Exploration de paramètres inconnus par des Q-grammaires

M. Delest et P. Duchon
LaBRI
Université Bordeaux 1 and CNRS
351 cours de la Libération
33405 Talence Cedex – France.
maylis,duchon@labri.u-bordeaux.fr

Résumé. Nous définissons la notion de paramètre Q-comptable à partir d'une extension de la méthodologie DSV. Nous montrons qu'ils peuvent être interprétés comme comptant des familles de sommets dans des arbres. Nous donnons un exemple montrant qu'ils peuvent exprimer des statistiques non algébriques comme des aires ou des moments d'inertie.

Abstract. From the DSV methodology, we define what we call Q-countable parameters. We show that they count some number of vertex sets in certain families of trees. An example shows that Q-countable parameters give a mean for catching inertia moments or area parameters.

1 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. . .

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. Lorsqu'on ne connait rien sur un paramètre plusieurs méthodes permettent de l'examiner. Des techniques d'approximation [4] recherchent, à partir de ses premières valeurs, une équation algébrique ou différentielle susceptible d'être satisfaite par la série génératrice. Dans la méthodologie DSV, développée initialement par Schützenberger dans [13, 14], 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. La notion de q-grammaire, introduite par Delest et Fédou dans [6], 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 [12]) 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.

Le travail présenté dans cet article se situe dans ce cadre. Nous étendons la notion de q-grammaire et montrons en quoi elles constituent un moyen d'approche de certains problèmes d'énumération suivant plusieurs paramètres. En particulier, le théorème 1 permet d'interpréter un paramètre en fonction de paramètres plus élémentaires. Cette méthode permet d'obtenir des q-équations dont les fonctions génératrices sont solutions. On trouvera en fin d'article, des exemples de q-équations obtenues.

2 Mots, grammaires, arbres

Cette section est un bref résumé des notions utiles pour comprendre la suite de ce texte. Des détails plus complets peuvent être trouvés dans [2], [11], [1].

Pour tout ensemble X, X^* désigne l'ensemble des suites finies d'éléments de X. Une telle suite w est appelée un *mot* de X^*, et sa longueur sera notée $|w|$. La suite de longueur nulle, appelée *mot vide*, sera notée ϵ. L'opération de concaténation fait de X^* un monoïde. 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 $|w|_A$. 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 $w_2 = \epsilon$.

On appelle *langage* sur X une partie L de X^*; X est alors appelé l'*alphabet* de L. Dans ce travail, nous nous intéresserons plus particulièrement aux langages qui sont engendrés par une *grammaire algébrique*. Une *grammaire algébrique* est un quadruplet

$$G = (X, N, \mathcal{R}, S)$$

tel que X et N sont deux alphabets disjoints appelés respectivement *alphabet terminal* et *non terminal*, S est un symbole non terminal appelé *axiome* et $\mathcal{R}$ est une partie finie de $N \times (X \cup N)^*$, dont chaque élément est appelé *règle de dérivation* et noté $U \to W$, avec $U \in N$ et $W \in (X \cup N)^*$. L'ensemble des mots engendrés par S est appelé langage algébrique engendré par G, on le note L(G). Nous nous intéressons à des grammaires non ambiguës.

Définition 1 *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 de $(X \cup N)^*$ tels que $S \overset{*}{\to} W_1 S_1 W_2 S_2 W_3$ ou $S \overset{*}{\to} W_1 S_2 W_2 S_1 W_3$.*

Dutour donne dans [9] 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 [9].

Exemple 1 [Langage de Dyck]

Un exemple fondamental de langage algébrique est le langage de Dyck, qui servira ici d'exemple. Le langage de Dyck est celui des systèmes correctement formés de parenthèses (Comtet [5]). 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 $G_1 = (\{a, b\}, \{D\}, \mathcal{R}, D)$, où $\mathcal{R}$ contient les deux règles de dérivation

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

A tout mot de Dyck, il est possible d'associer un chemin dans le plan. La lettre a se traduit par un pas Nord-Est, et la lettre b se traduit par un pas Sud-Est. Le chemin part de (0,0) et se termine alors sur la droite d'équation $y = 0$, sans avoir de sommet en-dessous de cette droite. Par définition, l'*aire* d'un chemin de Dyck est l'aire comprise entre ce chemin et l'axe d'équation $y = 0$.

