soit $G = (\{a, b\}, \{T, P, N, F, G\}, \mathcal{R}, T)$ la grammaire définie par ses règles de dérivation :

$$\begin{array}{rcl} T & \to & \epsilon + aPbT + bNaT + F + G \\ P & \to & \epsilon + aPbP \\ N & \to & \epsilon + bNaN \\ F & \to & aP + aPF \\ G & \to & bN + bNG \end{array}$$

Cette grammaire engendre le langage $\{a, b\}^*$, et le paramètre p compte les T-dérivations; il est donc Q-comptable de rang 1 dans cette grammaire.

Supposons que, dans une grammaire G', le paramètre $p+p'$ soit Q-comptable. Il est alors de rang minimal 1. Nous montrerons plus tard, et nous l'admettrons pour le moment, que lorsqu'un paramètre est de rang minimal k, il existe toujours une grammaire engendrant le même langage et dans laquelle le même paramètre est de rang k. Dans le cas du paramètre $p+p'$, la proposition 2.16 implique alors que le paramètre $p+p'$ peut être représenté par une lettre c supplémentaire. Afin de montrer que $p+p'$ ne peut être Q-comptable, il nous suffit donc de montrer le lemme suivant :

Lemme 3.2. *Il n'existe pas de langage algébrique non ambigu $L \subset \{a,b,c\}^*$ tel que*

- *la projection φ de $\{a,b,c\}^*$ sur $\{a,b\}^*$ établit une bijection de L sur $\{a,b\}^*$;*
- *pour tout mot $w \in L$, $|w|_c = (p+p')\,(\varphi(w))$.*

Preuve. Supposons qu'un tel langage algébrique L existe. Pour chaque mot $w \in L$, nous noterons $w' = \varphi(w)$ le mot obtenu en effaçant les c de w. Lorsque w' est défini, w désigne alors son antécédent par φ dans L.

Nous utilisons le lemme d'Ogden sur les langages algébriques (voir [65, 56, 1, 11]). Soit donc $w'_1 = a^{2N}b^Na^{2N}$, avec N assez grand pour appliquer le lemme au mot w_1 correspondant. Nous avons alors une factorisation $w_1 = \alpha u \beta v \gamma$ telle que u' (ou v', ce qui est équivalent par image miroir) soit de la forme $u' = b^k$, et tel que, pour tout $n \geq 0$, $w_n = \alpha u^{n+1}\beta v^{n+1}\gamma \in L$.

Il est facile de voir que $(p+p')(w'_1) = 2$; par conséquent, $|w_n|_c = 2 + n|uv|_c$. Sachant que $|uv|_c$ vaut 0, 1 ou 2, la suite $(|w_n|_c)_{n\geq 0}$, soit est constante, soit tend vers $+\infty$. Nous allons montrer que $(p+p')(w'_n)$ ne peut avoir ce comportement.

Les différents cas possibles pour α', u', β', v' et γ', également représentés figure 3.2, sont les suivants :

a. $\alpha' = a^{2N}b^{k_1}$, $u' = b^{k_2}$, $\beta' = b^{k_3}$, $v' = b^{k_4}$, $\gamma' = b^{k_5}a^{2N}$ avec $k_1+k_2+k_3+k_4+k_5 = N$ et $k_2+k_4 > 0$. Dans ce cas, $w'_n = a^{2N}b^{N+n(k_2+k_4)}a^{2N}$ et donc, pour n assez grand, $(p+p')(w'_n) = 4$.

b. $\alpha' = a^{2N}b^{k_1}$, $u' = b^{k_2}$, $\beta' = b^{k_3}a^{k_4}$, $v' = a^{k_5}$, $\gamma' = a^{k_6}$ avec $k_1+k_2+k_3 = N$, $k_4+k_5+k_6 = 2N$, et $k_2 > 0$. Dans ce cas, $w'_n = a^{2N}b^{N+nk_2}a^{2N+nk_5}$. Pour n assez grand, $(p+p')(w'_n)$ vaut 4 (ou 6 si $k_2 = k_5$).

c. $\alpha' = a^{2N}b^{k_1}$, $u' = b^{k_2}$, $\beta' = b^{k_3}$, $v' = b^{k_4}a^{k_5}$, $\gamma' = a^{k_6}$ avec $k_1+k_2+k_3+k_4 = N$, $k_5+k_6 = 2N$, et $k_2+k_4 > 0$. Dans ce cas, $w'_n = a^{2N}b^{N+nk_2}(a^{k_5}b^{k_4})^{n-1}a^{2N}$. Si $k_5 \geq k_4$, $(p+p')(w'_n) = 4$ lorsque n est assez grand; si $k_4 > k_5$, $(p+p')(w'_n)$ est ultimement constante avec une valeur plus grande que 4 (voir figure 3.2).

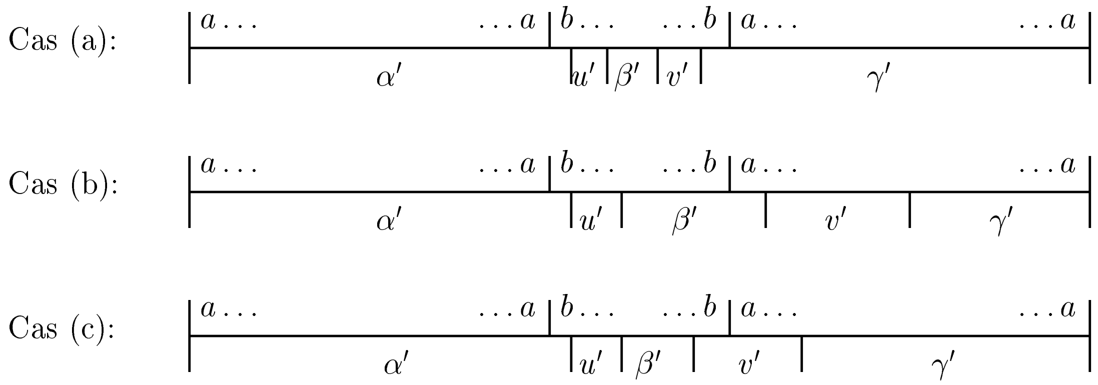


FIG. 3.2: *Différents cas*

Dans tous les cas, le comportement de $(p + p')(w'_n)$ est différent de celui de $|w_n|_c$, ce qui donne une contradiction et termine donc la preuve du lemme. $\square$

Une conséquence de ce lemme (et de la propriété 2.16) est qu'il n'existe pas de grammaire non ambiguë engendrant $\{a, b\}^*$ dans laquelle $p + p'$ soit un paramètre Q-comptable. Par conséquent, puisque la somme de deux paramètres Q-comptables dans la même grammaire est Q-comptable dans cette grammaire, les paramètres p et p' ne peuvent être Q-comptables dans la même grammaire.

3.2 Grammaire plus fine qu'une autre

Comme le prouve l'exemple précédent, un paramètre peut être Q-comptable dans une grammaire, mais pas dans une autre qui engendre le même langage.

Définition 3.3. Soient G et G' deux grammaires engendrant le même langage L. G' est *plus fine* que G si, pour tout paramètre p, Q-comptable dans G, il existe un paramètre p', Q-comptable dans G', tel que, pour tout mot $w \in L$, $p(w) = p'(w)$.

Définition 3.4. Deux grammaires G et G' sont *Q-équivalentes* si G est plus fine que G' et réciproquement.

Remarque. Dire qu'une grammaire est plus fine qu'une autre revient à dire que tout paramètre Q-comptable de la seconde coïncide, sur le langage engendré, avec un paramètre Q-comptable de la première. Dire que tout paramètre Q-comptable de l'une est Q-comptable dans l'autre serait un abus de langage, les paramètres étant a priori définis sur tous les langages auxiliaires. Une telle définition serait trop restrictive et perdrait son intérêt pratique, car cela exigerait que les langages auxiliaires des deux grammaires soient les mêmes.

Remarque. Dans le cadre des grammaires d'objets, Dutour [36] donne une définition de l'isomorphisme de grammaires d'objets qui est basée sur l'identité de systèmes caractéristiques de polynômes. Notre définition de la Q-équivalence de grammaires est plus restrictive, et les résultats sur les grammaires d'objets ne semblent pas pouvoir se transposer dans notre cadre. En effet, notre définition de la Q-équivalence de grammaires impose que les paramètres soient conservés pour chaque mot, et non pas pour le langage engendré pris dans son ensemble. Nous verrons sur un exemple (voir section 3.3) que la Q-équivalence de deux grammaires ne se réduit pas à l'isomorphisme de deux grammaires d'objets.

Il est clair qu'une condition nécessaire et suffisante pour que G' soit plus fine que G est que tout paramètre *élémentaire* de G, lorsqu'on ne le considère que sur les mots du langage engendré par G et G', soit Q-comptable dans G'.

L'exemple précédent nous montre qu'il est possible d'avoir deux grammaires engendrant le même langage, sans qu'il existe une grammaire plus fine que chacune d'elles.

Nous donnons maintenant quelques lemmes qui permettent, étant donnée une grammaire, d'en construire d'autres plus fines.

Etant donnée une grammaire G, nous allons montrer qu'il est possible de construire :

- une grammaire plus fine, de forme 1-2;
- une grammaire plus fine, où l'on a itéré l'une des règles de dérivation;
- une grammaire plus fine, où chaque sommet des arbres de dérivation est également marqué suivant la présence ou non, dans la branche qui le relie à la racine, de certaines étiquettes (marquage supérieur);
- une grammaire plus fine, où chaque sommet est marqué suivant l'existence, dans l'un de ses sous-arbres, de certaines étiquettes (marquage inférieur);
- une grammaire plus fine, où le rang formel d'un ou de chaque paramètre de G est égal à son rang minimal.

3.2.1 Passage en forme 1-2

Une grammaire est dite de forme 1-2 si toutes ses règles de dérivation sont d'arité 0, 1 ou 2. Le fait que tout langage algébrique non ambigu peut être engendré par une grammaire de forme 1-2 est bien connu. Nous montrons essentiellement que le passage en forme 1-2 se fait en transformant la grammaire en une grammaire Q-équivalente.

L'outil de base est la transformation décrite dans la proposition suivante, qui diminue strictement le nombre de règles de dérivation d'arité supérieure à 2.

Proposition 3.5. *Soit G une grammaire engendrant un langage L, et soit $R : U \to u_0 U_1 u_1 \dots U_k u_k$ une règle de dérivation de G, d'arité $k \geq 3$.*

Soit G' la grammaire obtenue en ajoutant $k-2$ symboles $V_1, \dots, V_{k-2}$, et en remplaçant la règle R par

$$\begin{array}{rrcl} (R'_0): & U & \to & u_0 U_1 u_1 V_1, \\ (R'_i): & V_i & \to & U_{i+1} u_{i+1} V_{i+1} \quad (1 \leq i < k-2), \\ (R'_{k-2}): & V_{k-2} & \to & U_{k-1} u_{k-1} U_k u_k. \end{array}$$

La grammaire G' engendre également L, et G et G' sont Q-équivalentes.

Preuve. Le passage de G à G' est une opération classique utilisée pour donner une grammaire en forme 1-2. Tout arbre de dérivation de G peut facilement être transformé en l'arbre de dérivation correspondant de G' en remplaçant les sommets étiquetés R par des sous-arbres étiquetés R', tout en rattachant les sous-arbres issus de ces sommets (voir figure 3.3).

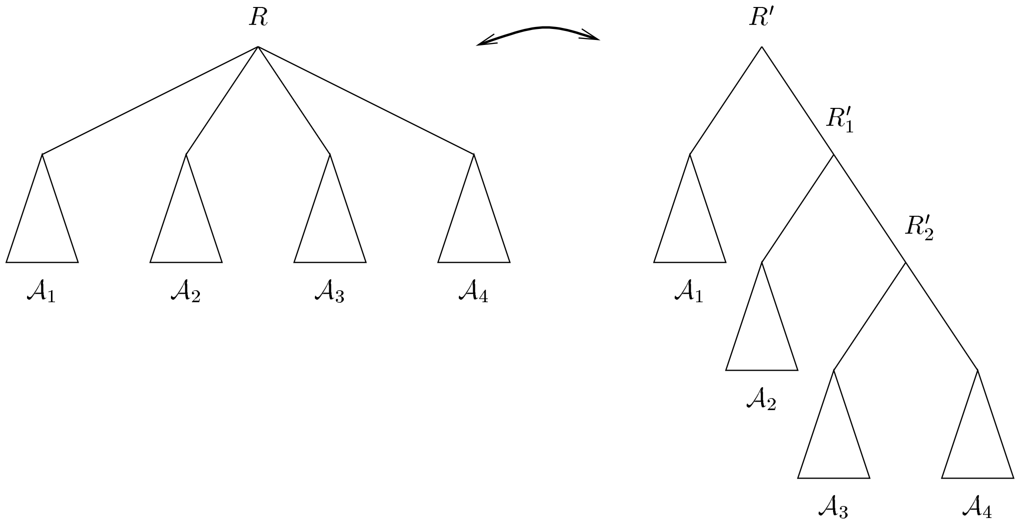


FIG. 3.3: *Passage en forme* 1-2

Cette transformation n'a aucune influence sur les sommets qui ne sont pas étiquetés R, donc les paramètres élémentaires de G qui ne font pas intervenir la règle R ne sont pas modifiés.

Afin de montrer que les grammaires G et G' sont bien Q-équivalentes, nous indiquons comment "traduire" dans G' le nom d'un paramètre élémentaire de G, et, inversement, comment "traduire" dans G le nom d'un paramètre élémentaire de G'.

Le passage de G à G' est extrêmement simple, puisqu'il suffit d'appliquer sur un nom de paramètre un morphisme alphabétique :

$$\begin{array}{rcll} \varphi(R) & = & R'_0, & \\ \varphi(R^{(i)}) & = & R'^{(1)}_{i-1} & (1 \leq i < k), \\ \varphi(R^{(k)}) & = & R'^{(2)}_{k-1}, & \\ \varphi(r) & = & r & (\text{pour tout autre } r \in \mathcal{R}_p \cup \mathcal{R}). \end{array}$$

Quel que soit le mot $W \in \mathcal{R}_p^* \mathcal{R}$, et quel que soit l'arbre de dérivation $\mathcal{A}$ de G, les chaînes de type W de $\mathcal{A}$ sont en bijection avec les chaînes de type $\varphi(W)$ de l'arbre de dérivation de G' correspondant à $\mathcal{A}$. Par conséquent, G' est plus fine que G.

Inversement, la transformation est légèrement plus compliquée, car un paramètre élémentaire de G' peut avoir dans G un équivalent qui n'est pas un paramètre élémentaire.

Nous noterons $\mathcal{R}'_0 = \varphi(\mathcal{R}_p \cup \mathcal{R})$, et $\mathcal{R}'_1 = (\mathcal{R}'_p \cup \mathcal{R}') - \mathcal{R}'_0$. L'ensemble $\mathcal{R}'_1$ est composé des dérivations pointées $R'^{(2)}_i$, pour $0 \leq i \leq k-1$, et des dérivations R'_i, pour $1 \leq i \leq k-2$.

Soit W' un nom de paramètre élémentaire dans G'. Si W' ne fait intervenir aucune des lettres de $\mathcal{R}'_1$, W' fait déjà partie des noms de paramètres qui correspondent à des paramètres élémentaires de G. Supposons donc que W' contient de telles lettres.

Afin de "traduire" un nom de paramètre élémentaire de G' vers G, nous définissons $\overline{\varphi}$ de la manière suivante :

- $\overline{\varphi}(r) = \varphi^{-1}(r)$ si $r \in \mathcal{R}'_0$;
- $\overline{\varphi}(R'_i) = R$;
- $\overline{\varphi}(R'^{(2)}_i) = R^{(i+2)} + \cdots + R^{(k)}$ (pour $0 \leq i < k-2$).

Soit, dans un arbre de dérivation $\mathcal{A}'$ de G', une chaîne S' de sommets, de type W'. Dans l'arbre de dérivation $\mathcal{A}$ de G qui correspond à $\mathcal{A}'$, cette chaîne devient une chaîne S, qui peut être plus courte que S', car plusieurs sommets de S' peuvent correspondre à un seul et même sommet de S – dans le second arbre de la figure 3.3, les sommets étiquetés R'_0, R'_1 et R'_2 proviennent du même sommet étiqueté R, et donc, si plusieurs d'entre eux font partie d'une chaîne de sommets de $\mathcal{A}'$, la chaîne de sommets correspondante dans $\mathcal{A}$ sera plus courte. Toutefois, si S est de même longueur que S', le type de S est l'un des mots de $\overline{\varphi}(W')$.

Afin de rendre compte de tous les cas possibles, nous dirons qu'une lettre de W' est *non maximale* si elle est de la forme $R_i'^{(2)}$ et est suivie, dans W', par une lettre de la forme $R_j'^{(1)}$, $R_j'^{(2)}$ ou R_j', avec $i < j$. Ainsi, une lettre d'un nom de paramètre W' est non maximale s'il est possible que, dans une chaîne de type W', les sommets correspondant à cette lettre et à la suivante proviennent d'un même sommet de l'arbre $\mathcal{A}$.

La figure 3.4 montre un exemple avec $k = 3$. L'arbre $\mathcal{A}'$ (représenté en haut) possède 7 chaînes de type $W' = R_0'^{(2)} R_1'^{(1)} r$, dont 3 seulement sont représentées. Dans le nom W', la lettre $R_0'^{(2)}$ est non maximale : cela se traduit par le fait que, dans l'arbre $\mathcal{A}$, les deux premiers sommets d'une chaîne de type W' peuvent n'en former qu'un. Dans ce cas, le type de la chaîne de $\mathcal{A}$ est $R^{(2)}r$. Sinon, le type est soit $R^{(2)}R^{(2)}r$, soit $R^{(3)}R^{(2)}r$.

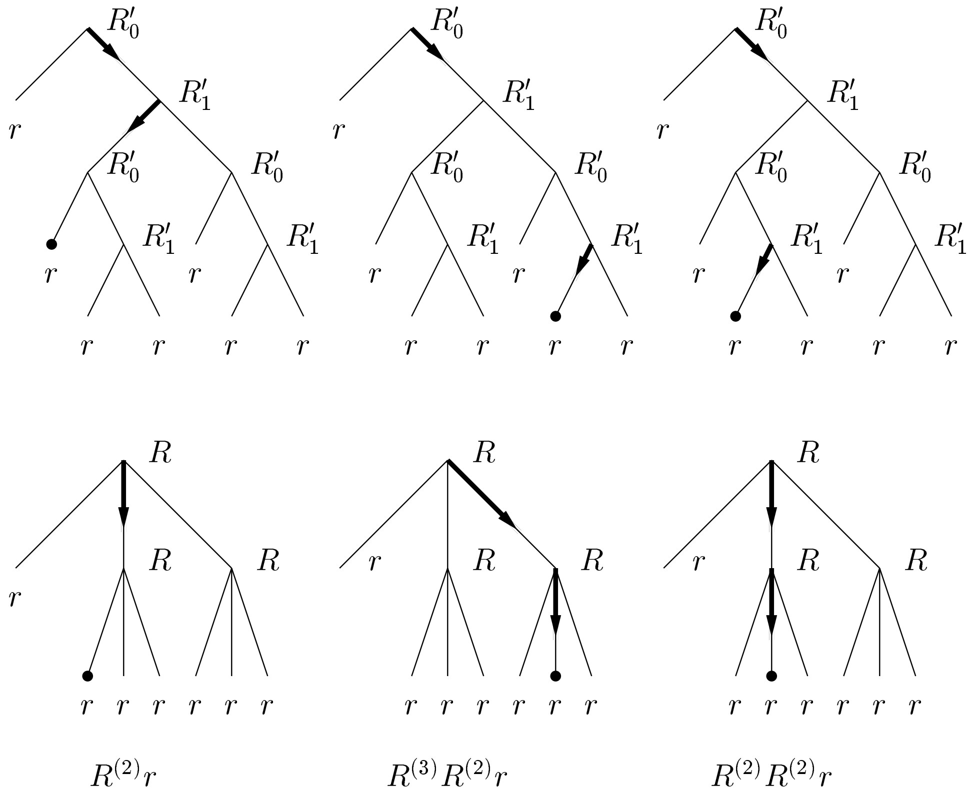


FIG. 3.4: *Transformation de chaînes de type* $R_0'^{(2)} R_1'^{(1)} r$

Au total, pour obtenir le nom du paramètre Q-comptable correspondant, dans G, au paramètre $p_{W'}$ de G', il faut donc traduire chaque lettre maximale r de W' par $\overline{\varphi}(r)$, et chaque lettre non maximale r par $\epsilon + \overline{\varphi}(r)$.

□

La traduction d'un nom de paramètre de G' en nom de paramètre de G peut également être définie de la manière suivante : un ordre partiel est défini sur $\mathcal{R}'_p \cup \mathcal{R}'$ par

$$\begin{cases} R_i'^{(2)} < R'_j & \Longleftrightarrow \quad i \leq j, \\ R_i'^{(2)} < R_j'^{(k)} & \Longleftrightarrow \quad i < j \ et\ 1 \leq k \leq 2. \end{cases}$$

Notons que, dans un nom de paramètre W' de G', une lettre est non maximale au sens de la preuve précédente lorsqu'elle est inférieure (strictement) à celle qui la suit dans W.

La fonction τ qui à tout nom de paramètre W' de G' associe le nom $\tau(W')$ de ce même paramètre dans G, peut alors être définie récursivement ainsi :

$$\begin{cases} \tau(r') & = \quad \overline{\varphi}(r') & \text{si } r' \in \mathcal{R}', \\ \tau(r'W'_1W') & = \quad \left(\overline{\varphi}(r') + 1_{r'<W'_1}\right)\tau(W'_1W') & \text{si } r' \in \mathcal{R}'_p, W'_1 \in \mathcal{R}'_p \cup \mathcal{R}'. \end{cases}$$

3.2.2 Itération d'une règle

Une autre opération permettant de changer de grammaire sans changer le langage engendré, est l'*itération* qui consiste à remplacer, dans le membre droit d'une règle, l'un des symboles par tous les membres droits des règles qui réécrivent ce symbole.

Proposition 3.6. *Soit G une grammaire, R une de ses règles d'arité $n \geq 1$, et $k \leq n$. Soit G' la grammaire engendrant le même langage, obtenue en retirant la règle R et en ajoutant, pour chaque $d(R,k)$-dérivation R', la règle obtenue en remplaçant, dans le membre droit de R, le k-ème symbole par le membre droit de R'.*

Dans ces conditions, la grammaire G' est plus fine que G; plus précisément, tout paramètre Q-comptable dans G est également Q-comptable dans G'.

Remarque. Une preuve simple de la proposition 3.6 consiste à remarquer que n'importe quel Q-analogue du système d'équations de la grammaire G peut être réécrit sous forme de Q-analogue du système d'équations de la grammaire G', sans évidemment changer les solutions. Le théorème 2.36 permet alors de conclure. Toutefois, la preuve que nous donnons ci-dessous présente l'avantage de fournir explicitement le "dictionnaire" qui traduit les noms de paramètres de G vers G'.

Preuve. Soit $R : U \to u_0U_1 \dots U_nu_n$ la règle à itérer, et soient $R_1, \dots, R_m$ les U_k-dérivations. Les m règles qui, dans G', remplacent la règle R, sont

$$R'_i : U \to u_0U_1 \dots u_{k-1}.d(R_i).u_{k+1} \dots U_nu_n \quad (1 \leq i \leq m).$$

Nous commençons par le cas le plus simple : celui où le k-ème symbole du membre droit de R est distinct de U, ce qui revient à dire que chaque R_i est distinct de R.

Dans ce cas, le passage d'un arbre de dérivation $\mathcal{A}$ de G à l'arbre de dérivation correspondant $\mathcal{A}'$ de G' est simple : chaque sommet étiqueté R et son k-ème fils, étiqueté R_i, sont remplacés par un seul sommet étiqueté R'_i, les autres fils de ces deux sommets étant attachés dans l'ordre préfixe comme indiqué figure 3.5.

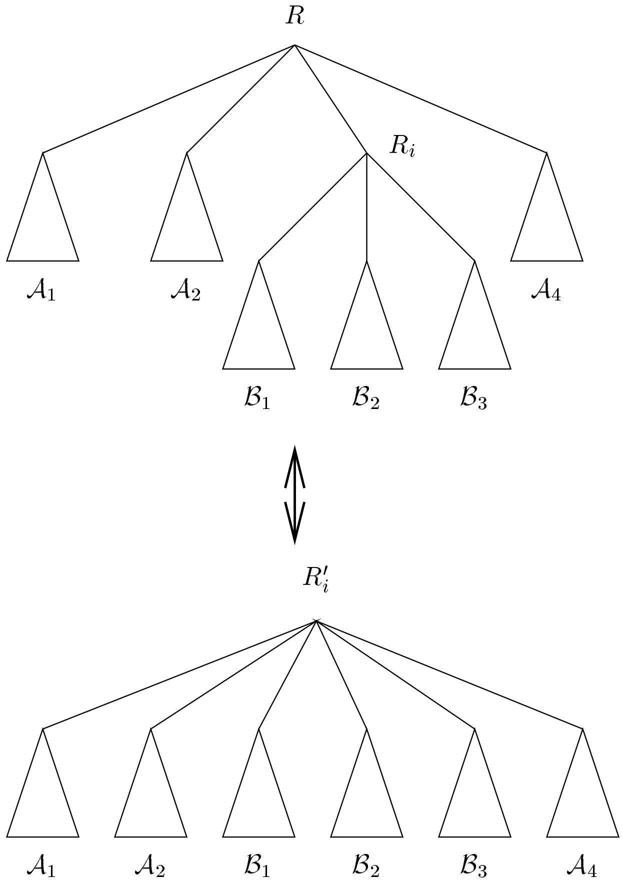


FIG. 3.5: *Itération de la règle R*

Si S est une chaîne de type W de l'arbre $\mathcal{A}$, elle devient dans $\mathcal{A}'$ une chaîne S'. Chaque sommet s d'étiquette R_i devient un sommet d'étiquette R'_i, et chaque sommet s d'étiquette R devient un sommet d'étiquette R'_i (l'indice i dépendant de l'étiquette du k-ème fils de s).

Chaque fois que deux lettres consécutives de W sont $R^{(k)}$ et $R_i^{(j)}$ (ou $R^{(k)}$ et R_i, en fin du mot W), il est possible que les deux sommets correspondants dans S deviennent, dans S', un seul et même sommet; dans ce cas, la chaîne S' est plus courte que S.

Comme dans la preuve de la proposition 3.5, nous dirons donc qu'une lettre de W est *non maximale* si cette lettre est $R^{(k)}$ et si la lettre suivante est une lettre R_i ou $R_i^{(j)}$.

