Grammaires synchrones

Parce qu'elles permettent de décrire et d'engendrer des langages — ou ensembles de mots — les grammaires formelles [Cho56] sont des outils centraux en théorie des langages [Aut94]. Le processus emprunté par une grammaire formelle pour engendrer un mot est simple : on commence à partir d'un mot particulier, appelé axiome, auquel on applique successivement des dérivations, c'est-à-dire des substitutions de certains de ses facteurs par d'autres mots, selon un ensemble fixé de règles, dites règles de substitutions. Les mots que l'on peut obtenir ainsi forment le langage engendré par la grammaire. Plusieurs sortes de grammaires ont été introduites et étudiées : les grammaires hors contexte, les grammaires contextuelles, les grammaires sans restrictions, pour ne citer que ces exemples, qui possèdent chacune d'elles leur propre niveau de généralité, et permettent d'engendrer et de décrire certains types de langages plutôt que d'autres [HMU00].

De manière similaire, certains types de grammaires sont pensés pour engendrer non plus des mots mais plutôt des arbres enracinés. Ces grammaires sont connues sous le nom de grammaires d'arbres [CDG+07]. Pour engendrer un arbre dans une telle grammaire, on commence à partir d'un arbre particulier, également appelé axiome, auquel on applique une série de substitutions en remplaçant ses feuilles par d'autres arbres, suivant ici aussi un ensemble de règles de substitution fixé.

L'objectif de ce chapitre est d'introduire un type particulier de grammaires dans le but d'engendrer des structures arborescentes selon un tout autre processus de génération. Dans nos grammaires, les substitutions doivent être *synchrones* : le principe est que, à chaque étape de substitution, *toutes* les feuilles de l'arbre sont substituées *simultanément* par de nouveaux arbres suivant un ensemble de règles de substitution fixé. Nous qualifions naturellement ces objets de *grammaires synchrones*. La motivation principale pour l'introduction de ces structures réside dans le fait que ces grammaires permettent d'engendrer des arbres tout en conservant un contrôle sur les hauteurs de chacun de ses sous-arbres. Rappelons à ce propos que la hauteur d'un arbre est la longueur du plus long chemin connectant sa racine à l'une de ses feuilles. De plus, sous certaines conditions, nous pouvons extraire, à partir d'une grammaire synchrone, une équation fonctionnelle de point fixe pour la série génératrice qui dénombre les arbres engendrés selon leur nombre de feuilles. Les résultats présentés ici sont utilisés à plusieurs reprises dans le chapitre 8, dans le but de dénombrer plusieurs types de structures arborescentes en rapport avec les arbres binaires équilibrés et le treillis de Tamari.

Ce chapitre est organisé comme suit. Dans le paragraphe 7.1, nous introduisons les arbres à bourgeons, les grammaires synchrones, et le principe de substitution synchrone. Le paragraphe 7.2 est consacré à la description d'un procédé qui permet d'obtenir une équation fonctionnelle de point fixe pour la série génératrice dénombrant les structures arborescentes engendrées par une grammaire synchrone. Nous décrivons aussi un algorithme qui permet de calculer ses premiers coefficients. Nous illustrons finalement dans le paragraphe 7.3, les notions présentées jusqu'alors en donnant trois exemples simples de grammaires synchrones : l'une engendre les arbres binaires parfaits, l'autre les arbres $2,3$-équilibrés, et la dernière, les arbres binaires équilibrés.

La plupart des résultats contenus dans ce chapitre ont été publiés dans [Gir10].

7.1 Arbres à bourgeons et grammaires synchrones

7.1.1 Arbres à bourgeons

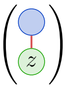
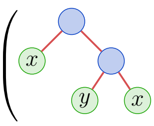
Rappelons qu'un *arbre plan* est un arbre enraciné plongé dans le plan. Autrement dit, les sous-arbres de chacun de ses nœuds sont totalement ordonnés de gauche à droite.

Définition 7.1.1. *Soit B un alphabet fini et non vide. Un* arbre à bourgeons *sur B — ou simplement* arbre à bourgeons *si le contexte est clair — est un arbre plan enraciné non vide dont les nœuds sans fils, appelés* bourgeons*, sont étiquetés sur B.*

Fixons pour ce chapitre un alphabet $B := \{\mathtt{b}_1, \dots, \mathtt{b}_k\}$ fini et non vide. Sauf mention contraire, tous les arbres à bourgeons considérés par la suite sont sur B. Soit D un arbre à bourgeons. Les *nœuds* de D sont ses sommets qui ne sont pas des bourgeons. L'ensemble des bourgeons de D est noté $\mathrm{Brg}(D)$. Si x est un bourgeon de D étiqueté par $\mathtt{b}_i$, nous dirons que x *est de type* $\mathtt{b}_i$. La *frontière* de D est la suite $(b_1, \dots, b_n)$ de ses bourgeons, lus de la gauche vers la droite. Si b est un bourgeon de D, son *évaluation* $\mathrm{ev}(b)$ est la lettre de B qui l'étiquette. Par extension, l'*évaluation* $\mathrm{ev}(D)$ de D est l'élément de l'algèbre commutative $\mathbb{Z}[B]$ des polynômes sur B défini par

$$\mathrm{ev}(D) := \prod_{b \,\in\, \mathrm{Brg}(D)} \mathrm{ev}(b). \tag{7.1.1}$$

Nous avons ainsi par exemple

$\mathrm{ev}\Big($  $\Big) = z$ et $\mathrm{ev}\Big($  $\Big) = x^2 y.$ (7.1.2)

La *taille* de D est le nombre de bourgeons qu'il contient. L'ensemble des arbres à bourgeons de taille n est noté $\mathcal{D}_n$, et l'ensemble de tous les arbres à bourgeons est noté $\mathcal{D}$. Nous seront amenés à considérer des *arbres à bourgeons étiquetés*, qui sont simplement des arbres à bourgeons dont les nœuds sont étiquetés sur un alphabet A auxiliaire.