L'ensemble des noms des règles de dérivation peut être identifié à un alphabet $\mathcal{R}$. Nous utiliserons la notion suivante :

Définition 2 *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 $R^{(k)}$.*

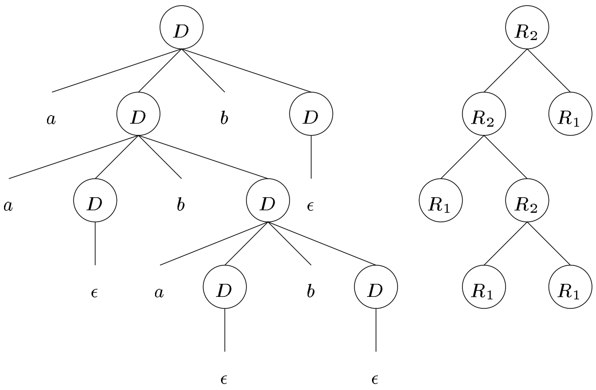
L'ensemble des noms des règles pointées constitue un nouvel alphabet $\mathcal{R}_p$.

Exemple 2 Dans la grammaire G_1 de l'exemple 1, $\mathcal{R} = \{R_1, R_2\}$. 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 $\mathcal{R}_p = \left\{R_2^{(1)}, R_2^{(2)}\right\}$.

Dans cet article, les arbres manipulés sont exclusivement des *arbres de dérivation* 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 étiqueté par des symboles non terminaux. Les feuilles sont étiquetées par des lettres terminales. Le mot associé à l'arbre est obtenu en lisant les étiquettes des feuilles dans l'ordre symétrique.

Dans cet article, nous étiquetons les nœuds des arbres de dérivation par les "noms" des règles de dérivation utilisées. La connaissance des règles de dérivation rendant alors superflues les feuilles et leurs étiquettes, nous les omettons systématiquement. Cette démarche est à rapprocher de celle de Dutour-Fédou sur les grammaires d'objets [9].

Exemple 3 Pour le langage de Dyck de l'exemple 1, les arbres de dérivation de $G1$ sont exactement les *arbres binaires complets* (les sommets internes, étiquetés R_2, ont 2 fils, et les feuilles sont étiquetées R_1). La figure 1 montre les deux formes possibles d'arbres de dérivation pour le mot $w = aababb$ dans la grammaire classique engendrant le langage de Dyck.



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

Fig. 1. Exemples d'arbres de dérivation

Soit $\mathcal{A}$ un arbre de dérivation de la grammaire G. Une *chaîne de longueur* k de $\mathcal{A}$ est un k-uplet $(s_1, \ldots, s_k)$ 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 $R_1^{(d_1)} \ldots R_{k-1}^{(d_{k-1})} R_k \in \mathcal{R}_p^{k-1}\mathcal{R}$, 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.

Exemple 4 Dans l'arbre de dérivation (b) de la figure 1, il y a deux chaines de type $R_2^{(1)} R_2$.

3 Grammaires attribuées, Q-grammaires et paramètres Q-comptables

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 [12].

Définition 3 *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* $\mathcal{D}_{U_1} \times \ldots \times \mathcal{D}_{U_k}$ *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 $\left(G, (T_U)_{U\in N}\right)$ 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. Dans la suite, 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 $1 \le i \le n$, tels que

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

Soit p un paramètre défini sur la réunion disjointe 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 4 *Soit u un mot d'un langage engendré par une U-dérivation, on appelle* terme de croissance de p pour *u, la quantité*

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

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. Notre travail a pour cadre le cas où chaque paramètre étudié peut être "simplement" défini au moyen de ses termes de croissance. Cela exclut donc les constructions utilisant des fonctions maximum ou minimum.

Exemple 5 Considérons de nouveau le langage de Dyck, engendré par les deux grammaires G_1 (voir exemple 1) 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 p' ("nombre de facteurs ab") 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 de ce 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 $w_1 = \epsilon$.

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. 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 5 (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 $\theta_p(u) = c_R + p_1(u_1) + \cdots + p_n(u_n)$.*

De plus, un paramètre de rang 1 a l'une au moins des constantes c_R non nulle et un paramètre de rang $k+1$ a l'un au moins des paramètres utilisés 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. Notons que 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.