La traduction d'un nom de paramètre élémentaire de G en son nom dans G' est alors essentiellement assurée par φ, définie ainsi :

$$\begin{array}{rcll}
\varphi(R) & = & R'_1 + \cdots + R'_m; & \\
\varphi(R_i) & = & R'_i & (1 \leq i \leq m); \\
\varphi(R^{(k)}) & = & \sum_{i=1}^{m} \sum_{j=1}^{\alpha(R_i)} R_i'^{(k-1+j)}; & \\
\varphi(R_i^{(j)}) & = & R_i'^{(k-1+j)} & (1 \leq i \leq m, 1 \leq j \leq \alpha(R_i)); \\
\varphi(r) & = & r & (\text{pour tout autre } r \in \mathcal{R}_p \cup \mathcal{R}).
\end{array}$$

Pour obtenir le nom, dans G', du paramètre p_W, il suffit de traduire chaque lettre maximale r de W par $\varphi(r)$, et chaque lettre non maximale $R^{(k)}$ par $\epsilon + \varphi(R^{(k)})$.

La situation est plus complexe à décrire lorsque $R = R_{i_0}$, car il est possible d'avoir, dans une même branche d'un arbre de dérivation $\mathcal{A}$, une succession de sommets étiquetés R, chacun étant le k-ème fils du précédent. Dans l'arbre de dérivation $\mathcal{A}'$, les premier et deuxième sommets se contractent en un seul, ainsi que le troisième et le quatrième, etc. Autrement dit, en conservant la même définition d'une lettre non maximale du mot W, dans tout facteur de W formé de lettres non maximales suivies d'une lettre maximale (de la forme $R^{(k)} \ldots R^{(k)} R_i^{(j)}$), chaque lettre peut être effacée à condition que la suivante ne le soit pas, et soit remplacée par son image par φ. Notons qu'il convient de revoir la définition donnée de $\varphi(R)$ (qui cumule les définitions précédentes de $\varphi(R)$ et de $\varphi(R_{i_0})$) et $\varphi(R^{(k)})$ (qui cumule les définitions précédentes de $\varphi(R^{(k)})$ et $\varphi(R_{i_0}^{(k)})$), ce qui donne :

$$\begin{array}{rcl}
\varphi(R) & = & R'_{i_0} + \sum_{i=1}^{m} R'_i \\
\varphi(R^{(k)}) & = & R_{i_0}'^{(2k-1)} + \sum_{i=1}^{m} \sum_{j=1}^{\alpha(R_i)} R_i'^{(k-1+j)}
\end{array}$$

□

Similairement au passage en forme 1-2, nous pouvons définir un ordre partiel sur $\mathcal{R}_p \cup \mathcal{R}$, correspondant à la définition d'une lettre non maximale, par

$$R^{(k)} \leq \begin{cases} R_i & (1 \leq i \leq m) \\ R_i^{(j)} & (1 \leq i \leq m, 1 \leq j \leq \alpha(R_i)). \end{cases}$$

La fonction de traduction τ se définit alors récursivement :

$$\begin{cases}
\tau(r) = \varphi(r) & \text{si } r \in \mathcal{R}, \\
\tau(rW) = \varphi(r)\tau(W) & \text{si } r \in \mathcal{R}_p, r \neq R^{(k)}, \\
\tau(R^{(k)}rW) = \varphi(R^{(k)})\tau(rW) + 1_{R^{(k)}<r} \cdot \varphi(r)\tau(W) & \text{si } r \in \mathcal{R}_p \cup \mathcal{R}, W \in \mathcal{R}_p^*\mathcal{R}.
\end{cases}$$

Contrairement à ce qui se passe avec le passage d'une grammaire en forme 1-2, l'itération d'une règle ne donne pas une grammaire équivalente, mais une grammaire plus fine, au moins dans le cas où le symbole itéré est le même que celui du membre gauche de la règle itérée. Nous allons le vérifier sur un exemple.

Exemple 3.7. Reprenons la grammaire G_1 engendrant les mots de Dyck. L'alphabet des dérivations pointées comportant deux lettres, il y a deux façons possibles d'itérer une règle dans cette grammaire. En choisissant d'itérer $R_2^{(1)}$, on obtient la grammaire G' définie par ses règles de dérivation :

$$\begin{array}{lccl} R_1 : & D & \to & \epsilon \\ R_1' : & D & \to & abD \\ R_2' : & D & \to & aaDbDbD \end{array}$$

La fonction φ est alors définie par :

$$\begin{array}{rcl} \varphi(R_1) & = & R_1 + R_1' \\ \varphi(R_2) & = & R_1' + 2.R_2' \\ \varphi(R_2^{(1)}) & = & 2.R_2'^{(1)} + R_2'^{(2)} \\ \varphi(R_2^{(2)}) & = & R_2'^{(2)} + R_2'^{(3)} \end{array}$$

En appliquant les règles de calcul de τ données dans la preuve de la proposition 3.6, nous obtenons pour le paramètre $p_{R_2^{(1)}R_2^{(1)}R_1}$:

$$\begin{array}{rcl} \tau(R_2^{(1)}R_2^{(1)}R_1) & = & \varphi(R_2^{(1)}R_2^{(1)}R_1) + \varphi(R_2^{(1)}R_1) + \varphi(R_2^{(1)})R_1' \\ & = & \left(2.R_2'^{(1)} + R_2'^{(2)}\right)\left(2.R_2'^{(1)} + R_2'^{(2)}\right)\left(R_1 + R_1'\right) \\ & & + \left(2.R_2'^{(1)} + R_2'^{(2)}\right)\left(R_1 + 2.R_1'\right) \end{array}$$

Le paramètre $p_{R_2^{(1)}R_2^{(1)}R_1}$ est compté par la variable s dans la solution du Q-système suivant, basé sur la grammaire G :

$$D(q, r, s) = q + D(qr, rs, s)D(q, r, s). \tag{1}$$

Dans ce système, q compte le paramètre p_{R_1}, et r, le paramètre $p_{R_2^{(1)}R_1}$. Nous utilisons l'équation 1 pour exprimer $D(qr, rs, s)$:

$$D(qr, rs, s) = qr + D(qr^2s, rs^2, s)D(qr, rs, s). \tag{2}$$

En reportant (2) dans (1), il vient alors :

$$D(q, r, s) = q + qrD(q, r, s) + D(qr^2s, rs^2, s)D(qr, rs, s)D(q, r, s).$$

Cette équation est une Q-équation basée sur la grammaire itérée G', et il est simple d'interpréter les paramètres comptés par q, r, et s :

– La variable q compte p_{W_1}, avec $W_1 = R_1 + R_1'$;

– la variable r compte p_{W_2}, avec $W_2 = R'_1 + (2.R'^{(1)}_2 + R'^{(2)}_2)W_1$;

– la variable s compte p_{W_3}, avec $W_3 = (2.R'^{(1)}_2 + R'^{(2)}_2)W_2$.

On retrouve ainsi le résultat donné en appliquant τ.

Afin de vérifier que la grammaire G' n'est pas Q-équivalente à G, il nous suffit d'observer la table donnant, pour les mots de longueurs 0, 2 et 4, les valeurs des paramètres élémentaires de rang 1 dans les deux grammaires.

	R_1	R_2	R_1	R'_1	R'_2
ϵ	1	0	1	0	0
ab	2	1	1	1	0
$aabb$	3	2	3	0	1
$abab$	3	2	1	2	0

Chacune des deux grammaires n'ayant qu'un seul symbole, chaque paramètre a un rang égal à son rang minimal. Par conséquent, si le paramètre $p_{R'_2}$ de G' était Q-comptable dans G, il devrait être combinaison linéaire (à coefficients entiers positifs) des paramètres p_{R_1} et p_{R_2} de G, ce qui n'est clairement pas le cas. Par conséquent, G n'est pas plus fine que G'.

Une autre façon de prouver cette propriété consiste à dire que les paramètres de rang 1 de G' sont capables de distinguer les mots $aabb$ et $abab$, alors que les paramètres de rang 1 de G ne le peuvent pas; par conséquent, il est impossible d'exprimer les paramètres de rang 1 de G' comme combinaisons linéaires de ceux de G.

Ce type d'arguments est naturel lorsqu'il s'agit de montrer qu'un paramètre n'est pas Q-comptable dans une grammaire donnée : il suffit généralement d'examiner les valeurs de ce paramètre pour les premiers mots engendrés par la grammaire, et de les comparer aux valeurs prises par les paramètres élémentaires dont le rang minimal ne dépasse pas celui du paramètre étudié.

Il existe une situation classique où l'itération d'une règle est une opération parfaitement naturelle, et que l'on risque d'effectuer sans même y penser. Si un symbole U ne peut être réécrit que d'une seule façon (lorsqu'il n'y a qu'une seule U-dérivation), il est naturel de faire disparaître ce symbole de la grammaire en remplaçant chaque apparition de U dans un membre droit de règle de dérivation, par le membre droit de la U-dérivation. Cela correspond en fait à itérer chacune des règles pointées qui font apparaître U. Cette opération est alors essentiellement l'inverse de celle décrite pour le passage en forme 1-2, et fournit une grammaire Q-équivalente.

3.2.3 Lemmes de marquage

Les lemmes suivants, que nous appelons lemmes de marquage, nous permettent de ne compter dans un paramètre que les chaînes vérifiant certaines conditions. Ils sont essentiels à la preuve du théorème de réduction du rang au rang minimal.

Lemme 3.8 (marquage supérieur). *Soient $G = (N, X, \mathcal{R}, S)$ une grammaire, et $\mathcal{R}_{p+}$ une partie de l'ensemble de ses règles de dérivation pointées $\mathcal{R}_p$.*

Pour chaque arbre de dérivation $\mathcal{A}$ de G, chaque règle de dérivation pointée $R \in \mathcal{R}_p$, et chaque nom de paramètre élémentaire $W \in \mathcal{R}_p^*\mathcal{R}$, soit $p_W^R(\mathcal{A})$ le nombre de chaînes de $\mathcal{A}$, de type W, qui sont des sous-chaînes droites d'au moins une chaîne de type RW.

Il existe une grammaire $G' = (N', X, \mathcal{R}', S)$, plus fine que G, qui vérifie les conditions suivantes :

1. *chaque paramètre G-Q-comptable élémentaire de rang k a, en tant que paramètre G'-Q-comptable, un rang au plus égal à k;*
2. *chaque paramètre p_W^R est G'-Q-comptable, de rang au plus $|W|$.*

Avant de donner la preuve de ce lemme, examinons un exemple.

Exemple 3.9. Considérons la grammaire $G = (\{D\}, \{a, b\}, \mathcal{R}, D)$, dont les règles de dérivation sont :

$$\begin{array}{llllllll} R_1 : & D & \to & ab & \quad R_3 : & D & \to & abD \\ R_2 : & D & \to & aDb & \quad R_4 : & D & \to & aDbD \end{array}$$

Cette grammaire, qui provient de la grammaire G_2 de l'exemple 2.3, engendre les mots de Dyck non vides.

Cette grammaire n'ayant qu'un seul symbole, tous les paramètres ont un rang minimal égal à leur rang. En particulier, le paramètre $p_{R_1+R_3}$ compte les “pics” (facteurs ab) des mots de Dyck engendrés. Mais que se passe-t-il si nous désirons compter les pics dont la hauteur n'est pas 1?

Un pic est représenté dans l'arbre de dérivation par un sommet s, étiqueté R_1 ou R_3. Sa hauteur est égale à 1, plus le nombre de dérivations pointées $R_2^{(1)}$ ou $R_4^{(1)}$ qui se trouvent sur la branche reliant ce sommet à la racine de l'arbre. Par conséquent, avec les notations du lemme, le nombre de pics de hauteur différente de 1 est le paramètre $p_{R_1}^{R_2^{(1)}} + p_{R_3}^{R_2^{(1)}} + p_{R_1}^{R_4^{(1)}} + p_{R_3}^{R_4^{(1)}}$: celui-ci compte en effet les sommets R_1 qui font partie d'une (ou plus) chaîne $R_2^{(1)}R_1$ ou $R_4^{(1)}R_1$, et les sommets R_3 qui font partie d'au moins une chaîne $R_2^{(1)}R_3$ ou $R_4^{(1)}R_3$.

Rien ne permet de penser *a priori* que ce paramètre est G-Q-comptable, les sommets étiquetés R_1 ou R_3 pouvant se trouver ou non “sous” les sommets R_2 ou R_4.

Nous voulons pouvoir, en comptant simplement les occurrences de certaines étiquettes dans l'arbre de dérivation d'un mot, dénombrer les sommets étiquetés R_1 ou R_3 qui font partie de chaînes convenables. Il faudrait pour cela que les étiquettes de ces sommets portent une information (une “marque”) sur la présence ou l'absence, dans la branche qui les relie à la racine, de sommets étiquetés R_2 ou R_4 (avec, dans le cas de R_4, la bonne direction, ce qui revient à dire que la branche contient une dérivation pointée $R_2^{(1)}$ ou $R_4^{(1)}$). La solution est de réétiqueter les sommets pour inclure cette information : tout sommet, étiqueté R_i, qui se trouve dans un sous-arbre d'un sommet étiqueté R_2, ou dans un sous-arbre gauche d'un sommet étiqueté R_4, est réétiqueté R_i'.

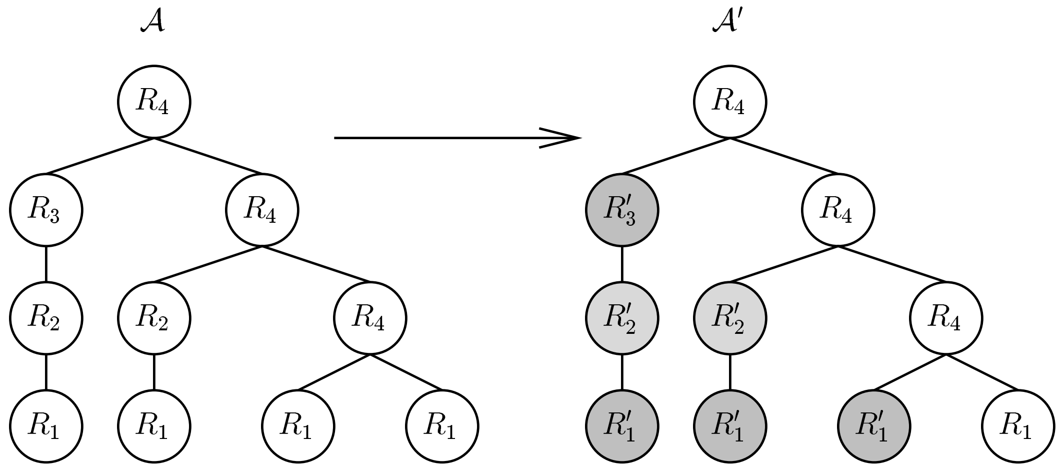


FIG. 3.6: *Marquage supérieur des règles* $R_2^{(1)}$ *et* $R_4^{(1)}$

La figure 3.6 montre l'arbre de dérivation du mot $w = aabaabbbaaabbbaabbab$, ainsi que la version réétiquetée. Le paramètre “nombre de pics de hauteur au moins 2” est égal au nombre de sommets étiquetés R_1' ou R_3' dans ce nouvel arbre. Par différence, le nombre de pics de hauteur 1 est le nombre de sommets étiquetés R_1 ou R_3 dans ce même arbre. Comme on peut le voir, le mot w possède 1 pic de hauteur 1, et 4 autres pics de hauteurs respectives 2, 3, 3, et 2.

Dans cet exemple, il n'a pas été nécessaire de marquer différemment les deux règles pointées $R_2^{(1)}$ et $R_4^{(1)}$, comme c'est le cas en général dans le lemme.

Preuve. Nous donnons tout de suite la grammaire G'. Les symboles de N' sont de la forme U_E, où U est l'un des symboles de N et E est une partie de l'ensemble $\mathcal{R}_{p+}$. Les règles de dérivation de G' sont formées à partir de celles de G de la manière suivante :

– pour chaque règle $R \in \mathcal{R}$ et chaque ensemble $E \subset \mathcal{R}_{p+}$, on forme une règle R_E qui est identique à R, à ceci près que chaque symbole U (du membre gauche comme du membre droit) est remplacé par U_E;

– pour chaque règle pointée spéciale $R^{(d)} \in \mathcal{R}_{p+}$, et pour chaque ensemble $E \subset \mathcal{R}_{p+}$, le d-ème symbole du membre droit de R_E, U_E, est remplacé par $U_{E\cup\{R^{(d)}\}}$.

Les indices ajoutés aux symboles ne modifient en rien les mots engendrés, en ce sens que, pour tout symbole $U \in N$ et tout $E \subset \mathcal{R}_{p+}$, $L_G(U) = L_{G'}(U_E)$. En revanche, la façon dont sont formées les règles de dérivation de G' assure que, dans chaque arbre de dérivation de G', chaque sommet a une étiquette qui nous renseigne sur la branche reliant ce sommet à la racine de l'arbre.

Soit $\mathcal{A}$ l'arbre de dérivation, dans G, d'un mot w. Pour chaque sommet s de $\mathcal{A}$, notons $(s_1, \ldots, s)$ la chaîne de sommets de $\mathcal{A}$ formée de tous les sommets ancètres de s, et $E(s)$ l'ensemble de toutes les règles pointées qui apparaîssent à la fois dans le type de $(s_1, \ldots, s)$ et dans $\mathcal{R}_{p+}$.

Soit $\mathcal{A}'$ l'arbre de dérivation du mot w dans la grammaire G'. Alors, $\mathcal{A}'$ est obtenu à partir de $\mathcal{A}$ en remplaçant chaque étiquette R de chaque sommet s, par $R_{E(s)}$. En effet, la définition que nous avons donnée des membres droits des règles R_E, assure que les ensembles indices $E(s)$ se propagent correctement.

La grammaire G' remplit les conditions indiquées. En effet, notons φ l'application "effacement des indices", définie sur N' par $\varphi(U_E) = U$ et sur $\mathcal{R}'$ par $\varphi(R_E) = R$. L'application φ est étendue naturellement aux dérivations pointées et, par morphisme, aux noms de paramètres de G'. Alors, il est clair que, dans un arbre de dérivation $\mathcal{A}$ de G, toute chaîne de type W devient, dans l'arbre $\mathcal{A}'$ correspondant de G', une chaîne dont le type appartient à $\varphi^{-1}(W)$. Réciproquement, lors du passage de $\mathcal{A}'$ à $\mathcal{A}$, une chaîne de type W' devient une chaîne de type $\varphi(W')$. Par conséquent, les paramètres p_W (dans G) et $p_{\varphi^{-1}(W)}$ (dans G') coïncident. Ainsi, G' est bien plus fine que G, et le rang des paramètres est conservé (condition 1).

Enfin, les chaînes comptées dans un arbre $\mathcal{A}$ par le paramètre $p_W^{R^{(d)}}$ sont exactement celles qui sont de type W et dont le premier sommet s_1 vérifie $R^{(d)} \in E(s_1)$. Or $E(s_1)$ est, par construction, l'indice de l'étiquette de s_1 dans l'arbre $\mathcal{A}'$. Il suffit donc de choisir pour W', les noms de $\varphi^{-1}(W)$ dont la première lettre est de la forme $R_E^{(d)}$, avec $R^{(d)} \in E$, pour assurer que $p_W^{R^{(d)}}$ (dans G) coïncide avec $p_{W'}$ (dans G'). De plus, le rang de $p_{W'}$ est bien la longueur du plus long mot de W', qui est $|W|$. $\square$

Exemple 3.10. Dans l'exemple 3.9, les règles pointées dont nous voulions repérer la présence étaient $R_2^{(1)}$ et $R_4^{(1)}$, qui forment donc l'ensemble $\mathcal{R}_{p+}$. Toutefois, il n'était nécessaire que de repérer la présence d'au moins l'une de ces deux règles, sans faire de distinction entre elles. Nous avons donc utilisé, comme étiquettes, R_i pour $R_{i,\emptyset}$ et R'_i pour $R_{i,E}$ avec $\emptyset \subsetneq E \subset \mathcal{R}_{p+}$.

La grammaire G' obtenue est donc :

$$\begin{array}{llll@{\qquad}llll}
R_1: & D & \to & ab & R'_1: & D' & \to & ab \\
R_2: & D & \to & aD'b & R'_2: & D' & \to & aD'b \\
R_3: & D & \to & abD & R'_3: & D' & \to & abD' \\
R_4: & D & \to & aD'bD & R'_4: & D' & \to & aD'bD'
\end{array}$$

L'arbre $\mathcal{A}'$ de la figure 3.6 est un arbre de dérivation de G'. Dans cette grammaire, le symbole D' n'est accessible à partir de D qu'au travers des dérivations pointées $R_2^{(1)}$ et $R_4^{(1)}$, et par conséquent, les pics de hauteur $h > 1$ sont ceux introduits par les dérivations R'_1 et R'_3, tandis que les pics de hauteurs 1 sont introduits par les dérivations R_1 et R_3.

Lemme de marquage inférieur

Le lemme de marquage supérieur nous permet de propager à chaque sommet d'un arbre de dérivation une information sur ce qui se trouve au-dessus de lui, c'est-à-dire dans la branche qui le relie à la racine. Le lemme ci-dessous permet un marquage similaire, mais portant sur ce qui se trouve en dessous de chaque sommet, c'est-à-dire dans les sous-arbres issus de ses fils.

Lemme 3.11 (marquage inférieur). *Soient $G = (N, X, \mathcal{R}, S)$ une grammaire, $\mathcal{R}_+ \subset \mathcal{R}$ une partie de ses règles de dérivation, et un entier $m > 0$.*

Pour chaque arbre de dérivation $\mathcal{A}$ de G, chaque nom de paramètre élémentaire $W \in \mathcal{R}_p^+$, chaque $R \in \mathcal{R}_+$, et chaque entier $n \le m$, notons $p_{W,n}^R(\mathcal{A})$ le nombre de chaînes de $\mathcal{A}$, de type W, qui sont des sous-chaînes gauches d'exactement n (ou d'au moins m, si $n = m$) chaînes de type WR.

Il existe une grammaire $G' = (N', X, \mathcal{R}', S')$, plus fine que G, et qui vérifie les conditions suivantes :

1. *chaque paramètre G-Q-comptable de rang k a, en tant que paramètre G'-Q-comptable, un rang au plus égal à k;*

2. *chaque paramètre $p_{W,n}^R$ est G'-Q-comptable de rang au plus $|W|$.*

La principale différence de forme entre les deux lemmes de marquage réside dans le fait que le lemme de marquage inférieur, tel qu'il est énoncé ci-dessus, utilise un entier m supplémentaire là où le lemme de marquage supérieur se contentait, en fait, de 0 ou 1 (le lemme de marquage supérieur permet de compter les chaînes qui sont sous-chaînes droites d'*au moins* une chaîne d'un type donné, tandis que le lemme de marquage inférieur permet de distinguer celles qui sont sous-chaînes gauches d'*exactement* $0, 1, \ldots, m-1$ telles chaînes). Il est possible d'énoncer le lemme de marquage supérieur de façon similaire, mais un tel raffinement complique les notations, et la version que nous avons démontrée est suffisante pour démontrer le théorème de réduction du rang.

Preuve. Soit $\mathcal{A}$ un arbre de dérivation de G, et s un sommet de $\mathcal{A}$. Pour chaque $R \in \mathcal{R}_+$, notons $R(s)$ le nombre de sommets étiquetés R dans le sous-arbre issu de s, avec la convention que, si ce nombre est supérieur à m, $R(s) = m$. Si s' est le i-ème fils de s, notons $R(s,i) = R(s')$.

Comme dans le cas du lemme de marquage supérieur, le passage à la grammaire G' consiste essentiellement à d'ajouter à l'étiquette de chaque sommet d'arité k, la liste des valeurs de $R(s)$ et $R(s,i)$, pour $R \in \mathcal{R}_+$ et $1 \leq i \leq k$.

La grammaire G' est définie comme suit. Chaque symbole est formé d'un symbole de G et d'un entier $n(R)$ pour chaque règle $R \in \mathcal{R}_+$:

$$N' = \left\{ \left(U, (n(R))_{R \in \mathcal{R}_+} \right), U \in N, 0 \leq n(R) \leq m \right\}.$$

Chaque entier $n(R)$ indique la valeur du paramètre p_R sur les mots engendrés par le symbole $\left(U, (n(R))_{R \in \mathcal{R}_+} \right)$:

$$L_{G'} \left(U, (n(R))_{R \in \mathcal{R}_+} \right) = \{ w \in L_G(U) : \forall R \in \mathcal{R}_+, n(R) = \min(m, p_R(w)) \}. \tag{3}$$

Pour chaque règle $R_0 : U \to w_0 U_1 w_1 \ldots U_k w_k \in \mathcal{R}$, la grammaire G' comprendra $(m+1)^{k|\mathcal{R}_+|}$ règles, dont les membres droits sont obtenus en remplaçant successivement, de toutes les manières possibles, chaque symbole U_i par un $\left(U_i, (n_i(R))_{R \in \mathcal{R}_+} \right)$. Le membre gauche de chaque règle est alors $\left(U, (n(R))_{R \in \mathcal{R}_+} \right)$ avec

$$n(R) = \min \left(m, \delta_{R,R_0} + \sum_{i=1}^{k} n_i(R) \right),$$

où δ_{R,R_0} vaut 1 si $R = R_0$, 0 sinon. Par une récurrence évidente sur la taille des arbres de dérivation, ces règles de dérivation engendrent le langage prévu pour chaque symbole.

Il convient de donner un axiome à G'; soit donc S_0 un nouveau symbole, et pour chaque symbole $\left(S, (n(R))_{R \in \mathcal{R}_+} \right)$, ajoutons une règle $S_0 \to \left(S, (n(R))_{R \in \mathcal{R}_+} \right)$: cela assurera que

$L_{G'}(S_0) = L_G(S)$. Le symbole S_0 n'étant accessible à partir d'aucun autre, ses dérivations n'apparaîtront qu'à la racine des arbres de dérivation.

Notons φ l'application qui envoie chaque règle de $\mathcal{R}'$ ainsi définie (autre qu'une S_0-dérivation) sur la règle $R \in \mathcal{R}$ dont elle est issue, et étendons φ naturellement aux règles pointées de $\mathcal{R}'_p$, puis, par morphisme, aux noms de paramètres dans G'. Le passage de G à G' s'effectue simplement en réétiquetant les sommets de l'arbre de dérivation, et en ajoutant une nouvelle racine. Donc, pour tout mot $w \in L_{G'}\left(U, (n(R))_{R\in\mathcal{R}_+}\right)$ et tout nom de paramètre W dans G, $p_W(w) = p_{\varphi^{-1}(W)}(w)$. Ainsi, G' est bien plus fine que G, et le rang des paramètres G-Q-comptables n'augmente pas lors du passage à G'.