Notons que l'ensemble $\mathcal{D}$ des arbres à bourgeons ne forme pas une classe combinatoire au sens de la définition 1.1.1 du chapitre 1. Il existe en effet pour tout $n \geqslant 1$ une infinité d'arbres à bourgeons de taille n. Nous pouvons par exemple construire une infinité d'arbres à bourgeons de taille 1 qui sont constitués d'une unique branche de longueur arbitrairement grande et qui possèdent un unique bourgeon.

7.1.2 Grammaires synchrones

Définitions et notations

Définition 7.1.2. *Une* grammaire synchrone *est un triplet $S := (B, a, R)$ où*
- *B est un alphabet fini et non vide, l'*alphabet des types de bourgeons*;*
- *a est un bourgeon étiqueté sur B, l'*axiome *de S ;*
- *$R \subseteq B \times \mathcal{D}$ est un ensemble fini tel que pour tout $\mathtt{b} \in B$, il existe au moins un arbre à bourgeons D tel que $(\mathtt{b}, D) \in R$. C'est l'*ensemble des règles de substitution *de S.*

Soit $S := (B, a, R)$ une grammaire synchrone. Par souci de lisibilité, nous allons utiliser la notation suivante pour les règles de substitution. Si $(\mathtt{b}, D)$ est une règle de substitution de S, nous la notons $\mathtt{b} \longmapsto_S D$ ou simplement $\mathtt{b} \longmapsto D$ si le contexte est fixé. De plus, par souci de concision, un ensemble de règles de substitution de la forme $\mathtt{b} \longmapsto_S D_1$, …, $\mathtt{b} \longmapsto_S D_n$ sera noté

$$\mathtt{b} \longmapsto_S D_1 + \cdots + D_n. \tag{7.1.3}$$

Dérivations et langages

Définition 7.1.3. *Soit $S := (B, a, R)$ une grammaire synchrone et D_0 un arbre à bourgeons de frontière $(b_1, \ldots, b_n)$ où $\mathrm{ev}(b_i) = \mathtt{b}_i$ pour tout $i \in [n]$. L'arbre à bourgeons D_1 est* dérivable *depuis D_0 dans S s'il existe dans R des règles de substitution $\mathtt{b}_1 \longmapsto T_1$, …, $\mathtt{b}_n \longmapsto T_n$ telles que, en substituant simultanément les bourgeons b_i de D_0 par la racine de T_i pour tout $i \in [n]$, on obtient D_1. Ceci est noté $D_0 \xrightarrow{S} D_1$.*

Définition 7.1.4. *Un arbre à bourgeons D est* engendré *par une grammaire synchrone $S := (B, a, R)$ s'il existe une suite $(D_1, \ldots, D_{\ell-1})$ d'arbres à bourgeons telle que*

$$a \xrightarrow{S} D_1 \xrightarrow{S} \cdots \xrightarrow{S} D_{\ell-1} \xrightarrow{S} D. \tag{7.1.4}$$

Nous dirons de plus que D est engendré par une dérivation de ℓ étapes *et que (7.1.4) constitue une dérivation de ℓ étapes. Nous dirons aussi dans ce cas que D est* engendré *par S.*

La grammaire synchrone S est *émondée* si pour tout $\mathtt{b} \in B$, il existe au moins un arbre à bourgeons D engendré par S qui contient un bourgeon de type $\mathtt{b}$. Dans ce qui suit, nous considérons uniquement des grammaires synchrones émondées, et de ce fait, nous ne mentionnerons plus ce qualificatif.

Définition 7.1.5. *Soit $S := (B, a, R)$ une grammaire synchrone. Le* langage de S d'ordre ℓ *est l'élément de* $\mathrm{Vect}(\mathcal{D})$*, l'espace vectoriel sur l'ensemble des arbres à bourgeons, défini par*

$$\mathcal{L}_S^{(\ell)} := \sum_{a \xrightarrow{S} D_1 \xrightarrow{S} \cdots \xrightarrow{S} D_\ell} D_\ell. \tag{7.1.5}$$

Par définition, le coefficient d'un arbre à bourgeons D dans $\mathcal{L}_S^{(\ell)}$ témoigne du nombre de façons d'engendrer D par des dérivations de ℓ étapes. Par extension, nous appelons *langage* de S l'ensemble $\mathcal{L}_S$ des arbres à bourgeons engendrés par S.

Graphes de génération

Définition 7.1.6. *Le* graphe de génération d'ordre ℓ *d'une grammaire synchrone S est le graphe orienté $\mathcal{G}_S^{(\ell)} := (V, E)$ défini par*

$$V := \bigcup_{0 \leqslant i \leqslant \ell} \left\{ D : \ D \textit{ apparaît dans } \mathcal{L}_S^{(i)} \right\}, \tag{7.1.6}$$

et

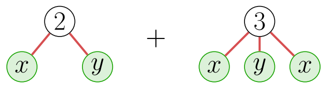
$$E := \left\{ (D_0, D_1) \in V^2 : D_0 \xrightarrow{S} D_1 \right\}. \tag{7.1.7}$$

Le graphe de génération *de S est le graphe $\mathcal{G}_S$ ainsi défini, avec $V := \mathcal{L}_S$.*

Le graphe $\mathcal{G}_S$ est potentiellement infini. Les graphes $\mathcal{G}_S^{(\ell)}$ et $\mathcal{G}_S$ possèdent exactement une source, l'axiome a.

Un exemple

Les notions introduites dans la suite du texte seront illustrées par l'intermédiaire de la grammaire synchrone $S_{\mathrm{epl}} := \left(\{x, y\}, x, R\right)$ où R contient les règles de substitution

$x \longmapsto$  , (7.1.8)

$$y \longmapsto x. \tag{7.1.9}$$

La Figure 7.1 montre une dérivation dans S_{epl}.