La définition ci-dessous généralise celle des q-grammaires définie dans [10, 6] :

Définition 6 *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 $p_1, \ldots, p_{i-1}$.*

Exemple 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

$$R : U \to w_0 U_1 w_1 \ldots U_n w_n$$

est la constante $\theta_p = c_R = |w_0 w_1 \ldots w_n|_{X'}$.

Exemple 7 Reprenons la grammaire G_1 de l'exemple 1 (grammaire classique engendrant le langage de Dyck).

L'aire des chemins de Dyck qui est l'aire comprise entre ce chemin et l'axe d'équation $y = 0$, est Q-comptable 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 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.

Définissons 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 donné par

- 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 a pour rang 3.

4 Interprétation de paramètres Q-comptables

Il est naturel de rechercher une description atomique des paramètres Q-comptables.

Définition 7 *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.*

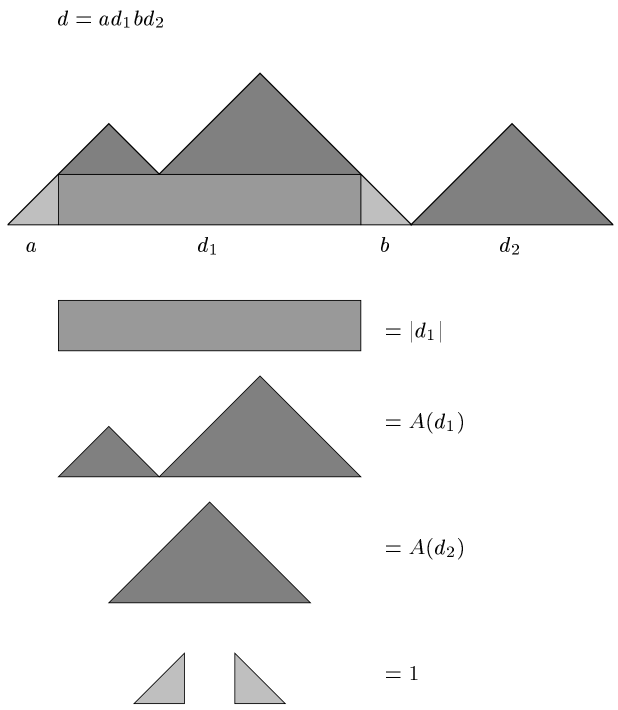


Fig. 2. Règle de calcul de l'aire d'un chemin de Dyck

Nous avons alors la

Proposition 1. *Tout paramètre élémentaire peut être décrit par un mot de $\mathcal{R}_p^*\mathcal{R}$.*

Preuve. Soit 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. Le paramètre élémentaire dont le nom est W est noté p_W. On a alors la proposition suivante :

Proposition 2. *Tout paramètre Q-comptable de rang k s'écrit 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

$$\theta_p(w_0u_1 \dots u_{\alpha(R)}w_{\alpha(R)}) = c_R + \sum_{i=1}^{\alpha(R)} p_{R,i}(u_i),$$

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 :

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

□

Nous allons donner une interprétation formelle de tout paramètre élémentaire et donc, par suite, nous aurons une interprétation de tout paramètre Q-comptable.

Proposition 3. *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 c'est immédiat.

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 $p'(s_1)$. Par suite, $p_W(w)$ est le nombre de chaînes de type W de l'arbre de dérivation de w. □

5 Propriétés des paramètres Q-comptables

Dans ce paragraphe, nous donnons quelques propriétés qui permettent pour un paramètre inconnu a priori de retrouver sa décomposition en paramètres élémentaires. Le lecteur trouvera dans [8] la nature des calculs asymptotiques que l'on peut effectuer sur des paramètres Q-comptables.

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

Proposition 4. *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,*

$$p(w) \leq K|w|^k. \tag{2}$$

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. □

On appellera *rang minimal* d'un paramètre Q-comptable, la plus petite constante positive k.

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

Théorème 1 *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.

Ce théorème se démontre en étudiant les propriétés d'accessibilité des symboles dans la grammaire. Calculer le rang minimal d'un paramètre se ramène alors à déterminer quels symboles sont accessibles à partir desquels, et quels couples de symboles sont accessibles à partir desquels.