Quant au paramètre $p^R_{W,n}$, si la dernière lettre de W est $R_1^{(d)} \in \mathcal{R}_p$, il compte dans les arbres de dérivation de G' les chaînes de sommets dont le type est dans $\varphi^{-1}(W)$, et dont le d-ème symbole au membre droit de la dernière dérivation vérifie $n(R) = n$. Cela revient à ne prendre, dans $\varphi^{-1}(W)$, que les mots dont la dernière lettre appartient à une partie convenable de $\varphi^{-1}(R_1)$. Ceci assure que $p^R_{W,n}$ est bien G'-Q-comptable de rang $|W|$.

□

Remarque. La grammaire G' construite dans la preuve du lemme de marquage inférieure peut parfaitement ne pas être complète; il est possible en effet que certains des langages engendrés soient vides.

Exemple 3.12. Reprenons les grammaires G_1 et G_2, qui engendrent tous deux le langage de Dyck. Nous allons voir que la grammaire G_2 peut être obtenue en appliquant la construction du lemme de marquage inférieur à G_1.

Dans la grammaire G_1, la règle R_1 est la seule règle terminale; par conséquent $p_{R_1}(w) \geq 1$ pour tout mot de Dyck w, et $p_{R_1}(w) = 1$ seulement pour le mot vide. Prenons $\mathcal{R}_+ = \{R_1\}$, et $m = 2$.

La construction donnée de la grammaire G' prévoit 4 symboles, S_0, $(D, 0)$, $(D, 1)$ et $(D, 2)$, et, pour chaque $i = 0, 1, 2$,

$$L_{G'}(D, i) = \{w \in D, i = \min(2, p_{R_1}(w))\}\,.$$

En particulier, $L_{G'}(D, 0) = \emptyset$, et $L_{G'}(D, 1) = \{\epsilon\}$, donc $L_{G'}(D, 2)$ est l'ensemble des mots de Dyck non vides.

Les règles de dérivation de la grammaire G' sont :

$$\begin{array}{rcl}
S_0 & \to & (D,0) \\
S_0 & \to & (D,2) \\
(D,0) & \to & a.(D,0).b.(D,0) \\
(D,1) & \to & a.(D,1).b.(D,0) \\
(D,2) & \to & a.(D,1).b.(D,1) \\
(D,2) & \to & a.(D,2).b.(D,0) \\
(D,2) & \to & a.(D,2).b.(D,2)
\end{array}
\qquad
\begin{array}{rcl}
S_0 & \to & (D,1) \\
(D,1) & \to & \epsilon \\
(D,1) & \to & a.(D,0).b.(D,1) \\
(D,2) & \to & a.(D,0).b.(D,2) \\
(D,2) & \to & a.(D,1).b.(D,2) \\
(D,2) & \to & a.(D,2).b.(D,1) \\
 & &
\end{array}$$

Il apparaît immédiatement que le seul symbole accessible à partir de $(D,0)$ est $(D,0)$, tandis que la seule règle terminale a pour membre gauche $(D,1)$; par conséquent $L_{G'}(D,0)$ est vide, et toutes les règles de dérivation faisant intervenir $(D,0)$ peuvent être éliminées. La grammaire alors obtenue, une fois les symboles $S_0,(D,1),(D,2)$ renommés en D,F,E respectivement, est :

$$\left\{\begin{array}{lllllll}
D & \to & F & \qquad & D & \to & E \\
F & \to & \epsilon & & E & \to & aFbF \\
E & \to & aFbE & & E & \to & aEbF \\
E & \to & aEbE & & & &
\end{array}\right.$$

La seule F-dérivation de cette grammaire est $F \to \epsilon$; en itérant chacune des dérivations pointées qui produisent le symbole F, on fait disparaître ce symbole de tous les membres droits, ce qui le rend inaccessible à partir de l'axiome D. La grammaire alors obtenue est

$$\left\{\begin{array}{lllllll}
D & \to & \epsilon & \qquad & D & \to & E \\
E & \to & ab & & E & \to & abE \\
E & \to & aEb & & E & \to & aEbE
\end{array}\right.$$

qui est exactement la grammaire G_2. Par conséquent, cette grammaire est plus fine que G_1.

3.2.4 Réduction du rang

Les lemmes de marquage ont pour principal but de permettre la preuve du théorème suivant :

Théorème 3.13. *Soit G une grammaire. Il existe une grammaire G', plus fine que G, dans laquelle tout paramètre G-Q-comptable non borné a un rang égal à son rang minimal.*

Preuve. Etant donnée la grammaire G, nous devons essentiellement trouver une grammaire G', plus fine que G, dans laquelle tout paramètre *élémentaire* de G, non borné, a dans G' un rang qui correspond à son rang minimal. Rappelons que, d'après la proposition (2.24), il est équivalent, pour un paramètre Q-comptable, d'être non borné et d'avoir rang minimal au moins 1. Il est donc normal que les paramètres bornés soient exclus du cadre de ce théorème.

La proposition (2.26) décrit exactement d'où peut provenir l'écart entre rang minimal et rang formel pour un paramètre élémentaire : il peut s'agir de dérivations pointées ne pouvant se retrouver à plus d'un exemplaire sur une même branche ($R^{(i)}$, lorsque $g(R)$ n'est pas accessible à partir de $d(R,i)$), ou de la dernière dérivation R du nom du paramètre, à condition que le paramètre (de rang formel 1) p_R soit borné, au moins sur $L_G(d(R_1,i))$, où $R_1^{(i)}$ est l'avant-dernière lettre du nom du paramètre.

Soit $\mathcal{R}_{p+}$ l'ensemble des règles pointées de la forme $R^{(k)}$, telles que $g(R)$ ne soit pas accessibles à partir de $d(R,k)$. Ces règles pointées sont exactement celles qui ne peuvent pas "boucler", en ce sens que le k-ème sous-arbre d'un sommet étiqueté R ne peut contenir de sommet étiqueté R. Soit G_1 la grammaire obtenue en appliquant le lemme de marquage supérieur à cet ensemble de règles $\mathcal{R}_{p+}$. Nous allons montrer que, dans G_1, l'écart entre rang formel et rang minimal pour les paramètres G-Q-comptables élémentaires est inférieur ou égal à 1.

Soit en effet $W \in \mathcal{R}_p^*\mathcal{R}$ un nom de paramètre G-Q-comptable élémentaire de rang minimal 1 ou plus. Posons $W = W_0 R_1^{(d_1)} W_1 \dots W_{k-1} R_k^{(d_k)} W_k R'$, où $R_i^{(d_i)} \in \mathcal{R}^+$ et $W_i \in (\mathcal{R}_p - \mathcal{R}_{p+})^*$. Notons encore $n_i = |W_i|$: le rang minimal de p_W est donc, d'après la proposition (2.26), $1 + n_1 + \cdots + n_k$ ou $n_1 + \cdots + n_k$, suivant que R' participe ou non au rang minimal.

Puisque p_W est supposé non borné, il n'est pas identiquement nul, et donc il existe un arbre de dérivation de G qui contient une chaîne de type W. Par construction de $\mathcal{R}_{p+}$, la règle $R_i^{(d_i)}$ ne peut apparaître qu'une fois dans le type d'une chaîne de sommets d'un arbre de dérivation, et par conséquent, si $R^{(d)}$ est une dérivation pointée qui apparaît dans W après $R_i^{(d_i)}$, aucune chaîne de type $R^{(d)}R_i$ ne peut exister. De même, si $R^{(d)}$ apparaît dans W avant $R_i^{(d_i)}$, aucune chaîne de type $R_i^{(d_i)}R$ ne peut exister. Donc, dans un arbre de dérivation $\mathcal{A}$ de G, les chaînes $(s_1, \dots, s_n)$ de type $W' = W_0 W_1 \dots W_k R'$ qui sont des sous-chaînes d'une chaîne de type W, sont exactement celles dont le dernier sommet s_n se trouve dans le d_i-ème sous-arbre d'un sommet étiqueté R_i, pour chaque $i \leq k$. Ces chaînes sont comptées, dans G_1, par un paramètre Q-comptable de rang $|W'| = 1 + \sum_i n_i$, ce qui assure que, dans G_1, l'écart entre rang minimal et rang formel des paramètres G-Q-comptables non bornés est au plus de 1.

Appliquons maintenant à la grammaire G_1 le lemme de marquage inférieur, en prenant comme ensemble de règles $\mathcal{R}_+$,

$$\mathcal{R}_+ = \left\{ R \in \mathcal{R}' : \exists U \in N', p_R \text{ est borné sur } L_{G_1}(U) \right\}$$

Nous prenons, comme entier m pour le lemme de marquage inférieur,

$$m = 1 + \max \left\{ p_R(w) : w \in L_G(U), p_R \text{ est borné sur } L_G(U) \right\}.$$

Soit G' la grammaire ainsi obtenue.

Tout paramètre G_1-Q-comptable élémentaire de rang n dont la dernière dérivation ne compte pas dans le rang minimal (d'après la proposition (2.26)), a, en tant que paramètre G'-Q-comptable, un rang au plus $n-1$. Etant donné que, pour chaque nom de paramètre élémentaire de G' qui intervient dans la décomposition d'un paramètre G-Q-comptable, seule la dernière lettre peut être source d'écart entre rang formel et rang minimal, il est donc clair que tous ces paramètres ont, dans G', un rang formel égal à leur rang minimal.

La grammaire G' remplit donc les conditions exigées. $\square$

Ce théorème permet de justifier la supposition faite dans l'exemple présenté section 3.1 : si un paramètre est de rang minimal 1, il est possible de trouver une grammaire plus fine dans laquelle il est de rang formel 1.

Remarque. Rien n'indique que *tous* les paramètres G'-Q-comptables aient un rang minimal égal à leur rang formel; seuls ceux qui apparaissent dans la décomposition des paramètres G-G-comptables sont tenus de vérifier une telle propriété.

Par ailleurs, remarquons que, pour chaque opération de changement de grammaire que nous avons étudiée, il est possible de définir, le plus souvent sous forme récursive, la façon dont un nom de paramètre de la grammaire de départ se transforme dans la nouvelle grammaire.

Enfin, notons que les constructions décrites dans les deux lemmes de marquage ont tendance à augmenter grandement la complexité des grammaires, le nombre de symboles et de règles de dérivation pouvant exploser rapidement.

L'intérêt du théorème 3.13 est principalement théorique. Il serait tentant de l'utiliser pour déterminer si un paramètre est Q-comptable dans une grammaire donnée (voir les commentaires sur la proposition 2.26), mais la taille des grammaires construites augmente trop vite pour que l'idée soit applicable en pratique.

Toutefois, ce théorème nous permet d'affirmer que les paramètres dont le rang minimal est strictement inférieur au rang n'ont pas un comportement fondamentalement différent de ceux pour lesquels le rang minimal est égal au rang. Cette remarque peut être précieuse pour démontrer des résultats négatifs, comme celui de la section 3.1.

3.3 Q-grammaires et grammaires d'objets

Dans [36], Dutour définit la notion de grammaires d'objets isomorphes. Une grammaire algébrique peut être considérée comme un cas particulier de grammaire d'objets, et il semble naturel de comparer les deux notions.

Dans le cas d'une grammaire n'ayant qu'un seul symbole non terminal, deux grammaires sont isomorphes si, pour chaque entier n, elles ont toutes deux le même nombre de règles d'arité n; il est alors possible de décrire, par un simple réétiquetage des arbres de dérivation, des bijections entre les familles d'objets engendrés. De plus, dans le cas de grammaires n'ayant qu'un seul symbole non terminal, Dutour montre que deux grammaires ayant chacune au moins une règle d'arité au moins égale à 2 ont des itérées isomorphes. La notion d'itération employée n'est pas très différente de celle que nous avons définie précédemment.

Dans le cas des Q-grammaires, toutefois, ces résultats n'ont pas d'équivalents simples. Considérons les deux grammaires G et G', engendrant toutes deux le langage de Dyck :

$$G : \left\{ \begin{array}{llcl} R_1 : & D & \to & \epsilon \\ R_2 : & D & \to & aDbD \end{array} \right. \qquad G : \left\{ \begin{array}{llcl} R'_1 : & D & \to & \epsilon \\ R'_2 : & D & \to & DaDb \end{array} \right.$$

Ces deux grammaires ont chacune une règle d'arité 0 et une règle d'arité 2, mais il est facile de montrer qu'elles ne sont pas Q-équivalentes. Le tableau ci-dessous indique, pour les mots de Dyck de longueur inférieure ou égale à 6, les valeurs des paramètres Q-comptables de rang au plus 2 dans chacune des deux grammaires. Les paramètres $R_2^{(d)R_1}$ et $R_2'^{(d)}R'_1$, pour $d = 1, 2$, ne sont pas mentionnés, car ils s'écrivent systématiquement comme combinaisons linéaires des autres paramètres (ainsi, $p_{R_2^{(1)}R_1} = p_{R_2^{(1)}R_2} + p_{R_2}$).

	G				G'			
w	R_1	R_2	$R_2^{(1)}R_2$	$R_2^{(2)}R_2$	R_{-1}	R'_2	$R_2'^{(1)}R'_2$	$R_2'^{(2)}R'_2$
ϵ	1	0	0	0	1	0	0	0
ab	2	1	0	0	2	1	0	0
$aabb$	3	2	1	0	3	2	0	1
$abab$	3	2	0	1	3	2	1	0
$aaabbb$	4	3	3	0	4	3	0	3
$aababb$	4	3	2	1	4	3	1	2
$aabbab$	4	3	1	1	4	3	2	1
$abaabb$	4	3	1	2	4	3	1	1
$ababab$	4	3	0	3	4	3	3	0

Dans ce tableau, il est impossible d'obtenir la colonne correspondant au paramètre $p_{R_2'^{(1)}R'_2}$ comme combinaison linéaire à coefficients entiers positifs des colonnes correspondant aux paramètres de la grammaire G; par conséquent, $p_{R_2'^{(1)}R'_2}$ n'est pas Q-comptable dans G (tous les paramètres élémentaires des deux grammaires ayant un rang minimal égal à leur rang, il est inutile de faire entrer les paramètres élémentaires de rang supérieur en ligne de compte).

Notons qu'il serait possible, en permutant les mots de même longueur entre eux, de faire coïncider les paramètres élémentaires des deux grammaires. Cette permutation (*aaabbb* devient *ababab*, *aababb* devient *aabbab*, *aabbab* devient *abaabb*, *abaabb* devient *aababb*, et *ababab* devient *aaabbb* pour les mots de longueur 6) correspond exactement à interpréter les arbres de dérivation de G comme arbres de dérivation de G' (après réétiquetage) : il n'est pas surprenant que cette transformation ne laisse pas inchangés les paramètres Q-comptables, qui sont définis mot par mot, et non pas globalement "à une bijection près".

En fait, il semble raisonnable de conjecturer que deux grammaires n'ayant chacune qu'un symbole non terminal, et ayant, pour chaque n, le même nombre de règles d'arité n, ne sont Q-équivalentes que si elles sont identiques (à un renommage des règles près).

3.4 Conclusion

Nous avons pu montrer dans ce chapitre que la plupart des transformations élémentaires sur les grammaires se comportent bien du point de vue des paramètres Q-comptables. Ainsi, une grammaire est Q-équivalente à sa forme 1-2, et, au pire, l'itération ne fait qu'ajouter des paramètres Q-comptables. Il est donc possible d'effectuer des modifications mineures dans la forme d'une grammaire, sans perte de paramètres Q-comptables. D'un point de vue pratique, de telles propriétés de stabilité sont confortables, comme nous le verrons au chapitre 5.

D'un point de vue plus théorique, le théorème de réduction du rang peut s'avérer précieux. Nous en avons vu un exemple lorsqu'il s'est agi de prouver qu'un paramètre donné ne pouvait être Q-comptable, lorsque le simple examen du rang minimal ne permettait pas de conclure.

Il pourrait être intéressant de savoir si les transformations de grammaires que nous avons définies, suffisent pour obtenir toutes les grammaires plus fines que la grammaire de départ; il est probable que non.

Une autre direction possible de recherche serait de déterminer quels sont, pour un langage donné, les paramètres qui sont Q-comptables dans toute grammaire. Nous savons déjà que tout paramètre qui s'exprime comme combinaison linéaires des nombres d'occurrences des lettres, est Q-comptable dans toute grammaire. Dans le cas du langage de Dyck, il semble vraisemblable que le paramètre “aire de Carlitz”, ou une de ses variantes, soit Q-comptable dans toute grammaire.

Dans le même registre, il serait intéressant de savoir s'il peut exister entre deux paramètres p et p' des relations du type “p' est Q-comptable dans toute grammaire où p est Q-comptable”. Un exemple serait, pour le langage de Dyck, de prouver que le paramètre “somme des hauteurs des pics” est Q-comptable dans toute grammaire où le paramètre “nombre de pics” l'est.

Enfin, il pourrait être intéressant, étant données deux grammaires, de savoir déterminer s'il existe une grammaire plus fine que chacune d'elles. Dans ce domaine, les techniques qui s'appliquent aux grammaires d'objets ne semblent pas pouvoir être transposées. Une question proche serait: étant donnés deux paramètres qui sont Q-comptables dans des grammaires différentes, existe-t-il une grammaire dans laquelle ces deux paramètres soient Q-comptables? Nous avons vu Section 3.1 que ce n'est pas vrai dans le cas général; toutefois, il semble que ce le soit pour toutes les grammaires issues d'une même grammaire initiale par les transformations décrites dans ce chapitre.

Chapitre 4

Statistiques et asymptotiques

Dans certains cas, bien qu'il soit possible d'obtenir des formules exactes d'énumération suivant certains paramètres pour une famille donnée d'objets combinatoires, une expression asymptotique apporte des informations beaucoup plus lisibles qu'une expression comportant des sommations multiples.

Nous nous concentrons dans ce chapitre sur le lien entre Q-comptabilité et deux aspects essentiels de la combinatoire énumérative : le calcul de valeurs moyennes de paramètres, et l'obtention d'expressions asymptotiques dans les problèmes d'énumération. En particulier, nous montrons que les séries de moments suivant un paramètre Q-comptable sont algébriques.

On trouvera une description généraliste de techniques d'énumération asymptotique dans les articles de Bender [9, 8], ainsi qu'une version multivariée dans l'article de Bender et Richmond [10]. L'étude asymptotique du nombre de mots de longueur donnée d'un langage algébrique peut être menée au moyen des techniques décrites par Flajolet et Sedgewick dans [45]. Drmota [34, 35] et Flajolet et Sedgewick [46] ont montré que, sous des hypothèses raisonnables, le nombre d'occurrences d'une lettre dans les mots de longueur n d'un langage algébrique admet une distribution limite gaussienne, d'espérance et de variance proportionnelles à n. De tels résultats s'appliquent directement aux paramètres Q-comptables de rang 1. La détermination de lois limites pour des paramètres Q-comptables de rang supérieur est beaucoup plus difficile. Prellberg [68] a montré que l'aire (paramètre de rang 2) des polyominos parallélogrammes de périmètre $2n$ admet une loi limite qui suit la *distribution d'Airy*.

4.1 Introduction et notations

Les questions d'ordre statistique relatives aux paramètres définis sur une famille A d'objets combinatoires, sont essentiellement : quelle est la valeur moyenne de tel paramètre $p(w)$, lorsque w est un objet pris au hasard dans A ? Comment se répartissent les valeurs de $p(w)$? Y a-t-il une corrélation entre les valeurs de $p_1(w)$ et celles de $p_2(w)$?

Avant tout, pour pouvoir parler d'objet pris “au hasard” dans A, il est indispensable de définir une loi de probabilité sur A. En général, les familles d'objets considérées sont infinies et dénombrables (mots d'un langage algébrique, différentes classes de polyominos ...); il n'existe pas sur de tels ensembles de lois de probabilités privilégiées, et en particulier pas de lois uniformes. Pour que chaque objet ait la même probabilité, il convient alors de se ramener à des ensembles finis. C'est ce que nous faisons en étudiant la distribution de paramètres Q-comptables pour les mots de *taille* donnée.

Dans ce chapitre, nous manipulerons des séries formelles à un grand nombre de variables. Si $F = F(q_1, \ldots, q_k)$ est une série formelle des variables $q_1, \ldots, q_k$, le coefficient de $q_1^{n_1} \ldots q_k^{n_k}$ dans la série F est noté

$$[q_1^{n_1} \ldots q_k^{n_k}]F.$$

Une opération très utile est la *projection* π de $\mathbb{Q}[[q_1, \ldots, q_{k'}]]$ vers $\mathbb{Q}[[q_1, \ldots, q_k]]$, avec $k' > k$. Elle correspond à donner la valeur 1 à chacune des variables $q_{k+1}, \ldots, q_{k'}$. La série ainsi obtenue sera simplement notée $F(q_1, \ldots, q_k) = \pi F(q_1, \ldots, q_{k'})$.

Formellement, on a

$$[q_1^{n_1} \ldots q_k^{n_k}]F(q_1, \ldots, q_k) = \sum_{n_{k+1} \geq 0} \ldots \sum_{n_{k'} \geq 0} [q_1^{n_1} \ldots q_{k'}^{n_{k'}}]F(q_1, \ldots, q_{k'}).$$

La projection d'une série n'est définie que si la somme ci-dessus est une somme finie; cette propriété sera toujours vraie dans les cas que nous étudierons.

4.2 Généralités

4.2.1 Distributions de paramètres

En général, il existe pour chaque classe d'objets au moins un paramètre “naturel”, qui possède la propriété que, pour chaque valeur possible du paramètre, il n'existe qu'un nombre fini d'objets pour lesquels le paramètre prend cette valeur. Nous appelons un tel paramètre une *taille*.

La longueur des mots d'un langage sur un alphabet fini, le périmètre ou l'aire des polyominos, sont des exemples de tailles. Le fait que l'un des paramètres suivant lesquels on établit une série génératrice soit une taille, est une condition suffisante pour que cette série génératrice ait un sens en tant que série formelle.

Dans un langage, il existe une taille plus naturelle que les autres : il s'agit de la longueur des mots. Toutefois, dans le cadre de la combinatoire énumérative, les mots sont souvent un moyen de coder d'autres objets, et il peut exister différents paramètres pouvant faire office de taille pour une même classe d'objets combinatoires – les plus classiques, pour des objets comme les polyominos, étant l'*aire* et le *périmètre*.

Soit D un ensemble d'objets, et soit $D(q_1, \ldots, q_k)$ sa série génératrice suivant k paramètres $\lambda_1, \ldots, \lambda_k$. Le premier paramètre λ_1 est supposé être une taille.

Sur chaque partie finie de D, comme par exemple chaque ensemble

$$D_{\lambda_1=n} = \{w \in D, \lambda_1(w) = n\}$$

il existe une probabilité uniforme, et donc chaque paramètre ou famille de paramètres a sur chaque $D_{\lambda_1=n}$ une loi de distribution; par conséquent, parler de valeurs moyennes, de variance, de corrélation entre deux paramètres a un sens, dès lors que l'on s'est restreint à une partie finie de D.

Nous nous bornerons donc, dans le cas général, à parler de valeur moyenne, ou de "loi", d'un paramètre, *connaissant la valeur d'un autre paramètre faisant office de taille.* L'absence de loi de probabilité privilégiée sur D nous interdira toutefois de parler de probabilités conditionnelles, bien que nous en utilisions à l'occasion les notations. Ainsi, la notation

$$P(A|\lambda_1 = n)$$

désigne la probabilité que l'événement A se produise lorsqu'un objet est pris aléatoirement et de manière uniforme parmi ceux qui vérifient $\lambda_1(w) = n$.

De même, nous utiliserons la notation

$$E(\lambda|\lambda_1 = n)$$

pour désigner l'espérance de $\lambda(w)$, lorsque l'objet w est pris aléatoirement et de manière uniforme parmi ceux qui vérifient $\lambda_1(w) = n$.

Les coefficients de la série D et de ses projections obtenues en fixant une ou plusieurs de ses variables à 1, contiennent toute l'information souhaitable sur ces lois conjointes.

Plus précisément, la probabilité pour un objet w de D, de taille $\lambda_1(w) = n$, d'avoir pour ses paramètres $(\lambda_i)_{2\leq i\leq k}$ les valeurs $(n_i)_{2\leq i\leq k}$, est

$$(1) \qquad P\left(\lambda_2 = n_2, \ldots, \lambda_k = n_k | \lambda_1 = n\right) = \frac{\left[q_1^n q_2^{n_2} \cdots q_k^{n_k}\right] D(q_1, \ldots, q_k)}{\left[q_1^n\right] D(q_1)}.$$

De manière plus générale, nous pouvons choisir une sous-famille $(\lambda_i)_{i\in I}$ de paramètres fixés (avec la condition que chaque partie $D_{(\lambda_i)_{i\in I}=(n_i)_{i\in I}}$ soit finie), et une sous-famille $(\lambda_j)_{j\in J}$ de paramètres "libres". Les ensembles I et J doivent être disjoints : si un paramètre est fixé, il est inutile d'étudier sa répartition. Nous pouvons alors étudier la répartition des paramètres $(\lambda_j)_{j\in J}$. En posant $(q_i)_{i\in I}^{(n_i)_{i\in I}} = \prod_{i\in I} q_i^{n_i}$, on a :

$$(2) \qquad P\left(\lambda_j = n_j, j \in J | \lambda_i = n_i, i \in I\right) = \frac{\left[(q_j)_{j\in J}^{(n_j)_{j\in J}} (q_i)_{i\in I}^{(n_i)_{i\in I}}\right] D\left(q_i, q_j\right)}{\left[(q_i)_{i\in I}^{(n_i)_{i\in I}}\right] D\left(q_i\right)}.$$

Dans la plupart des cas, le calcul de coefficients de séries à un grand nombre de variables est trop complexe pour être mené à bien, et ne donne pas de résultats "lisibles". On peut alors se contenter d'obtenir, pour un paramètre donné, sa moyenne, voire sa variance, et pour deux paramètres, leur covariance.

Les opérations essentielles pour le calcul de paramètres moyens sont l'extraction de coefficients d'une série génératrice, et la différentiation suivant une variable.

4.2.2 Différentiation

Soit $F = F(q_1, \ldots, q_k)$ la série génératrice, suivant k paramètres $\lambda_1, \ldots, \lambda_k$, d'un ensemble d'objets noté également F.