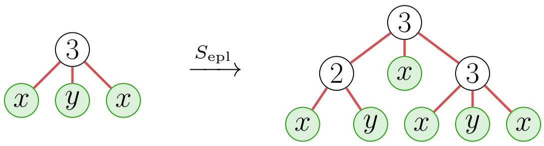


FIGURE 7.1 – Une dérivation dans la grammaire synchrone S_{epl}.

La Figure 7.2 montre le graphe de génération d'ordre 2 de S_{epl}.

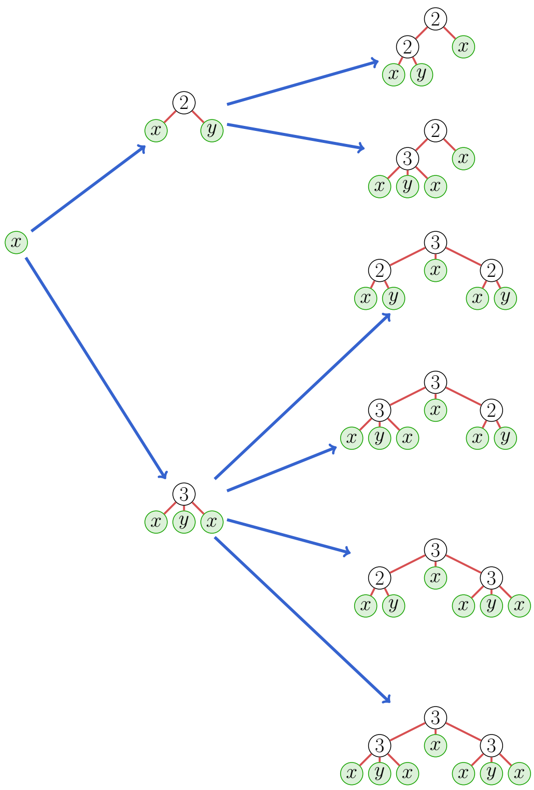


FIGURE 7.2 – Le graphe de génération d'ordre 2 de la grammaire synchrone S_{epl}.

7.1.3 Conditions sur les grammaires synchrones

Notre attention se porte maintenant sur les grammaires synchrones qui vérifient deux propriétés bien spécifiques que nous allons définir.

Grammaires synchrones localement finies

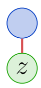
Définition 7.1.7. *Une grammaire synchrone $S := (B, a, R)$ est* localement finie *si pour tout $n \geqslant 1$, l'ensemble*

$$\left\{a \xrightarrow{S} D_1 \xrightarrow{S} \cdots \xrightarrow{S} D_\ell : \ell \geqslant 0 \text{ et } |D_\ell| = n\right\} \tag{7.1.10}$$

est fini. En d'autres termes, une grammaire synchrone est localement finie s'il existe, pour tout $n \geqslant 1$, uniquement un nombre fini de dérivations qui engendrent des arbres à bourgeons de taille n.

Ainsi, et à plus forte raison, une grammaire synchrone localement finie engendre pour tout $n \geqslant 1$ un nombre fini d'arbres à bourgeons de taille n ainsi qu'un nombre fini d'arbres à bourgeons ayant une évaluation donnée.

Il est clair que lorsque S est localement finie, son graphe de génération $\mathcal{G}_S$ est acyclique. En effet, par contraposée, si $\mathcal{G}_S$ admettait un cycle, il existerait alors une infinité de dérivations qui emprunteraient ce cycle pour engendrer un même arbre à bourgeons. Remarquons cependant que la réciproque est fausse : le graphe de génération de la grammaire synchrone $S := \left(\{z\}, z, R\right)$ où R contient la règle de substitution

$z \longmapsto$  , (7.1.11)

est constitué d'une unique branche, mais S engendre une infinité d'arbres à bourgeons différents, tous de taille 1.

Avant de donner un critère suffisant permettant d'affirmer qu'une grammaire synchrone est localement finie, rappelons ce qu'est un ordre monomial. Un *ordre monomial* est un ordre total $\leqslant$ défini sur un ensemble M de monômes, et qui vérifie les deux conditions suivantes :

(i) pour tous $x, y, z \in M$, $x \leqslant y$ implique $x \cdot z \leqslant y \cdot z$, où $\cdot$ est le produit de monômes de M ;

(ii) tout ensemble non vide de monômes de M possède un plus petit élément.

Un ordre total $\leqslant_B$ défini sur les lettres de B donne lieu à un ordre monomial sur les monômes de $\mathbb{Z}[B]$, que nous notons aussi $\leqslant_B$ par extension. En effet, si la relation $\leqslant_B$ est définie sur B par $\mathtt{b}_i \leqslant_B \mathtt{b}_j$ pour tout $1 \leqslant i \leqslant j \leqslant k$, nous avons, pour tous monômes $x := \mathtt{b}_1^{\alpha_1} \dots \mathtt{b}_k^{\alpha_k}$ et $y := \mathtt{b}_1^{\beta_1} \dots \mathtt{b}_k^{\beta_k}$,

$$x \leqslant_B y \qquad \text{si et seulement si} \qquad \begin{array}{c} x = y \\ \text{ou} \\ |x| < |y| \\ \text{ou} \\ \alpha_j < \beta_j \text{ pour un } j \in [k], \text{ et } \alpha_i = \beta_i \text{ pour tout } i \in [j-1]. \end{array} \tag{7.1.12}$$

Lemme 7.1.8. *Soit $S := (B, a, R)$ une grammaire synchrone. S'il existe un ordre monomial $\leqslant_B$ sur les monômes de $\mathbb{Z}[B]$ tel que, pour toute dérivation dans S de la forme*

$$a \xrightarrow{S} D_1 \xrightarrow{S} \cdots \xrightarrow{S} D, \tag{7.1.13}$$

nous avons

$$\mathrm{ev}(a) <_B \mathrm{ev}(D_1) <_B \cdots <_B \mathrm{ev}(D), \tag{7.1.14}$$

alors S est localement finie.