Soit p un paramètre défini sur un langage indépendament de toute grammaire. Supposons de plus que la distribution de ce paramètre ne soit pas connue. Le théorème 1 permet dans la pratique de déterminer si p, 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 généralement 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.

Exemple 8 Considérons, pour le langage de Dyck engendré par la grammaire G_2, le paramètre *hauteur maximale* h_m, défini par

$$h_m(w) = \max\left\{|w'|_a - |w'|_b : w' \text{ facteur gauche de } w\right\}$$

(le nom de hauteur maximale provient de l'interprétation des mots de Dyck comme chemins : $h_m(w)$ est alors la plus grande hauteur atteinte par le chemin).

Le rang minimal de ce paramètre est 1 : en effet, $h_m(a^n b^n) = n$, et n est la plus grande hauteur que puisse atteindre un mot de Dyck de longueur $2n$. Il convient donc, pour déterminer si h_m est ou non un paramètre Q-comptable, de chercher parmi les combinaisons linéaires de paramètres élémentaires de rang 1. Le tableau des valeurs de ces paramètres, pour les mots de Dyck de longueurs 0, 2, 4 et 6, ainsi que le mot $w = aabbaabb$, est le suivant :

w	h_m	p_{R_1}	p_{R_2}	p_{R_3}	p_{R_4}	p_{R_5}	p_{R_6}
ϵ	0	1	0	0	0	0	0
ab	1	0	1	1	0	0	0
$abab$	1	0	1	1	1	0	0
$aabb$	2	0	1	1	0	1	0
$ababab$	1	0	1	1	2	0	0
$abaabb$	2	0	1	1	1	1	0
$aababb$	2	0	1	1	1	1	0
$aabbab$	2	0	1	2	0	0	1
$aaabbb$	3	0	1	1	0	2	0
$aabbaabb$	2	0	1	3	0	1	1

Les 9 premières lignes ne laissent comme seules possibilités de décomposition de h_m, que $p_{R_2} + p_{R_5} + p_{R_6}$ et $p_{R_3} + p_{R_5}$ (ces deux possibilités n'en font qu'une, car p_{R_3} est identiquement égal à $p_{R_2} + p_{R_6}$. La dernière ligne suffit à éliminer ces deux possibilités, et par conséquent h_m n'est pas un paramètre Q-comptable de la grammaire G_2. Ce résultat confirme l'idée qu'un paramètre défini au moyen d'une fonction maximum a peu de chances d'être Q-comptable.

Un autre exemple est celui du paramètre *somme des hauteurs de pics*, défini comme la somme des ordonnées des "pics" (sommets situés entre un pas Nord-Est et un pas Sud-Est) des chemins de Dyck. Ce paramètre correspond, pour un codage classique des polyominos parallélogrammes par les mots de Dyck, à l'em aire de ces polyominos. Ce paramètre, similaire à l'aire, est de rang minimal 2 (le mot $w = a^{n-1}(ab)^n b^{n-1}$, de longueur $4n - 2$, a pour somme des hauteurs de pics n^2), et par conséquent il convient de lui chercher une décomposition comme combinaison linéaire des paramètres élémentaires de rangs 1 et 2. Après élimination des redondances immédiates ($p_{R_2^{(1)} R_3} = p_{R_3}$, etc.), le nombre de ces paramètres élémentaires est 22.

En construisant un tableau similaire au précédent, incorporant les mots de longueur inférieure ou égale à 10, on obtient pour ce paramètre une seule décomposition possible :

$$p_{R_3} + p_{R_4} + p_{R_5^{(1)} R_3} + p_{R_5^{(1)} R_4} + p_{R_6^{(1)} R_3} + p_{R_6^{(1)} R_4}.$$

Il est alors immédiat, en procédant comme dans l'exemple 7, de vérifier que les règles de croissance correspondant à cette décomposition définissent bien la somme des hauteurs des pics.

6 Q-équations liées aux Q-grammaires

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 [10, 3, 6]. Nous décrivons une généralisation de ce genre de transformation, dans le cas où les variables formelles sont plus nombreuses.

Definition 1. *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

$$\sigma_{(x_i \leftarrow A_i)} F(x_1, \ldots, x_n) = F(A_1, \ldots, A_n). \tag{3}$$

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

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

Théorème 2 *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.