En dérivant formellement, par rapport à la variable q_i, puis en multipliant par q_i, la définition de la série

$$F(q_1, \ldots, q_k) = \sum_{w\in F} q_1^{\lambda_1(w)} \cdots q_k^{\lambda_k(w)},$$

on obtient

$$(3) \qquad q_i \frac{\partial}{\partial q_i} F(q_1, \ldots, q_k) = \sum_{w\in F} \lambda_i(w) q_1^{\lambda_1(w)} \cdots q_k^{\lambda_k(w)}.$$

Cette nouvelle série peut être interprétée comme la série génératrice de F avec comme valuation, pour chaque objet w, $v(w) = \lambda_i(w) v_0(w)$, si $v_0(w)$ est la valuation initiale. C'est également la série génératrice, suivant les mêmes paramètres, d'un nouvel ensemble d'objets F', obtenu en remplaçant chaque objet w par $\lambda_i(w)$ copies de lui-même. Si le paramètre

λ_i compte des éléments distincts de w, cela correspond, pour obtenir F', à distinguer de toutes les façons possibles l'un de ces éléments : ainsi, si par exemple F est l'ensemble des chemins de Dyck, et λ_i le paramètre "nombre de pics", F' est l'ensemble des chemins de Dyck (non vides) dont un pic a été distingué.

Définition 4.1. Soient $q_1, \dots, q_n$ des variables formelles. L'opérateur Δ_{q_i} est défini sur l'algèbre de séries formelles $\mathbb{Q}[[q_1, \dots, q_n]]$ par

$$\Delta_{q_i} F(q_1, \dots, q_n) = q_i \frac{\partial F}{\partial q_i}(q_1, \dots, q_n) \tag{4}$$

Notation 4.2. Lorsqu'il sera nécessaire de composer les opérateurs Δ, nous noterons de manière générale,

$$\Delta_{q_{i_1} \dots q_{i_k}} = \Delta_{q_{i_1}} \circ \dots \circ \Delta_{q_{i_k}}.$$

Les opérateurs Δ_{q_i} commutent entre eux, et sont des dérivations sur l'algèbre de séries formelles $\mathbb{Q}[[q_1, \dots, q_n]]$. Leur interprétation sur les séries génératrices est donnée par la proposition suivante :

Proposition 4.3. *Lorsque $F(q_1, \dots, q_k)$ est la série génératrice de F suivant la valuation $v_0(w) = q_1^{\lambda_1(w)} \dots q_k^{\lambda_k(w)}$, la série $\Delta_{q_{i_1} \dots q_{i_m}} F$ est la série génératrice de F avec valuation*

$$v(w) = \left(\prod_{j=1}^{m} \lambda_{i_j}(w) \right) v_0(w).$$

Preuve. La proposition est immédiatement prouvée par récurrence sur m, chaque application de Δ_{q_i} multipliant la valuation de chaque objet w par $\lambda_i(w)$. $\square$

La proposition 4.3 permet de comprendre pourquoi il est avantageux de considérer comme opérateurs, $\Delta_{q_i} = q_i \frac{\partial}{\partial q_i}$ plutôt que $\frac{\partial}{\partial q_i}$: l'utilisation de ces derniers compliquerait l'interprétation des composées de tels opérateurs. Avec deux variables q_1 et q_2, comptant respectivement des paramètres λ_1 et λ_2,

$$\begin{aligned}
[q_1^{n_1} q_2^{n_2}] \left(\Delta_{q_2}\right)^k F(q_1, q_2) &= n_2^k \, [q_1^{n_1} q_2^{n_2}] \, F(q_1, q_2), \\
[q_1^{n_1} q_2^{n_2}] \left(\frac{\partial}{\partial q_2}\right)^k F(q_1, q_2) &= (n_2 + 1) \dots (n_2 + k) \left[q_1^{n_1} q_2^{n_2 + k}\right] F(q_1, q_2).
\end{aligned}$$

Ainsi, la première forme, plus simple, est préférable.

Cette proposition se traduit, en termes de coefficients de séries, de la manière suivante :

Proposition 4.4. *Soit $F(q_1, \dots, q_n)$ la série génératrice de F suivant les paramètres $\lambda_1, \dots, \lambda_n$. Alors, pour tout paramètre $\lambda = \lambda_{i_1}^{\alpha_1} \dots \lambda_k^{\alpha_{i_k}}$, la moyenne de λ sur $F_{\lambda_1=m}$ est*

$$E(\lambda(w)|\lambda_1(w) = m) = \frac{[q_1^m]\Delta_{i_1}^{\alpha_1} \dots \Delta_{i_k}^{\alpha_k} F(q_1, 1, \dots, 1)}{[q_1^m]F(q_1, 1, \dots, 1)}. \tag{5}$$

Preuve. Il est clair, d'après la proposition (4.3), que l'on a

$$[q_1^m]\Delta_{i_1}^{\alpha_1} \dots \Delta_{i_k}^{\alpha_k} F(q_1, 1, \dots, 1) = \sum_{w \in F_{\lambda_1=m}} \lambda(w). \tag{6}$$

Puisque le coefficient $[q_1^m]F(q_1, 1, \dots, 1)$ est exactement le cardinal de $F_{\lambda_1=m}$, nous obtenons directement

$$\frac{[q_1^m]\Delta_{i_1}^{\alpha_1} \dots \Delta_{i_k}^{\alpha_k} F(q_1, 1, \dots, 1)}{[q_1^m]F(q_1, 1, \dots, 1)} = \sum_{w \in F_{\lambda_1=m}} \lambda(w) P(w|\lambda_1(w) = m),$$

qui donne bien l'espérance du paramètre λ. □

Dans la pratique, il est fréquent de fixer à 1 toutes les variables formelles d'énumération, sauf la première, qui compte la taille des objets. Nous noterons π la projection de l'algèbre de séries formelles $\mathbb{Q}[[q_1, \dots, q_k]]$ dans $\mathbb{Q}[[q_1]]$[1]; pour chaque opérateur Δ obtenu en composant des opérateurs Δ_{q_i}, nous noterons $\Delta' = \pi \circ \Delta$. Dans le cadre des Q-grammaires, les relations que nous obtiendrons en associant opérateurs Δ et substitutions de variables se simplifieront nettement en remplaçant l'opérateur Δ par Δ'.

Définition 4.5. Lorsque $U(q_1, \dots, q_n)$ est la série génératrice d'un langage L suivant des parmètres $\lambda_1, \dots, \lambda_n$, la série $\Delta'_i(U)$ est appelée *série de moments* du langage L suivant le paramètre λ_i.

4.2.3 Calculs asymptotiques

Il est particulièrement instructif de s'intéresser au comportement de la valeur moyenne d'un paramètre pour les objets de taille n, lorsque n tend vers $+\infty$; c'est le principe de l'analyse asymptotique de paramètres.

En général, on cherche pour les coefficients a_n d'une série génératrice, un développement asymptotique dans l'échelle des fonctions

$$\mathcal{H}_{\mu,\theta}(n) = \mu^n n^{-\theta}. \tag{7}$$

1. π n'est pas définie pour *toutes* les séries formelles, mais seulement sur une sous-algèbre qui contient les séries génératrices pour lesquelles q_1 compte une taille.

Dans la pratique, on cherche un développement de la forme

$$a_n = \mu^n P(1/n) + O(\mu^n n^{-\theta}), \tag{8}$$

où $P(x) = C_1 x^{\theta_1} + \cdots + C_k x^{\theta_k}$, avec $\theta_1 < \ldots < \theta_k < \theta$ (P n'est pas forcément un polynôme, les exposants θ_i n'étant généralement pas entiers).

La règle de Cauchy-Hadamard permet d'affirmer que, lorsqu'un tel développement existe, $1/\mu$ est le rayon de convergence de la série génératrice. Cette série ayant des coefficients positifs, $1/\mu$ en est alors une singularité dominante. De plus, si la série est algébrique, μ est forcément un nombre algébrique, et les exposants θ_i sont rationnels [57, 40].

Une variante relativement fréquente de cette forme consiste à séparer les coefficients suivant différentes progressions arithmétiques. Ainsi, lorsque la série $f(x)$ peut s'écrire $f(x) = g(x^2)$ pour une autre série g, seuls les coefficients d'indice pair de f peuvent vérifier une relation de la forme (8); les coefficients d'indice impair sont évidemment nuls. Le plus souvent, toutefois, de telles propriétés sont détectées à l'avance, et la série génératrice considérée est $g(x)$ plutôt que $f(x)$. Ainsi, la série génératrice habituellement considérée pour les mots de Dyck est $f(x) = \sum_n C_n x^n$, où le n-ième nombre de Catalan C_n est le nombre de mots de Dyck de longueur $2n$.

En utilisant la proposition (4.3), le calcul asymptotique de la valeur moyenne d'un paramètre sur les objets de taille n, lorsque n tend vers $+\infty$, se ramène en fait au calcul de développements asymptotiques pour les coefficients de deux séries : la série génératrice $D(q_1)$ suivant la taille, et une série $(\Delta'_{q_i} D)(q_1)$. Le quotient de ces deux développements asymptotiques, donnera le comportement de la valeur moyenne lorsque n tend vers $+\infty$.

Un avantage de l'échelle de développements asymptotiques de la forme (8) est qu'un quotient de deux telles expressions se présente sous une forme semblable. Dans le cas de calcul de valeurs moyennes de paramètres, ceci permet d'obtenir des expressions asymptotiques simples pour ces valeurs moyennes.

Sous certaines conditions, le comportement asymptotique des coefficients d'une série à une seule variable, peut être décrit assez précisément en examinant les singularités de la fonction analytique définie par cette série (voir par exemple [45]).

La recherche d'expressions asymptotiques pour les coefficients d'une série génératrice peut généralement être résumée ainsi :

- Déterminer les singularités dominantes (de plus petit module) de la série. Lorsque les coefficients de la série sont positifs, l'une au moins de ces singularités est réelle positive; très souvent, on obtient une singularité dominante unique ρ.

- Déterminer un développement asymptotique de la série au voisinage de la singularité dominante; pour des séries algébriques, on obtient une expression de la forme $F(x) = F_0 + C.(\rho - x)^\alpha + o\left((\rho - x)^\alpha\right)$, où α est un nombre rationnel.
- A partir d'une telle expression, il est généralement possible d'écrire automatiquement le développement de a_n.

Ce processus est automatisé dans le logiciel $\Lambda_\Upsilon\Omega$ (`luo`) [42, 43], conçu pour l'analyse en moyenne de *structures décomposables*, dont les langages algébriques sont un cas particulier. Ce logiciel permet donc, dans le cadre des paramètres Q-comptables, de traiter les paramètres de rang 1.

Recherche de singularités : développements de Puiseux

Nous donnons ici un aperçu de la méthode du *polygone de Newton* utilisée pour obtenir le *développement de Puiseux* d'une série algébrique d'une variable. On trouvera une description plus complète dans l'ouvrage de Dieudonné [33].

Nous nous plaçons dans le cas où la série $F(x)$ dont nous recherchons les singularités est algébrique, et ne dépend que d'une seule variable x. Elle est donc solution d'une équation

$$P(x, F(x)) = 0 \tag{9}$$

où $P(x, y)$ est un polynôme.

Le théorème des fonctions implicites prévoit qu'au voisinage de tout point (x_0, y_0) tel que $P(x_0, y_0) = 0$, l'équation (9) admet une solution analytique dès que $\frac{\partial}{\partial y} P(x_0, y_0) \neq 0$. Par conséquent, pour que la série F ait une singularité en x_0, il faut que (x_0, y_0) soit solution de

$$\left\{ \begin{array}{lcl} 0 & = & P(x, y) \\ 0 & = & \frac{\partial P}{\partial y}(x, y) \end{array} \right. \tag{10}$$

Pour rechercher la (ou les) singularités dominantes, il faut chercher, parmi les solutions de (10), celles qui ont une coordonnée x de module minimal *et* qui correspondent à une singularité d'une branche analytique à l'origine de la solution de (9). Cette partie du problème ne peut être résolue de manière automatique dans le cas général; toutefois, dans le cas de séries génératrices (qui ne font intervenir que des coefficients positifs ou nuls), l'une au moins des singularités dominantes est assurée d'être un nombre réel positif, ce qui peut aider à éliminer les “fausses” solutions.

Par un changement de variables de la forme

$$\left\{\begin{array}{rcl} F & = & y_0 - f, \\ x & = & x_0 - h, \end{array}\right.$$

on se ramène à chercher une solution au voisinage de $(0,0)$ de l'équation

$$P(x_0 - h, y_0 - f) = 0. \tag{11}$$

Le fait que la solution recherchée soit une série génératrice permet de ne prendre en compte que des solutions pour $h \geq 0$ et $f \geq 0$, car la fonction $F(x)$ est forcément croissante sur $[0, x_0]$.

L'équation (11) se développe sous la forme

$$0 = \sum_{i=1}^{k} c_i h^{\alpha_i} f^{\beta_i}, \tag{12}$$

où les coefficients c_i sont non nuls, et les exposants α_i et β_i, entiers positifs, ne peuvent être nuls simultanément. En factorisant cette équation par $h^{\alpha_0} f^{\beta_0}$, on peut également supposer que l'un au moins des exposants α_i est nul, ainsi que l'un des exposants β_i.

Plaçons dans le plan les points $(A_i)_{1 \leq i \leq k}$ de coordonnées respectives (α_i, β_i). Supposons que les indices sont ordonnés de telle sorte que A_1 est le point le plus bas sur l'axe $\alpha = 0$ ($\alpha_1 = 0$ et $\beta_1 \leq \beta_i$ pour tout i tel que $\alpha_i = 0$), A_p est le point le plus à gauche sur l'axe $\beta = 0$ ($\beta_p = 0$ et $\alpha_p \leq \alpha_i$ pour tout i tel que $\beta_i = 0$), et le parcours dans le sens trigonométrique de l'enveloppe convexe du nuage de points $(A_i)_{1 \leq i \leq k}$, entre A_1 et A_p, passe par les points $A_2, \ldots, A_{p-1}$. Cette ligne polygonale $(A_1, \ldots, A_p)$ est appelée *polygone de Newton* de l'équation (11).

Pour chaque segment de ce polygone de Newton, on obtient, en ne conservant de (12) que les termes qui correspondent à des points situés sur ce segment (le plus souvent, il n'y a que 2 tels points pour chaque segment), une équation approchée dont la résolution donne le premier terme d'un développement asymptotique d'une branche de la solution de (11), à condition que ce développement soit compatible avec les hypothèses sur le signe de f et h. La forme générale de ce développement asymptotique pour le segment $[A_i, A_j]$, est

$$f = C.h^{\rho} + o\left(h^{\rho}\right), \tag{13}$$

où ρ est le nombre rationnel solution de l'équation

$$\alpha_i + \beta_i \rho = \alpha_j + \beta_j.\rho.$$

En théorie, le polygone de Newton peut donner un assez grand nombre de branches. Dans la pratique, il est assez fréquent que l'équation (12) comporte un terme en $h^1 f^0$ et un terme en $h^0 f^2$, auquel cas $A_1 = (0, 2)$, $A_2 = (1, 0)$, et le polygone de Newton se réduit au segment $[A_1 A_2]$. On a alors une branche unique avec comportement $f = C.h^{1/2} + o(h^{1/2})$.

4.3 Un exemple de calcul de moyenne

Nous donnons ici un exemple de calcul de valeur moyenne de paramètre, en reprenant l'étude de l'aire des chemins de Dyck. Bien que la série génératrice des chemins de Dyck soit un exemple extrêmement classique de série qu'il est possible de calculer explicitement et dont les coefficients s'expriment simplement, tout le travail asymptotique peut être fait sans connaître d'expressions exactes.

Les calculs qui suivent ne présentent aucune difficulté technique. Toutefois, la forme des résultats peut être généralisée à n'importe quel système de Q-équations données par une Q-grammaire, comme nous le verrons par la suite.

La série génératrice des chemins de Dyck comptés suivant la demi-longueur (par x) et l'aire (par q) vérifie l'équation :

$$D(x, q) = 1 + xqD(xq^2, q)D(x, q) \tag{14}$$

En remplaçant q par 1 dans cette équation, on retrouve l'équation algébrique satisfaite par la série génératrice suivant la demie longueur seule :

$$D(x) = 1 + xD(x)^2. \tag{15}$$

Les coefficients de la série génératrice suivant la demi-longueur seule sont bien connus : il s'agit des nombres de Catalan $C_n = \frac{1}{n+1}\binom{2n}{n}$.

L'équation algébrique (15) s'écrit sous la forme implicite

$$P(x, D(x)) = 0,$$

avec $P(x, y) = 1 - y + xy^2$. Les singularités des fonctions solutions sont donc solutions du système

$$0 = P(x, y) = \frac{\partial P}{\partial y}(x, y).$$

Dans notre cas, ce système s'écrit

$$0 = 1 - y + xy^2 = -1 + 2xy,$$

et a pour solution unique, $(x_0, y_0) = (1/4, 2)$. La branche de la solution de (15) qui nous intéresse doit être analytique au voisinage de 0, et être croissante sur $[0, 1/4]$; par conséquent, nous pouvons écrire $D(1/4 - h) = 2 - d(h)$, avec $h \geq 0$ et $d(h) \geq 0$: le développement asymptotique de $D(x)$ au voisinage de $1/4$ nous sera donné par celui de $d(h)$ au voisinage de 0.

En écrivant $P(1/4 - h, 2 - d)$, nous obtenons à partir de (15),

$$0 = \frac{1}{4}d^2 - 4h + 4hd - hd^2. \tag{16}$$

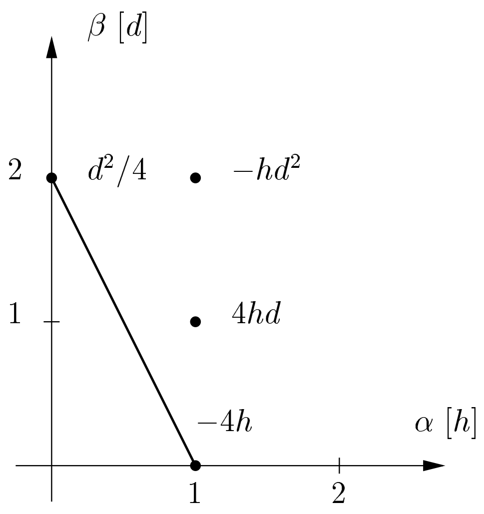


FIG. 4.1: *Polygone de Newton de l'équation* $d^2/4 - 4h + 4hd - hd^2$

Le polygone de Newton de l'équation (16) se réduit au segment $\alpha + \beta/2 = 1, \alpha \geq 0, \beta \geq 0$ (voir figure 4.1), ce qui nous donne immédiatement le développement asymptotique de $d(h)$:

$$d = 4h^{1/2} + o\left(h^{1/2}\right).$$

Le développement asymptotique de la série $D(x)$ au voisinage de sa singularité dominante[2] est par conséquent,

$$D(x) = 2 - 4\sqrt{\frac{1}{4} - x} + o\left(\sqrt{\frac{1}{4} - x}\right) = 2 - 2\sqrt{1 - 4x} + o\left(\sqrt{1 - 4x}\right).$$

2. Bien entendu, un tel développement eût été plus simple à obtenir à partir de la formule close $D(x) = \left(1 - \sqrt{1 - 4x}\right)/2x$.

De cette expression, on déduit immédiatement une forme asymptotique des coefficients de la série génératrice $D(x)$:

$$C_n \sim \frac{4^n n^{-3/2}}{\sqrt{\pi}}.$$

Pour obtenir l'aire moyenne des chemins de Dyck de longueur $2n$, il nous faut reprendre l'équation (14), et dériver par rapport à q. En notant $D_x(x) = \partial D/\partial x(x,q)$ et $D_q(x) = \partial D/\partial q(x,q)$ (la deuxième variable q est implicite), nous obtenons :

$$D_q(x) = xD(xq^2)D(x) + xqD(x)\left(D_q(xq^2) + 2xqD_x(xq^2)\right) + xqD(xq^2)D_q(x). \tag{17}$$

Lorsque $q = 1$, $D_x(x,q)$ devient la dérivée de la série génératrice à une seule variable $D(x) = D(x,1)$; en dérivant (15), il vient :

$$\begin{aligned} D'(x) &= D^2(x) + 2xD(x)D'(x) \\ &= \frac{D^2(x)}{1-2xD(x)}. \end{aligned}$$

Par conséquent, (17) devient pour $q = 1$,

$$\begin{aligned} D_q(x,1) &= xD^2(x,1) + xD(x,1)\left(D_q(x,1) + 2x\frac{D^2(x,1)}{1-2xD(x,1)}\right) \\ &\quad + xD(x,1)D_q(x,1) \\ &= \frac{xD^2(x,1)}{1-2xD(x,1)} + \frac{2x^2D^3(x,1)}{(1-2xD(x,1))^2} \\ &= \frac{xD^2(x,1)}{(1-2xD(x,1))^2}. \end{aligned}$$

Remarquons que $D_q(x,1)$, tout comme $D_x(x,1)$, peut s'écrire comme une fraction rationnelle en x et $D(x,1)$. Ce n'est pas le cas de la série à deux variables $D_q(x,q)$. En connaissant le développement asymptotique de $D(x,1)$ au voisinage de $x = 1/4$, on en déduit donc celui de $D_q(x,1)$:

$$D_q(x,1) \sim \frac{1}{1-4x}. \tag{18}$$

On en déduit immédiatement une expression asymptotique pour les coefficients de $D_q(x,1)$ (à savoir, $[x^n]D_q(x) \sim 4^n$), et surtout un équivalent de l'aire moyenne $\overline{A}_n$ des chemins de Dyck de longueur $2n$:

$$\overline{A}_n \sim \sqrt{\pi}n^{3/2}. \tag{19}$$

Bien entendu, en acceptant d'utiliser la forme exacte de $D(x,1)$, et en posant $J(x) = 1-2xD(x,1) = \sqrt{1-4x}$, on peut aisément exprimer $D_x(x,1)$ et $D_q(x,1)$ comme fractions rationnelles en J :

$$D_x(x,1) = \frac{1-J}{J(1+J)}, \tag{20}$$

$$D_q(x,1) = \frac{1-J}{J^2(1+J)}, \tag{21}$$

ce qui permet d'obtenir une expression exacte pour les coefficients de $D_q(x,1)$:

$$D_q(x,1) = \sum_{n\geq 0} (4^n - (2n+1)C_n)\, x^n.$$

On retrouve ici le résultat de Chottin et Cori [20] (ajouter $(2n+1)C_n$ pour obtenir 4^n, correspondrait à surélever de 1 chacun des $2n+1$ sommets des chemins, ce qui revient à compter les ordonnées à partir de 1 et non 0).

Nous pouvons pousser plus loin les calculs portant sur l'aire, afin d'obtenir également la *variance* de l'aire des chemins de Dyck de longueur $2n$. En utilisant les opérateurs Δ et Δ' définis en 4.2.2, le moment d'ordre 2 de l'aire est donné par les coefficients de $(\Delta'_{qq}D)(x)$. L'équation (17) peut se réécrire en :

(22)

$$(\Delta_q D)(x,q) = xq(q+1)D(x,q)D(xq^2,q) + xqD(x,q)\left((\Delta_q D)(xq^2,q) + (\Delta_x D)(xq^2,q)\right)$$

Appliquer Δ_q à cette equation fait apparaître les séries $\Delta_{xx}D(x)$, $\Delta_{xx}D(xq^2)$, $\Delta_{xq}D(x)$ et $\Delta_{xq}D(xq^2)$. En fixant $q = 1$, nous n'avons plus que $\Delta'_{xx}D(x)$ et $\Delta'_{xq}D(x)$. Nous devons donc également calculer ces séries, de la même manière que fait précédemment pour $\Delta'_q D(x) = D_d(x)$.

Pour calculer $\Delta'_{xx}D(x)$ et $\Delta'_{xq}D(x)$, il suffit en fait de travailler avec des séries à une variable, en appliquant Δ_x aux équations donnant $\Delta'_x D(x)$ et $\Delta'_q D(x)$, respectivement. Pour obtenir $\Delta'_{qq}D(x)$, en revanche, il est indispensable de conserver la variable q et de ne la faire disparaître qu'à la dernière étape du calcul.

Une fois de plus, la deuxième variable q est implicite dans les expressions qui suivent. F désigne $F(x,q)$; $F(xq)$ désigne $F(xq,q)$, et $F(xq^2)$ représente $F(xq^2,q)$. Les équations obtenues sont :

$$\begin{aligned}
\Delta'_{xx}D(x) &= xD^2(x) + 4xD(x)\Delta'_x D(x) + 2x\left(\Delta'_x D(x)\right)^2 + 2xD(x)\Delta'_{xx}D(x)\\
&= \frac{xD^2(x) + 4xD(x)\Delta'_x D(x)}{1-2xD(x)},\\
\Delta'_{xq}D(x) &= xD^2(x) + 4xD(x)\Delta'_x D(x) + 2xD(x)\Delta'_q D(x) + 2x\Delta'_x D(x)\Delta'_q D(x)\\
&\quad +2xD\Delta'_{xq}D(x) + 2x\left(\Delta'_x D(x)\right) + 2xD(x)\Delta'_{xx}D(x)\\
&= x\frac{D(x)\left(D(x) + 4\Delta'_x D(x) + 2\Delta'_{xx}D(x)\right) + 2\Delta'_x D(x)\left(\Delta'_x D(x) + \Delta'_q D(x)\right)}{1-2xD(x)},\\
\Delta_{qq}D &= xq\left(D.D(xq^2) + \Delta_q D.D(xq^2) + D.\Delta_q D(xq^2) + 2D.\Delta_x D(xq^2)\right.\\
&\quad +\Delta_q D.D(xq^2) + \Delta_{qq}D.D(xq^2) + \Delta_q D.\Delta_q D(xq^2) + 2\Delta_q D.\Delta_x D(xq^2)\\
&\quad +D.\Delta_q D(xq^2) + \Delta_q D.\Delta_q D(xq^2) + D.\Delta_{qq}D(xq^2) + 2D.\Delta_{xq}D(xq^2)\\
&\quad \left.+2D.\Delta_x D(xq^2) + 2\Delta_q D.\Delta_x D(xq^2) + 2D.\Delta_{xq}D(xq^2) + 4D.\Delta_{xx}D(xq^2)\right).
\end{aligned}$$

La dernière équation se simplifie quelque peu lorsque l'on fixe $q=1$, et donne :

$$\Delta'_{qq}D = x\frac{D.\left(D + 4\Delta'_x D + 4\Delta'_q D + 4\Delta'_{xq}D + 4\Delta'_{xx}D\right) + 2\Delta'_q D.\left(\Delta'_q D + 2\Delta'_x D\right)}{1-2xD}.$$

Une fois de plus, en posant $J(x) = 1 - 2xD(x)$, les trois séries que nous venons de calculer s'expriment comme fractions rationnelles en J :

$$\Delta'_{xx}D = \tfrac{(1-J)(1+2J-J^2)}{2J^3(1+J)}, \tag{23}$$

$$\Delta'_{xq}D = \tfrac{(1-J)(1+J-J^2)}{J^4(1+J)}, \tag{24}$$

$$\Delta'_{qq}D = \tfrac{(1-J)(5+2J-5J^2)}{2J^5(1+J)}. \tag{25}$$

L'équation (25) nous permettrait d'obtenir une expression exacte pour les coefficients de $\Delta'_{qq}D(x)$, et, partant, pour le moment d'ordre 2 de l'aire des chemins de Dyck de longueur $2n$. Nous nous contenterons d'une version asymptotique :

$$\Delta_{qq}D(x) \sim \frac{5}{2}(1-4x)^{-5/2}, \tag{26}$$

$$[x^n]\,\Delta_{qq}D(x) \sim \frac{10}{3\sqrt{\pi}}4^n n^{3/2}. \tag{27}$$

Par conséquent, l'écart-type σ_n de l'aire des chemins de Dyck de longueur $2n$, est asymptotiquement,

$$\sigma_n \sim n^{3/2}\sqrt{\frac{10}{3} - \pi}. \tag{28}$$

L'écart-type du paramètre aire est donc du même ordre de grandeur que l'espérance du paramètre lui-même. En conséquence, ce paramètre ne peut avoir, après normalisation ($A^* = (A - \mu_n)/\sigma_n$), une loi limite gaussienne, puisque le support de A^* reste borné inférieurement.