Démonstration. Soit u un mot sur B. Étant donné que l'ensemble B des types de bourgeons est fini, et que d'après (7.1.14), chaque dérivation dans S augmente strictement l'évaluation de tout arbre à bourgeon sur lequel elle est appliquée, il existe un entier ℓ tel que toute dérivation qui engendre un arbre à bourgeon ayant u comme évaluation fasse moins de ℓ étapes. Maintenant, comme l'ensemble R des règles de substitution est fini, le nombre de dérivations qui engendrent des arbres à bourgeons ayant u comme évaluation est fini. Le résultat suit finalement du fait que comme B est fini, il existe un nombre fini d'évaluations de longueur n, et donc, également un nombre fini de dérivations qui engendrent des arbres à bourgeons de taille n. □

Voici la condition suffisante qui permet d'affirmer qu'une grammaire synchrone est localement finie :

Lemme 7.1.9. *Soit $S := (B, a, R)$ une grammaire synchrone. S'il existe un ordre total $\leqslant_B$ sur B tel que, pour toute règle de substitution $\mathtt{b} \longmapsto D$ de R où D est un arbre à bourgeons de taille* 1*, nous avons $\mathtt{b} <_B \mathrm{ev}(D)$, alors S est localement finie.*

Démonstration. On étend tout d'abord l'ordre total $\leqslant_B$ sur B en un ordre monomial $\leqslant_B$ sur l'ensemble des monômes de $\mathbb{Z}[B]$ suivant (7.1.12).

Considérons maintenant un arbre à bourgeons D_0 engendré par S et un arbre à bourgeons D_1 dérivable depuis D_0. Montrons que l'on a bien $\mathrm{ev}(D_0) <_B \mathrm{ev}(D_1)$. Nous avons pour cela deux cas à considérer.

Cas 1. S'il existe au moins un bourgeon de D_0 qui est substitué par un arbre à bourgeons de taille supérieure à 2, alors $|\,\mathrm{ev}(D_0)| < |\,\mathrm{ev}(D_1)|$ et par conséquent, $\mathrm{ev}(D_0) <_B \mathrm{ev}(D_1)$.

Cas 2. Sinon, D_0 et D_1 sont de la même taille. Par hypothèse, tous les bourgeons de la frontière $(b_1, \ldots, b_n)$ de D_0 sont substitués par n arbres à bourgeons de taille 1 contenant les bourgeons c_1, …, c_n tels que $\mathrm{ev}(b_i) <_B \mathrm{ev}(c_i)$ pour tout $i \in [n]$. Ainsi, nous avons $\mathrm{ev}(D_0) <_B \mathrm{ev}(D_1)$.

Le résultat suit finalement du lemme 7.1.8. □

À titre d'exemple, la grammaire synchrone S_{epl}, en ordonnant son alphabet de bourgeons $B := \{x, y\}$ par $y <_B x$, vérifie la condition du lemme 7.1.9 et est ainsi localement finie.

Grammaires synchrones non ambiguës

Définition 7.1.10. *Une grammaire synchrone S est* non ambiguë *si pour tout arbre à bourgeons D, il existe au plus un entier positif ℓ et une suite $(D_1, \ldots, D_{\ell-1})$ telle que (7.1.4) est établie.*

Étant donné qu'il existe par définition dans le graphe de génération $\mathcal{G}_S$ d'une grammaire synchrone non ambiguë S exactement un chemin qui connecte l'axiome de S à tout arbre à bourgeons engendré par S, $\mathcal{G}_S$ est un arbre enraciné en son axiome. De plus, la réciproque est vraie : toute grammaire synchrone dont le graphe de génération est un arbre est d'emblée non ambiguë.

Nous donnons ici un critère suffisant pour affirmer qu'une grammaire synchrone localement finie est non ambiguë.

Lemme 7.1.11. *Soit $S := (B, a, R)$ une grammaire synchrone localement finie. Si pour tout $\mathtt{b} \in B$ et pour toutes règles de substitution $\mathtt{b} \longmapsto T_0$ et $\mathtt{b} \longmapsto T_1$ de R où $T_0 \neq T_1$, il existe deux nœuds différents dans T_0 et T_1 qui se trouvent aux mêmes positions dans leurs arbres respectifs, alors, S est non ambiguë.*

Démonstration. Soit D un arbre à bourgeons engendré par S, et D_0 et D_1 deux arbres à bourgeons différents et dérivables depuis D. Parmi d'autres substitutions, D_0 (resp. D_1) est obtenu en substituant un de ses bourgeons par un arbre à bourgeons T_0 (resp. T_1), et par hypothèse, il y a aux mêmes positions dans T_0 et T_1 deux nœuds différents. Ceci montre que tout arbre à bourgeons obtenu en réalisant une suite de dérivations depuis D_0 et tout arbre à bourgeons obtenu en réalisant une suite de dérivations depuis D_1 sont différents puisqu'il existe dans ces arbres deux nœuds différents aux mêmes positions. De plus, comme S est localement finie, son graphe de génération est acyclique, ce qui montre que S est non ambiguë. □

À titre d'exemple, le lemme 7.1.11 montre que la grammaire S_{epl} est non ambiguë. En effet, celle-ci est à la fois localement finie et de plus, seuls les bourgeons de type x peuvent être substitués par des arbres à bourgeons différents et chacun d'eux possède une racine différente : l'une est d'arité 2 tandis que l'autre est d'arité 3.

7.2 Grammaires synchrones et dénombrement

La principale raison qui motive l'introduction des grammaires synchrones réside dans le fait qu'elles constituent un nouvel outil pour dénombrer des familles d'objets combinatoires. Dans cet objectif, nous montrons dans ce paragraphe comment obtenir, à partir d'une grammaire synchrone localement finie et non ambiguë, une série génératrice qui dénombre selon leur évaluation les arbres à bourgeons que celle-ci engendre.