Nous donnons ici la méthode pour obtenir à partir d'une Q-grammaire, 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

$$R: \; p_m(w) = \sum_{i=1}^{a(R)} p_m(w_i) + C_{R,m} + \sum_{\substack{1 \leq j < m \\ 1 \leq i \leq a(R)}} C_{R,m,i,j} . p_j(w_i). \tag{4}$$

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 $w \in L_G(U)$,

$$v(w) = \prod_{1 \leq m \leq N+M} q_m^{p_m(w)} \tag{5}$$

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. Posons

$$\sigma_{R,i} = \sigma_{\left(q_j \leftarrow q_j \prod_{m=j+1}^{N+M} q_m^{C_{R,m,i,j}}\right)}. \tag{6}$$

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. On a :

$$v(w,U) = \left(\prod_{m=1}^{N+M} q_m^{C_{R,m}}\right) \prod_{i=1}^{k} \sigma_{R,i}(v(w_i,U_i)). \tag{7}$$

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. Nous avons donc,

$$U(q_1,\ldots,q_{N+M}) = \sum_{R\in\mathcal{R}(U)} \left(\prod_{m=1}^{N+M} q_m^{C_{R,m}}\right) \prod_{i=1}^{a(R)} \sigma_{R,i}\left(d(R,i)\right). \tag{8}$$

Le système formé des équations (8) (pour $U \in N$), est un Q-analogue du système algébrique fourni par la grammaire. □

Des détails de la preuve peuvent être consultés dans [7].

Exemple 9 Reprenons la grammaire G_2 de l'exemple 5. Dans les Q-équations, les variables suivantes comptent des paramètres 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*.

$$\begin{cases} D(x,q,r,s) = 1 + E(x,q,r,s), \\ E(x,q,r,s) = xqrs + xqrsE(x,q,r,s) + xqE(xq^2,q,rs,s) \\ \qquad\qquad\qquad +xqE(xq^2,q,rs,s)E(x,q,r,s). \end{cases}$$

A partir de la grammaire G_1 de l'exemple 1, en utilisant les résultats de l'exemple 7, on obtient en codant *l'aire* par q et *le moment d'inertie* par r :

$$D(x,q,r) = 1 + xqrD(x,q,r)D(xqr,qr,r)$$

References

1. A.V. Aho and J.D. Ullman. *The Theory of Parsing, Translation and Compiling. Vol. 1: Parsing*. Prentice-Hall, 1972.
2. J. Berstel. *Transductions and Context-Free Languages*. Teubner Studienbucher, Stuttgart, 1979.
3. M. Bousquet-Mélou. q-énumération de polyominos convexes. Thèse de doctorat, Université Bordeaux 1, 1991.
4. R. Brak and A.J. Guttmann. Algebraic approximants: A new method of series analysis. *J. Phys. A: Math. Gen.*, 23(24):1331–1337, 1990.
5. L. Comtet. *Advanced Combinatorics*. Reidel, 1974.
6. M. Delest and J.-M. Fédou. Attribute grammars are useful for combinatorics. *J. Theor. Comput. Sci.*, 98:65–76, 1992.
7. P. Duchon. Q-grammaires : un outil pour l'énumération. Thèse de doctorat, Université Bordeaux 1, 1998.
8. P. Duchon. q-grammars and wall polyominoes. Submitted to *Annals of Combinatorics*, 1998.
9. I. Dutour. Grammaires d'objets: énumération, bijections et génération aléatoire. Thèse de doctorat, Université Bordeaux 1, 1996.
10. J.-M. Fédou. Grammaires et q-énumérations de polyominos. Thèse de doctorat, Université Bordeaux 1, 1989.
11. S. Ginsburg. *The mathematical theory of context free languages*. McGraw-Hill, New York, 1966.
12. D.E. Knuth. Semantics of context-free languages. *Math. Systems Theory*, 2:127–145, 1968.
13. M.-P. Schützenberger. Certain elementary families of automata. In *Proc. Symp. on Mathematical Theory of Automata*, pages 139–153. Polytechnic Institute of Brooklyn, 1962.
14. M.-P. Schützenberger. On context-free languages and push-down automata. *Information and Control*, 6:246–264, 1963.