4.4 Opérateurs Δ et substitutions de variables

La série génératrice d'un langage algébrique suivant les différents paramètres Q-comptables d'une Q-grammaire, ne nous est a priori connue que par l'intermédiaire d'un système de Q-équations faisant intervenir des substitutions de variables σ.

Lors du calcul précédent, nous avons pu voir que tout se résume à appliquer des opérateurs Δ ou Δ' à des équations faisant intervenir des substitutions de variables (dans le cas de l'aire des chemins de Dyck, la seule substitution est $\sigma_{x\leftarrow xq^2}$). L'ensemble des calculs est simplifié lorsque l'on remarque que, pour une série $F(x,q)$ quelconque,

$$\begin{aligned}
\left(\Delta_x \circ \sigma_{x\leftarrow xq^2}\right) F &= \left(\sigma_{x\leftarrow xq^2} \circ \Delta_x\right) F \\
\left(\Delta_q \circ \sigma_{x\leftarrow xq^2}\right) F &= \left(\sigma_{x\leftarrow xq^2} \circ \Delta_q\right) F + 2\left(\sigma_{x\leftarrow xq^2} \circ \Delta_x\right) F.
\end{aligned}$$

Il est donc naturel d'examiner en toute généralité quels sont les liens possibles entre les opérateurs σ et les opérateurs Δ, et les conséquences que nous pouvons en tirer sur les statistiques de paramètres Q-comptables.

Lemme 4.6. *Soient $q_1, \ldots, q_n$ des variables formelles. Notons $\sigma_{i,j} = \sigma_{q_i \leftarrow q_i q_j}$ (pour $1 \leq i < j \leq n$). On a alors :*

- *si $k \neq j$, Δ_{q_k} et $\sigma_{i,j}$ commutent;*
- *$\Delta_j \sigma_{i,j} = \sigma_{i,j}\Delta_j + \sigma_{i,j}\Delta_{q_i}$*

Preuve. Il est clair que, pour n'importe quelle série formelle $U = U(q_1, \ldots, q_n)$, lorsque k est distinct de i et de j, $\Delta_{q_k}\sigma_{i,j}U = \sigma_{i,j}\Delta_{q_k}U$. Il reste à examiner les cas $k = i$ et $k = j$.

Lorsque $k = i$, le calcul est très simple :

$$\begin{aligned}
\frac{\partial}{\partial q_i}\sigma_{i,j}U &= \frac{\partial}{\partial q_i}U(q_1, \ldots, q_{i-1}, q_i q_j, q_{i+1}, \ldots, q_n) \\
&= q_j \frac{\partial U}{\partial q_i}(q_1, \ldots, q_{i-1}, q_i q_j, q_{i+1}, \ldots, q_n) \\
\Delta_{q_i}\sigma_{i,j}U &= q_i \frac{\partial}{\partial q_i}\sigma_{i,i}U \\
&= q_i q_j \frac{\partial U}{\partial q_i}(q_1, \ldots, q_{i-1}, q_i q_j, q_{i+1}, \ldots, q_n) \\
&= \sigma_{i,i}\Delta_{q_i}U
\end{aligned}$$

Dans le cas où $k = j$, on obtient :

$$\begin{aligned}
\frac{\partial}{\partial q_j}\sigma_{i,j}U &= \frac{\partial}{\partial q_j}U(q_1,\dots,q_{i-1},q_iqj,q_{i+1},\dots,q_n)\\
&= \frac{\partial U}{\partial q_j}(q_1,\dots,q_{i-1},q_iq_j,q_{i+1},\dots,q_n)+q_i\frac{\partial U}{\partial q_i}(q_1,\dots,q_{i-1},q_iq_j,q_{i+1},\dots,q_n)\\
\Delta_{q_j}\sigma_{i,j}U &= q_j\frac{\partial U}{\partial q_j}(q_1,\dots,q_{i-1},q_iqj,q_{i+1},\dots,q_n)+q_iq_j\frac{\partial U}{\partial q_i}(q_1,\dots,q_iq_j,\dots,q_n)\\
&= \sigma_{i,j}\Delta_{q_i}U+\sigma_{i,j}\Delta_{q_j}U
\end{aligned}$$

□

Le résultat du lemme (4.6) peut également s'écrire sous forme matricielle, à condition de définir quelques notations.

Notons $\overrightarrow{\Delta}$ le vecteur-ligne d'opérateurs $(\Delta_{q_1},\dots,\Delta_{q_n})$. Le produit d'un vecteur-ligne avec une matrice de substitution de variables se fait de la manière usuelle, de telle sorte que

$$(\Delta_{q_1},\Delta_{q_2},\Delta_{q_3})\begin{pmatrix}1&1&1\\0&1&2\\0&0&1\end{pmatrix}=(\Delta_{q_1},\Delta_{q_1}+\Delta_{q_2},\Delta_{q_1}+2\Delta_{q_2}+\Delta_{q_3}).$$

Enfin, si F est une série formelle, σ une substitution de variables, $\overrightarrow{\Delta}=(\Delta_1,\dots,\Delta_n)$ un vecteur-ligne d'opérateurs, et $(U_1,\dots,U_n)$ un vecteur-ligne de séries formelles, les calculs suivants se distribuent composante par composante :

$$\begin{aligned}
\overrightarrow{\Delta}(U) &= (\Delta_1(U),\dots,\Delta_n(U));\\
\sigma(U_1,\dots,U_n) &= (\sigma(U_1),\dots,\sigma(U_n)).
\end{aligned}$$

Alors, le lemme (4.6) peut s'exprimer de la manière suivante :

$$\overrightarrow{\Delta}\,(\sigma_{ij}(U))=\left(\sigma_{ij}\circ\overrightarrow{\Delta}\right)(U).$$

Cette formulation est en fait valable quelle que soit la substitution de variables σ. En effet, toute matrice triangulaire supérieure, à coefficients entiers positifs, ne présentant que des 1 sur sa diagonale, peut s'écrire comme produit de matrices de la forme $M_{i,j}$; ou, ce qui revient au même, toute substitution de variables peut s'écrire comme composée de substitutions de la forme $\sigma_{i,j}$. Le lemme (4.6) s'étend alors, par récurrence, à n'importe quelle substitution de variables :

Proposition 4.7. *Soit σ une substitution de variables, représentée par la matrice M. Alors*

(29) $$\overrightarrow{\Delta}.(\sigma U)=\sigma\left(\left(\overrightarrow{\Delta}.M\right)(U)\right)$$

Il est peut-être plus simple d'exprimer ce résultat en fonction des coefficients de la matrice : si $M = (a_{i,j})_{1\leq i,j\leq n}$, l'équation (29) s'écrit

$$\Delta_{q_j}(\sigma U) = \sigma\left(\sum_{i=1}^{j} a_{i,j}\Delta_{q_i}U\right). \tag{30}$$

Pour obtenir des résultats statistiques, nous devons restreindre nos séries génératrices à une seule variable, ce qui se fait par l'intermédiaire de l'opérateur de projection π. Nous devons donc donner de la proposition (4.7) une version ne concernant que les opérateurs Δ'.

Remarquons que, pour toute substitution de variables σ, on a $\pi\circ\sigma = \pi$. Cette remarque nous donne l'équivalent de la proposition (4.7) pour les opérateurs Δ' :

Proposition 4.8. *Soit σ une substitution de variables, représentée par la matrice M. Notons, pour une série formelle $F = F(q_1, \ldots, q_n)$, $\overrightarrow{\Delta'}$ le vecteur-ligne $(\Delta'_{q_1}, \ldots, \Delta'_{q_n})$. Alors*

$$\overrightarrow{\Delta'}(\sigma(F)) = \left(\overrightarrow{\Delta'}.M\right)(F) \tag{31}$$

Preuve. L'équation (31) s'obtient directement à partir de (29) en appliquant π de part et d'autre. □

Exemple 4.9. Considérons une série formelle $F(x, y, z)$, et la substitution de variables $\sigma = \sigma_{x\leftarrow xy, y\leftarrow yz^2}$, représentée par la matrice

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

La proposition (4.8) implique :

$$\begin{aligned}
\left(\Delta'_z\circ\sigma\right)(F) &= \Delta'_z(F) + 2\Delta'_y(F) \\
\left(\Delta'_y\circ\sigma\right)(F) &= \Delta'_y(F) + \Delta'_x(F) \\
\left(\Delta'_x\circ\sigma\right)(F) &= \Delta'_x(F) \\
&= x\frac{d}{dx}(\pi(F))
\end{aligned}$$

Remarque. La relation $\pi\circ\sigma = \pi$, qui permet d'obtenir la proposition (4.8), est également vraie si π n'est pas la projection dans $\mathbb{Q}[[q_1]]$, mais dans $\mathbb{Q}[[q_1, \ldots, q_{n'}]]$, à condition que les substitutions σ considérées, lorsqu'elles sont réduites aux variables $q_1, \dots, q_{n'}$, soient l'identité. Dans le cadre des substitutions apparaissant dans le système de Q-équations d'une Q-grammaire, c'est le cas lorsque les variables formelles $q_1, \dots, q_{n'}$ comptent toutes des paramètres de rang 1.

Chaque fois qu'une telle relation est vraie, la proposition (4.8) l'est aussi.

La proposition (4.8) exprime $\Delta'_{x_j}\sigma$ comme combinaison linéaire des Δ'_{x_i} $(i \leq j)$, les coefficients étant ceux de la j-ème colonne de la matrice M_σ. Ainsi, le coefficient de Δ'_{x_j} est toujours 1.

Dans la pratique, la proposition (4.8), plus simple que (4.7), sera la plus utile; la proposition (4.7) est toutefois indispensable lorsqu'il s'agit de différencier plus d'une fois (pour un calcul de moment d'ordre 2 par exemple).

4.5 Moyennes de paramètres Q-comptables

Nous en venons maintenant à l'évaluation des valeurs moyennes de paramètres Q-comptables.

4.5.1 Cas général

Supposons donnée une Q-grammaire G, d'axiome D_1, à m symboles $D_1, \dots, D_m$. Cette Q-grammaire nous fournit un système de m Q-équations portant sur les séries génératrices $D_i(q_1, \dots, q_n)$ des différents langages engendrés par la grammaire, suivant les n paramètres Q-comptables $\lambda_1, \dots, \lambda_n$.

Les n' premiers paramètres $\lambda_1, \dots, \lambda_{n'}$ sont supposés être de rang 1, et π désignera ici la projection de $\mathbb{Q}[[q_1, \dots, q_n]]$ dans $\mathbb{Q}[[q_1, \dots, q_{n'}]]$. Ainsi, pour chaque symbole D_j, $\pi(D_j)$ est la série génératrice (algébrique) du langage $L_G(D_j)$ suivant les n' paramètres de rang 1.

Pour chaque mot $w \in D_j$, le n'-uplet $(\lambda_1(w), \dots, \lambda_{n'}(w))$ sera appelé *composition* de w; cette composition nous servira de taille. Il est en effet fréquent que chacun des paramètres de rang 1 compte une lettre de l'alphabet.

Pour obtenir la valeur moyenne du paramètre λ_i sur les mots de composition $L = (l_1, \dots, l_{n'})$, il nous faut comparer les coefficients de $q_1^{l_1} \dots q_{n'}^{l_{n'}}$ dans les séries $\pi(D_1)$ et $\Delta'_{q_i}(D_1)$. Nous allons donc nous intéresser de près à $\Delta'_{q_i}(D_j)$.

Les m Q-équations peuvent, d'après le théorème (2.36), être présentées sous la forme

$$0 = \tilde{P}\left((q_j)_{1 \leq j \leq n}, (D_j)_{1 \leq j \leq m}, (\sigma_k(D_j))_{1 \leq k \leq p, 1 \leq j \leq m}\right). \tag{32}$$

Rappelons que, si chaque $\sigma_k(D_j)$ est remplacé par D_j, et si chaque variable qui n'est pas une des lettres de l'alphabet prend la valeur 1, l'équation redonne l'équation algébrique fournie par la grammaire, que nous écrivons sous la forme

$$0 = P\left((q_j)_{1\leq j\leq n'}, (D_j)_{1\leq j\leq m}\right). \tag{33}$$

En appliquant $\Delta_{q_i} = q_i\frac{\partial}{\partial q_i}$ à l'équation (32), il vient

$$0 = q_i\frac{\partial\tilde{P}}{\partial q_i} + \sum_{j=1}^{m}\Delta_{q_i}(D_j).\frac{\partial\tilde{P}}{\partial D_j} + \sum_{j=1}^{m}\sum_{k=1}^{p}\Delta_{q_i}\left(\sigma_k(D_j)\right)\frac{\partial\tilde{P}}{\partial\sigma_k(D_j)}. \tag{34}$$

En appliquant maintenant la projection π, cette équation devient

$$0 = \pi\left(q_i\frac{\partial\tilde{P}}{\partial q_i}\right) + \sum_{j=1}^{m}\Delta'_{q_i}(D_j).\pi\left(\frac{\partial\tilde{P}}{\partial D_j}\right) + \sum_{j=1}^{m}\sum_{k=1}^{p}\Delta'_{q_i}\left(\sigma_k(D_j)\right)\pi\left(\frac{\partial\tilde{P}}{\partial\sigma_k(D_j)}\right). \tag{35}$$

Ici, $\pi(P)$ doit être interprété de la manière suivante : chaque variable q_j avec $j > n'$ est remplacée par 1, et chaque variable $\sigma_k(D_j)$ est remplacée par D_j – ce qui correspond bien à l'idée que chaque variable q_i (pour $i > n'$) vaut 1. Cette projection transforme les polynômes $\tilde{P}_i$ du Q-système, en les polynômes P_i du système algébrique de départ. On a donc

$$\pi\left(\frac{\partial\tilde{P}}{\partial D_j}\right) + \sum_{k=1}^{p}\pi\left(\frac{\partial\tilde{P}}{\partial\sigma_k(D_j)}\right) = \frac{\partial P}{\partial D_j} \tag{36}$$

où $P(q_1,\dots,q_{n'},D_1,\dots,D_m)$ est le polynôme du système algébrique de départ, dont est issu (lors du passage aux Q-équations) le polynôme $\tilde{P}$.

Or, la proposition (4.8) exprime chaque terme $\Delta'_{q_i}\left(\sigma_k(D_j)\right)$ de (35) comme combinaison linéaire, à coefficients entiers positifs, de termes $\Delta'_{q_\ell}(D_j)$, avec $\ell \leq i$, le coefficient de $\Delta'_{q_i}(D_j)$ étant 1. Par conséquent, on peut réécrire (35) en regroupant d'une part les termes faisant intervenir Δ'_{q_i}, et d'autre part ceux faisant intervenir $\Delta'_{q_{i'}}$ avec $i' < i$:

$$0 = \pi\left(q_i\frac{\partial\tilde{P}}{\partial q_i}\right) + \sum_{j=1}^{m}\Delta'_{q_i}(D_j)\frac{\partial P}{\partial D_j} + \sum_{j=1}^{m}\sum_{i'=1}^{i-1}B_{j,i'}(q_1,\dots,q_{n'},D_1,\dots,D_m)\Delta'_{q_{i'}}(D_j) \tag{37}$$

Chaque $B_{j,i}$ est un polynôme de ses $n'+m$ variables, à coefficients entiers, obtenu en sommant tous les termes faisant intervenir $\Delta'_{q_{i'}}(D_j)$.

Exemple 4.10. Reprenons l'équation

$$D(x,q) = 1 + xD(x,q)D(xq,q) \tag{38}$$

correspondant à l'énumération des mots de Dyck suivant la demi-longueur (x) et l'aire de Carlitz (q). La seule substitution de variables présente est $\sigma = \sigma_{x \leftarrow xq}$, et l'équation s'écrit

$$\tilde{P}\left(x, D, \sigma(D)\right) = 0,$$

avec $\tilde{P}(x, q, y, y') = 1 - y + xyy'$. En appliquant Δ'_q, l'équation (38) devient

$$0 = -\Delta'_q D + x \Delta'_q D.D + xD.(\Delta'_x D + \Delta'_q D),$$

si bien que nous avons, pour cette équation,

$$B_{2,1}(x, D) = x.D.$$

La série $\Delta'_q(D)$ peut alors être calculée de la même manière que pour l'aire au paragraphe 4.3.

Plus précisément, l'examen des équations (37) permet de montrer la proposition suivante :

Proposition 4.11. *Soit, pour $1 \leq j \leq m$, $D_j = D_j(q_1, \ldots, q_{n'})$ la série génératrice du langage D_j suivant les seuls paramètres de rang 1 (précédemment notée $\pi(D_j)$). Notons également $P_j = P_j(q_1, \ldots, q_{n'}, D_1, \ldots, D_m)$ le polynôme correspondant à la j-ème équation du système algébrique donné par la grammaire sur laquelle est basée la Q-grammaire G.*

Soit également

$$J = \det \left(\frac{\partial P_i}{\partial D_j} \right)_{1 \leq i \leq m, 1 \leq j \leq m} \tag{39}$$

le jacobien du système algébrique.

Alors, si $q = q_{i_0}$ est une variable formelle qui compte un paramètre de rang k, la série formelle $\Delta'_q(D_j)$ est de la forme

$$\Delta'_q(D_j) = \frac{N_{i_0,j}(q_1, \ldots, q_{n'}, D_1, \ldots, D_m)}{J^k} \tag{40}$$

où $N_{i_0,j}$ est un polynôme qui dépend de la Q-grammaire.

Preuve. Pour chaque entier ℓ, $1 \leq \ell \leq m$, la Q-grammaire nous donne une équation, caractérisée par un polynôme $\tilde{P}_\ell$.

Les m équations (37) forment un système d'équations affines portant sur les séries $\Delta_{q_i}(D_j)$, dont les coefficients sont des polynômes en les variables q_j (pour $1 \leq j \leq n'$), $\pi(D_j)$ (pour $1 \leq j \leq m$). Le second membre de ces équations affines regroupe alors tous les termes faisant apparaître $\Delta'_{q_{i'}}(D_j)$ (pour $1 \leq j \leq m$, et $q_{i'}$ comptant un paramètre $\lambda_{i'}$ dont le rang est strictement inférieur à celui de λ_i).

Il est alors remarquable que le coefficient de $\Delta_{q_i}(D_j)$ dans chaque équation, $\partial P_\ell / \partial D_j$, ne dépende ni de i, ni du système de Q-équations ou des substitutions σ_k, mais seulement de j et du polynôme P_ℓ, qui est donné par la grammaire algébrique sous-jacente à la Q-grammaire G. En d'autres termes, lorsque i varie, seul le second membre du système affine change, mais pas son déterminant.

Lorsque $q = q_{i_0}$ compte un paramètre de rang 1 (c'est-à-dire, lorsque $i \leq n'$), les équations (37) s'écrivent :

$$\sum_{j=1}^{m} \frac{\partial P_i}{\partial D_j} \Delta'_q(D_j) = -q \frac{\partial P_i}{\partial q} \tag{41}$$

Le déterminant de ce système étant J, les formules de Cramer donnent directement la forme (40).

Procédons par récurrence sur le rang du paramètre compté par q : supposons (40) vraie pour tout paramètre de rang strictement inférieur à k, et soit $q = q_{i_0}$ une variable comptant un paramètre de rang k. Les m équations (37) forment encore un système dont le déterminant est J. Le second membre de ce système, fait intervenir des séries $\Delta_{q_l}(D_j)$, où q_l compte un paramètre de rang au plus $k-1$, et par conséquent, par hypothèse de récurrence, ce second membre peut se mettre sous la forme d'une fraction rationnelle des variables $q_1, \ldots, q_{n'}$ et $D_1, \ldots, D_m$, avec dénominateur J^{k-1}. Dès lors, il est clair que les formules de Cramer donnent (40). □

La proposition (4.11) a des conséquences importantes pour le calcul des singularités des séries génératrices et, partant, des valeurs moyennes de paramètres Q-comptables. Le comportement d'une série $\Delta'_q(D_j)$ au voisinage de ses singularités dominantes nous est en effet donné par la comparaison des termes dominants de J^k et de $N_{i_0,j}(D_1, \ldots, D_m)$.

Le numérateur dépend fortement du système de Q-équations considérées. Il est déterminé par les substitutions de variables employées dans le système, autant que par les polynômes $\tilde{P}_\ell$ eux-mêmes. En revanche, le dénominateur J^k ne dépend que très peu du paramètre Q-comptable considéré : J est entièrement déterminé par le système d'équations *algébriques* fournies par la grammaire sous-jacente à la Q-grammaire, et k est le rang du paramètre.

Par ailleurs, le jacobien J s'annule en chaque singularité des séries algébriques, et son développement asymptotique au voisinage de ces singularités (ou tout au moins le terme dominant) découle naturellement du développement de Puiseux cherché pour chaque série au voisinage de la singularité dominante – pour peu que les contributions liées aux différents termes dominants ne se compensent pas, auquel cas il sera nécessaire d'obtenir un développement asymptotique plus fin des différentes séries génératrices.

Dans le cas le plus fréquent, le corollaire suivant donne le comportement asymptotique de paramètres Q-comptables :

Corollaire 4.12. *Supposons que la singularité dominante unique du système soit x_0, et que l'on ait pour J un équivalent de la forme $J \sim K(x_0 - x)^{\alpha}$.*

Alors, si le numérateur $N_{i,j}(D_1, \dots, D_m)$ a une valeur finie non nulle en x_0, la forme asymptotique des coefficients de $\Delta'_{q_i}(D_j)$ est :

$$[x^n]\Delta'_{q_i}(D_j) \sim K' x_0^{-n} n^{\alpha k - 1}. \tag{42}$$

Preuve. Sous les hypothèses indiquées, la série $\Delta'_{q_i}(D_j)$ a un équivalent de la forme $K.(x_0 - x)^{-k\alpha}$, et

$$[x^n]\,(x_0 - x)^{-\beta} \sim \frac{x_0^{-n} n^{\beta-1}}{\Gamma(\beta)}.$$

□

Dans le cas le plus fréquent, $\alpha = 1/2$, et les séries génératrices $D_j(x)$ ont également un développement asymptotique de la forme $D_j(x) = D_{j,0} + K_j(x_0 - x)^{1/2} + o(x_0 - x)^{1/2}$. Nous avons alors le corollaire suivant :

Corollaire 4.13. *Sous les hypothèses du corollaire précédent, et si de plus*

$$D_j(x) = D_{j,0} + K_j(x_0 - x)^{1/2} + o(x_0 - x)^{1/2},$$

alors la valeur moyenne A_n d'un paramètre de rang k, parmi les mots de taille n, vérifie

$$A_n \sim K'' n^{(k+1)/2}.$$

Ceci donne aux paramètres de rang 1 un comportement moyen linéaire, chaque rang au-dessus de 1 multipliant l'ordre de grandeur de la valeur moyenne par $\sqrt{n}$. Ce comportement moyen est à comparer à l'ordre de grandeur maximal, qui est donné par le rang minimal du paramètre : ainsi, il semblerait qu'un paramètre Q-comptable dont l'ordre de grandeur *maximal* est n^k pour les mots de longueur n, ait tendance à avoir un ordre de grandeur *moyen* de $n^{(k+1)/2}$.

4.5.2 Décomposition en paramètres élémentaires

Le calcul de statistiques sur les paramètres Q-comptables est compatible avec leur décomposition en combinaison linéaire de paramètres élémentaires. En effet, soit λ un paramètre, compté par une variable q, et qui s'écrit comme combinaison linéaire d'autres paramètres :

$$\lambda(w) = \alpha_1\lambda_1(w) + \cdots + \alpha_k\lambda_k(w), \quad \forall w \in D_1 \cup \ldots \cup D_m.$$

Si chaque paramètre λ_i est compté par la variable q_i, on a la même relation entre les séries :

$$\Delta'_q(D_j) = \alpha_1\Delta'_{q_1}(D_j) + \cdots + \alpha_k\Delta'_{q_k}(D_j) \tag{43}$$

Par conséquent, pour obtenir les séries $\Delta'_q(D_j)$, nous pouvons nous contenter de les calculer pour des variables qui comptent des paramètres élémentaires. Ainsi, on pourra limiter le calcul des numérateurs $N_{i_0,j}$ de la proposition 4.11 au cas où la variable q_{i_0} compte un paramètre élémentaire.

4.5.3 Série de moments suivant un paramètre élémentaire

Soit G une grammaire algébrique, et soit $p = p_{R_k^{(a_k)} R_{k-1}^{(a_{k-1})} \ldots R_1}$ un paramètre élémentaire[3]. Soit, pour $\ell = 1, \ldots, k$,

$$p_\ell = p_{R_\ell^{(a_\ell)} \ldots R_1}$$

(les paramètres p_ℓ sont tous les paramètres élémentaires qu'il est indispensable d'incorporer à une Q-grammaire pour pouvoir obtenir $p = p_k$).

Soient $x_1, \ldots, x_n$ des variables comptant des paramètres de rang 1, et $q_1, \ldots, q_k$ des variables supplémentaires, q_ℓ comptant le paramètre p_ℓ (techniquement, q_1 compte également un paramètre de rang 1).

La projection π est ici la projection de $\mathbb{Q}[[x_1, \ldots, x_n, q_1, \ldots, q_k]]$ dans $\mathbb{Q}[[x_1, \ldots, x_n]]$, et les opérateurs Δ' sont définis en conséquence.