Les séries génératrices obtenues à partir des grammaires synchrones s'expriment en réalité comme solutions d'équations fonctionnelles de point fixe. Le calcul des coefficients de ces séries se fait par *itération*. Nous proposons de ce fait un procédé qui permet de les calculer.

Nous terminons ce paragraphe en illustrant le fait que les grammaires synchrones sont des objets qui se prêtent particulièrement bien à des spécialisations et à des raffinements dans les objets qu'elles dénombrent. Il est ainsi possible de dénombrer une famille d'objets selon des statistiques.

7.2.1 Série génératrice d'une grammaire synchrone

Définition 7.2.1. *Soit $S := (B, a, R)$ une grammaire synchrone. La* série génératrice d'ordre ℓ *de S, notée $\mathcal{S}_S^{(\ell)}$, est le polynôme de $\mathbb{Z}[B]$ défini par*

$$\mathcal{S}_S^{(\ell)}(\mathtt{b}_1, \dots, \mathtt{b}_k) := \mathrm{ev}\left(\mathcal{L}_S^{(\ell)}\right), \tag{7.2.1}$$

où le symbole ev *du membre droit de (7.2.1) désigne l'évaluation sur les arbres à bourgeons, étendue par linéarité. De plus, dans le cas où S est localement finie, la* série génératrice *de S, notée $\mathcal{S}_S$, est l'élément de $\mathbb{Z}[[B]]$ défini par*

$$\mathcal{S}_S(\mathtt{b}_1, \dots, \mathtt{b}_k) := \sum_{\ell \geqslant 0} \mathcal{S}_S^{(\ell)}(\mathtt{b}_1, \dots, \mathtt{b}_k). \tag{7.2.2}$$

Soit $S := (B, a, R)$ une grammaire synchrone localement finie. Comme mentionné précédemment, le fait que S est localement finie implique que le nombre d'arbres à bourgeons engendrés par S ayant une évaluation donnée est fini. Ceci montre que la série $\mathcal{S}_S$ est bien définie, *i.e.*, tous les coefficients de $\mathcal{S}_S$ sont dans $\mathbb{N}$. De plus, remarquons que si S est également non ambiguë, alors pour tout monôme $u := \mathtt{b}_1^{\alpha_1} \dots \mathtt{b}_k^{\alpha_k}$, le coefficient $[u]\mathcal{S}_S$ est le nombre d'arbres à bourgeons engendrées par S qui ont u comme évaluation, *i.e.*, une frontière constituée de α_i occurrences de bourgeons de type $\mathtt{b}_i$, pour tout $i \in [k]$.

Par exemple, les premières séries génératrices d'ordre ℓ de S_{epl} sont

$$\mathcal{S}^{(0)}_{S_{\mathrm{epl}}}(x,y) = x, \tag{7.2.3}$$
$$\mathcal{S}^{(1)}_{S_{\mathrm{epl}}}(x,y) = xy + x^2y, \tag{7.2.4}$$
$$\mathcal{S}^{(2)}_{S_{\mathrm{epl}}}(x,y) = x^2y + x^3y + x^3y^2 + 2x^4y^2 + x^5y^2. \tag{7.2.5}$$

et sa série génératrice $\mathcal{S}_{S_{\mathrm{epl}}}$ est de la forme

$$\mathcal{S}_{S_{\mathrm{epl}}}(x,y) = x + xy + 2x^2y + x^3y + x^3y^2 + 2x^4y^2 + x^5y^2 + \cdots . \tag{7.2.6}$$

7.2.2 Calculer la série génératrice d'une grammaire synchrone

Soit $S := (B, a, R)$ une grammaire synchrone. Posons maintenant la notation suivante. Pour tout $\mathtt{b} \in B$, $\mathrm{subs}(\mathtt{b})$ désigne le polynôme de $\mathbb{Z}[B]$ défini par

$$\mathrm{subs}(\mathtt{b}) := \sum_{(\mathtt{b},D) \in R} \mathrm{ev}(D). \tag{7.2.7}$$

Nous avons par exemple pour S_{epl},

$$\mathrm{subs}(x) = xy + x^2y, \tag{7.2.8}$$
$$\mathrm{subs}(y) = x. \tag{7.2.9}$$

Le lemme suivant donne un moyen de calculer la série génératrice d'une grammaire synchrone d'ordre ℓ par *itération*. En d'autre termes, le procédé décrit ici permet le calcul de la série génératrice d'ordre $\ell + 1$ en se basant sur la série génératrice d'ordre ℓ.

Lemme 7.2.2. *Soit $S := (B, a, R)$ une grammaire synchrone. Pour tout $\ell \geqslant 0$, $\mathcal{S}^{(\ell)}_S$ vérifie*

$$\mathcal{S}^{(\ell)}_S(\mathtt{b}_1, \dots, \mathtt{b}_k) = \begin{cases} \mathrm{ev}(a) & \textit{si } \ell = 0, \\ \mathcal{S}^{(\ell-1)}_S(\mathrm{subs}(\mathtt{b}_1), \dots, \mathrm{subs}(\mathtt{b}_k)) & \textit{sinon.} \end{cases} \tag{7.2.10}$$

Démonstration. Si $\ell = 0$, le seul arbre à bourgeons engendré par une dérivation de zéro étapes dans S est son axiome a. De ce fait, le lemme est correct dans ce cas.