Nous nous intéressons au calcul de $\Delta'_{q_k}(D_j)$, pour chaque série génératrice D_j d'un langage engendré par la grammaire G.

Afin de pouvoir exprimer facilement cette série, nous avons besoin de quelques notations :

- i_ℓ désigne l'indice du symbole gauche de la règle R_ℓ;

3. Pour des raisons de notations, l'ordre des indices est ici inversé par rapport à notre habitude.

– pour $\ell > 1$, j_ℓ désigne l'indice du a_ℓ-ème symbole droit de la règle R_ℓ (j_1 n'est pas défini);

– pour chaque règle $R \in \mathcal{R}$, T_R désigne le terme (monôme) introduit dans l'une des équations de la grammaire G, par la règle R (il s'agit du produit commutatif des symboles et lettres présents au second membre de R);

– pour chaque règle pointée $R^{(i)} \in \mathcal{R}_p$, si D est le i-ème symbole du membre droit de R, $T'_{R,i} = T_R/D$;

– M est la matrice jacobienne $(\partial P_i/\partial D_j)$ du système d'équations de la grammaire;

– $M_{i,j}$ est la matrice M, privée de sa i-ème ligne et de sa j-ème colonne;

– $J = \det M$, $J_{i,j} = (-1)^{i+j} \det M_{i,j}$.

Notons que J et les mineurs $J_{i,j}$ ne dépendent pas de la Q-grammaire, mais seulement de la grammaire algébrique sous-jacente, de même que les monômes T_R et $T'_{R,i}$. Toutes ces expressions s'expriment comme polynômes en les variables $x_1, \dots, x_n$ et en les séries génératrices algébriques $D_1, \dots, D_m$.

Théorème 4.14. *La série $\Delta'_{q_k}(D_j)$ est donnée par :*

$$\Delta'_{q_k}(D_j) = \frac{(-1)^k}{J^k} T_{R_1} T'_{R_2,a_2} \dots T'_{R_k,a_k} J_{i_1,j_2} J_{i_2,j_3} \dots J_{i_{k-1},j_k} J_{i_k,j}. \tag{44}$$

Preuve. La preuve est par récurrence sur k.

Lorsque $k = 1$, la variable q_1 n'apparaît dans le système d'équations de la Q-grammaire que dans le monôme T_{R_1}, avec degré 1; par conséquent, le système d'équations (37) s'écrit :

$$\begin{cases} 0 = \sum_{j=1}^m \frac{\partial P_i}{\partial D_j} \Delta'_{q_1}(D_j) & (i \neq i_1); \\ -T_{R_1} = \sum_{j=1}^m \frac{\partial P_{i_1}}{\partial D_j} \Delta'_{q_1}(D_j). \end{cases}$$

La résolution de ce système donne immédiatement

$$\Delta'_{q_1}(D_j) = \frac{-T_{R_1} J_{i_1,j}}{J}.$$

Supposons maintenant la formule vraie pour k, et montrons qu'elle est également vraie pour $k+1$. La variable q_{k+1} n'apparaît dans le système d'équations de la Q-grammaire que grâce à la substitution de variables $\sigma_{q_k \leftarrow q_k q_{k+1}}$, et ce, dans un seul terme (qui donne $T_{R_{k+1}}$

dans le système algébrique sous-jacent). Comme précédemment, le système (37) s'écrit

$$\begin{aligned} 0 &= \sum_{j=1}^{m} \frac{\partial P_i}{\partial D_j} \Delta'_{q_{k+1}}(D_j) \qquad (i \neq i_{k+1}) \\ -T'_{R_{k+1},a_{k+1}} \Delta'_{q_k}(D_{j_{k+1}}) &= \sum_{j=1}^{m} \frac{\partial P_{i_{k+1}}}{\partial D_j} \Delta'_{q_{k+1}}(D_j). \end{aligned}$$

En résolvant, nous obtenons en utilisant l'hypothèse de récurrence

$$\begin{aligned} \Delta'_{q_{k+1}}(D_j) &= \frac{-T'_{R_{k+1},a_{k+1}} \Delta'_{q_k}(D_j)}{J} J_{i_{k+1},j} \\ &= \frac{(-1)^{k+1}}{J^{k+1}} T_{R_1}.\left(\prod_{l=2}^{k+1} T'_{R_l,a_l} J_{i_{l-1},j_l}\right).J_{i_{k+1},j} \end{aligned}$$

qui termine la récurrence. □

Remarque sur la forme de l'équation (44) **:** La forme donnée précédemment pour l'équation (44) montre que toutes les séries de moments suivant des paramètres Q-comptables s'expriment au moyen de m^2 mineurs et des termes T_R et $T'_{R,a}$. En revanche, elle n'est pas très “parlante” lorsqu'il s'agit de l'appliquer à une série et un paramètre donnés. Nous pouvons la réécrire en utilisant comme indices les symboles non terminaux :

- si la i-ème ligne de la matrice M correspond à l'équation définissant la série U, et si la j-ème colonne correspond à la différentiation suivant la série V, le mineur $J_{i,j}$ peut être noté $J_{U,V}$;
- si U est le i-ème symbole du membre droit de la règle R, le terme $T'_{R,i}$ peut être noté $T'_{R/U}$.

Avec ces notations, et en prenant comme paramètre élémentaire[4] $p = p_{R_1^{a_1}\dots R_{k-1}^{a_{k-1}} R_k}$, notons V_i le symbole du membre gauche de la règle R_i (pour $1 \leq i \leq k$), et U_i le a_i-ème symbole du membre droit de R_i (pour $1 \leq i \leq k-1$). Le théorème (4.14) s'exprime alors sous la forme suivante :

$$\Delta'_q(U_0) = (-1)^k \left(\prod_{i=1}^{k-1} \frac{T'_{R_i/U_i} J_{V_i,U_{i-1}}}{J}\right) \frac{T_{R_k} J_{V_k,U_{k-1}}}{J}.$$

Le théorème (4.14) répond de manière raisonnablement satisfaisante au problème de l'énumération suivant la somme des valeurs d'un paramètre Q-comptable : connaissant la

4. L'ordre des indices est ici conforme à notre notation usuelle.

décomposition de ce paramètre en combinaison linéaire de paramètres élémentaires, il est relativement simple d'exprimer la série génératrice comme fraction rationnelle des séries génératrices initiales, la fraction rationnelle faisant intervenir le jacobien J et des mineurs de la matrice jacobienne du système.

Une autre conséquence intéressante de ce théorème se situe au niveau des asymptotiques. Chaque paramètre élémentaire ne prenant que des valeurs positives, et les paramètres Q-comptables étant formés par combinaisons linéaires à coefficients positifs de ces paramètres élémentaires, il n'y a pas à craindre que des contributions provenant de paramètres élémentaires se compensent. Dès lors, pour estimer la valeur moyenne d'un paramètre Q-comptable, il nous suffit de faire le même travail pour chaque paramètre élémentaire apparaissant dans sa décomposition.

Dans la pratique, il devient possible de calculer le développement asymptotique de n'importe quelle série $\Delta'_{q_i}(D_j)$, quel que soit le rang du paramètre compté par la variable q_i, une fois calculés ceux des m séries algébriques D_i, celui du jacobien J, et ceux des m^2 mineurs $J_{i,j}$.

Exemple 4.15 (somme des hauteurs de pics des chemins de Dyck). Nous reprenons la grammaire G_2 de l'exemple (2.3). Dans cette grammaire, considérons les trois paramètres Q-comptables p_1, p_2 et p_3, comptant respectivement la demi-longueur des chemins, leur nombre de pics, et la somme des hauteurs de pics. Leurs décompositions en paramètres élémentaires sont :

Paramètre	Nom
p_1 (demi-longueur)	$R_3 + R_4 + R_5 + R_6$
p_2 (nombre de pics)	$R_3 + R_4$
p_3 (somme des hauteurs de pics)	$(\epsilon + R_5^{(1)} + R_6^{(1)})(R_3 + R_4)$

Le paramètre p_3 est donc une somme de 6 paramètres élémentaires, dont 2 sont de rang 1 et 4 de rang 2. Grâce au théorème (4.14), lorsque la variable q compte l'un des ces paramètres élémentaires, la série $\Delta'_q(D)$ peut être obtenue sans écrire une seule Q-équation.

En effet, du système algébrique

$$\begin{cases} D &= 1 + E \\ E &= x + 2xE + xE^2 \end{cases}$$

nous tirons la matrice jacobienne M :

$$M = \begin{pmatrix} -1 & 1 \\ 0 & -1 + 2x + 2xE \end{pmatrix}.$$

Nous avons donc $J = 1 - 2x(1+E)$; nous retrouvons le même jacobien que lors de l'énumération suvant l'aire. Les mineurs, quant à eux, sont (en indiçant lignes et colonnes par les symboles D et E plutôt que par 1 et 2) :

$$\begin{array}{rclrcl} J_{D,D} &=& -J, \qquad & J_{D,E} &=& 0, \\ J_{E,D} &=& -1, \qquad & J_{E,E} &=& -1. \end{array}$$

Tous les paramètres élémentaires apparaissant dans la décomposition des paramètres qui nous intéressent, n'utilisent que des E-dérivations, et donc seuls les mineurs $J_{E,D}$ et $J_{E,E}$ (tous deux égaux à -1) apparaîtront dans les calculs.

Commençons par appliquer le théorème au calcul de $\Delta'_x(D)$. La variable x compte la demi-longueur, soit $p_{R_3+R_4+R_5+R_6}$; nous avons donc

$$\begin{array}{rcl} \Delta'_x(D) &=& -\dfrac{(T_{R_3}+T_{R_4}+T_{R_5}+T_{R_6})\,J_{E,D}}{J} \\ &=& \dfrac{x+2xE+xE^2}{J} = \dfrac{E}{J}. \end{array}$$

De même, si la variable q compte le nombre de pics, le théorème donne pour $\Delta'_q(D)$:

$$\begin{array}{rcl} \Delta'_q(D) &=& -\dfrac{T_{R_3}J_{E,D}}{J} - \dfrac{T_{R_4}J_{E,D}}{J} \\ &=& \dfrac{x(E+1)}{J}. \end{array}$$

Si, maintenant, la variable r compte la somme des hauteurs de pics, il faut prendre en compte les contributions dues aux paramètres élémentaires $p_{R_5^{(1)}R_3}$, $p_{R_5^{(1)}R_4}$, $p_{R_6^{(1)}R_3}$ et $p_{R_6^{(1)}R_4}$. Pour les deux règles R_5 et R_6, le premier symbole du membre droit est toujours E, donc, pour chacun de ces 4 paramètres élémentaires, $j_2 = E$ (l'indice j_ℓ indique, pour chaque lettre $R_\ell^{(d_\ell)}$ du nom de paramètre, quel est le d_ℓ-ème symbole du membre droit de R_ℓ). Le théorème donne donc :

$$\begin{array}{rcl} \Delta'_r(D) &=& \Delta'_q(D) + \dfrac{\left(T'_{R_5,1}.J_{E,E} + T'_{R_6,1}.J_{E,E}\right)(T_{R_3}J_{E,D} + T_{R_4}.J_{E,D})}{J^2} \\ &=& \dfrac{x(E+1)}{J} + \dfrac{x^2(E+1)^2}{J^2}. \end{array}$$

En récrivant les résultats comme fractions rationnelles en J (on a $x(E+1) = (1-J)/2$ et $E = (1-J)/(1+J)$), nous obtenons les expressions suivantes :

$$\Delta'_x(D) = \frac{1-J}{J(1+J)}, \tag{45}$$

$$\Delta'_q(D) = \frac{1-J}{2J}, \tag{46}$$

$$\Delta'_r(D) = \frac{1-J^2}{4J^2}. \tag{47}$$

L'expression (45) est bien celle qui a été trouvée au paragraphe 4.3 : il s'agit de la même série.

Les coefficients des séries $\Delta'_q(D)$ et $\Delta'_r(D)$ peuvent être calculés explicitement. Pour $\Delta'_q(D)$, nous avons

$$\begin{aligned}\Delta'_q(D) &= \frac{1-J}{1+J}\frac{1+J}{2J} \\ &= E\left(\frac{1}{2}+\frac{1}{2J}\right) \\ &= \frac{1}{2}\left(-1+D+\Delta'_x(D)\right).\end{aligned}$$

Par conséquent, puisque $D = \sum_{n\geq 0} C_n x^n$ et $\Delta'_x(D) = \sum_{n\geq 0} nC_n x^n$, nous avons

$$\Delta'_q(D) = \sum_{n\geq 0} \frac{n+1}{2} C_n x^n,$$

et le nombre moyen de pics des chemins de Dyck de longueur $2n$ est exactement $(n+1)/2$.

En reprenant l'expression $J = \sqrt{1-4x}$ calculée au paragraphe 4.3, on trouve pour $\Delta'_r(D)$ l'expression extrêmement simple

$$\Delta'_r(D) = \frac{x}{1-4x}. \tag{48}$$

On retrouve aisément un résultat bien connu : la somme des hauteurs de pics des chemins de Dyck de longueur $2n$ est 4^{n-1}.

Enfin, l'expression (47) peut être comparée à (21), qui correspond à l'énumération suivant l'aire. Le rapport entre les deux séries est de x, ce qui, pour une singularité en $x = 1/4$, donne un rapport proche de 4 entre les moyennes des aires et de la somme des hauteurs de pics.

L'utilité du théorème (4.14) comme moyen pratique de calcul peut être mesurée en comparant les calculs effectués ci-dessus, à ceux effectués section 4.3. Il est à noter, toutefois, que nous n'avons pas de ce théorème une version portant sur les séries non projetées. Par conséquent, il ne nous permet pas de calculer les séries de moments d'ordre 2 ou plus.

Chapitre 5

Application à l'énumération de polyominos

Dans ce chapitre, nous mettons en œuvre la théorie des Q-grammaires sur différentes familles de polyominos codés par des langages algébriques. Dans la pratique, nous nous concentrons sur différentes classes de polyominos *verticalement convexes*, dont la frontière forme un chemin qu'il est relativement aisé de coder par les mots d'un langage algébrique.

Pour chaque famille de polyominos, nous donnons un codage par les mots d'un langage algébrique, et nous indiquons, parmi les paramètres classiques d'étude, lesquels sont Q-comptables dans la grammaire donnée.

5.1 Paramètres étudiés

Soit P un polyomino. Nous notons ∂P sa frontière, qui est formée d'un certain nombre de segments reliant des points adjacents du plan $\mathbb{Z}^2$; ces segments séparent chacun deux cellules dont l'une appartient à P et l'autre non.

Nous pouvons définir un certain nombre de paramètres :

– Le *périmètre* de P est la longueur de ∂P. Ce périmètre peut être décomposé en la somme du *périmètre vertical* (le nombre de segments verticaux qui forment ∂P) et du *périmètre horizontal* (le nombre de segments horizontaux qui forment ∂P). Chacun de ces périmètres étant pair, nous notons $pe(P)$ (respectivement, $ph(P)$, $pv(P)$) le demi-périmètre (respectivement, le demi-périmètre horizontal, le demi-périmètre vertical) de P.

– L'*aire* de P est le nombre de cellules qui le composent; nous la notons $a(P)$.

- La *largeur* (respectivement *hauteur*) de P est la largeur (respectivement hauteur) du plus petit rectangle contenant P. Ces deux paramètres sont respectivement notés $\ell(P)$ et $h(P)$. Notons que les polyominos verticalement convexes sont caractérisés par $\ell(P) = ph(P)$, et les polyominos convexes, par $h(P) = pv(P)$ et $\ell(P) = ph(P)$.

- Le *périmètre de sites* est le nombre total de cellules n'appartenant pas à P, mais qui sont adjacentes (le long d'une arête) à au moins une cellule de P. Ce périmètre de sites, noté $ps(P)$, est toujours inférieur ou égal au périmètre $2pe(P)$.

- Le *nombre d'angles rentrants* de P, est le nombre de sommets de ∂P pour lesquels, parmi les 4 cellules adjacentes qui partagent ce sommet, trois appartiennent à P et une ne lui appartient pas. Il est possible de distinguer, parmi les angles rentrants, des angles *Nord-Est, Sud-Est, Sud-Ouest, et Nord-Ouest*, suivant la position de la cellule extérieure par rapport aux trois autres. Pour les polyominos convexes, le nombre d'angles rentrants est égal à la différence entre le périmètre et le périmètre de sites.

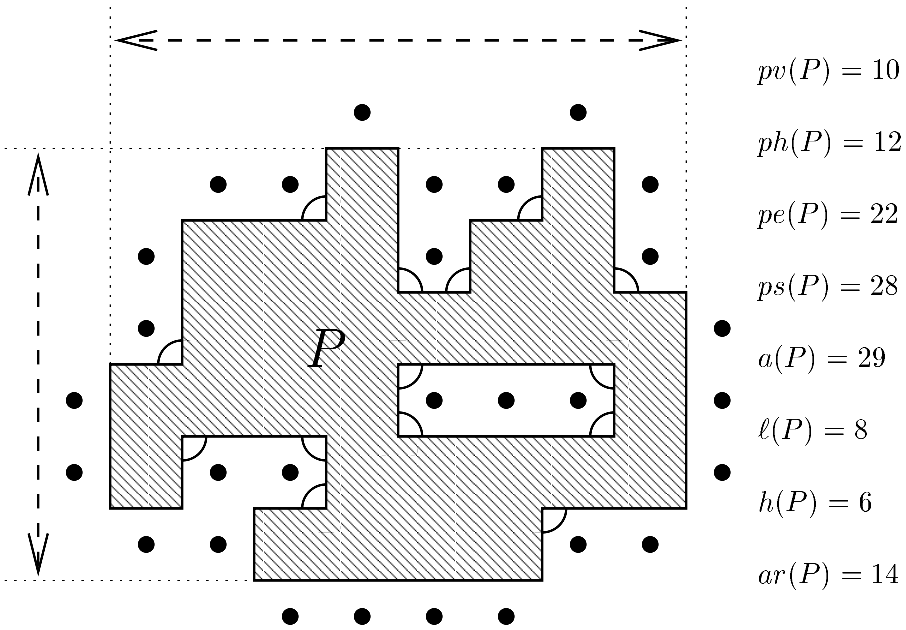


FIG. 5.1: *Un exemple de polyomino*

Pour chaque classe de polyominos, nous pouvons également définir un autre paramètre, le *périmètre de croissance* $pc(P)$, qui est le nombre de cellules c n'appartenant pas à P, mais telles que $P \cup \{c\}$ soit toujours un polyomino de la même classe. Pour les polyominos généraux, dont l'étude sort du cadre de ce travail, ce périmètre de croissance coïncide avec le périmètre de site, mais, en règle générale, $pc(P) \leq ps(P)$.

La figure 5.1 montre un exemple de polyomino général, et les valeurs de ces différents paramètres.

5.2 Polyominos parallélogrammes

Les polyominos parallélogrammes étant convexes, leur demi-périmètre vertical est égal à leur hauteur, et leur demi-périmètre horizontal, à leur largeur.

5.2.1 Codage

Le codage des polyominos parallélogrammes par des mots de Dyck est classique, et n'est rappelé ici que pour mémoire.

Définition 5.1. Soit P un polyomino parallélogramme. Nous notons k son nombre de colonnes, a_i $(1 \leq i \leq k)$ la hauteur de sa i-ème colonne, et b_i $(1 \leq i \leq n-1)$ le nombre de cellules suivant lesquelles les i-ème et $(i+1)$-ème colonnes sont accolées.

Le mot de Dyck codant P est le mot de Dyck $w = \psi_1(P)$, ayant k pics et $k-1$ creux, le i-ème pic étant de hauteur a_i et le i-ème creux, de hauteur $b_i - 1$.

Ce codage vérifie les propriétés suivantes :

Proposition 5.2. *Le codage ψ_1 établit une bijection entre l'ensemble $\mathcal{P}$ des polyominos parallélogrammes et le langage D des mots de Dyck; de plus,*

- *le nombre de colonnes de P est le nombre de pics de $\psi_1(P)$;*
- *si $p(P)$ est le périmètre de P, et si P n'est pas le polyomino vide, $p(P) = |\psi_1(P)| + 2$;*
- *si $\mathcal{A}(P)$ est l'aire de P et $S(w)$ la somme des hauteurs des pics de w, $\mathcal{A}(P) = S(\psi_1(P))$.*

Dans ce qui suit, nous acceptons comme polyomino parallélogramme, le polyomino vide, codé par le mot vide ϵ. L'exclure reviendrait à coder les polyominos parallélogrammes par les mots de Dyck non vides; dans les deux cas, la série génératrice suivant le demi-périmètre n'est pas exactement celle des mots de Dyck suivant la longueur, en raison du décalage de 1 entre longueur des mots non vides et demi-périmètre.

5.2.2 Grammaire

Pour que la grammaire puisse fournir d'autres types de paramètres que le simple périmètre, il faut que le nombre de pics des mots de Dyck soit Q-comptable. Nous avons vu dans les chapitres précédents qu'une grammaire convenable est la grammaire G_2, d'axiome D :

$$\left\{ \begin{array}{llll} R_1 : & D & \to & \epsilon \\ R_2 : & D & \to & E \\ R_3 : & E & \to & ab \\ R_4 : & E & \to & abE \\ R_5 : & E & \to & aEb \\ R_6 : & E & \to & aEbE \end{array} \right.$$

5.2.3 Paramètres Q-comptables

Dans la grammaire G_2, les paramètres suivants sont facilement décomposables en paramètres élémentaires :

- Hauteur : $h = p_{R_2} + p_{R_5} + p_{R_6}$.
- Largeur : $\ell = p_{R_3} + p_{R_4}$.
- Aire : $\mathcal{A} = p_{R_5^{(1)} R_3} + p_{R_5^{(1)} R_4} + p_{R_6^{(1)} R_3} + p_{R_6^{(1)} R_4}$.

Il est facile de voir que les paramètres "nombre d'angles rentrants" et "nombre d'angles Nord-Ouest rentrants" ne sont pas Q-comptables dans la grammaire G_2. En effet, étant tous deux inférieurs au périmètre, ils devraient être de rang 1 (dans la grammaire G_2, tous les paramètres Q-comptables non identiquement nuls et dont le nom ne commence pas par $R_2^{(1)}$ ont un rang minimal égal à leur rang formel, et si leur nom commence par $R_2^{(1)}$, cette lettre peut être retirée du nom sans changer la valeur du paramètre). Or, on vérifie immédiatement, en examinant les deux polyominos de la figure 5.2, qu'aucun paramètre de rang 1 ne convient : les arbres de dérivation correspondants utilisent les mêmes règles, mais les polyominos n'ont pas le même nombre d'angles Nord-Ouest rentrants. Le même argument prouve que le périmètre de sites n'est pas non plus Q-comptable, puisque, pour chaque polyomino parallélogramme P, $ps(P) = 2pe(P) - ar(P)$.

En revanche, le nombre d'angles rentrants Sud-Est est p_{R_6}. Les angles rentrants correspondent, sur le mot de Dyck, aux creux qui sont précédés immédiatement par au moins deux occurrences de b; ces creux apparaissent lors de l'utilisation de la règle R_6, et jamais autrement.

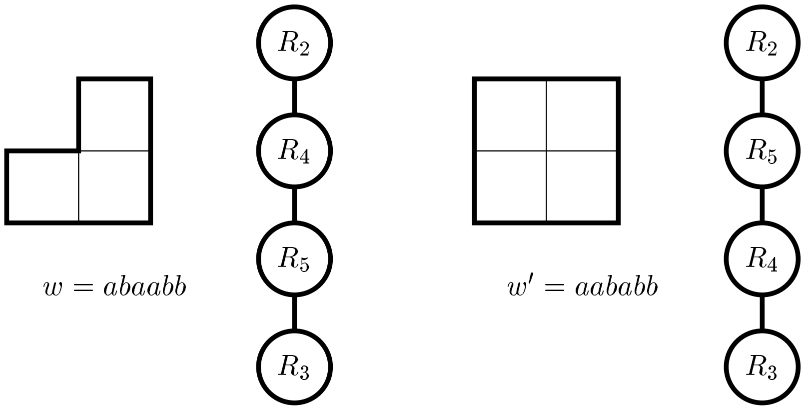


FIG. 5.2: *Deux polyominos parallélogrammes et leurs arbres de codage*

Compte tenu de la simplicité de la grammaire, nous pouvons calculer aisément la série génératrice suivant la hauteur (comptée par x), la largeur (comptée par p), et le nombre d'angles rentrants Sud-Est (comptés par r) :

$$\begin{cases} D(x,p,r) &= 1 + xE(x,p,r) \\ E(x,p,r) &= p + pE(x,p,r) + xE(x,p,r) + xrE(x,p,r)^2 \end{cases}$$

qui donne, après résolution d'une équation du second degré,

$$D(x,p,r) = \frac{1-p-x+2r-\sqrt{(1-p-x)^2-4xpr}}{2r}.$$

Dans cette situation, le jacobien (en ne gardant que les variables x et r) vaut $J = 1-p-x-2xE = 3-p-x-2D$.

En prenant comme *taille* le couple formé de la hauteur et de la largeur (puisque aucun de ces deux paramètres ne constitue à lui seul une taille), le calcul de la série de moments est également très simple : l'application du théorème 4.14 donne

$$\begin{aligned} \Delta'_q(D) &= -\frac{J_{E,D}T_{R_6}}{J} \\ &= \frac{x^2E^2}{1-p-x-2xE} \\ &= \frac{(D-1)^2}{3-p-x-2D}. \end{aligned}$$

Le nombre d'angles rentrants Sud-Est a la même distribution (par rapport à la hauteur et à la largeur) que le nombre d'angles rentrants Nord-Ouest : une rotation d'angle π échange ces deux paramètres sans modifier les différents périmètres ni l'aire. Par conséquent, la série génératrice et la série de moments correspondant à ce paramètre sont les mêmes que pour le nombre d'angles rentrants Sud-Est. Si la variable s compte le nombre total d'angles rentrants (qui n'est pas non plus Q-comptable), la série de moments $\Delta'_s(D)$ est le double de $\Delta'_r(D)$:

$$\Delta'_s(D) = 2\Delta'_r(D) = \frac{2(D-1)^2}{3-p-x-2D}.$$

Toutefois, le nombre d'angles rentrants n'étant pas Q-comptable, nous ne pouvons donner aussi simplement la série génératrice bivariée suivant le périmètre et le nombre d'angles rentrants.

De même que le nombre d'angles rentrants, la hauteur de la première ou de la dernière colonne du polyomino ne forment pas des paramètres Q-comptables. L'examen des mêmes polyominos permet de s'en convaincre rapidement, tout comme le simple fait que, lors de l'application de la règle R_4, la hauteur de la première colonne peut décroître : la première colonne du polyomino codé par $w = abw'$ est de hauteur 1, alors même que le mot w' peut coder un polyomino dont la première colonne est plus haute.

Toutefois, nous pouvons utiliser le lemme de marquage supérieur sur la règle pointée $R_6^{(1)}$ afin d'obtenir la grammaire G', plus fine que G_2 :

$$\left\{\begin{array}{llcllllcl}
R_1 : & D & \to & \epsilon & \quad & R_2 : & D & \to & E \\
R_3 : & E & \to & ab & & R'_3 : & F & \to & ab \\
R_4 : & E & \to & abE & & R'_4 : & F & \to & abF \\
R_5 : & E & \to & aEb & & R'_5 : & F & \to & aFb \\
R_6 : & E & \to & aFbE & & R'_6 : & F & \to & aFbF
\end{array}\right.$$

Dans cette nouvelle grammaire, le paramètre "hauteur du dernier pic" est Q-comptable, et vaut $p_{R_3} + p_{R_5}$. Si la variable t compte ce paramètre, et toujours en utilisant comme taille le couple formé de la largeur et de la hauteur, le théorème 4.14 donne, après calculs,

$$\Delta'_t(D)(x,p) = \frac{xE(x,p)}{1-x\,(1+E(x,p))}.$$

Pour obtenir la hauteur du premier pic comme paramètre Q-comptable, il faut appliquer le lemme de marquage supérieur aux règles $R_4^{(1)}$ et $R_6^{(2)}$; on obtient alors une autre expression pour la série de moments (qui est la même série, puisque le premier pic devient, par image miroir, le dernier; cela correspond, sur les polyominos parallélogrammes, à effectuer une symétrie centrale) :

$$\Delta'_t(D)(x,p) = \frac{xp + x^2E(x,p)}{1-p-x\,(1+F(x,p))}.$$

Dans les deux cas, en fixant $p = x$ (énumération suivant le périmètre seul), on obtient en développant en série entière :

$$\Delta_t'(D)(x,x) = \sum_{n\geq 1} (C_n - C_{n-1})\, x^n.$$

Ainsi, la hauteur moyenne de la première colonne des polyominos parallélogrammes de périmètre $2n$, est exactement $C_n/C_{n-1} - 1$, qui tend vers 3 lorsque n tend vers $+\infty$.

5.3 Polyominos verticalement convexes

Dans [25], Delest décrit un codage des polyominos verticalement convexes par des mots de Dyck colorés formant un langage algébrique, et donne une grammaire engendrant leur langage, ainsi qu'une expression pour la série génératrice. Ce codage est inspiré de celui des polyominos parallélogrammes, et la série génératrice obtenue énumère les polyominos verticalement convexes suivant le paramètre *périmètre*.

Dans [14], Bousquet-Mélou obtient, par une méthode ne faisant pas appel au codage par des mots, une expression pour la série génératrice de ces polyominos suivant les paramètres *largeur*, *périmètre vertical*, *aire*, et *hauteurs des première et dernière colonnes*. Les calculs font intervenir des q-équations d'une forme proche de celles des équations fournies par les Q-grammaires, mais où interviennent directement des projections de séries.

Plus récemment, Feretić [38, 39] a donné, pour la série génératrice des polyominos verticalement convexes suivant la largeur et le périmètre vertical, une expression nettement plus simple que celle donnée dans [25], et dont la preuve ne fait intervenir que des équations du second degré. La méthode passe par l'énumération d'objets formés à partir des polyominos verticalement convexes, et par une bijection avec les *polyominos murs*, qui codent les *compositions* d'un entier – nous les étudierons au paragraphe 5.4.

5.3.1 Codage

Nous reprenons, en le modifiant très légèrement, le codage des polyominos verticalement convexes par des mots de Dyck colorés tel qu'il est décrit dans [25].

Nous codons la frontière des polyominos verticalement convexes par des mots utilisant les lettres a, b, a', b' et p. La première condition que doivent remplir nos mots est qu'en leur appliquant le morphisme φ défini par

$$\begin{cases} \varphi(a) = \varphi(a') = a \\ \varphi(b) = \varphi(b') = b \\ \varphi(p) = ab \end{cases}$$

on doit obtenir des mots de Dyck non vides. De plus, ces mots ne doivent contenir aucun facteur ab, $a'b$, ab' ou $a'b'$: tous les pics du mot de Dyck $\varphi(w)$ doivent provenir des occurrences de la lettre[1] p.

Enfin, les mots de L doivent être de la forme forme

$$w = w_0 p w_1 p \dots w_{k-1} p w_k$$

où les mots w_i vérifient les conditions suivantes :

- $w_0 \in a^*$;
- $w_k \in b^*$;
- pour $1 \leq i \leq k-1$, $w_i \in b^*a^* \cup b'^*a'^* \cup b^*b'^* \cup a'^*a^*$.

Le langage L peut également être décrit comme le langage des mots de Dyck, où tous les pics sont remplacés par la lettre p, et où les facteurs $b^i a^j$ (les "vallées") qui séparent deux pics consécutifs sont colorés de différentes manières suivant les lettres qu'ils contiennent :

- si $i > 0$ et $j > 0$ (la vallée contient au moins un pas descendant et au moins un pas montant), toutes les lettres sont de la même "couleur" : $b^i a^j$ ou $b'^i a'^j$;
- si $i = 0$ (la vallée ne contient en fait que des pas montants), les lettres a' précèdent les lettres a : $a'^{j_1} a^{i_2}$;
- si $j = 0$ (la vallée ne contient que des pas descendants), les lettres b' sont placées après les lettres b : $b^{i_1} b'^{i_2}$.

Par analogie avec les mots et chemins de Dyck, nous appelons *pic* d'un mot de L, chaque occurrence de la lettre p dans ce mot. Si $w = w_1 p w_2$, la *hauteur* de ce pic est $|w_1|_{a,a'} - |w_1|_{b,b'} + 1$.

Soit P un polyomino verticalement convexe. Nous appelons *coin Sud-Ouest* de P, le coin inférieur gauche A de la plus basse cellule de la première colonne de P, et *coin Nord-Est* de P, le coin supérieur droit C de la plus haute cellule de la dernière colonne de P. Soit B le point situé immédiatement au-dessus de A, et D le point situé immédiatement au-dessous de C (voir figure 5.3). Le polyomino P est parfaitement décrit par un chemin allant de A à D (chemin inférieur) et un chemin allant de B à C (chemin supérieur). Ces chemins sont semi-dirigés (ils ne font que des pas Nord, Sud et Est), et ne se rencontrent pas; le chemin

1. Dans [25], le codage est fait avec 4 lettres x, y, $\overline{x}$, $\overline{y}$, les pics étant laissés sous forme $x\overline{x}$; il semble toutefois plus naturel de les distinguer comme nous le faisons ici.

AD commence forcément par un pas Est, et le chemin BC se termine forcément par un pas Est.

Le codage de P par un mot de L se fait de la manière suivante : chaque colonne est codée par un pic (lettre p) de hauteur égale à celle de la colonne. Entre deux pics successifs, on code les pas des chemins inférieur et supérieur qui se situent à la jonction entre les deux colonnes codées, dans l'ordre suivant :

- les éventuels pas Nord du chemin inférieur, par autant de b;
- les éventuels pas Sud du chemin supérieur, par autant de b';
- les éventuels pas Sud du chemin inférieur, par autant de a';
- les éventuels pas Nord du chemin supérieur, par autant de a.

Ainsi, les lettres a' et b' codent des pas qui ne peuvent apparaître dans un polyomino parallélogramme : si P est un polyomino parallélogramme, le mot $\psi_2(P)$ ne diffère du mot $\psi_1(P)$ que par le fait que les facteurs ab ont été remplacés par p. En ce sens, le codage ψ_2 peut être considéré comme une extension aux polyominos verticalement convexes du codage ψ_1 des polyominos parallélogrammes. Comme pour le codage des polyominos pa-

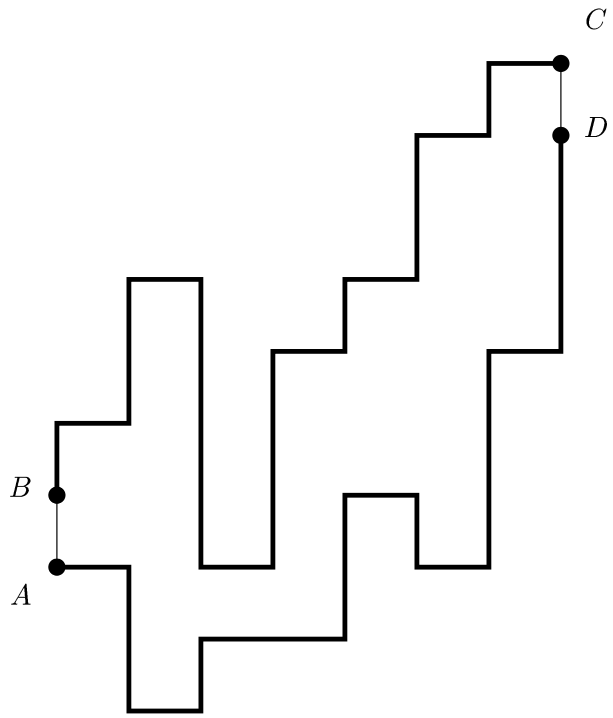


FIG. 5.3: *Le polyomino codé par* $w = apaaa'a'pb'b'b'b'bpaaapbbapa'aapbbbapbbb$

rallélogrammes, remplacer le mot w par son image miroir (lue de droite à gauche, en remplaçant a par b, b par a, a' par b', et b' par a') donne un mot w' qui code un nouveau polyomino verticalement convexe; ces deux polyominos verticalement convexes sont images l'un de l'autre par une symétrie centrale.

D'après la définition que nous avons donnée de $\psi_2(P)$, il est clair qu'un polyomino de largeur k et de demi-périmètre vertical n, est codé par un mot $\psi_2(P)$ vérifiant $|\psi_2(P)|_p = k$ et $|\psi_2(P)|_{a,a'} = |\psi_2(P)|_{b,b'} = n-1$. Enfin, l'*aire* de P devient, dans $\psi_2(P)$, la *somme des hauteurs des pics.*

5.3.2 Grammaire

Une grammaire algébrique non ambiguë qui engendre L, est donnée, de manière incomplète, dans [25]; une fois écrite *in extenso*, nous obtenons les équations (non commutatives) :

$$\left\{\begin{array}{rcl}
OO & = & p + pGO + aOOb(\epsilon + OO) + aODb'p(\epsilon + GO) + a(\epsilon + OD)pb'\overline{G}O \\
 & & +aO\overline{D}b'\overline{G}O \\
OD & = & p + pGD + aOOb(\epsilon + OD) + aODb' + aODb'p(\epsilon + GD) \\
 & & +a(\epsilon + OD)pb'\overline{G}D + aO\overline{D}b'\overline{G}D \\
O\overline{D} & = & pG\overline{D} + aOObO\overline{D} + a(\epsilon + OD)pb' + aO\overline{D}b' + aODb'pG\overline{D} \\
 & & +a(\epsilon + OD)pb'\overline{GD} + aO\overline{D}b'\overline{GD} \\
\overline{G}O & = & a'pb(\epsilon + OO) + a'pGOb(\epsilon + OO) + a'\overline{G}Ob(\epsilon + OO) + a'\overline{G}Db'pGO \\
 & & +a'p(\epsilon + GD)b'pGO + a'(p + pp + pG\overline{D} + \overline{G}Dp + pGDp + \overline{GD})b'\overline{G}O \\
\overline{G}D & = & a'p(\epsilon + GO)b(\epsilon + OD) + a'\overline{G}Ob(\epsilon + OD) + a'p(\epsilon + GD)b' + a'\overline{G}Db' \\
 & & +a'p(\epsilon + GD)b'p(\epsilon + GD) + a'\overline{G}Db'p(\epsilon + GD) \\
 & & +a'(p + pp + pG\overline{D} + \overline{G}Dp + pGDp + \overline{GD})b'\overline{G}D \\
\overline{GD} & = & a'p(\epsilon + GO)bO\overline{D} + a'\overline{G}ObO\overline{D} \\
 & & +a'(p + pp + pG\overline{D} + \overline{G}Dp + pGDp + \overline{GD})b'(\epsilon + \overline{GD}) \\
 & & +a'p(\epsilon + GD)b'pG\overline{D} + a'\overline{G}Db'pG\overline{D} \\
GO & = & OO + a'GOb(\epsilon + OO) + a'GDb'p(\epsilon + GO) + a'(\epsilon + GD)pb'\overline{G}O \\
 & & +a'G\overline{D}b'\overline{G}O \\
GD & = & OD + a'GOb(\epsilon + OD) + a'GDb' + a'GDb'p(\epsilon + GD) + a'(\epsilon + GD)pb'\overline{G}D \\
 & & +a'G\overline{D}b'\overline{G}D \\
G\overline{D} & = & O\overline{D} + a'GObO\overline{D} + a'(\epsilon + GD)pb' + a'G\overline{D}b' + a'GDb'pG\overline{D} \\
 & & +a'(\epsilon + GD)pb'\overline{GD} + a'G\overline{D}b'\overline{GD}
\end{array}\right.$$

L'axiome de la grammaire est le symbole OO; dans chaque symbole, la première lettre (O, G ou $\overline{G}$) indique quelles lettres précèdent la première occurrence de p, et la deuxième lettre (O, D ou $\overline{D}$) indique ce qui suit la dernière occurrence de p :

- la lettre O indique une montée (ou descente) composée uniquement de a (ou de b), éventuellement vide;
- la lettre G (respectivement D) indique une montée (éventuellement vide) composée de a', puis de a (respectivement, de b, puis de b');
- la lettre $\overline{G}$ (respectivement $\overline{D}$) indique une montée (respectivement descente) non vide, composée uniquement de a' (respectivement b').

Dans le codage, la largeur (nombre de colonnes) des polyominos verticalement convexes devient le nombre d'occurrences de p, et le demi-périmètre vertical devient le nombre total d'occurrences de a et a' (auquel il faut ajouter 1). Par conséquent, les paramètres largeur, périmètre vertical, et périmètre total, sont tous Q-comptables de rang 1. Par ailleurs, l'aire du polyomino correspond à la somme des hauteurs des pics du mot. C'est donc un paramètre Q-comptable de rang 2, pour lequel la seule substitution de variables utilisée sera $\sigma_{p \leftarrow pq}$. Le système de q-équations est obtenu en remplaçant, dans le système algébrique, chaque symbole U (respectivement, chaque p) apparaissant entre deux lettres a (ou a') et b (ou b'), par $\sigma_{p \leftarrow pq}(U)$ (respectivement, par pq).

Cette grammaire comporte 9 symboles et 108 règles de dérivation, et n'est pas une grammaire propre; itérer les règles qui n'écrivent aucune lettre pour la rendre propre ferait encore augmenter le nombre de nombre de règles. Tel quel, le jacobien du système algébrique est trop gros pour être calculé par Maple. Il est donc hors de question, pour des raisons pratiques, d'exploiter directement ce système.

Toutefois, le calcul sur les séries génératrices (à variables commutatives) permet de ramener ce système à des proportions plus raisonnables. La première simplification consiste à remarquer que les langages GO et OD (respectivement, $\overline{G}O$ et $O\overline{D}$; $\overline{G}D$ et $G\overline{D}$) sont images miroir l'un de l'autre. Nous en déduisons l'identité suivante :

$$OD(a, a', b, b', p) = GO(b, b', a, a', p),$$

ainsi que des identités similaires pour les deux autres couples de langages. Si l'on renonce à différencier les lettres a et a' d'une part, et b et b' d'autre part (ce qui implique de ne pas distinguer, dans le périmètre vertical, la contribution du chemin inférieur, codé par les lettres a' et b, de celle du chemin supérieur, codé par a et b'), on obtient, en posant

$a = a' = b = b' = x^{1/2}$,

$$\begin{cases} OD(x,p) &= GO(x,p) \\ O\overline{D}(x,p) &= \overline{G}O(x,p) \\ G\overline{D}(x,p) &= \overline{G}D(x,p) \end{cases}$$

qui permet d'éliminer 3 des 9 équations du système de départ.

La seconde étape de simplification revient à remarquer qu'il existe une bijection naturelle entre OO et $\overline{G}O + p + p.GO$, laquelle consiste à remplacer tous les a initiaux d'un mot $w \in OO$, par des a'. La même transformation établit également une bijection entre OD et $\overline{G}D + p + p.GD$. Les identités qui en découlent sur les séries génératrices sont :

$$\begin{cases} \overline{G}O(x,p) &= OO(x,p) - p - p.GO(x,p) \\ \overline{G}D(x,p) &= OD(x,p) - p - p.GD(x,p). \end{cases}$$

Le système peut alors s'écrire en n'utilisant que les séries OO, OD et GD (la série $\overline{GD}$, n'apparaissant plus dans les 3 équations, est abandonnée) :

$$\begin{array}{lrcl} (1) & OO &=& p + pOD + x\underline{OO}(1+OO) + xp\underline{OD}(1+OD) \\ & & & +x\underline{p}(1+\underline{OD})(OO-p-pOD) + x(\underline{OO}-\underline{p}-\underline{pOD})(OO-p-pOD) \\ (2) & OD &=& p + pGD + x\underline{OO}(1+OD) + x\underline{OD} + xp\underline{OD}(1+GD) \\ & & & +x\underline{p}(1+\underline{OD})(OD-p-pGD) + x(\underline{OO}-\underline{p}-\underline{pOD})(OD-p-pGD) \\ (3) & GD &=& OD + x\underline{OD}(1+OD) + x\underline{GD} + xp\underline{GD}(1+GD) \\ & & & +x\underline{p}(1+\underline{GD})(OD-p-pGD) + x(\underline{OD}-\underline{p}-\underline{pGD})(OD-p-pGD) \end{array}$$

Dans le système ci-dessus, les termes soulignés sont ceux qui, dans le q-système correspondant à l'énumération suivant l'aire, doivent être remplacés par leur image par $\sigma_{p \leftarrow pq}$.

5.3.3 Paramètres Q-comptables

Nous avons déjà vu que les paramètres *largeur* (ou *demi-périmètre horizontal*) et *demi-périmètre vertical* sont Q-comptables de rang 1 dans la grammaire présentée ci-dessus (il s'agit respectivement, sur les mot, de $|w|_p$ et de $|w|_{a,a'}$), et que le paramètre *aire* est Q-comptable de rang 2.

Comme dans le cas des polyominos parallélogrammes, les paramètres *nombre d'angles rentrants*, *périmètre de sites* et *hauteur de la première colonne* ne sont pas Q-comptables; il suffit d'ailleurs, pour le prouver, d'écrire les arbres de dérivation des mots codants les deux polyominos de la figure 5.2, qui utilisent toujours les mêmes règles.

Ici encore, il serait possible d'utiliser le lemme de marquage supérieur pour obtenir une grammaire plus fine dans laquelle les hauteurs des première et dernière colonnes soient Q-comptables; malheureusement, la taille de la grammaire initiale rendrait cette transformation totalement inexploitable. Toutefois, il est possible d'accéder aux séries de moments de la hauteur de la dernière colonne, ou du produit des hauteurs des première et dernière colonnes : il s'agit des séries génératrices des langages OD et GD.

En effet, si un mot $w \in OO$ s'écrit $w = a^i pw'pb^j$ $(i \geq 0,\, j \geq 0)$, il code un polyomino dont la première colonne est de hauteur $i+1$, et la dernière, de hauteur $j+1$. Or, nous pouvons associer à w, $j+1$ mots de OD :

$$w_{j'} = a^i pw'pb^{j-j'} b'^{j'} b^{j-j'} \quad (0 \leq j' \leq j)$$

et $(i+1)(j+1)$ mots de GD :

$$w'_{i',j} = a'^{i'} a^{i-i'} pw'pb^{j-j'} b'^{j'} \quad (0 \leq i' \leq i, 0 \leq j' \leq j).$$

Tous les mots $w_{j'}$ et $w_{i',j'}$ vérifient

$$\begin{cases} |w_{j'}|_p &= |w_{i',j'}|_p &= |w|_p \\ |w_{j'}|_{a,a'} &= |w_{i',j'}|_{a,a'} &= |w|_{a,a'} \end{cases}$$

et apportent donc la même contribution à leurs séries génératrices respectives. Par conséquent, la série $OD(x,p)$ est la série de moments de la hauteur de la dernière colonne, et $GD(x,p)$, la série de moments du produit des hauteurs de la première et de la dernière colonne.

5.3.4 Séries génératrices

Il est possible de résoudre explicitement le système d'équations $(1)-(3)$ pour obtenir la série génératrice, mais l'expression obtenue est plus complexe que celle, remarquablement simple, obtenue par Feretić dans [38, 39] (après les changements de variables appropriés) :

$$OO(x,p) = \frac{1-x}{x}\left(1 - \frac{4}{6 - \sqrt{2(1+p) + 2\sqrt{(1-p)^2 - 16\frac{xp}{(1-x)^2}}}}\right).$$

Le système d'équations algébriques fourni par la grammaire ne permet pas d'obtenir une expression aussi élégante, mais il rend possible la comparaison à la série $OD(x,p)$. La méthode n'est ici qu'une version à deux variables de celle utilisée dans [25]; les expressions données par Maple sont également plus simples que celles fournies par Macsyma.

Après élimination de GD (qui n'apparaît qu'au degré 1 dans l'équation 2), on obtient un système algébrique portant sur OO et OD. Pour chaque inconnue, on se ramène alors à une équation de degré 6, à chaque fois factorisable en un produit de deux équations de degrés 2 et 4. Parmi les solutions explicitement fournies par Maple, une seule est alors analytique à l'origine, et les premiers termes de son développement correspondent bien à ceux obtenus en comptant directement les polyominos verticalement convexes de petit périmètre.

Les séries obtenues sont alors :

$$\begin{aligned} OO(x,p) &= \frac{T_1+T_2+T_3}{4xA}, \\ OD(x,p) &= \frac{T_1'+T_2'+T_3'}{4xA}, \end{aligned}$$

avec

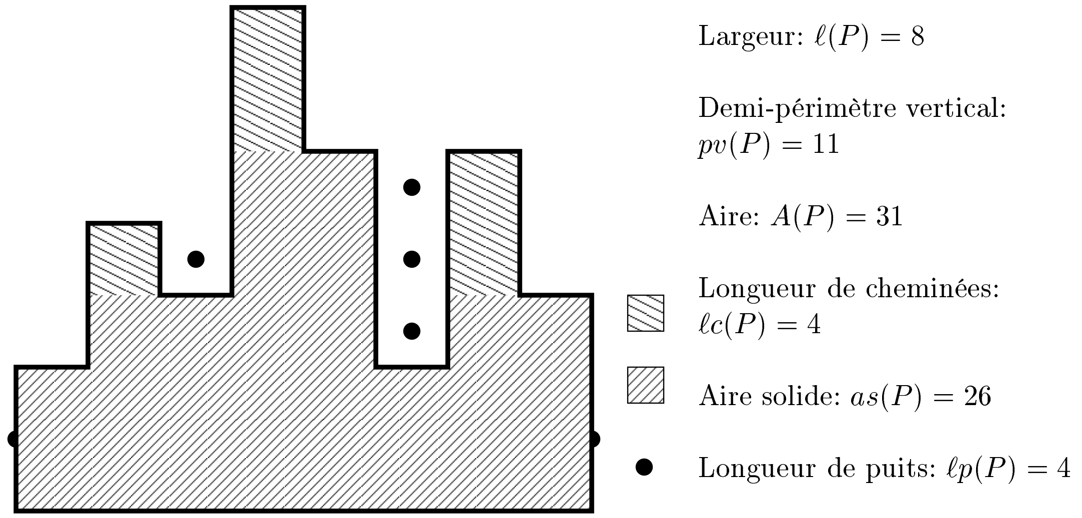
$$\begin{aligned}
A &= 18(1-x)^2 - p(2-5x+2x^2) \\
B &= 1-2x-2p-12xp+x^2+p^2-2xp^2-2x^2p+x^2p^2 \\
T_1 &= (1-x)\left\{21(1-x)^2 - p(5-14x+5x^2)\right\} \\
T_1' &= -(1-x)^2(17+38x)+p(1-2x)(1+6x-3x^2) \\
T_2 &= -3(1-x)^2\sqrt{B} \\
T_2' &= -(1-x)(1-2x)\sqrt{B} \\
T_3 &= -(1-x)^2\sqrt{2\left(P+Q\sqrt{B}\right)} \\
T_3' &= (1-x)^3\sqrt{\frac{2}{p}\left(P'+Q'\sqrt{B}\right)} \\
P &= (1-x)^2\left(81(1-x)^2+p(46-232x+26x^2)+p^2(1+x)^2\right) \\
Q &= (1-x)\left(81(1-x)^2-p(1+x)^2\right) \\
P' &= 144(1-x)^3-3p(1-x)(5-14x)(1+10x) \\
&\quad -2p^2(1-2x)(1-3x+14x^2)+p^3(1-x)(1-2x)^2 \\
Q' &= -144(1-x)^2+3p(1-2x)(11-14x)-p^2(1-2x)^2
\end{aligned}$$

5.4 Polyominos murs

Les polyominos murs ont été étudiés par Feretić dans [38]; leur série génératrice suivant les périmètres vertical et horizontal sert d'intermédiaire de calcul pour obtenir celle des polyominos verticalement convexes. Prellberg et Brak [69] ont également étudié leur série génératrice suivant l'aire et les périmètres vertical et horizontal, au moyen de q-équations similaires à celles que nous donnons plus loin.

Les polyominos murs sont les polyominos verticalement convexes dont toutes les colonnes ont leur plus basse cellule à la même hauteur; ce sont également les polyominos verticalement convexes qui sont à la fois dirigés suivant les directions Sud-Ouest/Nord-Est *et* Sud-Est/Nord-Ouest. La figure 5.4 montre un exemple de polyomino mur.

De manière classique, un polyomino mur d'aire n est codé par une *composition* de n, c'est-à-dire une suite (ordonnée) d'entiers strictement positifs dont la somme est n. La bijection est immédiate, chaque colonne du polyomino étant codée par sa hauteur. Toutefois, afin d'avoir un codage par un langage, et de pouvoir étudier simultanément l'aire et les périmètres, nous codons leur frontière par des mots et obtenons l'aire comme paramètre Q-comptable de rang 2.



$w = apaapb'paaaapb'b'pb'b'b'paaapb'b'pb'b'$

FIG. 5.4: *Un polyomino mur*

5.4.1 Codage et grammaire

Comme sous-classe de polyominos verticalement convexes, les polyominos murs peuvent être codés de la même manière que les polyominos verticalement convexes; un mot du langage OO code un polyomino mur s'il ne comporte pas la lettre a' (puisque cette lettre code les pas verticaux descendants du chemin inférieur), ni la lettre b, sauf éventuellement après la dernière occurrence de p (puisque la lettre b code les pas verticaux montants du chemin inférieur).