Soit $\ell \geqslant 1$ et supposons qu'il existe la suite de dérivations suivante dans S :

$$a \xrightarrow{S} D_1 \xrightarrow{S} \cdots \xrightarrow{S} D_{\ell-1} \xrightarrow{S} D_\ell. \tag{7.2.11}$$

Soit n la taille de l'arbre à bourgeons $D_{\ell-1}$. Par définition, D_ℓ est obtenu en substituant les bourgeons b_i de $D_{\ell-1}$ par des arbres à bourgeons T_i pour tout $i \in [n]$. Du point de vue des polynômes, le monôme $\mathrm{ev}(D_\ell)$ est obtenu par les substitutions polynomiales $\mathrm{ev}(b_i) \hookleftarrow \mathrm{ev}(T_i)$ dans $\mathcal{S}^{(\ell-1)}_S$. Par conséquent, le polynôme $\mathcal{S}^{(\ell)}_S$ est obtenu à partir de $\mathcal{S}^{(\ell-1)}_S$ en réalisant la substitution polynomiale $\mathtt{b} \hookleftarrow \mathrm{subs}(\mathtt{b})$ pour tout $\mathtt{b} \in B$, ce qui montre (7.2.10). □

Proposition 7.2.3. *Soit $S := (B, a, R)$ une grammaire synchrone localement finie. La série génératrice $\mathcal{S}_S$ vérifie l'équation fonctionnelle de point fixe*

$$\mathcal{S}_S(\mathtt{b}_1, \dots, \mathtt{b}_k) = \mathrm{ev}(a) + \mathcal{S}_S(\mathrm{subs}(\mathtt{b}_1), \dots, \mathrm{subs}(\mathtt{b}_k)). \tag{7.2.12}$$

Démonstration. En utilisant le lemme 7.2.2, nous obtenons

$$
\begin{aligned}
\mathcal{S}_S(\mathtt{b}_1, \ldots, \mathtt{b}_k) &= \sum_{\ell \geqslant 0} \mathcal{S}_S^{(\ell)}(\mathtt{b}_1, \ldots, \mathtt{b}_k) && (7.2.13)\\
&= \mathcal{S}_S^{(0)}(\mathtt{b}_1, \ldots, \mathtt{b}_k) + \sum_{\ell \geqslant 1} \mathcal{S}_S^{(\ell)}(\mathtt{b}_1, \ldots, \mathtt{b}_k) && (7.2.14)\\
&= \mathrm{ev}(a) + \sum_{\ell \geqslant 0} \mathcal{S}_S^{(\ell+1)}(\mathtt{b}_1, \ldots, \mathtt{b}_k) && (7.2.15)\\
&= \mathrm{ev}(a) + \sum_{\ell \geqslant 0} \mathcal{S}_S^{(\ell)}(\mathrm{subs}(\mathtt{b}_1), \ldots, \mathrm{subs}(\mathtt{b}_k)) && (7.2.16)\\
&= \mathrm{ev}(a) + \mathcal{S}_S(\mathrm{subs}(\mathtt{b}_1), \ldots, \mathrm{subs}(\mathtt{b}_k)). && \square
\end{aligned}
$$

La proposition 7.2.3 donne une équation fonctionnelle de point fixe pour la série génératrice d'une grammaire synchrone S localement finie, et le lemme 7.2.2 donne, quant à lui, un moyen de l'approximer jusqu'à un ordre n désiré en sommant les polynômes $\mathcal{S}_S^{(\ell)}$ pour tout $0 \leqslant \ell \leqslant n$.

Dans notre exemple, la série génératrice de S_{epl} vérifie l'équation fonctionnelle de point fixe

$$\mathcal{S}_{S_{\mathrm{epl}}}(x, y) = x + \mathcal{S}_{S_{\mathrm{epl}}}(xy + x^2y, x). \tag{7.2.17}$$

7.2.3 Spécialisations et raffinements

Dans bon nombre de situations, certains bourgeons d'une grammaire synchrone sont nécessaires uniquement dans un rôle de *catalyseur*, *i.e.*, les objets que l'on souhaite dénombrer sont ceux exempts de ces bourgeons, ces derniers étant uniquement utiles dans le processus de dérivation. Il est utile dans ce cas de spécialiser à 0 une variable $\mathtt{b}$ qui joue un rôle de catalyseur dans la série génératrice $\mathcal{S}_S$. Ainsi, les monômes qui correspondent à cette variable sont annihilés, et les arbres à bourgeons qui possèdent des bourgeons de type $\mathtt{b}$ ne sont de ce fait pas dénombrés.

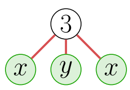
De même, on peut souhaiter dénombrer les arbres à bourgeons engendrés par S non plus selon leur évaluation, mais plus grossièrement selon leur taille. Dans ce cas, la spécialisation à t de chacune des variables $\mathtt{b}$ où t est une paramètre formel convient. De plus, il est facile d'accorder des tailles différentes à chacun des types de bourgeons qui constituent les arbres à bourgeons du langage de S en spécialisant chacune des variables $\mathtt{b}$ par t^n où n est la taille des bourgeons de type $\mathtt{b}$.

Un procédé similaire permet de raffiner $\mathcal{S}_S$ en tenant compte du nombre d'applications de certaines règles de substitution qui interviennent pour engendrer les arbres à bourgeons du langage de S. En effet, pour tenir compte du nombre d'applications d'une règle de substitution $\mathtt{b} \mapsto D$ intervenant dans la génération d'un arbre à bourgeons, il suffit de poser

$$\mathrm{subs}_\xi(\mathtt{b}) := \mathrm{subs}(\mathtt{b}) + (\xi - 1)\,\mathrm{ev}(D), \tag{7.2.18}$$

où ξ est un paramètre formel, et d'utiliser $\mathrm{subs}_\xi(\mathtt{b})$ à la place de $\mathrm{subs}(\mathtt{b})$ dans l'expression de la série génératrice de S. Ainsi, le paramètre ξ compte le nombre d'applications de la règle de substitution $\mathtt{b} \mapsto D$. De cette manière, une grammaire synchrone permet de dénombrer les éléments de son langage suivant des statistiques.