Le plus simple, pour coder un polyomino mur, est en fait d'utiliser un mot de OD ne comportant ni a' ni b, ce qui correspond à changer en b' tous les b finaux du mot de OO correspondant. Le mot w codant un polyomino mur P correspond alors à une simple lecture du chemin reliant le coin Nord-Ouest de P à son coin Nord-Est : chaque pas Nord est codé par la lettre a, chaque pas Sud, par la lettre b', et chaque pas Est, par la lettre p.

Nous obtenons donc une grammaire engendrant un codage des polyominos murs, en reprenant la grammaire pour les polyominos verticalement convexes, et en effaçant toutes les règles faisant apparaître l'une des lettres a' et b; l'axiome est, bien entendu, OD. La grammaire obtenue est :

$$\left\{\begin{array}{lcl}
OO & = & p + pGO + aODb'p(\epsilon + GO) + a(\epsilon + OD)pb'\overline{G}O \\
 & & +aO\overline{D}b'\overline{G}O \\
OD & = & p + pGD + aODb' + aODb'p(\epsilon + GD) \\
 & & +a(\epsilon + OD)pb'\overline{G}D + aO\overline{D}b'\overline{G}D \\
O\overline{D} & = & pG\overline{D} + a(\epsilon + OD)pb' + aO\overline{D}b' + aODb'pG\overline{D} \\
 & & +a(\epsilon + OD)pb'\overline{GD} + aO\overline{D}b'\overline{GD} \\
\overline{G}O & = & \emptyset \\
\overline{G}D & = & \emptyset \\
\overline{GD} & = & \emptyset \\
GO & = & OO \\
GD & = & OD \\
G\overline{D} & = & O\overline{D}
\end{array}\right.$$

En retirant les règles faisant apparaître les symboles $\overline{G}O$, $\overline{G}D$ et $\overline{GD}$, et en remplaçant les symboles GO, GD et $G\overline{D}$ par, respectivement, OO, OD et $O\overline{D}$, nous obtenons une grammaire où seul le symbole OD est accessible à partir de lui-même. Par conséquent, le langage codant les polyominos murs est engendré par la grammaire suivante :

$$\left\{\begin{array}{llcl}
R_1 : & OD & \to & p \\
R_2 : & OD & \to & pOD \\
R_3 : & OD & \to & aODb' \\
R_4 : & OD & \to & aODb'p \\
R_5 : & OD & \to & aODb'pOD
\end{array}\right.$$

Cette grammaire correspond exactement à la décomposition donnée dans [38], où elle est présentée sous la forme d'une grammaire d'objets.

5.4.2 Paramètres Q-comptables

La grammaire utilisée pour coder les polyominos murs étant, en quelque sorte, une restriction de celle utilisée pour les polyominos verticalement convexes, il est clair que la largeur et le demi-périmètre vertical sont toujours Q-comptables de rang 1, et que l'aire est Q-comptable de rang 2.

Les hauteurs des première et dernière colonnes ne sont pas Q-comptables, mais, comme dans le cas des polyominos parallélogrammes, le lemme de marquage supérieur peut être utilisé pour obtenir une grammaire plus fine dans laquelle ils sont Q-comptables de rang 1.

Les angles rentrants Nord-Est du polyomino correspondent exactement, sur le mot qui le code, aux facteurs $b'p$. Il est facile de voir que de tels facteurs sont engendrés par les règles R_4 et R_5, et par elles seules; le nombre d'angles rentrants Nord-Est est donc Q-comptable. En revanche, le nombre d'angles rentrants Nord-Ouest n'est pas Q-comptable. Il est toutefois facile d'écrire une grammaire plus fine dans laquelle ce paramètre est Q-comptable, tout en restant à un niveau de complexité raisonnable :

$$\left\{\begin{array}{lllllllll} R'_1: & U & \rightarrow & P & \qquad & R'_2: & U & \rightarrow & P \\ R'_3: & P & \rightarrow & p & & R'_4: & P & \rightarrow & pA \\ R'_5: & P & \rightarrow & pP & & & & & \\ R'_6: & A & \rightarrow & aUb' & & R'_7: & A & \rightarrow & aUb'p \\ R'_8: & A & \rightarrow & aUb'pA & & R'_9: & A & \rightarrow & aUb'pbP \end{array}\right.$$

Dans cette grammaire d'axiome U, le langage A contient tous les mots de U dont la première lettre est a, et le langage P, ceux dont la première lettre est p. Ainsi, le nombre d'angles rentrants Nord-Ouest (ou nombre d'occurrences du facteur pa) est $p_{R'_4}+p_{R'_8}+p_{R'_9}$, et le nombre d'angles rentrants Nord-Est (ou de facteurs $b'p$) est $p_{R'_7}+p_{R'_8}+p_{R'_9}$.

Dans le travail de Feretić, seuls les polyominos murs ayant un nombre *impair* de colonnes sont réellement utilisés, et, pour une énumération suivant l'aire, seule la contribution à l'aire des colonnes de rang impair doit être considérée. Il n'est pas possible, dans les grammaires présentées ci-dessus, de déterminer si une colonne donnée (codée par une occurrence de p) est de rang pair ou impair. Il n'est donc pas étonnant que le paramètre *somme des hauteurs des colonnes de rang impair* ne soit pas Q-comptable. Nous pouvons toutefois modifier légèrement le codage, de telle sorte que les colonnes de rang impair soient codées par la lettre p, et celles de rang pair, par une nouvelle lettre p'. En distinguant quatre langages auxiliaires suivant que les mots qui les composent ont un nombre pair (P et P') ou impair (I et I') de lettres p ou p', et suivant que la première de celles-ci est un p (I et P)

ou un p' (I' et P'), nous adaptons très simplement la grammaire initiale pour en produire une nouvelle, dans laquelle les paramètre *somme des hauteurs des colonnes de rang impair* est Q-comptable[2] de rang 2 :

$$\begin{cases} I &= p + pP' + aIb' + aPb'p + aIb'p'I + aPb'pP' \\ I' &= p' + p'P + aI'b' + aP'b'p' + aI'b'pI' + aP'b'p'P \\ P &= pI' + aPb' + aIb'p' + aIb'p'P + aPb'pI' \\ P' &= p'I + aP'b' + aI'b'p + aI'b'pP' + aP'b'p'I \end{cases}$$

Un exemple de paramètre intéressant, mais qui semble difficile à obtenir comme paramètre Q-comptable, est le périmètre de sites. La différence entre le périmètre et le périmètre de sites se compose de la somme de deux autres paramètres, dont l'un est le nombre d'angles rentrants (que nous avons pu "attraper" au prix d'une modification raisonnable de la grammaire la plus "simple"), et l'autre est ce que nous pouvons appeler la *longueur totale de puits* : le nombre de cellules extérieures au polyomino, mais dont les deux cellules situées immédiatement à droite et à gauche appartiennent au polyomino – voir figure 5.4. Sur notre codage des polyominos murs, ce paramètre peut être compté en sommant, pour chaque facteur $b'^i pa^j$ maximal, le plus petit parmi i et j. Lorsqu'un paramètre est défini par un *minimum* (ou par un maximum), il y a peu d'espoir de le rendre Q-comptable; il est donc probable que, s'il est possible de rendre ce paramètre Q-comptable, ce sera au moyen d'une grammaire, voire d'un codage, radicalement différents.

Un autre paramètre, en apparence proche de la longueur totale de puits, est la *longueur totale de cheminées* : le nombre de cellules appartenant au polyomino, mais dont aucun des voisins Est et Ouest n'appartient au polyomino. Un tel paramètre est, sur notre langage, très similaire au *poids de pyramides* étudié sur le langage de Dyck par Denise et Simion [32]. Nous pouvons aisément raffiner notre grammaire pour en obtenir une dans laquelle la longueur totale de cheminées est un paramètre Q-comptable de rang 1 :

$$\begin{cases} R_1: & M \to P & R_2: & M \to N \\ R_3: & P \to p & R_4: & P \to aPb' \\ R_5: & N \to pP & R_6: & N \to pN \\ R_7: & N \to aPb'p & R_8: & N \to aNb'p \\ R_9: & N \to aPb'pP & R_{10}: & N \to aPb'pN \\ R_{11} & N \to aNb'pP & R_{12}: & N \to aNb'pN \end{cases}$$

2. Cette nouvelle grammaire n'est pas à proprement parler plus fine que l'originale, car le langage engendré n'est pas le même; en remplaçant p' par p, on obtient une grammaire plus fine que l'originale.

Dans cette nouvelle grammaire (P est le langage des mots n'ayant qu'une occurrence de la lettre p, et N est celui des mots qui en ont au moins 2; cette grammaire est obtenue en effectuant un marquage inférieur des règles qui écrivent la lettre p), la longueur totale de cheminée correspond au paramètre $p_{R_1} + p_{R_4} + p_{R_7} + p_{R_9} + p_{R_{10}}$.

Dans cette même grammaire, l'aire des polyominos peut, sans difficulté, s'exprimer comme paramètre Q-comptable de rang 2, avec comme nom de paramètre $R_1 + \alpha\beta$, où:

$$\begin{cases} \alpha &= 1 + R_4^{(1)} + R_7^{(1)} + R_8^{(1)} + R_9^{(1)} + R_{10}^{(1)} + R_{11}^{(1)} + R_{12}^{(1)}, \\ \beta &= R_3 + R_4 + R_5 + R_6 + R_7 + R_8 + R_9 + R_{10} + R_{11} + R_{12}. \end{cases}$$

Sous cette forme, il n'apparaît pas immédiatement que la différence entre l'aire du polyomino et sa longueur totale de cheminées (qui est le nombre de cellules du polyomino ayant au moins un voisin Est ou Ouest dans le polyomino), est également Q-comptable : si l'on fait formellement la différence entre les deux noms de paramètres, nous obtenons une décomposition faisant intervenir des coefficients négatifs. Toutefois, il convient de remarquer que le symbole N n'est pas accessible à partir de P dans cette grammaire. Par conséquent, les paramètres $p_{R_i^{(1)} R_j}$, pour $i = 4, 7, 9, 10$ et $5 \leq j \leq 12$, sont identiquement nuls, et $p_{R_i^{(1)} R_3} = p_{R_i}$ pour $i = 4, 7, 9, 10$. Nous obtenons alors une décomposition de l'aire en somme de deux paramètres Q-comptables, la longueur de cheminées ℓc et la partie "solide" as (les cellules ayant au moins un voisin latéral dans le polyomino) :

$$\begin{cases} aire &= \ell c + as \\ \ell c &= p_{R_1} + p_{R_4} + p_{R_7} + p_{R_9} + p_{R_{10}} \\ as &= p_{\left(R_8^{(1)} + R_{11}^{(1)} + R_{12}^{(1)}\right).(R_3+R_5+R_6+R_7+R_8+R_9+R_{10}+R_{11}+R_{12})} \\ & \quad + p_{R_3+R_5+R_6+R_7+R_8+R_9+R_{10}+R_{11}+R_{12}} \end{cases}$$

Les équations fournies par cette grammaire étant de faible degré, il n'est pas difficile de les résoudre et d'obtenir les séries génératrices suivant largeur et périmètre vertical.

Bibliographie

[1] A.V. Aho and J.D. Ullman. *The Theory of Parsing, Translation and Compiling. Vol. 1: Parsing.* Prentice-Hall, 1972.

[2] D. André. Solution d'un problème posé par M. Bertrand. *C.R. Acad. Sc.*, pages 436–437, 1887.

[3] G.E. Andrews. Identities in combinatorics II: A q-analog of the Lagrange inversion theorem. *Proc. Amer. Math. Soc*, 53:240–245, 1975.

[4] G.E. Andrews. *q-series: their Development and application in Analysis, Number Theory, Combinatorics, Physics, and Computer Algebra.* American Mathematical Society, 1986.

[5] J.-M. Autebert. *Langages algébriques.* Masson, 1987.

[6] E. Barcucci, A. Del Lungo, E. Pergola, and R. Pinzani. A construction for enumerating k-coloured Motzkin paths. In *Computing and Combinatorics, First annual conference, COCOON 95, Proceedings*, number 959 in Lecture Notes in Computer Science, pages 254–263. Springer, 1995.

[7] E. Barcucci, A. Del Lungo, E. Pergola, and R. Pinzani. A methodology for plane tree enumeration. *Discrete Math.*, 180(1-3):45–64, 1998.

[8] E.A. Bender. Central and local limit theorems applied to asymptotic enumeration. *J. Combin. Theory Ser. A*, 15:91–111, 1973.

[9] E.A. Bender. Convex n-ominoes. *Discrete Math.*, 8:219–226, 1974.

[10] E.A. Bender and L.B. Richmond. Central and local limit theorems applied to asymptotic enumeration II: Multivariate generating functions. *J. of Comb. Th. A*, 34:255–265, 1983.

[11] J. Berstel. *Transductions and Context-Free Languages.* Teubner Studienbucher, Stuttgart, 1979.

[12] J. Bertrand. Solution d'un problème. *C.R. Acad. Sc.*, page 369, 1887.

[13] M. Bousquet-Mélou. q-énumération de polyominos convexes. Thèse de doctorat, Université Bordeaux 1, 1991.

[14] M. Bousquet-Mélou. A method for the enumeration of various classes of column-convex polygons. *Discrete Math.*, 154:1–25, 1996.

[15] M. Bousquet-Mélou. Rapport d'habilitation. Technical report, LaBRI, Université Bordeaux 1, 1996.

[16] M. Bousquet-Mélou and J.-M. Fédou. The generating function of convex polyominoes: The resolution of a q-differential system. *Discrete Math.*, 137(1-3):53–75, 1995.

[17] R. Brak and A.J. Guttmann. Algebraic approximants: A new method of series analysis. *J. Phys. A: Math. Gen.*, 23(24):1331–1337, 1990.

[18] N. Chomsky and M.-P. Schützenberger. The algebraic theory of context-free languages. In P. Braffort and D. Hirschberg, editors, *Computer Programming and Formal Systems*, pages 118–161. North-Holland, 1963.

[19] L. Chottin. Etude syntaxique de certains langages solutions d'équations avec opérateurs. *Theor. Comp. Sci.*, 5:51–84, 1977.

[20] L. Chottin and R. Cori. Une preuve combinatoire de la rationalité d'une série génératrice associée aux arbres. *RAIRO Informatique théorique et applications*, 16(2):113–128, 1982.

[21] L. Comtet. Calcul pratique des coefficients de Taylor d'une fonction algébrique. *Enseign. Math.*, 10:267–270, 1964.

[22] L. Comtet. *Advanced Combinatorics.* Reidel, 1974.

[23] R. Cori. Un code pour les graphes planaires et ses applications. *Astérisque*, 27, 1975.

[24] R. Cori and J. Richard. Enumération des graphes planaires à l'aide des séries formelles en variables non commutatives. *Discrete Math.*, 2:115–162, 1972.

[25] M. Delest. Generating functions for column-convex polyominoes. *J. of Comb. Th. A*, 48(1):12–31, 1988.

[26] M. Delest. Polyominoes and animals: some recent results. *J. of Math. Chem.*, 8:3–18, 1991.

[27] M. Delest, J.-P. Dubernard, and I. Dutour. Parallelogram polyominoes and corners. *J. Symbolic Computation*, 20(5–6):503–515, 1995.

[28] M. Delest and J.-M. Fédou. Attribute grammars are useful for combinatorics. *J. Theor. Comput. Sci.*, 98:65–76, 1992.

[29] M. Delest and J.-M. Fédou. Enumeration of skew Ferrers diagrams. *Discrete Math.*, 112:65–79, 1993.

[30] M. Delest, D. Gouyou-Beauchamps, and B. Vauquelin. Enumeration of parallelogram polyominoes with given bond and site perimeter. *Graphs and Combinatorics*, 3(4):325–339, 1987.

[31] M. Delest and X. Viennot. Algebraic languages and polyominoes enumeration. *Theor. Comp. Sci.*, 34:169–206, 1984.

[32] A. Denise and R. Simion. Two combinatorial statistics on Dyck paths. *Discrete Math.*, 137:155–176, 1995.

[33] J. Dieudonné. *Calcul Infinitésimal.* Hermann, Paris, 1968.

[34] M. Drmota. Asymptotic distributions and a multivariate Darboux method in enumeration problems. *J. Combin. Theory Ser. A*, 67:169–184, 1994.

[35] M. Drmota. Systems of functional equations. *Random Structures and Algorithms*, 10:103–124, 1997.

[36] I. Dutour. Grammaires d'objets: énumération, bijections et génération aléatoire. Thèse de doctorat, Université Bordeaux 1, 1996.

[37] J.-M. Fédou. Grammaires et q-énumérations de polyominos. Thèse de doctorat, Université Bordeaux 1, 1989.

[38] S. Feretić. The column-convex polyominoes perimeter generating function for everybody. *Croatica Chemica Acta*, 69:741–756, 1996.

[39] S. Feretić. A new way of counting the column-convex polyominoes by perimeter. *Discrete Math.*, 180:173–184, 1998.

[40] P. Flajolet. Analytic models and ambiguity of context-free languages. *Theor. Comp. Sci.*, 49:283–309, 1987.

[41] P. Flajolet and A. Odlyzko. Singularity analysis of generating functions. *SIAM J. Discrete Math.*, 3:216–240, 1990.

[42] P. Flajolet, B. Salvy, and P. Zimmermann. Lambda-Upsilon-Omega: An assistant algorithms analyzer. In T. Mora, editor, *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes*, volume 357 of *Lect. Notes Comput. Sci.*, pages 201–212. Springer Verlag, 1989.

[43] P. Flajolet, B. Salvy, and P. Zimmermann. Lambda-Upsilon-Omega: The 1989 cookbook. Rapport technique 1073, Institut National de Recherche en Informatique et en Automatique, 1989.

[44] P. Flajolet, B. Salvy, and P. Zimmermann. Automatic average-case analysis of algorithms. *Theor. Comp. Sci.*, 79(1):37–109, 1991.

[45] P. Flajolet and R. Sedgewick. The average case analysis of algorithms: Complex asymptotics and generating functions. Rapport de recherche 2026, Institut National de Recherche en Informatique et en Automatique, 1993.

[46] P. Flajolet and R. Sedgewick. The average case analysis of algorithms: Multivariate asymptotics and limit distributions. Rapport de recherche 3162, Institut National de Recherche en Informatique et en Automatique, 1997.

[47] P. Flajolet, P. Zimmermann, and B. Van Cutsem. A calculus for the random generation of combinatorial structures. *Theor. Comp. Sci.*, 132:1–35, 1994.

[48] A. Garsia. A q-analogue of the Lagrange inversion formula. *Houston Journal of Mathematics*, 7:205–237, 1981.

[49] I. Gessel. A noncommutative generalization and q-analog of the Lagrange inversion formula. *Trans. Amer. Math. Soc*, 257:455–482, 1980.

[50] I. Gessel. A combinatorial proof of the multivariable Lagrange inversion formula. *J. of Comb. Th. A*, 45:178–195, 1987.

[51] I. Gessel and D. Stanton. Applications of q-Lagrange inversion to basic hypergeometric series. *Trans. Amer. Math. Soc*, 277(1):173–201, 1983.

[52] S.W. Golomb. Checker boards and polyominoes. *Amer. Math. Monthly*, 61(10):675–682, 1954.

[53] I.J. Good. Generalizations to several variables of Lagrange's expansion, with applications to stochastic processes. *J. Proc. Camb. Philos. Soc.*, pages 367–380, 1960.

[54] I.P. Goulden and D.M. Jackson. *Combinatorial Enumeration*. John Wiley and Sons, 1983.

[55] A.J. Guttmann. Planar polygons: Regular, convex, almost convex, staircase and row convex. In *Proceedings of the 1991 International Symposium in Statistical Physics*, volume 248 of *Published Conference Proceedings*, pages 12–33. America Institute of Physics, 1991.

[56] M.A. Harrison. *Introduction to Formal Language Theory*. Addison-Wesley, Reading, Mass., 1978.

[57] K. Kendig. *Elementary Algebraic Geometry*. Springer, New York, 1977.

[58] D.A. Klarner. Some results concerning polyominoes. *Fibonacci Quart.*, 3:9–20, 1965.

[59] D.A. Klarner. Cell growth problems. *Canad. J. Math.*, 19:851–863, 1967.

[60] D.A. Klarner and R.L. Rivest. Asymptotic bounds for the number of convex n-ominoes. *Discrete Math.*, 8:31–40, 1974.

[61] D.E. Knuth. Semantics of context-free languages. *Math. Systems Theory*, 2:127–145, 1968.

[62] C. Krattenthaler. A new q-Lagrange formula and some applications. *Proc. Amer. Math. Soc.*, 90:338–344, 1984.

[63] M. Marcus. Cartes, hypercartes et diagrammes de cordes. Thèse de doctorat, Université de Bordeaux 1, 1997.

[64] A. Odlyzko. Periodic oscillations of coefficients of power series that satisfy functional equations. *Adv. in Math.*, 44:180–205, 1982.

[65] W. Ogden. A helpful result for proving inherent ambiguity. *Math. Syst. Theory*, 2:191–194, 1968.

[66] R.J. Parikh. On context-free languages. *J. Assoc. Comput. Mach*, 13:570–580, 1966.

[67] G. Pólya. On the number of certain lattice polygons. *J. Combinatorial Theory*, 6:102–105, 1969.

[68] T. Prellberg. Uniform q-series asymptotics for staircase polygons. *J. Phys. A: Math. Gen.*, 28:1289–1304, 1995.

[69] T. Prellberg and R. Brak. Critical exponents from non-linear functional equations for partially directed cluster models. *J. Statist. Phys.*, 78:701–730, 1995.

[70] G. N. Raney. Functional composition patterns and power series reversion. *Trans. Amer. Math. Soc.*, 94:441–451, 1960.

[71] R.C. Read. Contributions to the cell growth problem. *Canad. J. Math.*, 14:1–20, 1962.

[72] D.H. Redelmeier. Counting polyominoes: Yet another attack. *Discrete Math.*, 36:191–203, 1981.

[73] M.-P. Schützenberger. Certain elementary families of automata. In *Proc. Symp. on Mathematical Theory of Automata*, pages 139–153. Polytechnic Institute of Brooklyn, 1962.

[74] M.-P. Schützenberger. On context-free languages and push-down automata. *Information and Control*, 6:246–264, 1963.

[75] N.J.A. Sloane. *A Handbook of integer sequences.* Academic Press, 1973.

[76] N.J.A. Sloane and S. Plouffe. *The Encyclopedia of Integer Sequences.* Academic Press, 1995.

[77] H.N.V. Temperley. Combinatorial problems suggested by the statistical mechanics of domains and of rubber-like molecules. *Phys. Rev.*, 103:1–16, 1956.

[78] W.T. Tutte. A census of planar maps. *Canad. J. Math*, 15:249–271, 1963.

[79] X. Viennot. A survey of polyomino enumeration. In P. Leroux and C. Reutenauer, editors, *Actes du 4e colloque Séries formelles et combinatoire algébrique*, volume 11 of *Publications du LaCIM*, pages 399–420, Montréal, 1992.

[80] T. R. S. Walsh and A. B. Lehman. Counting rooted maps by genus. III: Nonseparable maps. *J. Comb. Theory, Ser. B*, 18:222–259, 1975.

[81] E.T. Whittaker and G.N. Watson. *A Course of Modern Analysis.* Cambridge University Press, 4 edition, 1935.

Q-grammaires: un outil pour l'énumération

Résumé : Cette thèse se situe dans le domaine de la combinatoire énumérative. Les Q-grammaires constituent une extension de la méthode DSV: les objets à énumérer sont codés par les mots d'un langage algébrique. Nous nous concentrons sur l'énumération suivant plusieurs paramètres d'objets codés par des mots. Le formalisme introduit permet d'écrire, pour les séries génératrices suivant plusieurs paramètres, des équations fonctionnelles non nécessairement algébriques, et dont les solutions ne sont généralement pas des séries algébriques.

Les paramètres d'énumération, appelés Q-comptables, sont des paramètres cumulatifs définis au moyen d'attributs synthétisés, et peuvent être interprétés comme comptant certaines familles de sommets dans les arbres de dérivation. Essentiellement, chaque utilisation d'une règle de dérivation fait croître un paramètre Q-comptable d'une quantité qui doit être un paramètre Q-comptable de rang inférieur. Les paramètres Q-comptables comprennent le nombre d'occurrences d'une lettre donnée, et, dans le cas de mots codant des chemins discrets, peuvent inclure des statistiques telles que l'aire de la surface délimitée par le chemin, ou son moment d'inertie.

Nous étudions dans quelles conditions la grammaire utilisée pour engendrer un langage peut être modifiée sans que certains paramètres ne perdent leur caractère Q-comptable. Nous montrons qu'il est possible de mettre les grammaires sous des formes normales sans perte de paramètres Q-comptables.

Sont également abordés le calcul de séries de moments destinées à l'évaluation de valeurs moyennes de paramètres Q-comptables. En particulier, les séries de moments "projetées" sont algébriques, et s'écrivent explicitement en fonction des séries génératrices algébriques des langages engendrés par la grammaire.

Mots-clés : Combinatoire, langages algébriques, q-analogues, énumération, grammaires attribuées, séries génératrices.

Q-grammars: a tool for enumeration

Abstract : Our field of interest is that of enumerative combinatorics. We define Q-grammars, which are an extension of the DSV methodology : the objects to be enumerated are coded by the words in an unambiguous context-free language. Our main concern is with enumeration according to several parameters. Our method allows us to write functional equations for the generating functions which are not algebraic, but rather "multiple-q"-analogs of algebraic equations. The generating functions themselves are usually not algebraic.

Our enumeration parameters, called *Q-countable parameters*, are cumulative parameters which we define through synthesized attributes, and count certain families of nodes in the corresponding derivation trees. Each occurrence of a given derivation rule "increases" a Q-countable parameter by an amount which must be a Q-countable parameter of a lower rank. Q-countable parameters include the number of occurrencs of any given letter; when the words code lattice paths, the area delimited by the path, as well as some moments of inertia, can be other examples of Q-countable parameters.

We describe conditions under which the grammar generating a given language can be changed with no loss of Q-countable parameters. Among other things, we prove that the grammar can be written in a variety of normal forms without such loss.

The problem of computing moment series for Q-countable parameters, as a way of obtaining mean values for these parameters, is also covered. We prove that "projected" moment series are algebraic, and can be written as explicit functions of the algebraic generating functions for the language generated by the context-free grammar.

Keywords : Combinatorics, algebraic languages, q-analogs, enumeration, attribute grammars, generating functions.