Par exemple, en ce qui concerne la grammaire synchrone S_{epl}, pour tenir compte du nombre d'applications de la règle de substitution

$x \longmapsto$  , (7.2.19) il suffit de poser

$$\mathrm{subs}_\xi(x) := xy + x^2 y\xi, \tag{7.2.20}$$

de sorte que la série génératrice $\mathcal{S}_{S_{\mathrm{epl}}}$, vérifiant l'équation fonctionnelle de point fixe

$$\mathcal{S}_{S_{\mathrm{epl}}}(x, y, \xi) = x + \mathcal{S}_{S_{\mathrm{epl}}}\left(xy + x^2 y\xi, x, \xi\right), \tag{7.2.21}$$

dénombre les arbres à bourgeons engendrés par S_{epl} selon la statistique associant à un arbre son nombre de nœuds d'arité 3.

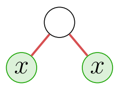
7.3 Exemples de grammaires synchrones

Nous considérons ici trois exemples de grammaires synchrones pour illustrer les concepts présentés dans les paragraphes précédents.

7.3.1 La grammaire synchrone des arbres binaires parfaits

Soit la grammaire synchrone $S_{\mathrm{perf}} := \left(\{x\}, x, R\right)$ où R contient l'unique règle de substitution

$$x \longmapsto$$



. (7.3.1)

En identifiant les bourgeons de type x avec des feuilles, le langage $\mathcal{L}_{S_{\mathrm{perf}}}$ est l'ensemble des *arbres binaires parfaits*, c'est-à-dire l'ensemble des arbres binaires qui apparaissent dans la suite $(T_i)_{i \geqslant 0}$ définie par $T_0 := \perp$ et $T_{i+1} := T_i \wedge T_i$.

Cette grammaire synchrone est localement finie car la taille d'un arbre à bourgeons engendré par S_{perf} augmente strictement lors de chaque étape de dérivation. De plus, comme S_{perf} est localement finie et que R contient une seule règle de substitution, le graphe de génération $\mathcal{G}_{S_{\mathrm{perf}}}$ contient un unique chemin de longueur maximale, ce qui montre que S_{perf} est non ambiguë. Ainsi, la série $\mathcal{S}_{S_{\mathrm{perf}}}$ est bien définie, et par la proposition 7.2.3, celle-ci vérifie l'équation fonctionnelle de point fixe

$$\mathcal{S}_{S_{\mathrm{perf}}}(x) = x + \mathcal{S}_{S_{\mathrm{perf}}}\left(x^2\right), \tag{7.3.2}$$

et dénombre les arbres binaires parfaits selon leur nombre de feuilles.

Les premiers polynômes $\mathcal{S}^{(\ell)}_{S_{\mathrm{perf}}}$ sont

$$\mathcal{S}^{(0)}_{S_{\mathrm{perf}}}(x) = x, \qquad \mathcal{S}^{(1)}_{S_{\mathrm{perf}}}(x) = x^2, \qquad \mathcal{S}^{(2)}_{S_{\mathrm{perf}}}(x) = x^4, \qquad \mathcal{S}^{(3)}_{S_{\mathrm{perf}}}(x) = x^8, \tag{7.3.3}$$

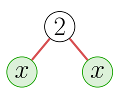
et sa série génératrice est de la forme

$$\mathcal{S}_{S_{\mathrm{perf}}}(x) = \sum_{n \geqslant 0} x^{2^n} = x + x^2 + x^4 + x^8 + \cdots. \tag{7.3.4}$$

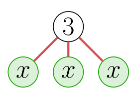
7.3.2 La grammaire synchrone des arbres $2, 3$-équilibrés

Soit la grammaire synchrone $S_{23} := \left(\{x\}, x, R\right)$ où R contient les règles de substitution

$$x \longmapsto$$



$+$



. (7.3.5)

En identifiant les bourgeons de type x avec des feuilles, le langage de S_{23} est l'ensemble des *arbres* $2, 3$*-équilibrés*, qui sont des arbres plans enracinés tels que tout nœud interne possède exactement 2 ou 3 fils, et tous les chemins connectant la racine aux feuilles sont de mêmes longueurs (voir [Odl82], [FS09]).

Étant donné que chaque étape de dérivation augmente strictement la taille de l'arbre à bourgeons sur lequel elle est appliquée, S_{23} est localement finie. De plus, S_{23} vérifie les hypothèses du lemme 7.1.11 et est ainsi non ambiguë. En effet, les deux arbres à bourgeons qui apparaissent dans les deux règles de substitution ont des racines différentes : l'une d'elles est d'arité 2 tandis que l'autre est d'arité 3. La série $\mathcal{S}_{S_{23}}$ est donc bien définie, et par la proposition 7.2.3, celle-ci vérifie l'équation fonctionnelle de point fixe

$$\mathcal{S}_{S_{23}}(x) = x + \mathcal{S}_{S_{23}}\left(x^2 + x^3\right), \tag{7.3.6}$$

et dénombre les arbres $2, 3$-équilibrés selon leur nombre de feuilles.

Les premiers polynômes $\mathcal{S}^{(\ell)}_{S_{23}}$ sont

$$\mathcal{S}^{(0)}_{S_{23}}(x) = x, \tag{7.3.7}$$

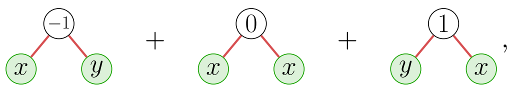
$$\mathcal{S}^{(1)}_{S_{23}}(x) = x^2 + x^3, \tag{7.3.8}$$

$$\mathcal{S}^{(2)}_{S_{23}}(x) = x^4 + 2x^5 + 2x^6 + 3x^7 + 3x^8 + x^9. \tag{7.3.9}$$

Une interprétation du polynôme $\mathcal{S}^{(2)}_{S_{23}}(x)$ est la suivante : par des dérivations de deux étapes depuis son axiome, S_{23} engendre un arbre à bourgeons de taille 4, deux arbres à bourgeons de taille 5, deux arbres à bourgeons de taille 6, trois arbres à bourgeons de taille 7, trois arbres à bourgeons de taille 8, et un arbre à bourgeons de taille 9.

7.3.3 La grammaire synchrone des arbres binaires équilibrés

Considérons maintenant la grammaire synchrone $S_{\mathrm{bal}} := \left(\{x, y\}, x, R\right)$ où R contient les règles de substitution

$x \longmapsto$ 

, (7.3.10)

$$y \longmapsto x. \tag{7.3.11}$$

Comme nous allons le voir, en annihilant les arbres à bourgeons qui contiennent des bourgeons de type y — qui jouent le rôle de catalyseurs — et en identifiant les bourgeons de type x avec des feuilles, le langage S_{bal} est l'ensemble des arbres binaires équilibrés (voir la définition 8.1.1 du chapitre 8).

La *mesure de déséquilibre* d'un nœud d'un arbre binaire est la différence de hauteur entre son sous-arbre droit et son sous-arbre gauche.

Proposition 7.3.1. *Soit D un arbre à bourgeons engendré par S_{bal} contenant uniquement des bourgeons de type x. Alors, les nœuds de D sont étiquetés par leur mesure de déséquilibre.*

Démonstration. Toute étape de dérivation menant à la génération de D substitue chaque bourgeon de type x par un arbre à bourgeons de hauteur 2, et chaque bourgeon de type y par un arbre à bourgeons de hauteur 1. Par conséquent, toute étape de dérivation incrémente la hauteur des sous-arbres qui contiennent un bourgeon de type x, et le rôle des bourgeons de type y est d'imposer un délai, pendant une étape de dérivation, sur la croissance de la branche qui le contient. Ainsi, des nœuds ayant des mesures de déséquilibre à -1 et 1 peuvent être créés. Finalement, comme D ne possède aucun bourgeon de type y, chaque délai de croissance est respecté, et ainsi, les étiquettes de D sont les mesures de déséquilibre de ses nœuds. ☐

La proposition 7.3.1 montre que les arbres à bourgeons engendrés par S_{bal} contenant uniquement des bourgeons de type x sont des arbres binaires équilibrés. De plus, il suit par induction structurelle sur l'ensemble des arbres binaires équilibrés que tout arbre binaire équilibré peut être engendré par S_{bal}. En effet, l'arbre vide peut être engendré puisqu'il correspond à son axiome, et si T est un arbre binaire équilibré et z sa racine, par hypothèse d'induction, son sous-arbre gauche et son sous-arbre droit peuvent être engendrés par S_{bal}. Pour engendrer T, il suffit de réaliser la première étape de substitution suivant la mesure de déséquilibre de z. La Figure 7.3 montre les étapes de dérivation qui engendrent un arbre binaire équilibré.

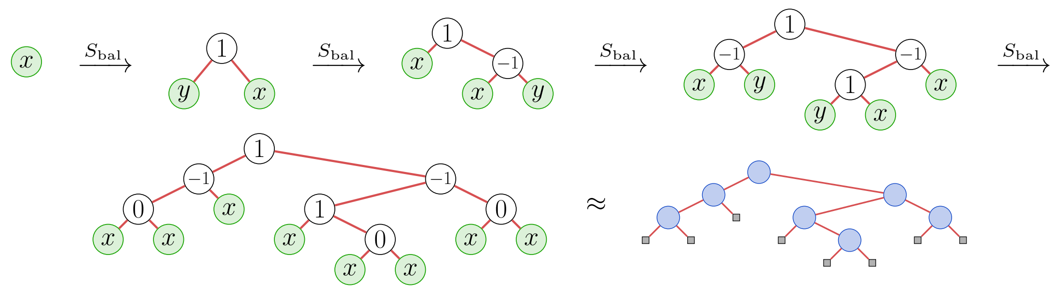


Figure 7.3 – Les étapes de dérivation pour la génération d'un arbre binaire équilibré par la grammaire synchrone S_{bal}.

En posant $y \leqslant_B x$, S_{bal} vérifie les conditions du lemme 7.1.9, ce qui montre qu'elle est localement finie. De plus, le lemme 7.1.11 montre que S_{bal} est non ambiguë puisque les arbres à bourgeons qui apparaissent dans ses règles de substitution possèdent une racine différente puisque leur étiquetage diffère. La proposition 7.3.1 montre que l'étiquetage des nœuds est consistant. Ainsi, la série génératrice $\mathcal{S}_{S_{\mathrm{bal}}}$ est bien définie. Par la proposition 7.2.3, la série génératrice qui dénombre les éléments de $\mathcal{L}_{S_{\mathrm{bal}}}$ vérifie l'équation fonctionnelle de point fixe

$$\mathcal{S}_{S_{\mathrm{bal}}}(x,y) = x + \mathcal{S}_{S_{\mathrm{bal}}}\left(x^2 + 2xy, x\right). \tag{7.3.12}$$

Les premiers polynômes $\mathcal{S}^{(\ell)}_{S_{\mathrm{bal}}}$ sont

$$\mathcal{S}^{(0)}_{S_{\mathrm{bal}}}(x,y) = x, \tag{7.3.13}$$

$$\mathcal{S}^{(1)}_{S_{\mathrm{bal}}}(x,y) = 2xy + x^2, \tag{7.3.14}$$

$$\mathcal{S}^{(2)}_{S_{\mathrm{bal}}}(x,y) = 4x^2y + 2x^3 + 4x^2y^2 + 4x^3y + x^4. \tag{7.3.15}$$

Comme nous l'avions mentionné, pour dénombrer les arbres binaires équilibrés, il faut rejeter les éléments de $\mathcal{L}_{S_{\mathrm{bal}}}$ qui contiennent des bourgeons de type y. Pour cette raison, la série génératrice qui dénombre les arbres binaires équilibrés selon leur nombre de feuilles est donnée par la spécialisation $\mathcal{S}_{S_{\mathrm{bal}}}(x,0)$. Notons que cette équation fonctionnelle de point fixe est obtenue également dans [BLL88], [BLL94], et [Knu98] dans un autre formalisme et par d'autres méthodes.